FastReport Benutzerhandbuch (.NET, Mono, WPF)

Version 2023.3

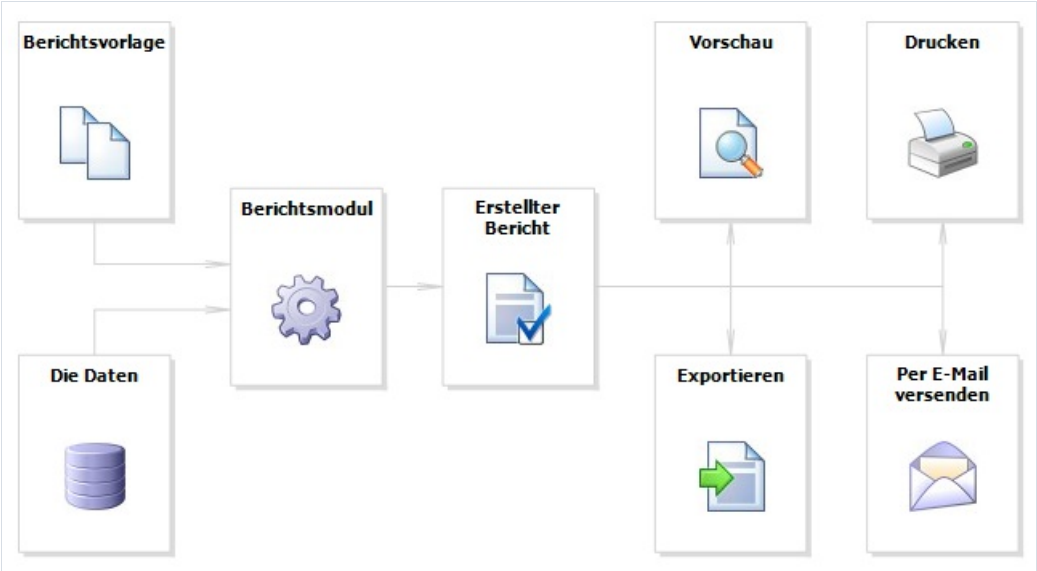
© 2008-2023 Fast Reports Inc.

Grundsätze

In diesem Kapitel werden die Prinzipien der Arbeit mit Berichten in FastReport vermittelt. Es werden Details zu Berichtselementen, wie Berichtsseiten, Leisten und Berichtsobjekten erläutert.

Der Bericht

Der Prozess der Berichtsbildung kann folgendermaßen dargestellt werden:



Berichtsvorlage (im weiteren Bericht): Diese wird im Designer angezeigt. Berichte werden als .FRX-Erweiterung gespeichert. Ein Bericht kann mit Hilfe des Designers oder programmatisch erstellt werden.

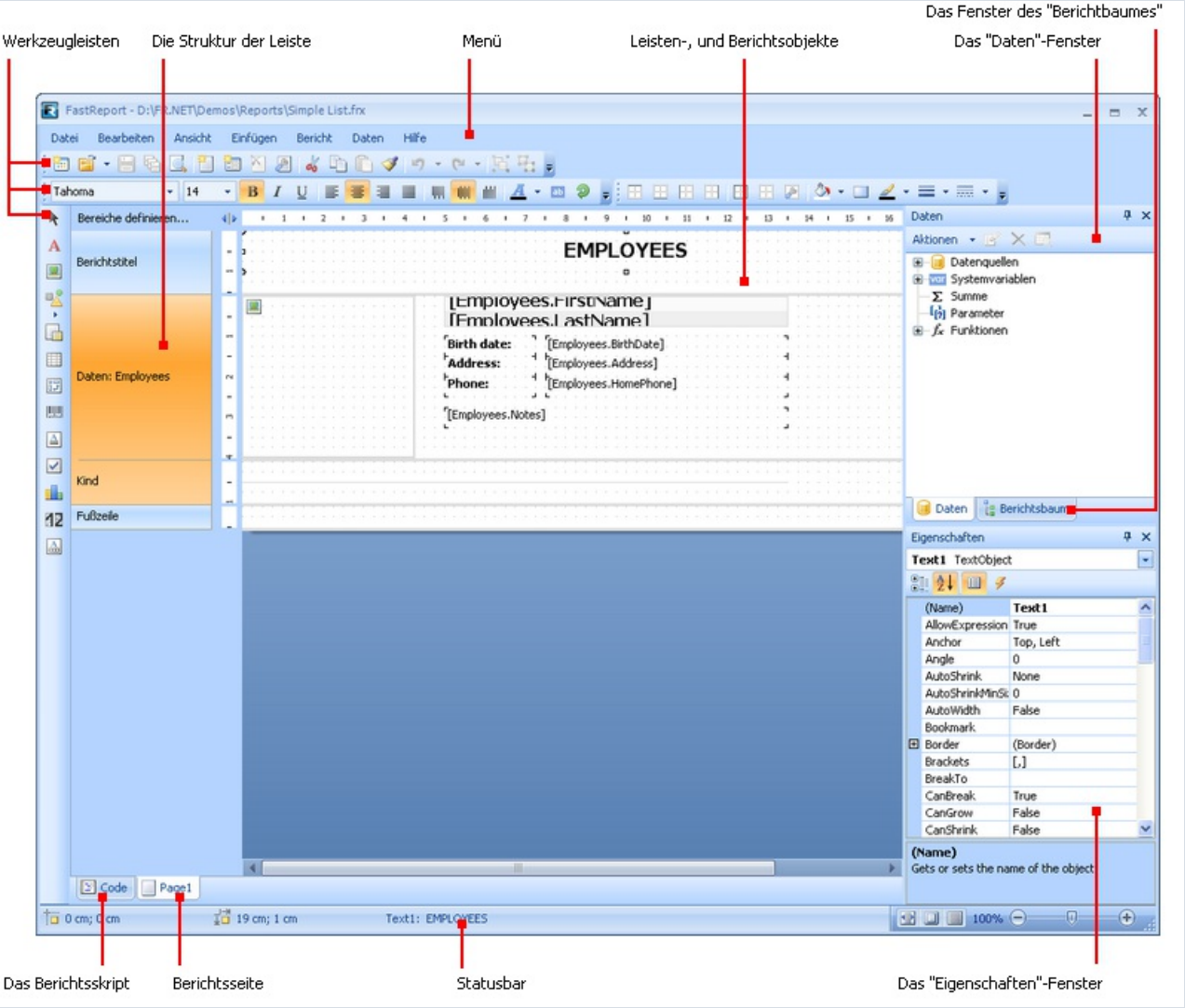
Die Daten können vielfältig sein: Daten, die in dem Programm festgelegt werden, oder Daten aus einer DBMS oder beispielsweise aus MS oder SQL. FastReport funktioniert auch mit Anwendungslogik-Objekten (im weiteren Anwendungslogik).

Erstellter Bericht: Wird im Vorschaufenster angezeigt. Ein erstellter Bericht kann in einer Vorschau angezeigt, gedruckt, in einem der unterstützen Formate (.doc, .xls, .pdf, etc.) gespeichert oder mittels E-Mail versandt werden.

Bericht-Designer

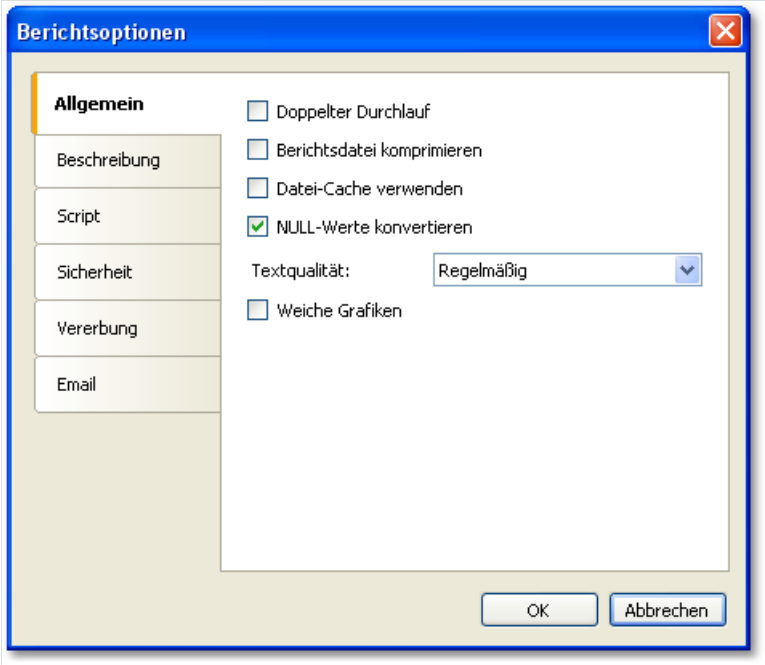
Eine Berichtsvorlage wird im Bericht-Designer erstellt. Der Designer bietet dem Benutzer bedarfsgerechte Hilfen zur Erstellung des Berichts und ermöglicht gleichzeitig eine Berichtsvorschau.

Der Bericht-Designer ist ein integraler Bestandteil von FastReport und ist nicht von der Entwicklungsumgebung (z.B. MS, Visual Studio) abhängig. Als Softwaredesigner können Sie den Bericht-Designer in Ihre Anwendung integrieren. Dies erlaubt Ihren Endnutzern den bestehenden Bericht zu ändern oder einen Neuen zu erstellen.



Berichtsoptionen

Ein Fenster mit Berichtsoptionen kann über das Menü "Bericht|Optionen" aufgerufen werden. Es wird ein Dialogfenster mit mehreren Registerkarten angezeigt:



Mittels der Registerkarte "Allgemein" kann man die folgenden Berichtsparameter steuern:

· Der Parameter "Doppelter Durchlauf" erlaubt die Aktivierung der zwei Bericht Durchläufe. Dies kann notwendig werden, wenn man die Systemvariable "Gesamtanzahl Seiten" nutzt;

· Der Parameter "Berichtsdatei komprimieren" erlaubt die komprimierte Speicherung eines Berichts. Zur Komprimierung wird ein Zip-Algorithmus verwendet. So kann der Originalinhalt bequem mit der Hilfe eines beliebigen Archivs extrahiert werden;

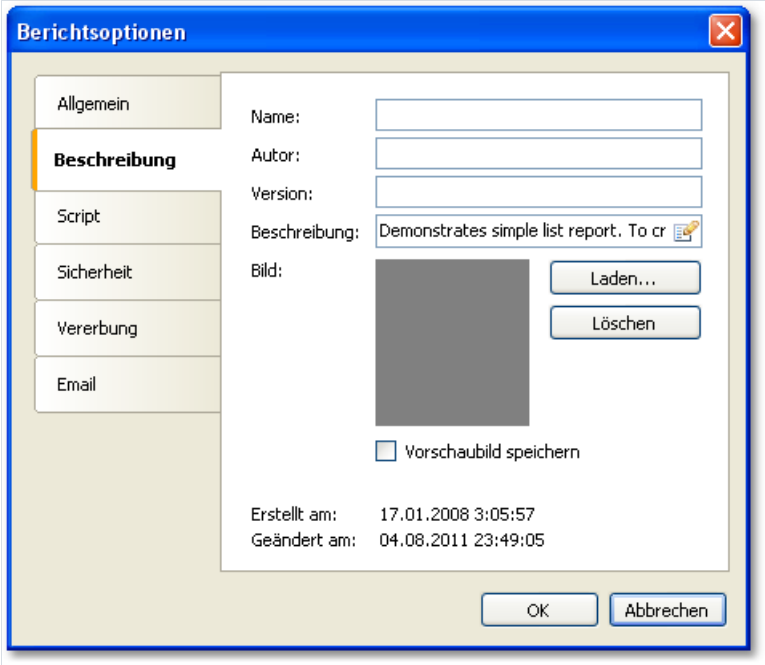
· Der Parameter "Datei-Cache verwenden" ermöglicht die Speicherung des Speichers bei der Berichtserstellung. Dies sollte für einen Bericht mit einer umfangreichen Seitenzahl genutzt werden;

· Der Parameter "NULL-Werte konvertieren" steuert die Konvertierung der nullwertigen Datenspalten im Standardwert (0, leere Zeichenfolge, falsch - abhängig vom Datentyp der Spalte);

· Der Parameter "Textqualität" ermöglicht die Auswahl der Textanzeige im Bericht. Dieser Modus hat keinen Einfluss auf den Druck des Berichtes;

· Der Parameter "Weiche Grafiken" aktiviert den Glättungsmodus bei der Zeichnung graphischer Objekte (Zeile, Rahmen, Bild).

Über die Registerkarte "Beschreibung" kann man die Beschreibung des Berichts eingeben. All diese Parameter sind nicht obligatorisch und dienen lediglich Informationszwecken:



Über die Registerkarte "Script" kann man die Skriptsprache des Berichts wählen. Detaillierte Informationen über Skript sind im Kapitel "Script" erläutert.

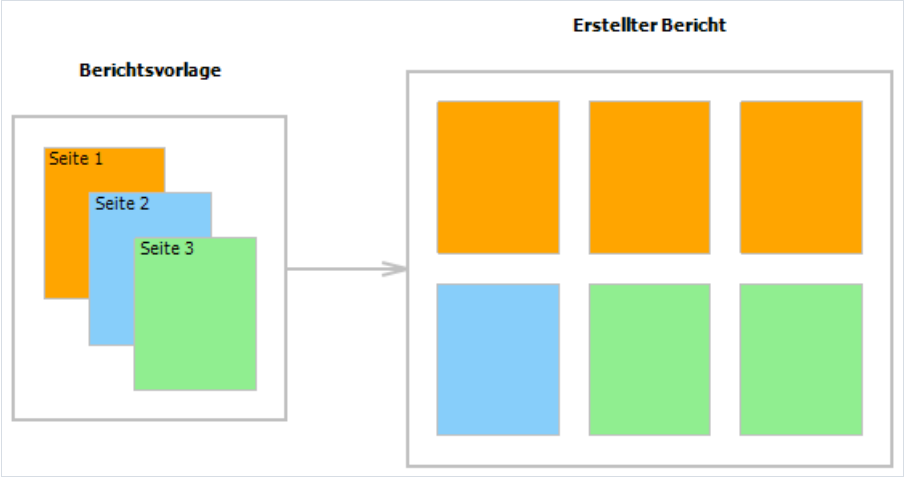
Über die Registerkarte "Sicherheit" kann man das, zur Öffnung des Berichts, erforderliche Passwort eingeben. Ein passwortgeschützter Bericht wird in verschlüsselter Form gespeichert: Man sollte daher sein Passwort nicht vergessen! In einem solchen Fall ist die Wiederherstellung eines Berichts nahezu unmöglich.

Über die Registerkarte "Vererbung" kann man die Vererbung steuern. Diese Funktionalität wird später genauer betrachtet.

Berichtsseiten

Eine Vorlage besteht (zumeist) aus einer oder mehreren Seiten. Die Berichtsseite wiederum enthält Leisten. Auf diesen befinden sich Berichtsobjekte, wie Text, Bild etc.:

Die Berichtsvorlage kann sich aus mehreren Seiten zusammensetzen. Mann kann beispielsweise eine Vorlage erstellen, die aus einer Titelseite und einer Datenseite besteht. Bei der Erstellung eines solchen Berichts, werden die Seiten in angegebener Reihenfolge gedruckt (erste Seite, zweite Seite etc.). Jede Vorlagenseite kann, abhängig von den enthaltenen Daten, eine oder mehrere Seiten eines Berichts generieren:



Berichtsseiten werden auch für die Arbeit mit Teilberichten genutzt. Im Gegensatz zu anderen Reportgeneratoren, werden Teilberichte in FastReport als separate Vorlageseite und nicht als separate Datei gespeichert.

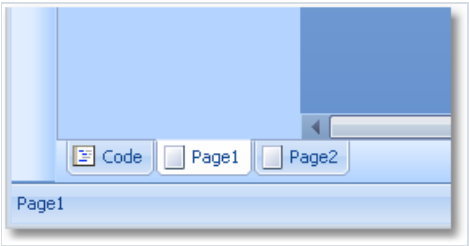
Neben den Berichtsseiten kann eine Vorlage eine oder mehrere Dialogmasken enthalten. Dialogmasken können zur Untersuchung bestimmter Parameter vor der Berichterstellung genutzt werden. Detaillierte Informationen zu Dialogmasken werden im Kapitel "Dialog" erläutert.

Seitenverwaltung

Ein neuer Bericht enthält bereits eine Seite mit mehreren Leisten. Zum Hinzufügen eine neuen Seite, klickt man auf die Schaltfläche . Durch Klicken der Schaltfläche und Auswahl "Neue Berichtsseite" in einem Fenster, lässt sich eine neue Seite hinzufügen.

Dialogmasken werden dem Bericht auf ähnliche Art hinzugefügt. Hierzu wird die Schaltfläche gedrückt.

Vorlageseiten werden im Designer als Registerkarten angezeigt:



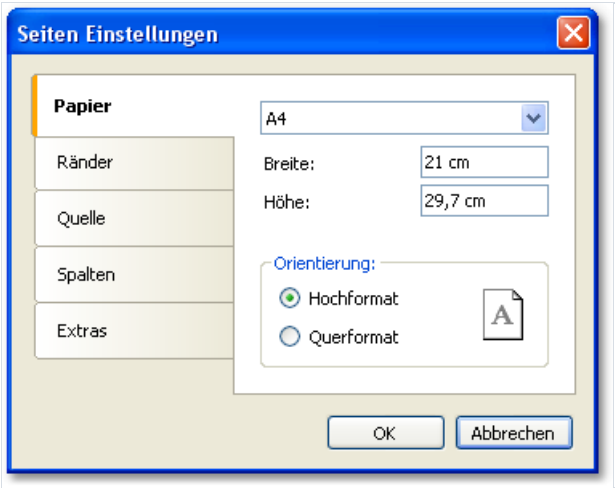
Die erste Registerkarte enthält den Berichtcode. Diese kann weder verschoben, noch gelöscht werden.

Zum Wechsel auf die benötigte Seite wird auf die entsprechende Registerkarte gedrückt. Die Reihenfolge der Seiten kann mit der Maus verändert werden. Hierzu klickt man mit der linken Maustaste auf die Registerkarte und zieht diese mit gedrückter Maustaste in die erwünschten Position.

Zum Löschen einer Seite klickt man die Schaltfläche . Bei einem einseitigen Bericht ist diese Schaltfläche ist nicht aktiviert.

Seiten Einstellungen

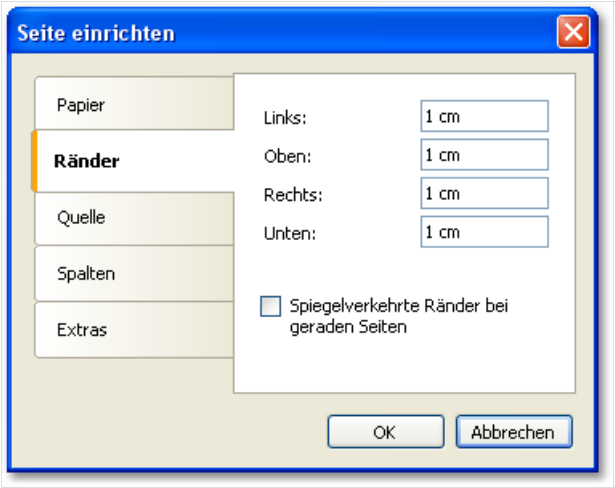
Für jede Berichtsseite können individuelle Eigenschaften, wie Papierformat, Ausrichtung (Hoch-, oder Querformat), Seitenränder, Spalten, Papierzufuhr etc. festgelegt werden. Berichtvorlagen können mehrere Seiten mit unterschiedlichen Ausrichtungen und Papierformaten umfassen. Das Fenster zur Seiteneinrichtung kann durch Klicken der Schaltfläche oder über den Menübefehl "Datei|Seiten Einstellung" aufgerufen werden.



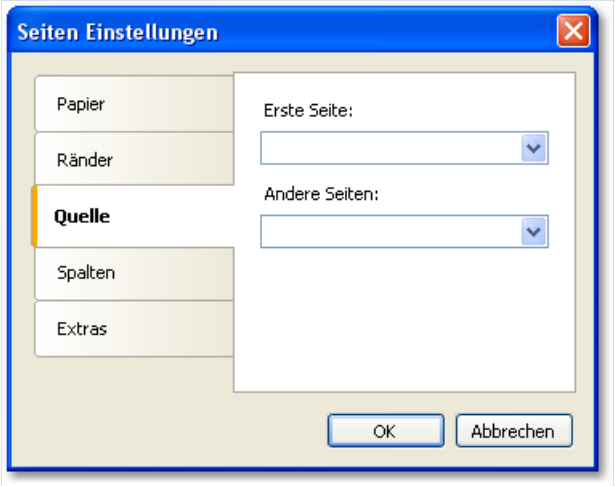
Die Gruppe "Papier" ermöglicht die Festlegung des Papierformats und der Ausrichtung. Die unterstützen Formate können über eine Dropdownliste ausgewählt werden. Diese enthält alle vom aktuellen Drucker unterstützen Papierformate.

Aktuelle Drucker können über "Datei|Druckereinstellung" konfiguriert werden.

Die Gruppe "Ränder" ermöglicht die Festlegung der Seitenränder. Die Option "Spiegelverkehrte Ränder bei geraden Seiten" wird für Broschüren verwendet.

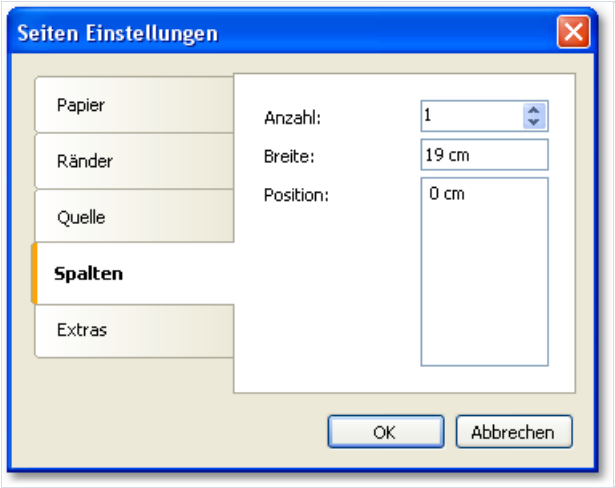


Die Gruppe "Quelle" ermöglicht eine Auswahl der Papierzufuhr. Das Papier kann für verschiedene Seiten unterschiedlich zugeführt werden, z.B. die erste erstelle Berichtsseite und die restlichen Seiten:

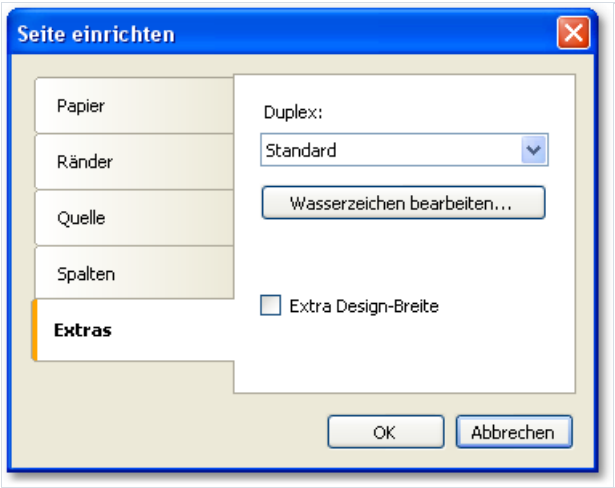


Die Zufuhr kann im Dialog "Drucken" ausgewählt werden.

Die Gruppe "Spalten" ermöglicht eine Festlegung der Spaltenparameter für mehrspaltige Berichte. Hierzu muss die Spaltenanzahl angegeben werden und (optional) kann die Spaltenbreite und die Positionierung einer jeden Spalte korrigiert werden.



Die Gruppe "Extras" ermöglicht die Festlegung weiterer hilfreicher Seiteneigenschaften. Sollte der Drucker diesen Modus unterstützen, lässt sich hier der Duplexmodus für den Duplexdruck festlegen. Hier lassen sich auch Wasserzeichen für den Druck auf den erstellen Berichtsseiten eingeben.



Das Kontrollkästchen "Extra Design-Breite" ermöglicht eine Vergrößerung der Seitenbreite im Entwurfsmodus. Eventuell ist die Arbeit mit Objekten, wie Tabelle oder Matrix, hilfreich.

Der Duplexmodus kann auch im Dialog "Drucken" ausgewählt werden.

Leisten

Die Leiste ist ein Objekt, das sich direkt auf der Berichtsseite befindet und als Container für andere Objekte, wie "Text", "Bild" etc., dient.

Insgesamt bietet FastReport 13 Leistentypen. Die Leiste wird, abhängig von der Art, an bestimmten Positionen im Bericht gedruckt.

Leiste	Art des Drucks
Berichtstitel	Dieser wird einmal zu Beginn des Berichts gedruckt. Die Druckreihenfolge (vor oder nach der Leiste "Kopfzeile") kann mittels der Seiteneigenschaft "TitleBeforeHeader" ausgewählt werden. Diese Eigenschaft kann mit Hilfe des Fensters "Eigenschaften" geändert werden. Die Eigenschaft entspricht standardmäßig wahr, d.h. der Bericht Titel wird vor der Kopfzeile gedruckt.
Berichtszusammenfassung	Diese wird einmal am Ende des Berichts, nach der letzten Datenreihe, jedoch vor der Leiste "Fußzeile", gedruckt.
Kopfzeile	Diese wird am oberen Rand jeder Seite des Berichts gedruckt.
Fußzeile	Diese wird am unteren Rand jeder Seite des Berichts gedruckt.
Spaltenkopf	Diese Leiste wird beim Druck eines mehrspaltigen Berichts (wenn die Anzahl der Spalten in der Seiteneinstellung > 1 ist) genutzt. Sie wird am oberen Rand einer jeden Spalte nach der Leiste Kopfzeile gedruckt.
Spaltenfuß	Wird am unteren Rand jeder Spalte, vor der Fußzeile, gedruckt.
Daten	Diese Leiste ist mit der Datenquelle verbunden und wird so häufig gedruckt, wie die Quelle Zeilen aufweist.
Kopfdaten	Diese Leiste ist mit der Leiste "Daten" verbunden und wird vor der ersten Datenreihe gedruckt.
Fußdaten	Diese Leiste ist mit der Leiste "Daten" verbunden und wird nach der letzten Datenreihe gedruckt.
Gruppenkopf	Diese wird zu Beginn jeder Gruppe, wenn sich der Wert der Gruppenbedingung ändert, gedruckt.
Gruppenfuß	Diese wird am Ende jeder Gruppe gedruckt.
Kind	Diese Leiste kann mit jeder anderen Leiste, eingeschlossenen einer Leiste eines anderen unterordneten Elementes, verbunden werden. Diese wird direkt nach dem entsprechenden übergeordneten Element gedruckt.
Overlay	Wird als Hintergrund auf jeder Berichtsseite gedruckt.

Leisten im Designer

Im Designer wird eine Leiste als rechteckiger Bereich angezeigt. Eine Leiste, wie viele andere Berichtsobjekte auch, kann einen Rahmen und eine Füllung haben. Beide sind standardmäßig deaktiviert. Ansonsten wird eine Leiste als Gitter angezeigt. Um den Gittermodus anzuzeigen, geht man zum Menü "Ansicht|Optionen" und wählt "Berichtsseite". Das Gitter kann über das Menü "Ansicht", aktiviert oder deaktiviert werden.

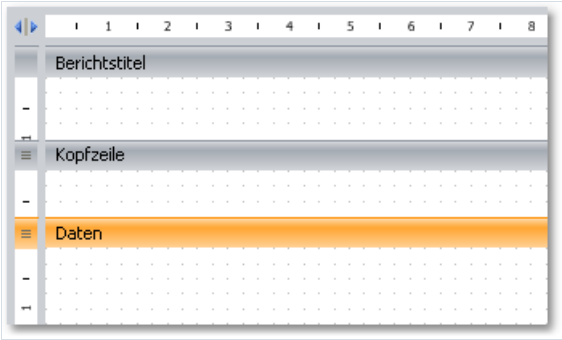
Die Leistenhöhe kann auf drei Arten festgesetzt werden:

· durch Platzierung des Mauszeigers auf den unteren Rand der Leiste. Die Form des Zeigers verwandelt sich in einen horizontalen "Teiler" und die Größe der Leiste kann verändert werden.

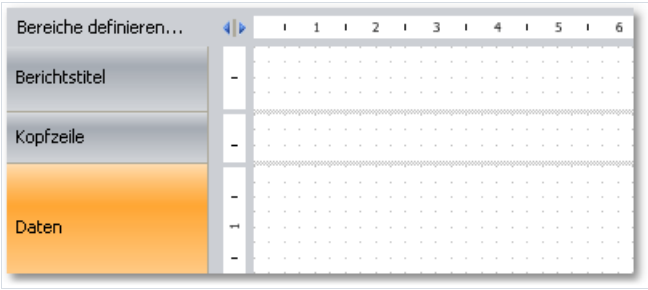
· durch Ziehen des Leistenziehpunktes im linken Lineal.

· durch Nutzung des Fensters "Eigenschaften" zur Festlegung der Höheneigenschaften des Bereichs.

Der Designer hat zwei Anzeigeleisten-Modi zwischen denen jederzeit gewechselt werden kann. Im ersten Modus hat jede Leiste eine Kopfzeile, die den Titel der Leiste und nützliche Informationen über diesen (z.B. Name der verbundenen Datenquelle) anzeigt.



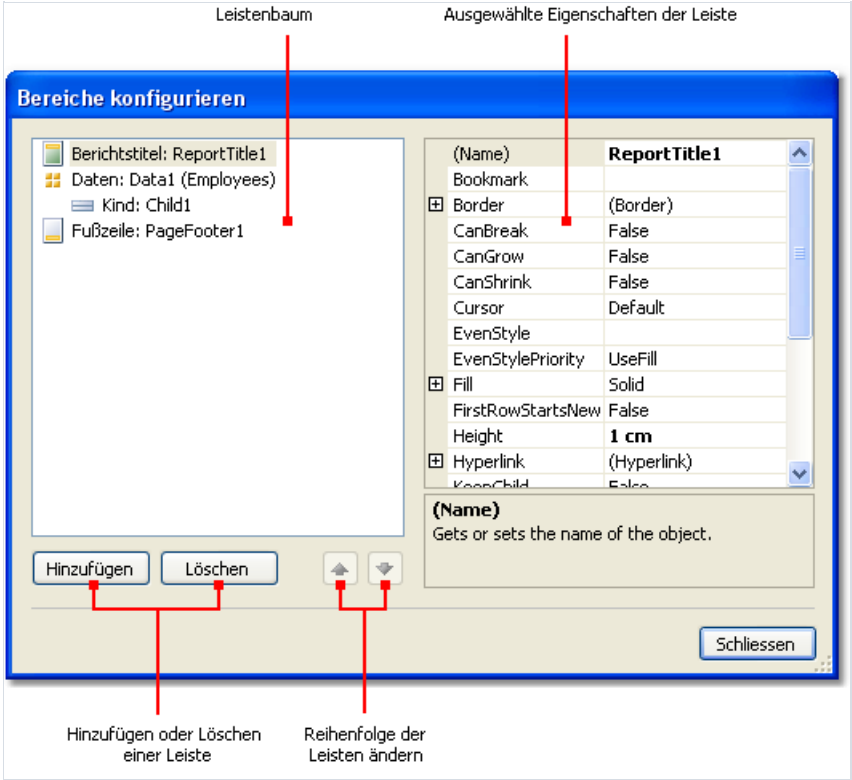
Der zweite Modus hat keine Kopfzeile; Stattdessen wird an der linken Seite des Fensters die Struktur der Leiste angezeigt. Dieser Modus führt zu einem besseren Verständnis des Berichts, speziell wenn dieser nicht von derselben Person erstellt wurde.



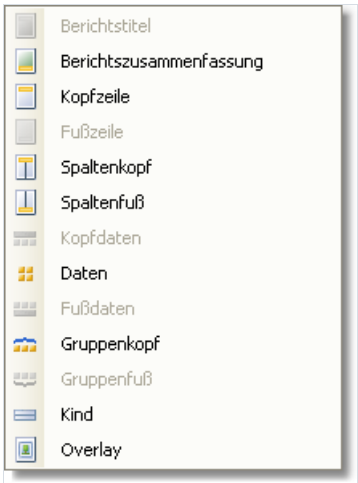
Durch Klicken der Schaltfläche kann zwischen den verschiedenen Modi gewechselt werden.

Konfiguration der Leisten

Die Leisten können in dem Fenster "Bereiche konfigurieren" konfiguriert werden. Dieses kann über das Menü "Bericht|Leistenkonfiguration" oder mittels der Schaltfläche "Leistenkonfiguration" (oberhalb des Leistenbaums) aufgerufen werden:

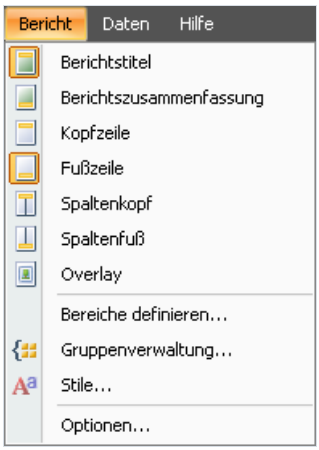


In diesem Fenster können Leisten zu dem Bericht hinzugefügt, diese gelöscht oder deren Reihenfolge geändert werden. Zum Hinzufügen einer Leiste klickt man auf die Schaltfläche "Hinzufügen" oder mit der rechten Maustaste auf den Leistenbaum. Es erscheint ein Kontextmenü mit einer Liste an Leisten. Eine Leiste, die nicht hinzugefügt werden kann, wird abgeblendet angezeigt.



Der Vorgang "Hinzufügen" ist von der Auswahl der Leisten auf dem Leistenbaum abhängig. So ist es beispielsweise nur möglich die Leisten "Datenkopf" und "Datenfuß" hinzuzufügen, wenn zuvor die Leiste "Daten" ausgewählt wurde.

Einige Leisten bieten eine weitere Möglichkeit der Konfiguration. Dazu nutzt man das Menü "Bericht":



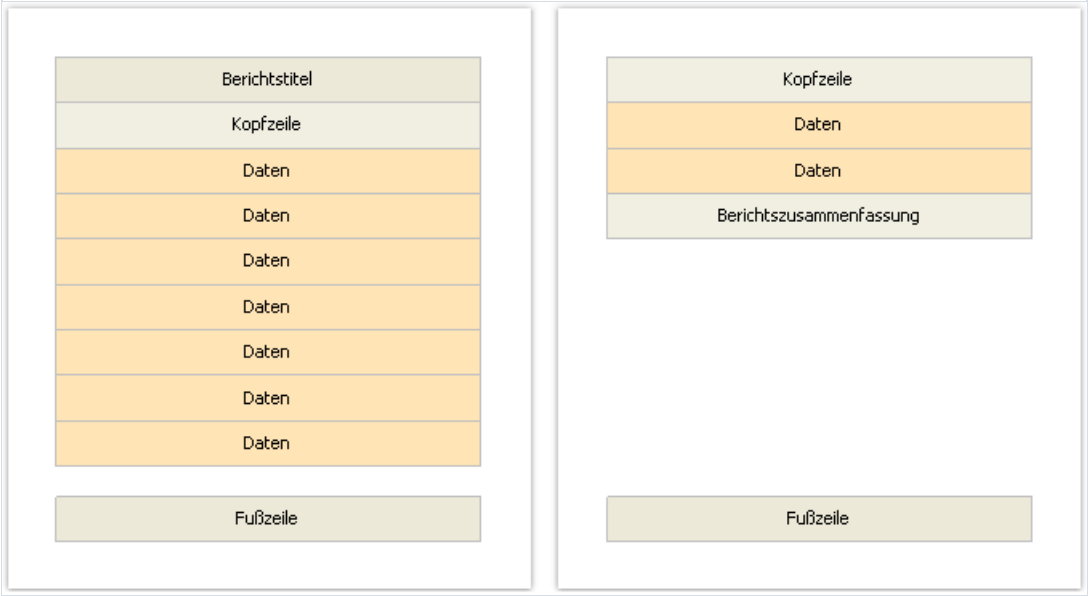
Um eine Leiste zu löschen, wird diese ausgewählt und die Taste "Entfernen" gedrückt.

Während der Konfiguration der Leisten, erlaubt FastReport keine Vorgänge, die zur Erstellung einer fehlerhaften Berichtvorlage führen würden. Beispielweise ist das Löschen der Leiste "Daten", die mit der Gruppe verbunden ist, nicht möglich; Hierzu muss zuerst die Gruppe gelöscht werden. Ebenso wird beim Löschen der Leiste "Daten", automatisch die entsprechende Kopf-, und Fußzeile gelöscht. Existiert nur eine Leiste auf der Seite, kann diese nicht gelöscht werden.

Druckleisten

Auf dieser Seite wird eine Vielzahl an Leisten angezeigt. Wie erstellt FastReport einen Bericht? Das folgende Beispiel soll dies erläutern:

Die Leiste "Berichtstitel" wird als erste gedruckt. Direkt im Anschluss wird die "Kopfzeile" gedruckt. Danach wird die Leiste "Daten", entsprechend der Anzahl der Zeilen in der Datenquelle mit der sie verbunden ist, gedruckt. Nachdem alle Zeilen der Leiste "Daten" gedruckt sind, wird die Leiste "Berichtszusammenfassung" gedruckt und, am Ende der Seite, die Leiste "Fußzeile". Hiernach ist der Druck des Berichts beendet. Ein fertiger Bericht sieht ungefähr folgendermaßen aus:



Während der Druckbearbeitung überprüft FastReport ob für den aufbereiteten Bericht ausreichend Platz auf der aktuellen Seite zur Verfügung steht, so dass die Leiste gedruckt werden kann. Ist ausreichend Platz vorhanden, wird das Folgende durchgeführt:

· Druck der Fußzeile;

· Hinzufügen einer neuen Seite;

· Druck der Kopfzeile;

· Fortführung des Drucks der Leisten, die nicht auf die vorherige Seite passten.

Leisteneigenschaften

Jede Leiste verfügt über mehrere Eigenschaften, die den Druckprozess betreffen. Diese können mittels des Kontextmenüs der Leiste konfiguriert werden. Hierzu wird mit der rechten Maustaste auf eine leere Stelle auf der Leiste gedrückt. Es ist auch möglich auf den Leistenkopf (im klassischen Anzeigemodus) oder auf die Leistenstruktur (in anderen Modi) zu drücken. Einen weitere Methode: Auswahl einer Leiste und Änderung der entsprechenden Eigenschaften im Fenster "Eigenschaften".

Eigenschaft	Beschreibung
CanGrow, CanShrink	Diese Eigenschaften legen fest, ob eine Leiste, abhängig von der Größe der in der Leiste beinhalteten Objekte, verkleinert oder vergrößert werden kann. Sind beide Eigenschaften aktiviert, weist die Leiste die im Designer festgelegten Eigenschaften auf. Weitere Information hierzu sind im Kapitel "Berichterstellung" erläutert.
CanBreak	Ist diese Eigenschaft aktiviert, wird FastReport versuchen einen Teil des Inhalts der Leiste auf dem verfügbaren Platz zu drucken, d.h. die Leiste "umzubrechen". Weitere Informationen hierzu werden im Kapitel "Berichterstellung" erläutert.
StartNewPage	Der Druck einer Leiste mit dieser Eigenschaft startet auf einer neuen Seite. Diese Eigenschaft wird üblicherweise genutzt, wenn Gruppen gedruckt werden - jede Gruppe wird dann auf eine neue Seite gedruckt.
PrintOnBottom	Eine Leiste mit diesen Eigenschaften wird am unteren Rand der Seite, vor der Leiste "Fußzeile" gedruckt. Dies kann beim Druck bestimmter Dokumente, bei denen die Gesamtanzahl der Seiten am unteren Rand der Seite gedruckt werden soll, nützlich sein.
RepeatOnEveryPage	Leisten die diese Eigenschaft aufweisen: "Datenfuß", "Gruppenkopf" und "Gruppenfuß". Ist der Druck der Daten erledigt, wird diese Art der Leiste auf jede neue Seite gedruckt. Weitere Informationen hierzu werden im Kapitel "Berichterstellung" erläutert.

Berichtsobjekte

In einem Bericht können eine Vielzahl an Objekten verwendet werden.

Symbol	Name	Beschreibung
	"Text" (TextObject)	Anzeige einer oder mehrerer Textzeilen.
	"Bild" (PictureObject)	Anzeige eines Bildes.
	"Zeile" (LineObject)	Anzeige einer Zeile Eine Zeile kann vertikal, horizontal oder diagonal sein.
	"Form" (ShapeObject)	Anzeige einer geometrischen Form: Rechteck, Ellipse, Dreieck etc.
	"Richt Text" (RichObject)	Anzeige eines formatierten Textes (als RTF-Format).
	"Barcode" (BarcodeObject)	Anzeige eines Barcodes.
	"Kontrollkästchen" (CheckBoxObject)	Anzeige eines Kontrollkästchens mit zwei Zuständen: "Aktiviert" und "Deaktiviert".
	"Tabelle" (TableObject)	Anzeige einer Tabelle mit Zeilen, Spalten und Zellen.
	"Matrix" (MatrixObject)	Anzeige einer Matrix (auch Kreuztabellenbericht genannt).
	"Diagramm"(MSChartObject)	Anzeige eines Diagramms.
	"Postleitzahl"(ZipCodeObject)	Anzeige einer Postleitzahl.
	"Zellentext"(CellularTextObject)	Anzeige eines jeden Zeichens in seiner entsprechenden Zelle.

Ein Objekt kann zur Anzeige einer Information ("Text"-Objekt) oder zur Verbesserung des Aussehens des Berichts ("Bild"-, "Zeile"-, "Form"-Objekt) genutzt werden. Komplexe Objekte, wie "Tabelle" oder "Matrix" können einfachere Objekte enthalten.

Eigenschaften allgemeiner Objekte

Alle Berichtsobjekte werden aus einer Basisklasse vererbt (ReportComponentBase) und haben eine bestimme Anzahl an allgemeinen Eigenschaften. Vor der detaillierten Erläuterung jedes Objektes, werden diese Eigenschaften näher betrachtet.

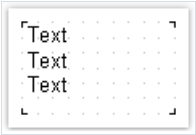
Die Eigenschaften können im Fenster "Eigenschaften" geändert werden. Einige Eigenschaften (zum Beispiel Rahmen und Füllung) können über das Kontextmenü des Objektes oder über die Symbolleiste geändert werden..

Eigenschaft	Beschreibung
Links, Oben, Breite, Höhe	Grundsätzlich ist ein Objekt rechteckig. Es hat Koordinaten (Eigenschaften Links, Oben) und Größeneigenschaften (Breite, Höhe).
Anker	Diese Eigenschaft bestimmt, wie das Objekt seine Position und/oder seine Größe ändert, wenn der Container auf dem es liegt vergrößert oder verkleinert wird. Durch die Nutzung eines Ankers kann das Objekt expandieren oder sich synchron mit dem Container verschieben. Weitere Informationen über diese Eigenschaft werden im Kapitel "Dynamisches Layout" erläutert.
Andocken	Diese Eigenschaft legt fest an welcher Seite des Containers das Objekt angedockt wird. Weitere Informationen über diese Eigenschaft werden im Kapitel "Dynamisches Layout" erläutert.
Rahmen, Füllung	Diese Eigenschaften steuern den Rahmen und die Füllung des Objektes. Diese können mittels der Symbolleiste "Rahmen und Füllung" geändert werden.
CanGrow, CanShrink	Diese Eigenschaften ermöglichen eine Anpassung der Objekthöhe an den Gesamttext. Weitere Informationen über diese Eigenschaft werden im Kapitel "Dynamisches Layout" erläutert.
ShiftMode	Ein Objekt, mit dieser aktivierten Eigenschaft, bewegt sich entsprechend der Vergrößerung oder Verkleinerung des darüber liegenden Objektes, hoch und runter. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Dynamisches Layout" erläutert.
GrowToBottom	Ein Objekt, für das diese Eigenschaft aktiviert ist, erstreckt sich bis zum Ende der Leiste. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Dynamisches Layout" erläutert.
CanBreak	Die Objekte "Text" und "Rich Text" weisen diese Eigenschaft auf. Sie bestimmt, ob die Inhalte des Objektes geteilt werden kann.
PrintOn	Diese Eigenschaft legt fest auf welchen Seiten das Objekt gedruckt werden kann. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Bericht: Broschüre" erläutert.
Zeiger	Diese Eigenschaft legt die Art des Zeigers fest, wenn sich dieser über einem Objekt befindet. Diese Eigenschaft funktioniert ausschließlich im Vorschaufenster.
Sichtbar	Diese Eigenschaft legt fest, ob das Objekt im Bericht angezeigt wird. Unsichtbare Objekte werden nie im Vorschaufenster angezeigt und nicht gedruckt.
Druckbar	Diese Eigenschaft legt fest, ob das Objekt gedruckt wird. Ist diese Eigenschaft deaktiviert, wird das Objekt zwar im Vorschaufenster angezeigt, jedoch nicht gedruckt.
Hyperlink	Dieses Objekt erlaubt die interaktive Gestaltung des Berichtsobjekts. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Interaktive Berichte" erläutert.

Eigenschaft	Beschreibung
Lesezeichen	Diese Eigenschaft wird in Kombination mit der Eigenschaft "Hyperlink" genutzt. Es kann beliebige Ausdrücke beinhalten. Eine Berechnung des Ausdrucks findet statt, wenn der Bericht arbeitet und sein Wert wird als Name des Lesezeichen genutzt.
Einschränkungen	Diese Eigenschaft erlaubt eine Einschränkung bestimmter Vorgänge, wie verschieben, Größe ändern oder löschen des Objektes.
Format	Hier kann der Eigenschaft einem Formatnamen zugeordnet werden. Nach Beendigung der Zuordnung ändert sich das Objekt entsprechend des angegebenen Formats. Ändern sich die Parameter des Formats, so wird sich auch das Aussehen des Objektes ändern.

Das "Text"-Objekt

Das "Text-Objekt" ist eines der Hauptobjekte und wird häufig genutzt. Es sieht folgendermaßen aus:



Das Objekt kann beliebige Textdaten anzeigen, im Besonderen:

· eine oder mehrere Textzeilen;

· Datenspalten;

· Berichtparameter;

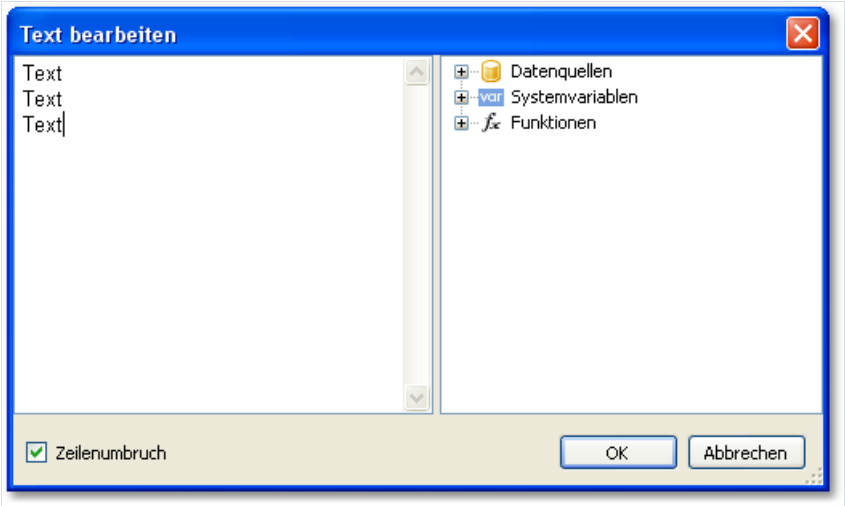
· Werte insgesamt;

· Ausdrücke;

· eine beliebige Kombination der oben genannten Elemente.

Textbearbeitung

Zur Bearbeitung des Textes eines Objektes muss es doppelt geklickt werden. Ein Texteditor wird eingeblendet.



Auf der rechten Seite des Editors befindet sich ein Datenbaum, dessen Elemente zum Text hinzugefügt werden können. Das Hinzufügen eines Elementes auf die erwünschte Position erfolgt mittels Ziehen (Drag) mit der Maus. Ein Element kann auch in den Text eingefügt werden, indem es doppelgeklickt wird; So wird es auf die aktuelle Position des Zeigers gesetzt.

Zur Speicherung der Änderungen und zum Schließen des Editorfensters, klickt man die Schaltfläche OK oder drückt die Strg+OK-Tasten.

Eine weitere Methode der Textbearbeitung stellt die direkte Bearbeitung dar. Hierzu wählt man das "Text"-Objekt und drückt die OK-Taste. Um die Bearbeitung zu beenden kann man auf einem beliebigen Platz außerhalb der Objektgrenzen klicken oder Strg+OK drücken. Durch Drücken der Esc-Taste werden die Änderungen verworfen.

Die direkte Bearbeitung eines Objektes ermöglicht eine Größenänderung mittels Maus.

Anzeige der Ausdrücke

Das "Text"-Objekt kann Nur-Text in Kombination mit Ausdrücken beinhalten. Zum Beispiel:

```
Heute ist [Datum]
```

Beim Druck eines solchen Objektes, werden alle im Text enthaltenen Ausrücke berechnet. Das Ergebnis kann folgendermaßen aussehen:

```
Heute ist der 12.09.2010.
```

Wie oben dargestellt, werden Ausdrücke durch rechteckige Klammern identifiziert. Dies wird in der Eigenschaft "Klammern" festgelegt. Diese enthält standardmäßig das Zeichen "[,]". Bei Bedarf können hier andere Symbole oder Zeichenfolgen verwendet werden, zum Beispiel "<,>" oder "<!,!>". Im letzteren Fall, wird der Ausdruck im Text folgendermaßen angezeigt:

```
Heute ist <!Datum!>
```

Es ist, darüber hinaus, möglich alle Ausdrücke zu deaktivieren. Hierzu muss die Eigenschaft AllowExpressions auf false gesetzt werden. Der Text wird dann "wie er ist" angezeigt.

Innerhalb der rechteckigen Klammern kann jeder beliebige Ausdruck genutzt werden. Weitere Informationen zu Ausdrücken werden im Kapitel "Ausdrücke" erläutert. Beispielweise ein Objekt mit dem folgenden Text:

```
2 * 2 = [2 * 2]
```

wird gedruckt als:

```
2 * 2 = 4
```

Häufig wird der Fehler begangen und der Ausdruck außerhalb der rechteckigen Klammern gesetzt. Man sollte beachten, dass es sich hier um einen Ausdruck handelt und dieser nur dann ausgeführt wird, wenn er sich innerhalb der rechteckigen Klammern befindet. Zum Beispiel:

```
2 * 2 = [2] * [2]
```

Dieser Text wird folgendermaßen gedruckt:

```
2 * 2 = 2 * 2
```

Innerhalb von Ausdrücken existieren Elemente die individuelle rechteckige Klammern erforderlich machen. Beispielweise bei der Referenzierung auf eine Systemvariable (siehe hierzu auch das Kapitel "Ausdrücke" für weitere Details). Das folgende Beispiel verdeutlicht dies:

```
Die nächste Seite: [[Seite] + 1]
```

Der Text beinhaltet einen Ausdruck [Seite] + 1. Seite ist eine Systemvariable, die die Nummer der aktuellen Berichtsseite wiedergibt. Diese hat ihre eigenen Klammern. Ungeachtet der "Text"-Objekteinstellungen muss diese in rechteckige Klammern gesetzt werden.

Streng genommen, hätten im obigen Beispiel, für die "Datum"-Systemvariable, zwei rechteckige Klammern verwendet werden müssen:

```
Heute ist [[Datum]]
```

Es ist jedoch in FastReport möglich ein unnötiges Klammerpaar wegzulassen, wenn der Ausdruck nur ein Element enthält.

Anzeige der Datenspalten

Die Datenspalten können folgendermaßen gedruckt werden:

```
[Datasource name.Column name]
```

Wie man sehen kann, werden auch hier rechteckige Klammern genutzt. Die Namen der Datenquelle und der Datenspalte werden durch Kommata getrennt. Zum Beispiel:

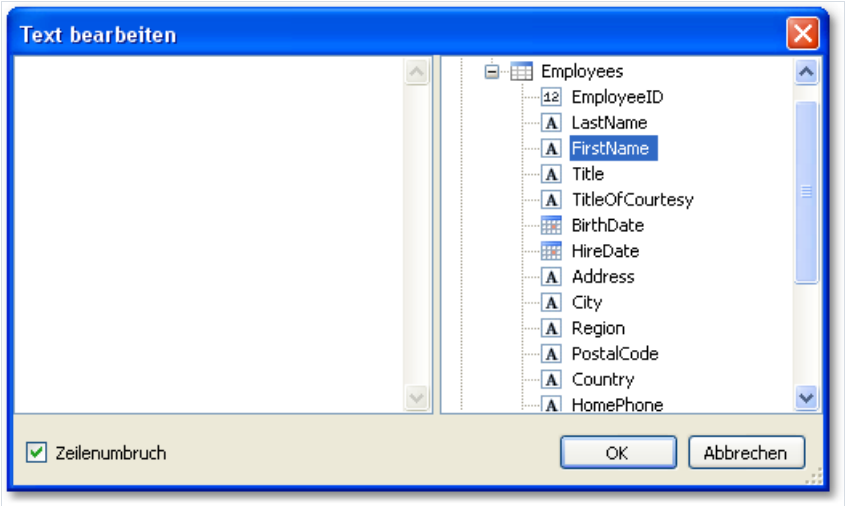
```
[Employees.FirstName]
```

Weitere Informationen zur Nutzung von Datenspalten werden im Kapitel "Ausdrücke" erläutert.

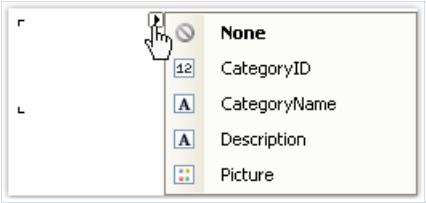
Eine Datenspalte kann auf mehrere Arten in ein "Text"-Objekt eingefügt werden:

1. Der Name der Datenspalte kann händisch im Editor des "Text"-Objekts eingegeben werden. Diese Methode ist jedoch am wenigsten geeignet, da diese sehr fehleranfällig ist.

2. Die benötigte Datenspalte wird im Objekt Editor ausgewählt und mittels drag&drop in den Text gezogen:



3. Ein Klick auf die kleine Schaltfläche in der oberen rechten Ecke des Objektes wählt die Datenspalte aus einer Liste aus:



4. Mittels drag&drop wird die Datenspalte aus dem Fenster "Daten" auf die Berichtsseite gezogen. In diesem Fall wird das "Text"-Objekt erstellt und mit der Spalte verlinkt.

HTML-Tags

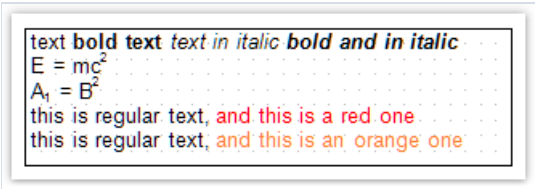
Im "Text"-Objekt können einfache HTML-Tags genutzt werden. Tags sind standardmäßig deaktiviert. Zur Aktivierung setzen Sie die Eigenschaft der "HTML-Tags" im Fenster "Eigenschaften" auf true. Hier eine Liste der unterstützten Tags:

Tag	Beschreibung
...	Fettdruck
...	Kursivdruck
<u>...</u>	Unterstrichener Text
<strike>... </strike>	Durchgestrichener Text
_{...}	Tiefgestellt
^{...}	Hochgestellt
**... **	Schriftfarbe Bei der Farbe kann es sich um eine benannte Farbe (DunkelGrau) oder einen hexadezimal Code im #RGB-Format handeln, zum Beispiel #FF8030.

Im Folgenden wird erläutert wie diese Tags genutzt werden können.

```
text <b>fettgedruckter </b> <i>kursiver text</i> <b><i>fett und kursiv</b></i>
E = mc<sup>2</sup>
A<sub>1</sub> = B<sup>2</sup>
Dies ist Standardtext, <font color=red>und dies ist Text in rot</font>
Dies ist Standardtext, <font color=#FF8030>und dies ist orangefarbener Text</font>
```

Der Text wird wie folgt angezeigt:



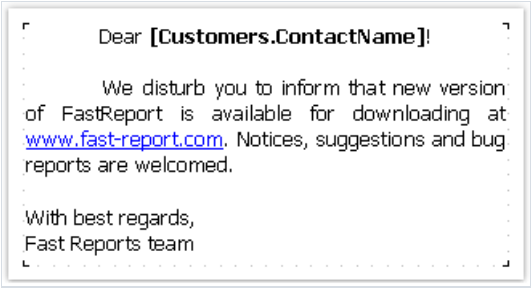
Objekteigenschaften

Eigenschaft	Beschreibung	
AllowExpressions	Diese Eigenschaft erlaubt eine Aktivierung/Deaktivierung von Ausdrücken. Standardmäßig ist diese aktiviert.	
Winkel	Diese Eigenschaft zeigt die Textdrehung in Grad an.	
AutoShrink	Diese Eigenschaft ermöglicht die automatische Textanpassung mittels Verkleinerung der Schriftgröße oder Schriftbreite.	
AutoShrinkMinSize	Diese Eigenschaft legt bei der Nutzung der AutoVerkleinern Eigenschaft, die Mindestgröße einer Schriftart, oder das Mindestverhältnis der Schriftbreite, fest.	
AutoWidth	Diese Eigenschaft erlaubt die automatische Berechnung der Breite eines Objektes.	
Klammern	Diese Eigenschaft beinhaltet ein Symbolpaar das einen Ausdruck bestimmt.	
BreakTo	Mit Hilfe dieser Eigenschaft kann der Textfluss von einem Objekt zum nächsten organisiert werden. Beispielweise haben wir ein Textobjekt "A" und ein Textobjekt "B". Das Objekt "A" enthält einen langen Text, der nicht in die Begrenzungen des Objektes passt. A.BreakTo B kann so festgelegt werden, dass das Objekt "B" den Teil des Textes anzeigt, der nicht in A passt.	
Clip	Diese Eigenschaft bestimmt ob es notwendig ist einen Text außerhalb der Begrenzungen des Objektes anzuheften. Standardmäßig ist diese Eigenschaft aktiviert.	
Duplicates	Diese Eigenschaft bestimmt den Druck wiederholter Werte. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Formatierung" erläutert.	
FirstTabOffset	Diese Eigenschaft bestimmt das Offset des ersten Tabstoppsymbols (in Pixel).	
FontWidthRatio	Diese Eigenschaft verbreitert oder verengt die Schriftart. Standardmäßig ist die Eigenschaft auf 1 gesetzt. Um die Schriftart zu verbreitern, wird die Eigenschaft auf einen Wert > 1 eingestellt. Um die Schriftart zu verengen, wird die Eigenschaft auf einen Wert zwischen 0 und 1 eingestellt.	
HideValue	Diese Eigenschaft ein Zeichenfolgetyp. Es erlaubt Werte, die dem Wert der Eigenschaft ähnlich sind, zu verbergen. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Formatierung" erläutert.	
HideZeros	Diese Eigenschaft ermöglicht das Verbergen von Nullwerten. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Formatierung" erläutert.	
Highlight	Diese Eigenschaft ermöglicht die Festlegung der bedingten Markierung. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Formatierung" erläutert.	
HorzAlign, VertAlign	Diese Eigenschaften legen die Textausrichtung fest.	

Eigenschaft	Beschreibung	
HtmlTags	Ermöglicht das Einfügen einfacher Html-Tags in den Text. Weitere Informationen zu dieser Eigenschaft werden im Kapitel "HTML-Tags" erläutert.	
LineHeight	Diese Eigenschaft erlaubt die explizite Festlegung der Höhe einer Textzeile. Diese ist standardmäßig auf 0 eingestellt, so dass der Standard-Zeilenabstand genutzt wird.	
NullValue	Der Text, der anstelle eines Nullwertes, gedruckt wird. Hierzu muss auch die Option "NULL-Values konvertieren" im Menü "Bericht	Optionen..." deaktiviert werden.
Padding	Diese Eigenschaft erlaubt die Einstellung des Abstandes (in Pixel).	
RightToLeft	Diese Eigenschaft legt fest, das der Text in rechter-zu-linker Leserichtung angezeigt wird.	
TabWidth	Diese Eigenschaft legt die Breite des Tabstoppsymbols (in Pixel) fest.	
Text	Diese Eigenschaft beinhaltet den Text des Objektes.	
TextFill	Diese Eigenschaft legt die Textfüllung fest. Der Editor dieser Eigenschaft bietet eine Auswahl verschiedener Füllarten.	
Trimming	Diese Eigenschaft legt fest, wie der Text, der nicht in die Begrenzungen des Objektes passt, getrimmt wird. Diese wird nur genutzt, wenn die Eigenschaft "Zeilenumbrüche" auf false eingestellt ist.	
Underlines	Diese Eigenschaft ermöglicht die Anzeige einer graphischen Linie nach jeder Textzeile. Diese Eigenschaft kann nur genutzt werden, wenn der Text nach oben ausgerichtet ist.	
WordWrap	Diese Eigenschaft bestimmt ob es notwendig ist einen Text nach Worten umzubrechen.	
Wysiwyg	Diese Eigenschaft ändert den Anzeigemodus des "Text"-Objektes: Bildschirmpassung und Druckansicht. Dieser Modus wird genutzt, wenn der Blocksatz oder die nicht-normierte Zeilenhöhe genutzt wird.	

Das "Rich Text"-Objekt

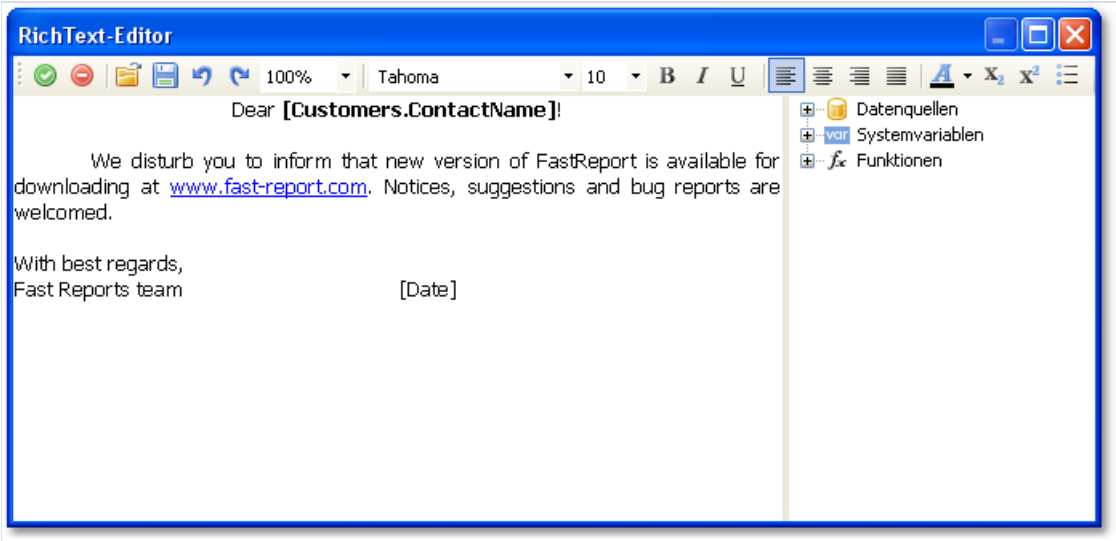
Dieses Objekt zeigt den formatierten Text im RTF-Format an. Dieser sieht folgendermaßen aus:



Soweit möglich, wird das "Text"-Objekt zur Anzeige eines Textes genutzt. Beim Exportieren des Berichts in einige Dokumentenformate, wird das "Rich Text"-Objekt als Bild exportiert.

Dieses Objekt unterstützt lediglich einfarbige (solide) Füllarten. Verlaufs-, Schraffur-, und Glasfüllungen werden nicht unterstützt.

Zum Bearbeiten des Objektes wird dieses doppelgeklickt. Das Editorfenster wird angezeigt:



Microsoft Word kann auch zur Texterstellung genutzt werden. Der Text muss dann im RTF-Format gespeichert werden. Als nächstes muss der Rich Text Editor geöffnet und die RFT-Datei, durch Drücken der Schaltfläche in den Editor geladen werden.

Das "Rich Text"-Objekt unterstützt nicht alle Microsoft Word Formatierungseigenschaften.

Daten können in diesem Objekt folgendermaßen angezeigt werden:

· Einfügen eines Ausdrucks in den Objekttext - gerade so wie im "Text"-Objekt. Einfügen der notwendigen Datenspalte in den Text;

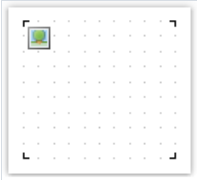
· Nutzung der Eigenschaft "DataColumn" um das Objekt an die Spalte zu binden.

Das Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
AllowExpressions	Diese Eigenschaft erlaubt eine Aktivierung/Deaktivierung von Ausdrücken. Standardmäßig ist diese aktiviert.
Brackets	Diese Eigenschaft beinhaltet ein Symbolpaar, dass einen Ausdruck bestimmt.
DataColumn	Die Datenspalte an die das Objekt gebunden ist.
Text	Diese Eigenschaft beinhaltet den RTF-Text.
Padding	Der Abstand (in Pixel).

Das "Bild"-Objekt

Das Objekt kann Graphiken der folgenden Formate anzeigen: BMP, PNG, JPG, GIF, TIFF, ICO, EMF, WMF. Das "Bild"-Objekt unterstützt den Druck eines Geschäftslogos, das Foto eines Mitarbeiters oder sonstige graphische Informationen. Das Objekt sieht folgendermaßen aus:

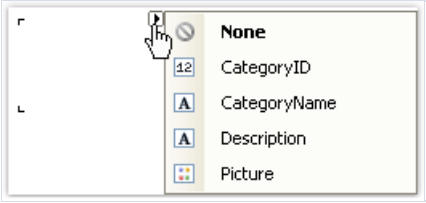


Ein Objekt kann Daten aus den folgenden Quellen anzeigen:

Quelle	Beschreibung
Datei mit Bild	Das Bild wird aus einer Datei geladen und ist im Bericht gespeichert. Das Bild wird in der Eigenschaft "Bild" gespeichert.
Datenspalte	Bild aus einer Datenspalte. Der Name der Spalte ist in der Eigenschaft "DataColumn" gespeichert.
Dateiname	Das Bild wird aus einer Datei mit dem angegebenen Namen geladen. Der Dateiname ist in der Eigenschaft "Bild" gespeichert. Das Bild wird nie im Bericht gespeichert. Man sollte die Bilddatei zusammen mit dem Bericht verteilen.
URL	Das Bild wird für jede Berichterstellung aus dem Internet geladen. Das Bild wird nie im Bericht gespeichert. Die URL ist in der Eigenschaft "ImageLocation" gepeichert.

Um einen Bildeditor aufzurufen muss das Objekt doppelgeklickt werden. Der Editor ermöglicht die Auswahl de Datenquelle für das Bild:

Um das Objekt mit einer Datenspalte zu verbinden, klickt man auf die schmale Schaltfläche in der oberen rechten Ecke des Objektes und wählt die Datenspalte aus einer Liste aus:



Man kann eine Datenspalte aus dem "Daten"-Fenster auf die Berichtsseite ziehen und diese am gewünschten Ort fallen lassen. In diesem Fall wird ein "Bild"-Objekt erstellt das mit der Spalte verlinkt ist. Die gezogene Spalte sollte dem "byte[]" Datentyp entsprechen.

Der Größenmodus kann im Kontextmenü des "Bild"-Objektes ausgewählt werden:

· AutoSize. Das Objekt entspricht der Größe des Bildes.

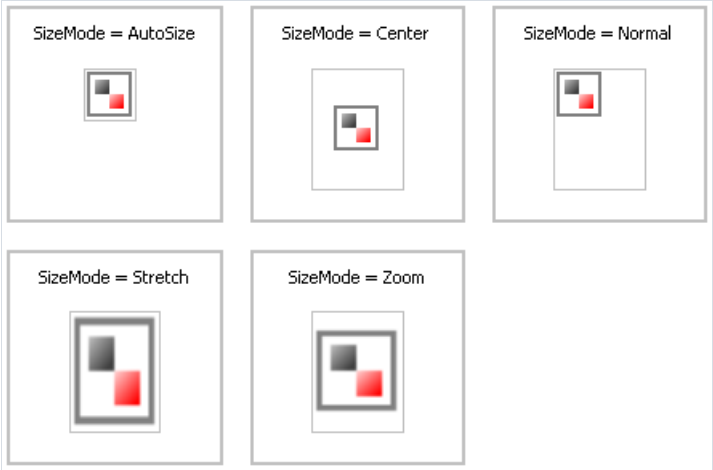
· CenterImage. Das Bild wird im Objekt zentriert.

· Normal. Das Bild wird in der linken Ecke des Objektes angezeigt.

· StretchImage. Das Bild wird auf die Größe des Objektes gestreckt.

· Zoom. Das Bild wird, entsprechend des Seitenverhältnisses, auf die Größe des Objektes gestreckt.

Die Unterschiede zwischen den Modi sind in der folgenden Abbildung dargestellt:



Das "Bild"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
Angle	Der Drehwinkel (in Grad). Diese Eigenschaft kann die Werte 0, 90, 180 und 270 annehmen.
SizeMode	Der Größenmodus.
Transparency	Der Transparenzgrad der Bilder. Die Eigenschaft kann Werte zwischen 0 und 1 annehmen. Der Standardwert 0 bedeutet dass das Bild nicht transparent ist.
TransparentColor	Die Farbe die, bei Anzeige des Bildes, transparent ist.
Image	Das Bild.
DataColumn	Die Datenspalte an die das Objekt gebunden ist.

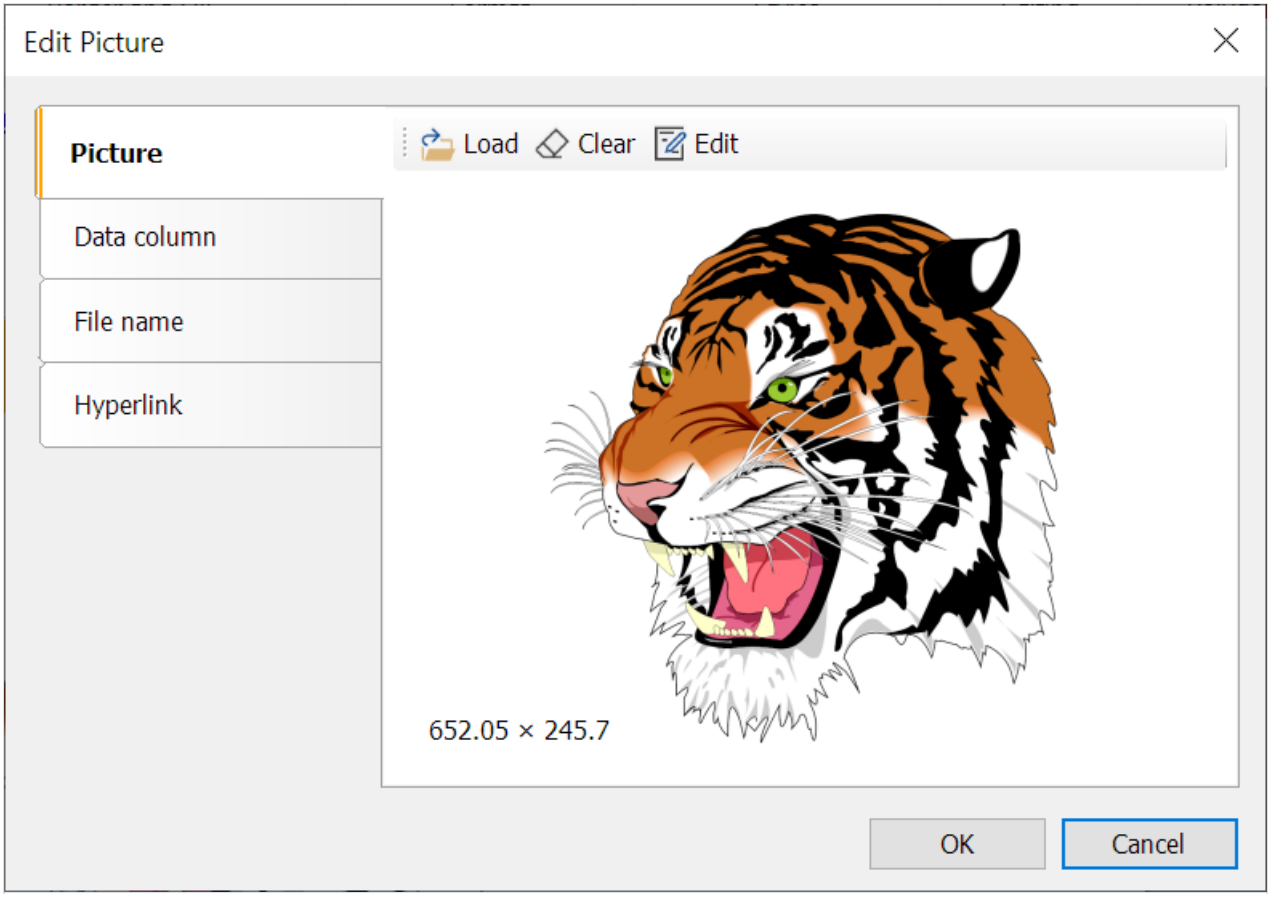
Eigenschaft	Beschreibung
ImageLocation	Die Eigenschaft kann den Dateinamen oder die URL beinhalten. Bei der Erstellung des Berichts wird das Bild aus dem Ort hochgeladen.
Padding	Der Abstand (in Pixel).
ShowErrorImage	Zeigt das "Ohne-Bild"-Bild an, für den Fall, dass das Bild leer gelassen wird. Es macht Sinn diese Eigenschaft zu nutzen, wenn das Bild aus dem Internet heruntergeladen wird.

"SVG"-Objekt

Dieses Objekt dient der Anzeige von Vektorbildern im SVG-Format. Unten ist ein Beispiel für ein solches Bild:

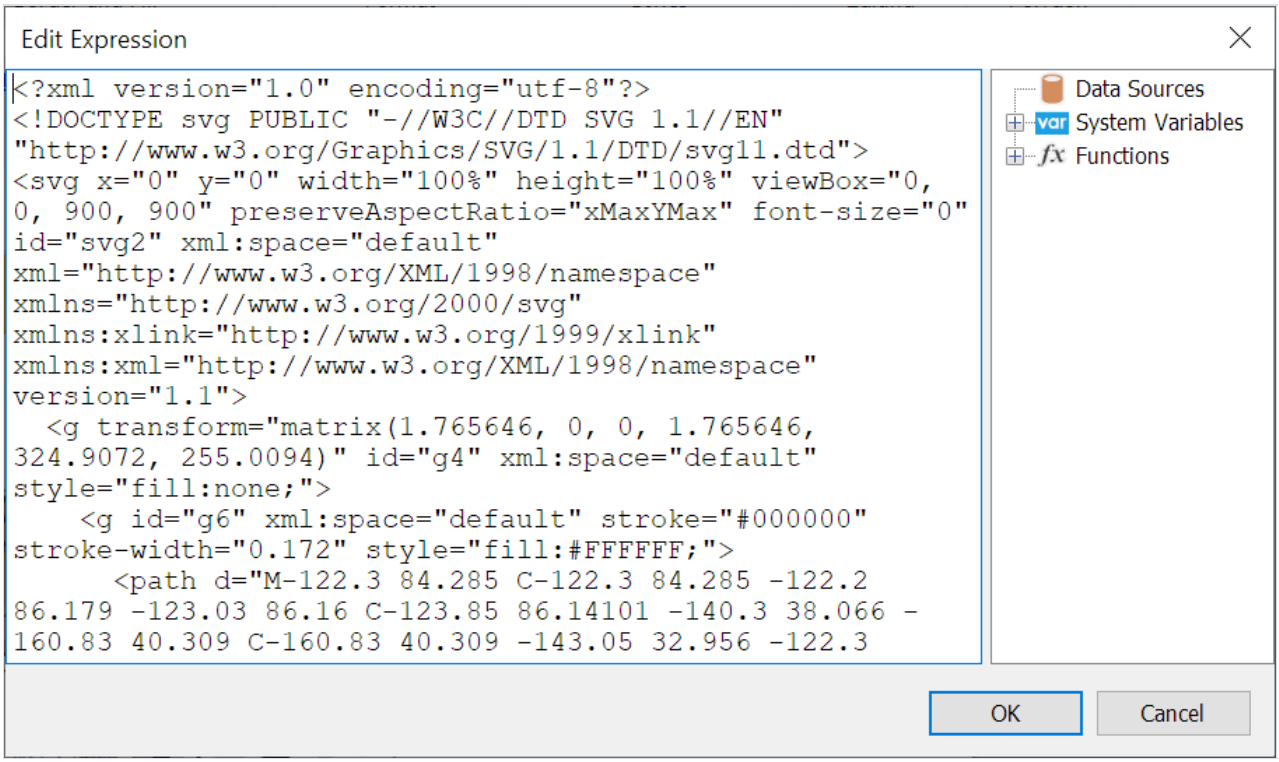
Um ein SVG-Bild hinzuzufügen, fügen Sie ein "SVG"-Objekt zu Ihrem Bericht hinzu, indem Sie die folgende Schaltfläche in der Objektleiste betätigen:

Doppelklicken Sie dann auf das Objekt, oder wählen Sie "Bearbeiten" aus dem Kontextmenü. Dadurch wird der Bildeditor geöffnet, der dem Objekteditor "Bild" ähnelt.



In diesem Editor können Sie das Bild festlegen, das als Objekt angezeigt wird. Sie können das Bild in die Berichtsvorlage einfügen (Schaltfläche "Öffnen" auf der Registerkarte "Bild"), ein SVG-Bild aus der Datentabelle auswählen (Registerkarte "Datenfeld"), eine Verbindung zu einer externen SVG-Datei herstellen (Registerkarte "Dateiname"; in diesem Fall müssen Sie diese Datei zusammen mit der Vorlage verteilen) oder einen Hyperlink definieren, über den das Bild abgerufen wird (Registerkarte "Hyperlink").

Wenn Sie ein Bild in die Berichtsvorlage einfügen, wird seine String-Darstellung in der SVGString-Eigenschaft gespeichert. Der Screenshot zeigt ein Fragment eines solchen Strings:

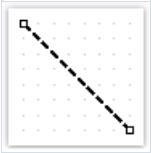


Das SVG-Objekt unterstützt die gleichen Größenmodi wie das "Bild"-Objekt: Automatische Größe (AutoSize), Zentrieren (CenterImage), Normal, Strecken (StretchImage), Zoomen (Zoom).

Eine detaillierte Beschreibung dieser Modi finden Sie in der Beschreibung des Objekts "Bild".

Das "Zeilen"-Objekt

Das "Zeilen"-Objekt kann horizontale, vertikale oder diagonale Zeilen anzeigen. Das Objekt ist:



Anstelle des "Zeilen"-Objektes sollte, wenn möglich, die Ränder des Objektes genutzt werden. Dies vereinfacht den Bericht und vermeidet potentielle Schwierigkeiten beim Exportieren in andere Formate.

Der FastReport-Designer bietet einfach zu bedienende Werkzeuge zur Zeichnung von Linien. Zum Einfügen einer Linie in einen Bericht klickt man die Schaltfläche auf der "Objekt"-Symbolleiste und wählt das "Zeilen"-Objekt oder "Diagonale Zeile". Der Mauszeiger wird an der Stelle positioniert an der die Linie beginnen soll. Durch einen linken Mausklick und dem Halten der Maus wird eine Linie gezogen. Hiernach kann die Linie erneut gezogen werden. Sind alle Linien erstellt, klickt man die Schaltfläche auf der "Objekt"-Symbolleiste.

Eine reguläre Linie kann nur vertikal oder horizontal verlaufen und unterscheidet sich hiermit von einer diagonalen Linie.

Für dieses Objekt sollte nicht der Stil "Doppellinie" gewählt werden. Dieser Stil wird nur auf den Rahmen des Objektes angewandt.

Das "Zeilen"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
Diagonal	Die Eigenschaft legt fest ob die Linie diagonal verläuft. Durch die Aktivierung dieser Eigenschaft kann eine normale Linie in eine diagonale Linie geändert werden.
StartCap, EndCap	Diese Eigenschaften erlaubt das Festlegen des Linienendes. Es stehen die folgenden Linienenden zur Verfügung:· Ellipse;· Rechteck;· Diamant;· Pfeil.Die Größe des Linienendes kann in den Breite-, und Höheneigenschaften des Linienendes festgelegt werden. Beide Linienenden können konfiguriert werden.

Das "Form"-Objekt

Das "Form"-Objekt kann folgende Formen annehmen:

· Rechteck;

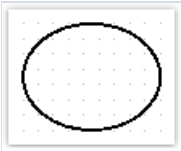
· Abgerundetes Rechteck;

· Ellipse;

· Dreieck;

· Diamant;

Das Objekt sieht folgendermaßen aus:



Zum Einfügen einer Form in einen Bericht klickt man die Schaltfläche auf der "Objekt"-Symbolleiste und wählt die benötigte Form aus.

Die Form, wie jedes andere Berichtsobjekt auch, hat eine Füllung und einen Rahmen. Im Gegensatz zum "Text"-Objekt kann nicht jede Rahmenlinie gesteuert werden. Von der Nutzung des "Doppel"-Linienstils wird abgeraten.

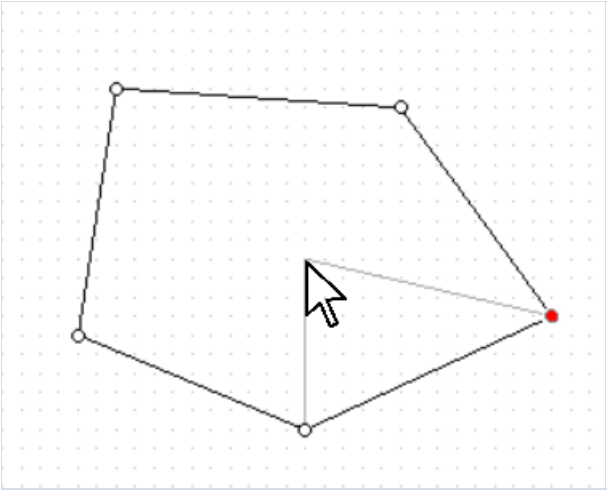
Anstelle der rechteckigen Form sollte der Rahmen des Objektes genutzt werden.

Das "Form"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
Shape	Diese Eigenschaft legt die Art der Form fest.
Curve	Diese Eigenschaft wird in Kombination mit der Eigenschaft "RoundRectangle" genutzt. Es erlaubt die Festlegung der Krümmung.

Objekte "Polylinie" und "Polygon"

Diese Objekte werden zur Darstellung einer Vielzahl von Figuren, wie z. B. Polygonen, verwendet. Sie gehören zur gleichen Kategorie wie die Objekte "Figur" oder "Linie". Um ein "Polygon"-Objekt zu platzieren, öffnen Sie das Menü unter der Schaltfläche und wählen Sie eines der Polygone aus. Sie können ein vorgefertigtes Fünf-, Sechs-, Sieben- oder Achteck platzieren. Sie können auch "Polygon" auswählen, um die Figur selbst zu zeichnen. Ein Beispiel dafür, wie ein Polygon aussehen könnte:

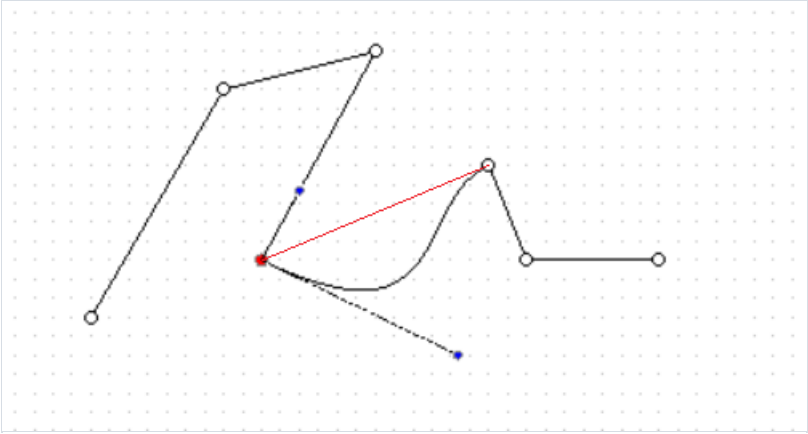


Die grauen Linien in der Abbildung veranschaulichen, wie das Objekt aussehen wird, wenn ein neuer Punkt hinzugefügt wird.

Nachdem Sie den ersten Punkt platziert haben, können Sie weitere hinzufügen. Wenn Sie mit dem Zeichnen des Polygons fertig sind, drücken Sie Escape oder wechseln Sie in den Bearbeitungsmodus. Das Bearbeitungsmodus-Bedienfeld für "Polylinie" und "Polygon" befindet sich auf der Registerkarte "Start".

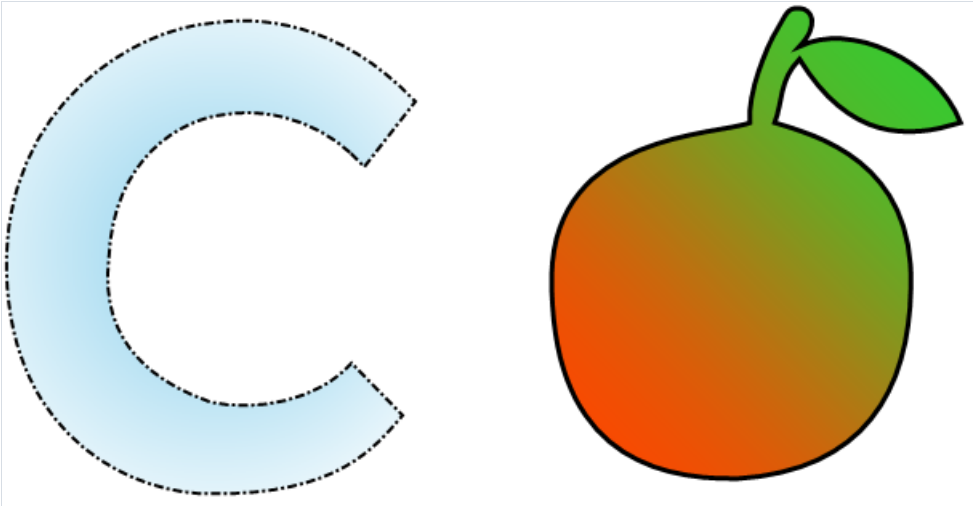
Bearbeitungsmodi für die Polylinie und das Polygon:

Schaltfläche	Modus	Beschreibung
	Hinzufügen von Punkten	Dieser Modus wird sofort nach dem Hinzufügen eines Polygons aktiviert. Um einen Punkt hinzuzufügen, wählen Sie diesen Modus und wählen Sie einen der Punkte auf dem Objekt. Bewegen Sie dann den Cursor über die gewünschte Stelle für den Punkt und klicken Sie mit der linken Maustaste. Der Punkt wird neben dem von Ihnen ausgewählten Punkt hinzugefügt.
	Verschieben und Größenänderung	In diesem Modus können Sie das gesamte Objekt verschieben oder es verkleinern oder strecken.
	Entfernen von Punkten	Um einen Punkt aus einem Polygon oder einer Polylinie zu entfernen, wählen Sie diesen Modus und klicken Sie auf den gewünschten Punkt. Alternativ können Sie ihn auch in einem anderen Modus auswählen und die Entf-Taste drücken.
	Mauszeiger	In diesem Modus können Sie alle zuvor hinzugefügten Objektpunkte anzeigen und verschieben.
	Referenzpunkt auf der Kurve	In diesem Modus können Sie der Linie neben dem ausgewählten Punkt einen Krümmungspunkt hinzufügen. Diese Linie wird zu einer Bézierkurve, und der hinzugefügte Punkt dient als Referenzpunkt.



So sieht eine Polylinie aus, wenn der Krümmungspunkt gesetzt ist (blau). Die rote Linie zeigt den Zustand der Linie, bevor sie in eine Kurve umgewandelt wurde.

Beide Objekte haben die Möglichkeit, ihre Grenzen anzupassen, und das Polygon-Objekt unterstützt die gleichen Füllmodi wie die anderen Objekte.



Das "Barcode"-Objekt

Dieses Objekt zeigt die Barcodes im Bericht an. Dieses sieht folgendermaßen aus:

Das Objekt unterstützt die folgenden Arten von Barcodes:

Code	Länge	Erlaubte Symbole
2 von 5 überlappend		0-9
2 von 5 Industrial		0-9
2 of 5 Matrix		0-9
Codabar		0-9, -$:/.+
Code128		128 ASCII Zeichen
Code39		0-9,A-Z, -. *$/+%
Code39 erweitert		128 ASCII Zeichen
Code93		0-9,A-Z, -. *$/+%
Code93 erweitert		128 ASCII Zeichen
EAN8	8	0-9
EAN13	13	0-9
MSI		0-9
PostNet		0-9
UPC A	12	0-9
UPC E0	6	0-9
UPC E1	6	0-9
2-Zahlergänzung	2	0-9
5-Zahlergänzung	5	0-9

Code	Länge	Erlaubte Symbole
PDF417		beliebig
Datenmatrix		beliebig

Eine detaillierte Beschreibung unterschiedlicher Barcodetypen wird im Internet erläutert:

http://www.barcodeisland.com

Die Barcodedaten in einem Objekt sind in der Art einer Zeichenfolge. Die Zeichenfolge kann beliebige Symbole, die für den Barcodetyp erlaubt sind, beinhalten. Man kann den Barcodetyp im Kontextmenü des "Barcode"-Objektes auswählen.

Man kann die Objekte mit den Daten, durch eine der folgenden Methoden, verbinden:

· Festlegen der Barcodedaten in den "Text"-Eigenschaften;

· Anbinden des Objektes an die Datenspalte mit Hilfe der Eigenschaft "Datenspalte";

· Festlegung der Ausdrücke die die Barcodedaten in die Eigenschaft "Ausdrücke" zurückführt.

Das "Barcode"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
Barcode	Diese Eigenschaft umfasst barcodespezifische Einstellungen. Eine Erweiterung der Eigenschaften erlaubt das Festlegen dieser Einstellungen.
Angle	Diese Eigenschaft legt die Drehung des Barcodes (in Grad) fest. Es stehen die folgenden Werte zur Verfügung: 0, 90, 180, 270.
Zoom	Diese Eigenschaft ermöglicht das Zoomen des Barcodes. Diese wird zusammen mit der Eigenschaft "AutoGröße" genutzt.
AutoSize	Ist die Eigenschaft aktiviert, streckt sich das Objekt um den gesamten Barcode anzuzeigen. Ist diese Eigenschaft deaktiviert erstreckt sich der Barcode an die Ränder des Objektes.
ShowText	Diese Eigenschaft bestimmt die Notwendigkeit der Anzeige visuell lesbaren Textes.
DataColumn	Die Datenspalte die mit diesem Objekt gebunden ist.
Expression	Der Ausdruck der die Barcodedaten zurückgibt.
Text	Das "Barcode"-Objekt
Padding	Der Abstand (in Pixel).

Die folgenden Eigenschaften sind spezifisch für den Barcodetypen. Zu deren Änderung wird der Barcode ausgewählt, das Fenster "Eigenschaften" aufgerufen und die "Barcode"-Eigenschaften erweitert.

Eigenschaft	Beschreibung

Eigenschaft	Beschreibung
WideBarRatio	Diese Eigenschaft betrifft alle linearen Barcodes. Es legt das breit-zu-schmal Verhältnis der Liste fest. Für die meisten Barcodes sollte diese Eigenschaft einen Wert zwischen 2 und 3 annehmen.
CalcCheckSum	Diese Eigenschaft betrifft alle linearen Barcodes. Es ermittelt die Notwendigkeit der automatischen Berechnung der Prüfsumme. Sollte diese Eigenschaft deaktiviert sein, muss die Prüfziffern in den Barcodedaten bereitgestellt werden.
AutoEncode	Diese Eigenschaft betrifft alle Code128 Barcodes. Dieser Code hat drei verschiedene Kodierungen: A, B, C. Die Kodierung sollte explizit im Barcode festgelegt, oder diese Eigenschaft auf wahr eingestellt werden. In diesem Fall wird die Kodierung automatisch ausgewählt. Nutzung der folgenden Kontrollcodes für die Barcodedaten: &A; START A / CODE A &B; START B / CODE B &C; START C / CODE C &S; SHIFT &1; FNC1 &2; FNC2 &3; FNC3 &4; FNC4 Ist die Eigenschaft "AutoEncode" auf wahr gesetzt, werden alle Kontrollcodes ignoriert. Beispiel der Nutzung der Kontrollcodes: &C;1234&B;ABC
AspectRatio	Diese Eigenschaft betrifft alle PDF417 Barcodes. Diese bestimmt das Höhe-zu-Breite Seitenverhältnis und wird zur automatischen Berechnung der Barcodegröße bestimmt (für den Fall, dass die "Spalten" und "Zeilen" Eigenschaften nicht festgelegt sind).
CodePage	Diese Eigenschaft betrifft alle PDF417 und Datenmatrix-Barcodes. Es legt die Codepage fest, die zur Konvertierung von ASCII-Zeichen genutzt wird. Die Standard Windows-Codepage lautet beispielsweise: 1251.
Columns, Rows	Diese Eigenschaft betrifft alle PDF417 Barcodes. Diese bestimmt die Anzahl der Spalten und Zeilen des Barcodes. Nehmen beide Eigenschaften den Wert 0 an, wird die Größe des Barcodes automatisch berechnet. In diesem Fall wird auch die Eigenschaft "Seitenverhältnis" genutzt.
CompactionMode	Diese Eigenschaft betrifft alle PDF417 Barcodes. Es bestimmt den Compaction-Modus der PDF417-Daten.
Fehlerkorrektur	Diese Eigenschaft betrifft alle PDF417 Barcodes. Diese legt das Niveau der Fehlerkorrektur fest.
PixelSize	Diese Eigenschaft betrifft alle PDF417 Barcodes. Sie bestimmt die Größe des Barcodeelements in Bildschirmpixel. Allgemein gilt, dass die Höhe des Elements mindestens dreimal so groß sein sollte, wie die Breite des Elements.
Codierung	Diese Eigenschaft betrifft alle Datenmatrix-Barcodes. Es bestimmt die Datencodierung der Datenmatrix.
PixelSize	Diese Eigenschaft betrifft alle Datenmatrix-Barcodes. Sie bestimmt die Größe des Barcodeelements (in Pixel).
SymbolSize	Diese Eigenschaft betrifft alle Datenmatrix-Barcodes. Diese bestimmt die Größe des Barcodesymbols.

Industrial 2 of 5

Industrial 2 of 5 (auch bekannt als Standard 2 of 5) ist ein linearer Barcode mit geringer Dichte zur Codierung von Zahlen beliebiger Länge. Für die Codierung werden nur Striche unterschiedlicher Dicke verwendet, wobei Zwischenräume nur als Trennzeichen zwischen ihnen dienen. Dies führt zu einem Barcode mit geringer Dichte und somit zu einer großen Barcodegröße.

Ein Zeichen wird durch 5 Strichen codiert, von denen zwei dicker sind. Dieses Merkmal gab dem Code seinen Namen.

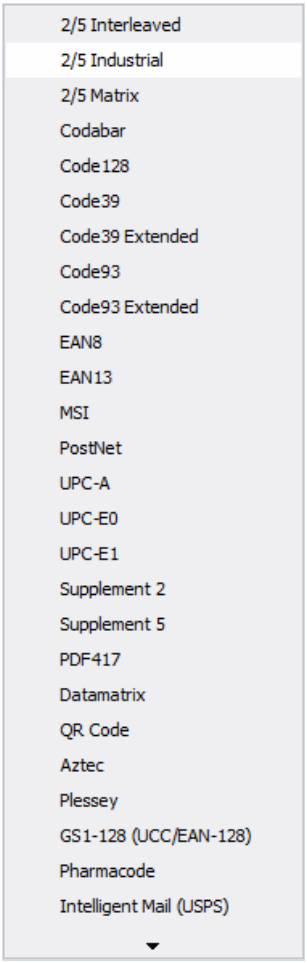
Dieser Barcode gilt derzeit als veraltet. Er war ein häufig verwendeter Code in der Lagertechnik, in Fotolabors und zur Nummerierung von Flugtickets.

Industrial 2/5 verfügt über eine Prüffziffer, die nach Modulo 10 berechnet wird, sie ist aber optional.

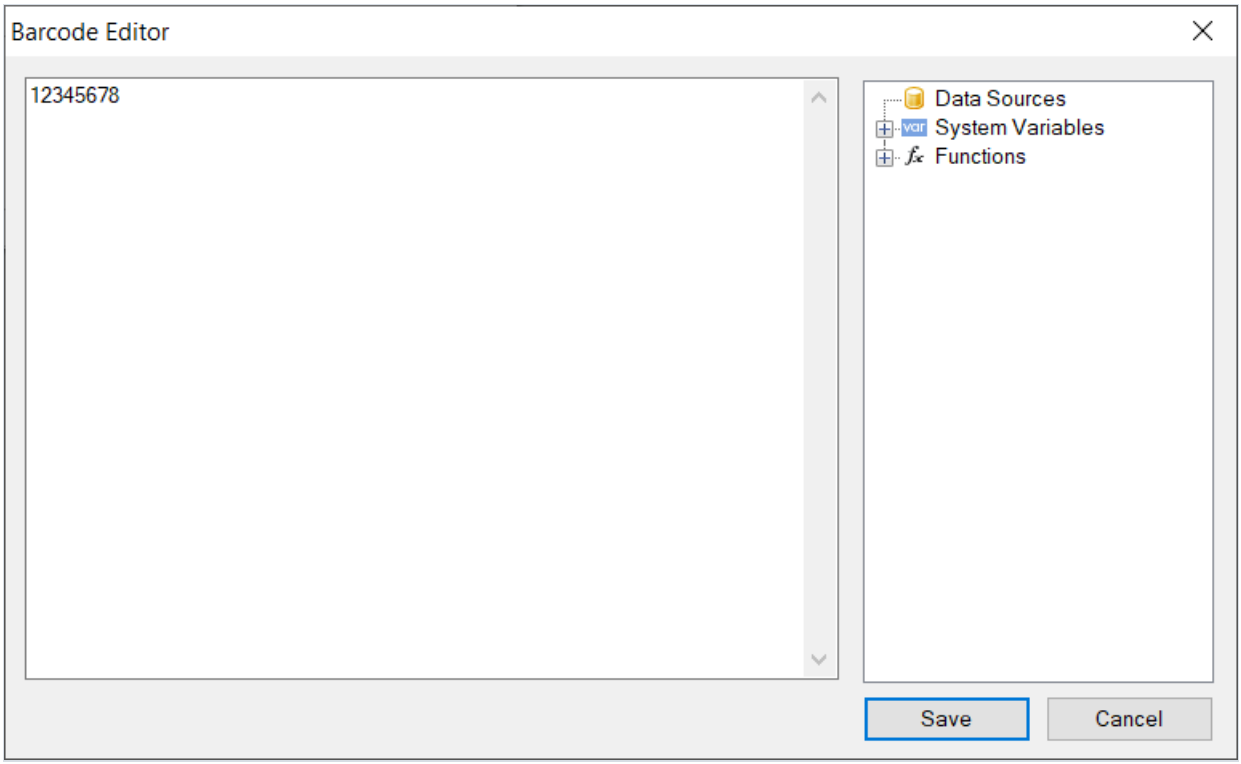
Der Standard 2 of 5 Barcode wird wie folgt aufgebaut:

- Start-Zeichen;
- Datensymbole;
- Optionales Prüfzeichen;
- Stopp-Zeichen.

Um der Berichtsseite einen Industrial 2 of 5 Barcode hinzuzufügen, wählen Sie den Barcode in der seitlichen Symbolleiste des Berichtsdesigners aus. Wählen Sie im Dropdown-Menü „2/5 Industrial“ aus und platzieren Sie das Objekt auf der Berichtsseite:



Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Barcode-Objekt. Oder klicken Sie mit der rechten Maustaste auf das Barcode-Objekt und wählen Sie Bearbeiten (Edit) im Kontextmenü aus.



Der Code kann auch ohne Text angezeigt werden. Dazu setzen Sie die ShowText Eigenschaft des Barcode-Objekts auf False:

2/5 Interleaved

Der Interleaved 2 of 5 (ITF 2/5) Barcode ist ein klassischer linearer Code zur Codierung von numerischen Informationen beliebiger Länge. Es wurde 1972 von David Alley entwickelt. Die Besonderheit dieses Barcodes besteht darin, dass eine gerade Anzahl von Zeichen codiert werden muss. Der Grund dafür ist, dass die Zeichen paarweise codiert werden. Wenn ein Code eine ungerade Anzahl von Ziffern hat, wird dem Barcode eine Null vorangestellt. Das erste Zeichen des Paares wird mit Strichen und das zweite mit Zwischenräumen codiert. Die Striche und Zwischenräume sind entweder schmal oder breit. Ein Zeichen wird mit fünf Strichen oder Zwischenräumen codiert. Zwei Striche sind breit und drei sind schmal. Das Gleiche gilt für Zwischenräume.

Dank dieser Codierungsmethode hat der Barcode eine hohe Dichte und nimmt weniger Platz ein als beispielsweise Code 39.

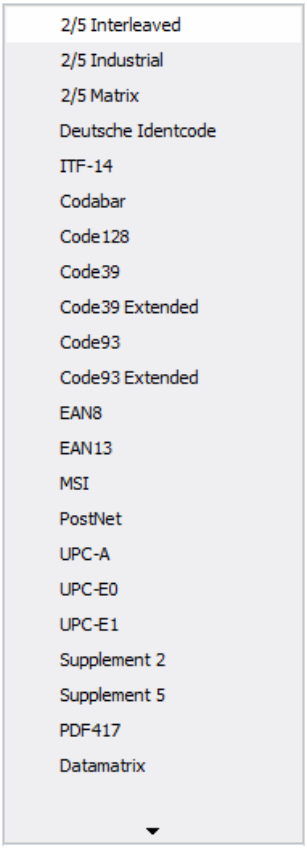
Der 2/5 Interleaved Barcode wird zur Kennzeichnung von Produkten im Lager sowie im Schiffsverkehr verwendet.

Aufbau des Codes:

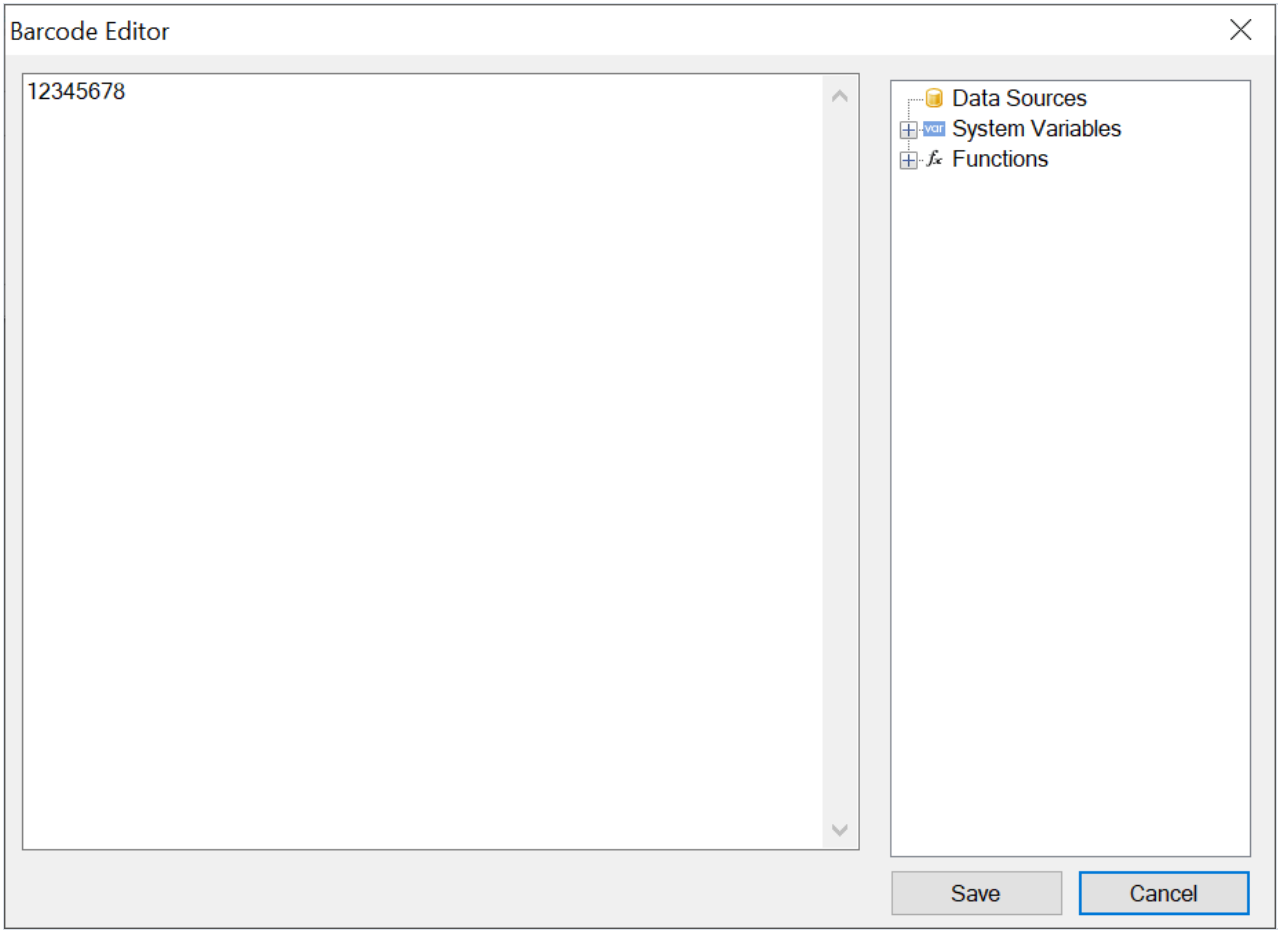
- ein Start-Zeichen zeigt den Beginn des Codes an;
- codierte Daten;
- optionale Prüfziffer;
- Stopp-Zeichen.

Der Barcode ist selbstüberwachend, daher ist eine Prüfziffer nicht erforderlich.

Um der Berichtsseite einen Interleaved 2 of 5 Barcode hinzuzufügen, klicken Sie mit dem Maus auf das Barcode-Symbol :



Platzieren Sie dann das Objekt auf der Berichtsseite. Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt. Alternativ kann der Code-Editor auch über das Kontextmenü des hinzugefügten Objekts aufgerufen werden, und zwar mit der Option Edit:



Um die Textanzeige unter dem Barcode zu deaktivieren, setzen Sie die ShowText-Eigenschaft im Eigenschafteninspektor des hinzugefügten Barcodes auf False:

2 of 5 Matrix

Es handelt sich um einen linearen Barcode mit hoher Dichte und beliebiger Länge zur Codierung numerischer Daten (0–9). Es wurde 1968 von der Fa. Identicon entwickelt. Die hohe Code-Dichte wird durch die Verwendung von Zwischenräumen zwischen den Strichen für die Codierung erreicht. Somit sind sowohl Striche als auch Zwischenräume an der Codierung beteiligt.

Der Code ist selbstüberwachend und kann in beiden Richtungen gelesen werden. Das heißt, der Code kann sowohl von links nach rechts als auch von rechts nach links gelesen werden.

Aufbau des Codes:

● Start-Zeichen;

● Daten;

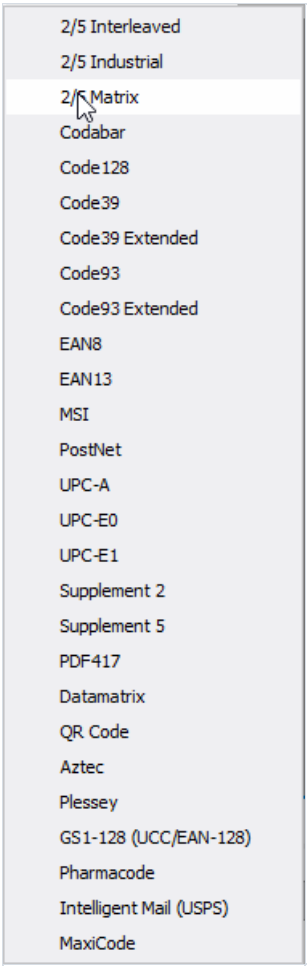
● Optionales Prüfzeichen wird nach Modulo 10 berechnet;

● Stopp-Zeichen.

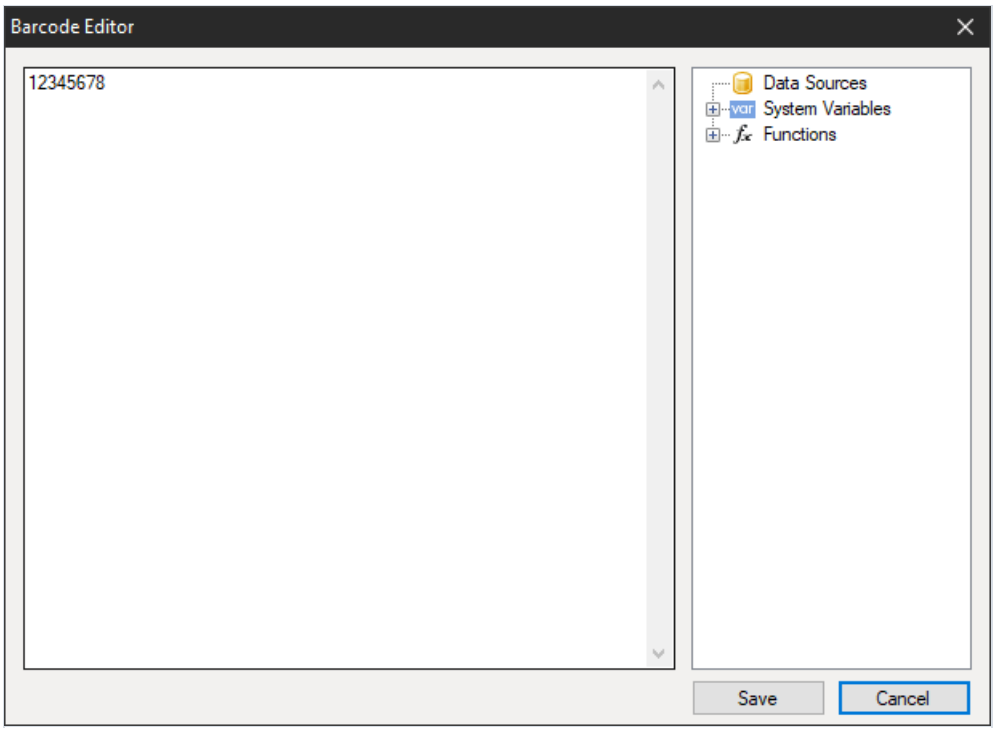
Das Symbol besteht aus 5 Strichen (Zwischenräumen), von denen zwei breit sind. Dies ist ein Symbol mit zwei Breiten. Das heißt, die Breite des schmalen Strichs (Zwischenraums) ist darin definiert, und alles, was breiter als dieser ist, wird als breiter Strich (Zwischenraum) betrachtet. Dadurch können Codes mit einer großen Toleranz auf minderwertigen Geräten gedruckt werden.

2 of 5 Matrix wird zur Kennzeichnung von Waren in Lagern, zur Kennzeichnung von Flugtickets, in Fotolabors verwendet.

Um der Berichtsseite einen 2 of 5 Matrix Barcode hinzuzufügen, wählen Sie das Element Barcode in der seitlichen Symbolleiste des Berichtsdesigners aus. Wählen Sie im Dropdown-Menü „2/5 Matrix“ aus und platzieren Sie das Objekt auf den Berichtsseiten:



Platzieren Sie den ausgewählten Code auf der Berichtsseite. Um ihn zu bearbeiten, doppelklicken Sie auf das Objekt, oder klicken Sie mit der rechten Maustaste auf das Objekt und wählen Sie Edit... aus.



Um die numerische Anzeige unter dem Code zu entfernen, wählen Sie das hinzugefügte Objekt aus. Setzen Sie die ShowText-Eigenschaft im Eigenschafteninspektor des Objektes auf False. Oder rufen Sie das Kontextmenü für das Objekt auf und deaktivieren Sie die Option ShowText:

ITF-14

Der ITF-14 ist ein Zahlencode, der aus zwei Balken besteht, auch bekannt als High-Density-Code, der nur Ziffer in gerader Anzahl codieren kann. Jeder dieser Barcodes codiert eine ungerade Ziffer mit einem dunklen Strich und eine gerade Ziffer mit einem Zwischenraum dazwischen. Um eine ungerade Anzahl von Ziffern zu codieren, müssen Sie die äußerste linke (höchste) Ziffer durch eine Null ergänzen.

Der ITF-14 wird folgendermaßen aufgebaut:

- Die erste Ziffer ist die Bezeichnung der Verpackungsebene für einen bestimmten Karton. Dieses eindeutige Präfix kann zwischen 0 und 8 liegen (z.B., 1 für Karton, 2 für Kiste usw.).
- Weitere 2-3 Ziffern stellen den Regionalcode (Präfix) des Landes dar, in dem diese Nummer registriert ist;
- Die nächsten 4-5 Ziffern sind die Registrierungsnummer des Unternehmens innerhalb der nationalen Organisation;
- die nächste Ziffergruppe bezeichnet die fortlaufende Nummer des Produkts innerhalb des Unternehmens;
- Die letzte Ziffer des ITF-14 ist die Prüfsumme bzw. -ziffer. Sie wird aus den vorherigen zwölf Ziffern nach dem Modulo 10 Algorithmus berechnet.

Der ITF-14-Barcode wird zur Automatisierung der Lagerbuchführung von Waren in Einzel- oder Sammeltransportverpackungen verwendet. Er wird vom EDV-Buchhaltungssystem verwendet, um nicht nur die Art der Waren in der Verpackung, sondern auch ihre Menge zu identifizieren.

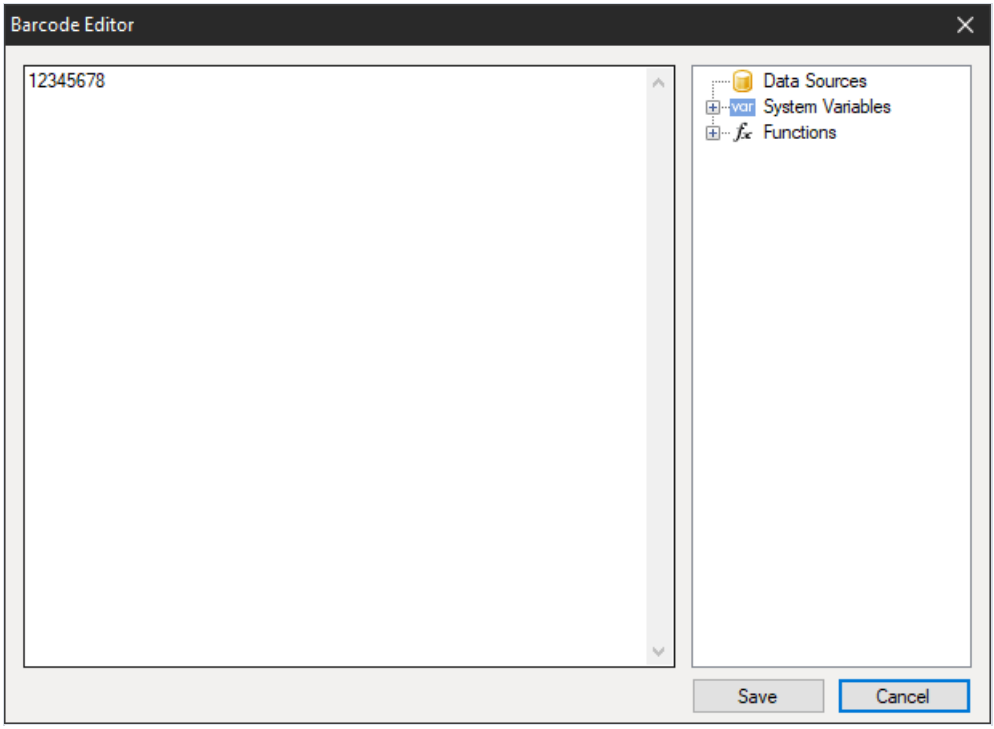
Der ITF-14 wird in der Regel auf Wellpappe gedruckt, zur Kennzeichnung von Kartons, Kisten oder Paletten verwendet. Er wird häufig von Einzelhändlern, Herstellern und Händlern für präzise Logistik und Lagerhaltung eingesetzt. Er kommt in Einsatz auch bei der Identifizierung von Gepäck auf Flughäfen, der Nummerierung von Flugtickets, der Identifizierung von Postsendungen.

Da der ITF-14-Barcode Waren in Transportverpackungen kennzeichnen soll, wird er nicht von POS-Terminals verarbeitet.

Der dicke feste Rahmen rundum den Barcode wird Bearer Bar genannt. Dieser Rahmen gleicht den Druckdruck über die Barcodeoberfläche aus und verbessert die Erfolgsrate beim Lesen, indem er die Wahrscheinlichkeit verringert, dass ein unvollständiges Zeichen gescannt wird. Der ITF-14, kann mit sichtbaren oder verdeckten senkrechten Bearer Bars ausgeführt werden.

Um der Berichtsseite einen ITF-14- Barcode hinzuzufügen, klicken Sie auf das Barcode-Symbol in der Seitenleiste und wählen Sie den ITF-14 im Dropdown-Menü aus.

Um einen Barcode zu bearbeiten, doppelklicken Sie darauf oder rufen Sie das Kontextmenü auf und wählen Sie Edit aus:



Eigenschaften des Barcode-Objektes:

- Angle – Mit dieser Eigenschaft können Sie die Drehung eines Objekts auf einen der festen Winkel - 0, 90, 180, 270 Grad - einstellen.
- Zoom – Die Eigenschaft legt die Skalierung des Barcodes fest. Diese Eigenschaft wird nur in Verbindung mit der AutoSize-Eigenschaft verwendet.
- AutoSize – Wenn diese Eigenschaft aktiviert ist, wird das Objekt so gestreckt, dass der gesamte Barcode angezeigt wird. Wenn die Eigenschaft deaktiviert ist, wird der Barcode auf die Größe des Objekts gestreckt.
- ShowText – Die Eigenschaft legt fest, ob der Text am unteren Rand des Barcodes angezeigt werden soll oder nicht.
- DataColumn – Datenfeld, aus dem der Objekttext geladen werden soll.
- Expression – Ausdruck, der den Objekttext zurückgibt.
- Text – Objekttext.
- Padding – Mit dieser Eigenschaft können Sie Einzüge von den Rändern des Objekts in Pixel festlegen.
- WideBarRatio – Diese Eigenschaft ist in allen linearen Barcodes vorhanden. Sie definiert die relative Größe der breiten Striche des Barcodes.
- CalcCheckSum – Diese Eigenschaft ist in vielen linearen Barcodes vorhanden. Sie bestimmt, ob die Prüfsumme automatisch berechnet werden soll. Wenn diese Eigenschaft deaktiviert ist, muss die Prüfsumme im Objekttext vorhanden sein.
- DrawVerticalBearerBars – Wenn diese Eigenschaft aktiviert ist, werden die Seitenstriche des Objekts angezeigt. Wenn die Eigenschaft deaktiviert ist, sieht der Barcode wie folgt aus:

Codabar

Dies ist ein linearer Barcode mit variabler Länge. Er wird am häufigsten zur Codierung von Seriennummern verwendet und wurde bereits 1972 von Pitney Bowes entwickelt. Die Striche ermöglichen ein genaues Lesen des Codes, auch wenn der Code auf einem Punktmatrixdruckern gedruckt wird.

Zeichensatz:

- Ziffern 0–9;
- Buchstaben A, B, C, D;
- Sonderzeichen -, $, /,., +.

Die Buchstaben können nur als Start- und Stopp-Zeichen des Codes verwendet werden. Sie können als zusätzliche Codeinformationen verwendet werden.

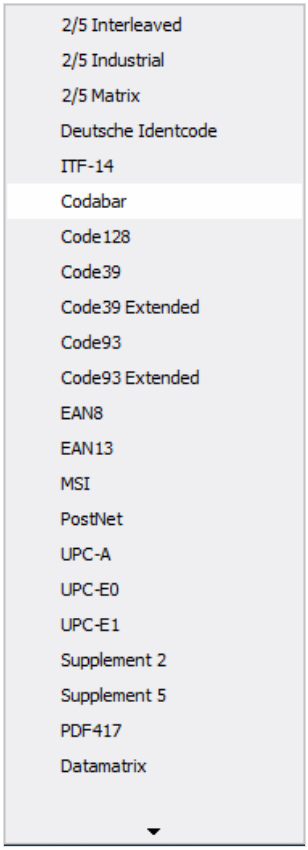
Der Code ist ganz einfach aufgebaut:

- Start-Zeichen bezeichnet den Beginn der Datenlesung;
- Zu codierende Daten;
- Stopp- Zeichen bezeichnet das Ende des Codes.

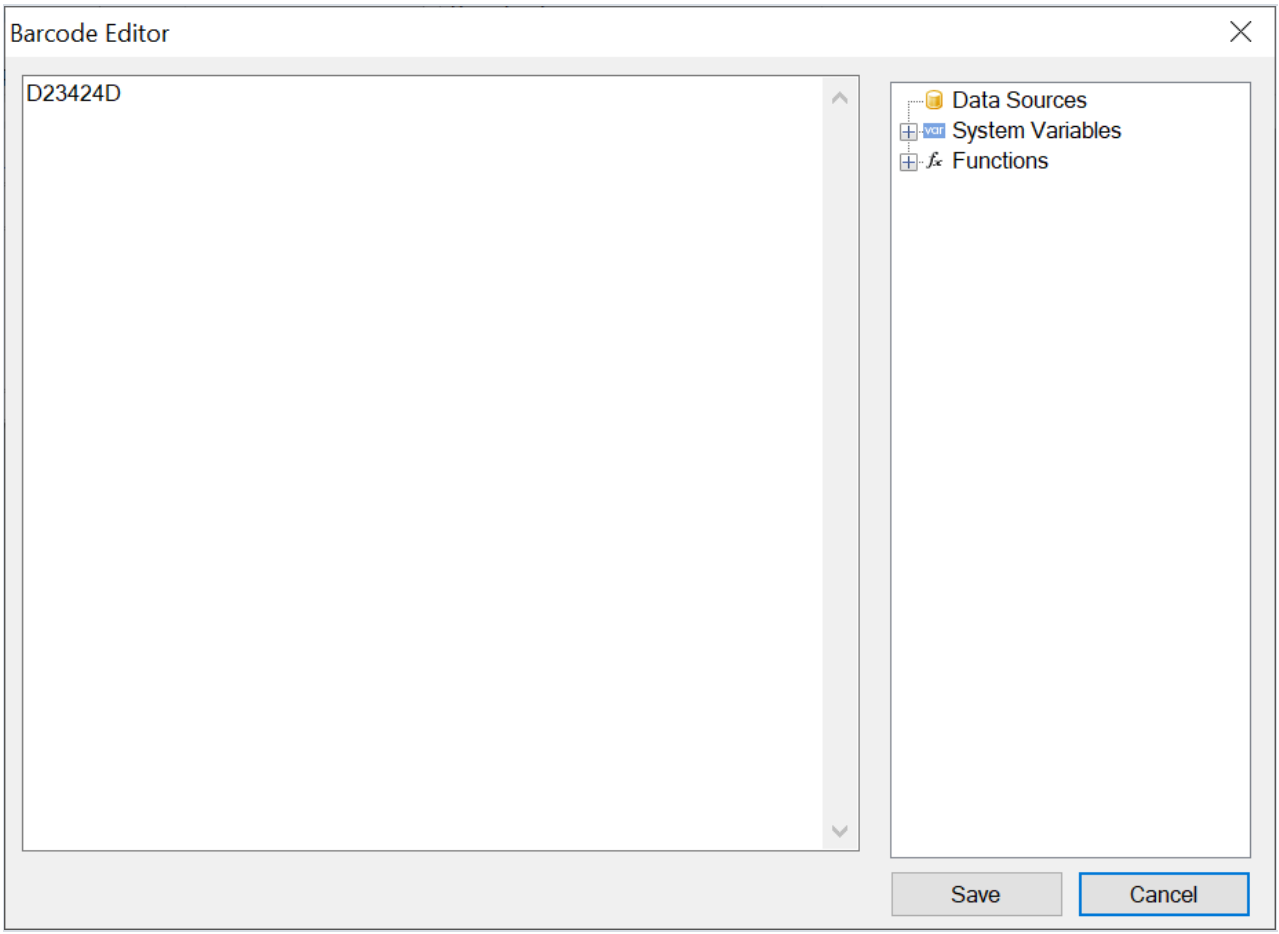
Kein Prüfsummenzeichen ist vorgesehen, da der Code über eine eingebaute Selbstprüfung verfügt.

Jedes Zeichen in Codabar wird mithilfe von 4 vertikalen Strichen und 3 Zwischenräme codiert.

Um der Berichtsseite einen Codabar hinzuzufügen, klicken Sie auf das Barcode-Symbol :



Platzieren Sie dann das Objekt auf der Berichtsseite. Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt. Alternativ kann der Code-Editor auch über das Kontextmenü des hinzugefügten Objekts aufgerufen werden, und zwar mit der Option Edit:



Um die Textanzeige unter dem Barcode deaktivieren, setzen Sie die ShowText-Eigenschaft im Eigenschafteninspektor des hinzugefügten Barcodes auf False:

Code 39

Der Code 39 ist ein linearer Barcode, der 1975 von Intermec entwickelt wurde. Dieser Code wurde wegen seiner Fähigkeit, alphanumerische Daten zu codieren, sehr beliebt. Er wird auch heute noch aktiv genutzt.

Ein wichtiges Merkmal dieses Codes ist die Fähigkeit, Daten beliebiger Länge zu codieren. Die Breite des Codes ist durch die Lesefähigkeit des Scanners begrenzt.

Jedes Zeichen des Codes besteht aus 9 Elementen und zwar aus fünf Strichen und vier Zwischenräumen. Diese große Anzahl von Elementen ist auf die eingebaute Selbstprüfung zurückzuführen, das heißt, jedes Zeichen des Codes wird geprüft und eine Prüfziffer ist nicht erforderlich. Sie ist jedoch weiterhin zulässig. Um eine höhere Code-Integrität zu sichern, können Sie auch eine Prüfziffer hinzufügen.

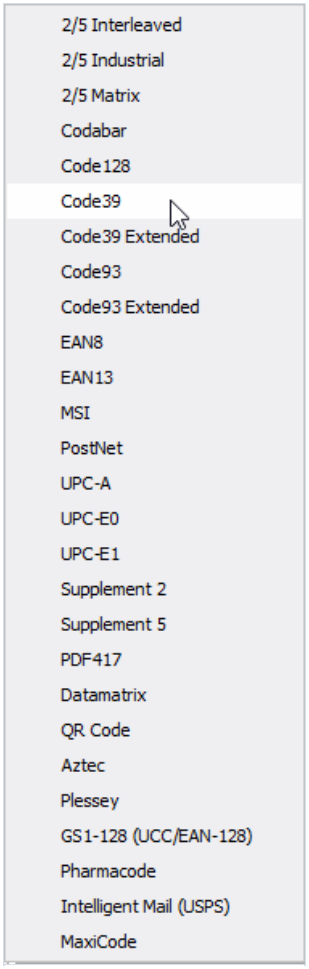
Wie bereits erwähnt, ermöglicht dieser Barcode die Codierung alphanumerischer Informationen. Es sind:

- Lateinische Großbuchstaben (A - Z);
- Ziffer (0–9);
- Sonderzeichen (-,.$/+%:) zugelassen.

Der Code 39 wird wie folgt aufgebaut:

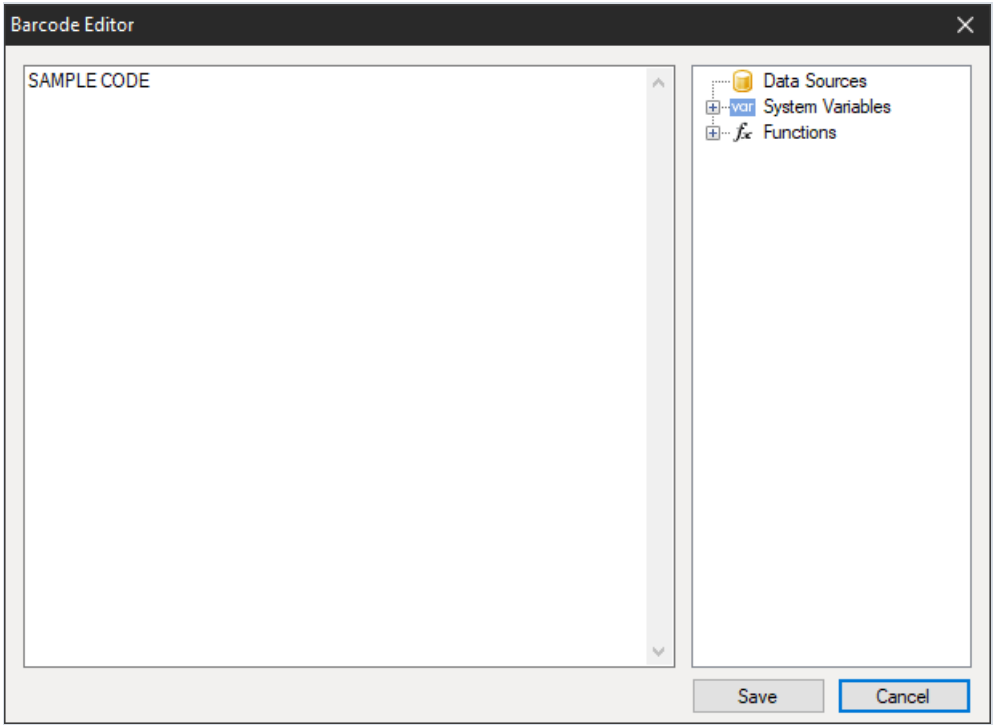
- Start-Zeichen, das mit dem "*" Zeichen codiert wird und den Anfang des Codes bezeichnet;
- Codierte Daten;
- Optionale Prüfsummenziffer;
- Stopp-Zeichen, das ebenso mit dem "*" Zeichen codiert wird.

Um der Berichtsseite einen Code 39 Extended hinzuzufügen, klicken Sie auf das Barcode-Objekt in der seitlichen Symbolleiste des Berichtsdesigners :



Wählen Sie dann Code 39 aus und platzieren Sie das Objekt auf der Berichtsseite.

Um den hinzugefügten Code zu bearbeiten, doppelklicken Sie auf das Objekt oder rufen Sie mit einem Rechtsklick das Kontextmenü auf und wählen Sie "Bearbeiten..." aus.



Sie können die Textanzeige unter dem Code deaktivieren, indem Sie die ShowText-Eigenschaft im Eigenschaftsinspektor des hinzugefügten Barcode-Objekts auf False setzen.

Code 39 Extended

Aus dem Namen geht hervor, dass es sich um eine erweiterte Version von Code 39 handelt. Er unterstützt den vollständigen Satz von ASCII-Zeichen, das heißt, neben den Großbuchstaben A-Z gibt es jetzt auch Kleinbuchstaben a-z sowie eine größere Anzahl von Sonderzeichen. Genau wie der Code 39 kann auch die erweiterte Version ein Prüfzeichen enthalten, das nach der Formel modulo 43 berechnet wird.

Der Barcode ermöglicht die Codierung von alphanumerischen Informationen. Es sind:

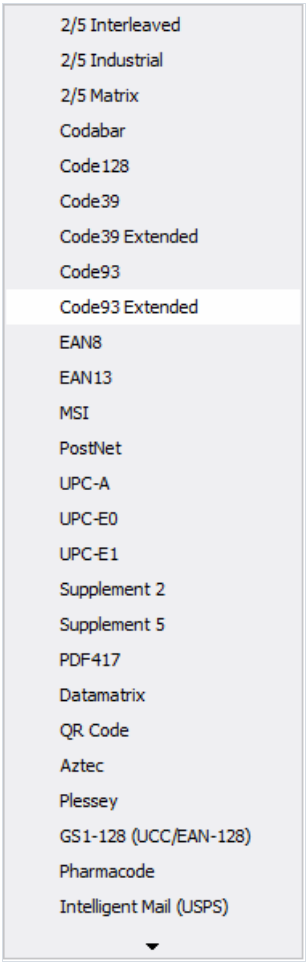
- Lateinische Klein- und Großbuchstaben (A - Z);
- Ziffern (0 - 9);
- Sonderzeichen zugelassen.

Code 39 Extended wird wie folgt aufgebaut:

- Start-Zeichen, das mit dem "*" Zeichen codiert wird;
- Codierte Daten;
- Optionale Prüfsummenziffer;
- Stopp-Zeichen wird ebenso mit dem "*"-Zeichen codiert.

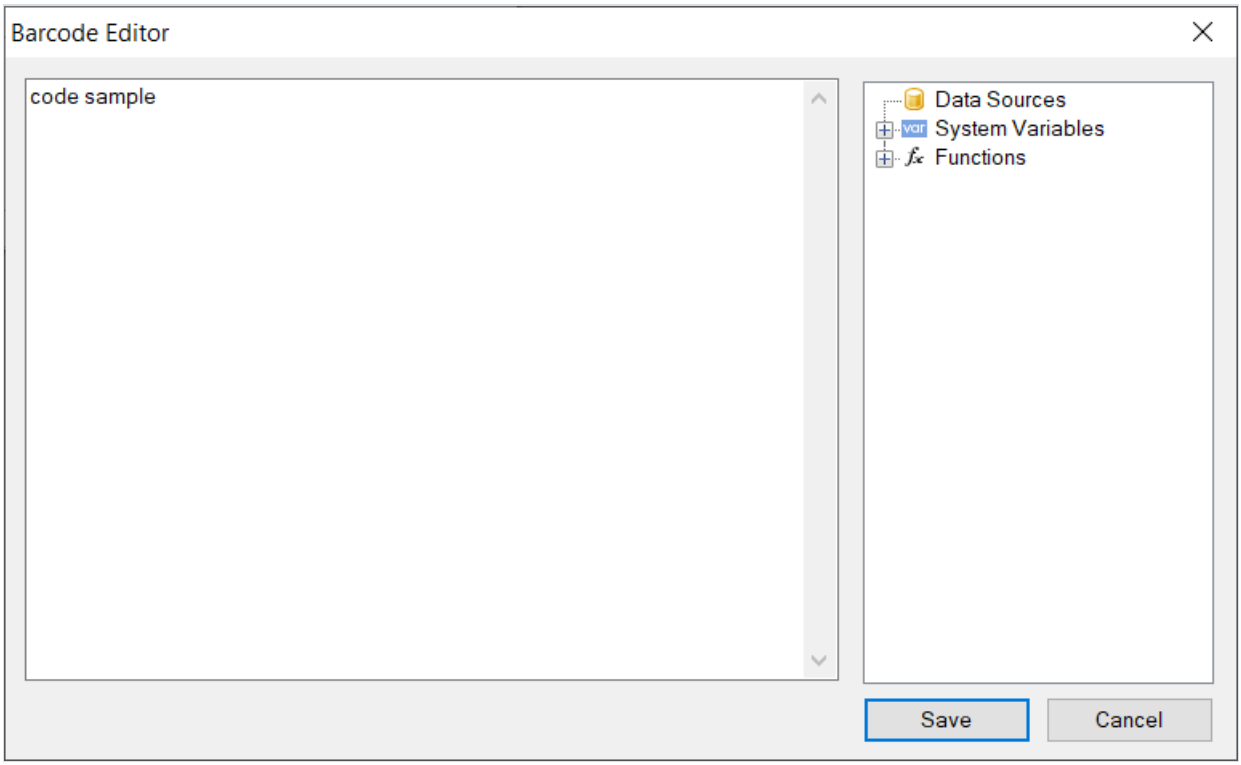
Zwei "+"-Zeichen und der gewünschte Großbuchstabe werden verwendet, um einen Kleinbuchstaben zu erstellen. Zur Codierung einiger Sonderzeichen wird auch eine Kombination aus zwei "%"-Zeichen und einem jeweiligen Großbuchstaben oder "/"-Zeichen und einem jeweiligen Großbuchstaben verwendet.

Um der Berichtsseite einen Code 39 Extended hinzuzufügen, klicken Sie auf das Barcode-Objekt in der seitlichen Symbolleiste des Berichtsdesigners :

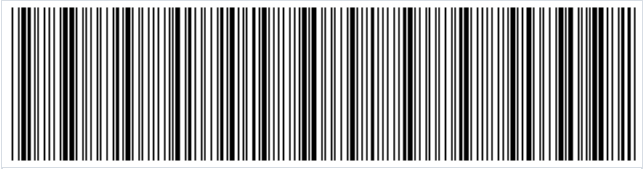


Wählen Sie dann Code 39 Extended aus und platzieren Sie das Objekt auf der Berichtsseite.

Um den hinzugefügten Code zu bearbeiten, doppelklicken Sie auf das Objekt oder rufen Sie mit einem Rechtsklick das Kontextmenü auf und wählen Sie "Bearbeiten..." aus.



Sie können die Textanzeige unter dem Code deaktivieren, indem Sie die ShowText-Eigenschaft im Eigenschaftsinspektor des hinzugefügten Barcode-Objekts auf False setzen.



Code 93

Der Code 93 (AIM-BC5-2000 Spezifikation) ist ein linearer Barcode, mit dem sowohl Zahlen als auch Buchstaben codiert werden können. Er wurde 1982 von Intermec entwickelt und basiert auf dem Code 39. Der Hauptvorteil gegenüber seinem Vorgänger ist seine geringere Größe.

Der Code 93 wird von der kanadischen Post zur Codierung von Paketinformationen und von der Automobilindustrie verwendet.

Die Länge des Codes ist auf 48 Zeichen begrenzt. Davon: 26 Großbuchstaben, 10 Ziffern, 7 Sonderzeichen und 5 Servicezeichen wie Anfang und Ende des Codes sowie Zeichen zur Erweiterung des Zeichensatzes. Dieser Code kann einen vollständigen Satz von ASCII-Zeichen darstellen, indem er eine Kombination von 2 Dienstzeichen verwendet.

Aufbau des Codes:

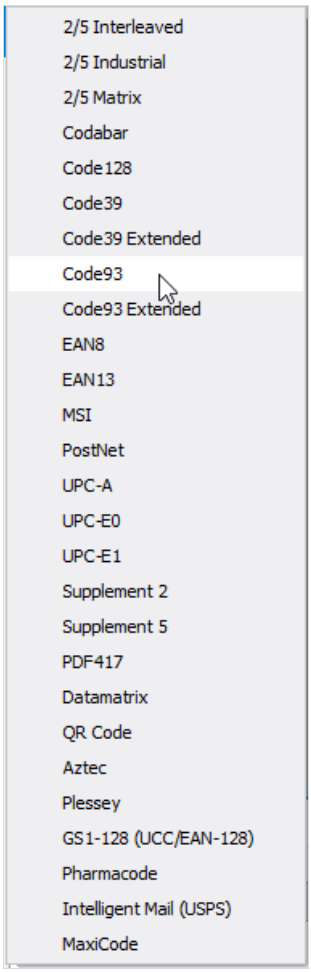
- Start-Zeichen;
- Codeinformation;
- Prüfsummenziffer, die nach Modulo "C" berechnet wird;
- Prüfsummenziffer, die nach Modulo "K" berechnet wird;
- Stopp- Zeichen;

Das Start-Zeichen wird als "*" dargestellt.

Jedes Datenzeichen besteht aus drei Strichen und drei Zwischenräumen. Die Breite der Striche oder Zwischenräume kann zwischen 1 und 4 Modulen liegen.

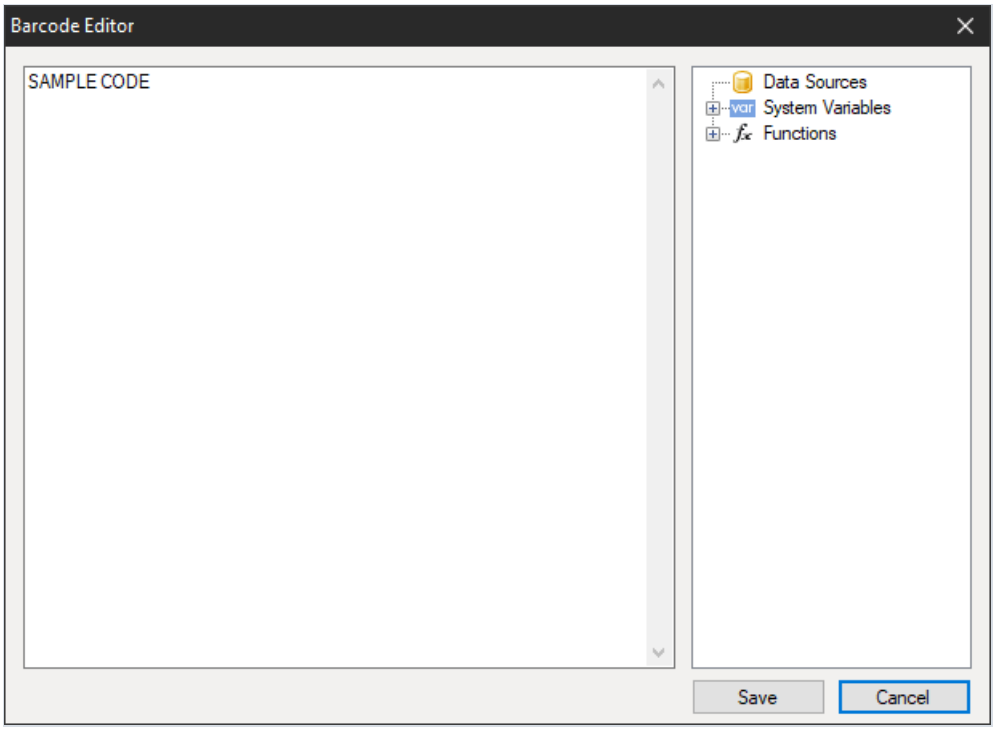
Die Prüfsummenziffern werden nicht im Text unter dem Code angezeigt. Die Prüfsumme besteht aus 2 Ziffern, die auf unterschiedliche Weise entweder nach Modulo "C" oder nach Modulo "K" berechnet werden. Die Formel ist der Rest modulo 47 der Summe der Gewichte der Datenwerte. Der Unterschied zwischen "C" und "K" liegt in der unterschiedlichen Gewichtung der Werte.

Um der Berichtsseite einen Code 93 hinzuzufügen, klicken Sie auf das Barcode-Symbol in der seitlichen Symbolleiste des Berichtsdesigners :



Wählen Sie dann den Code 93 Extended aus und platzieren Sie das Objekt auf der Berichtsseite.

Um den hinzugefügten Code zu bearbeiten, doppelklicken Sie auf das Objekt oder rufen Sie mit einem Rechtsklick das Kontextmenü auf und wählen Sie "Bearbeiten..." aus.



Sie können die Textanzeige unter dem Code deaktivieren, indem Sie die ShowText-Eigenschaft im Eigenschaftsinspektor des hinzugefügten Barcode-Objekts auf False setzen.

Code 93 Extended

Der Code 93 Extended ist eine Erweiterung von Code 93 und kann 128 ASCII-Zeichen anstelle der 48 des Basis-Barcodes codieren.

Zeichensatz:

- Ziffer 0–9;
- Großbuchstaben A-Z;
- Sonderzeichen: $,%, /, +.

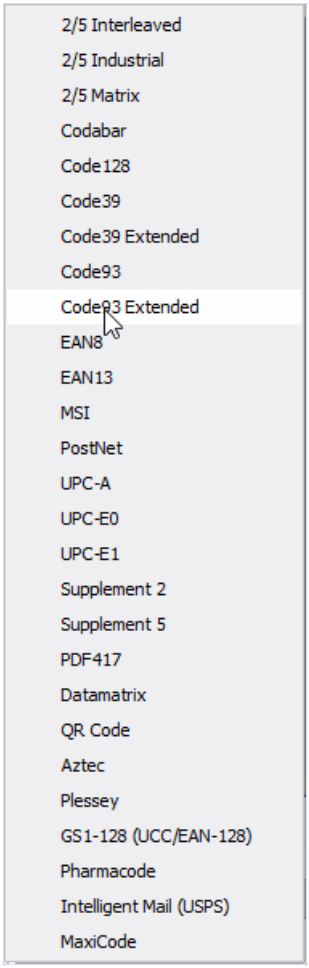
Aufbau des Codes:

- Start-Zeichen (*);
- Codeinformation – das Datenzeichen besteht aus drei Strichen und drei Zwischenräumen. Die Breite der Striche oder Zwischenräume kann zwischen 1 und 4 Modulen liegen;
- Prüfsummenziffer, die nach Modulo "C" berechnet wird;
- Prüfsummenziffer, die nach Modulo "K" berechnet wird;
- Stopp- Zeichen (*).

Die Prüfsummenziffern werden nicht im Text unter dem Code angezeigt. Die Prüfsumme besteht aus 2 Ziffern, die auf unterschiedliche Weise entweder nach Modulo "C" oder nach Modulo "K" berechnet werden. Die Formel ist der Rest modulo 47 der Summe der Gewichte der Datenwerte. Der Unterschied zwischen "C" und "K" liegt in der unterschiedlichen Gewichtung der Werte.

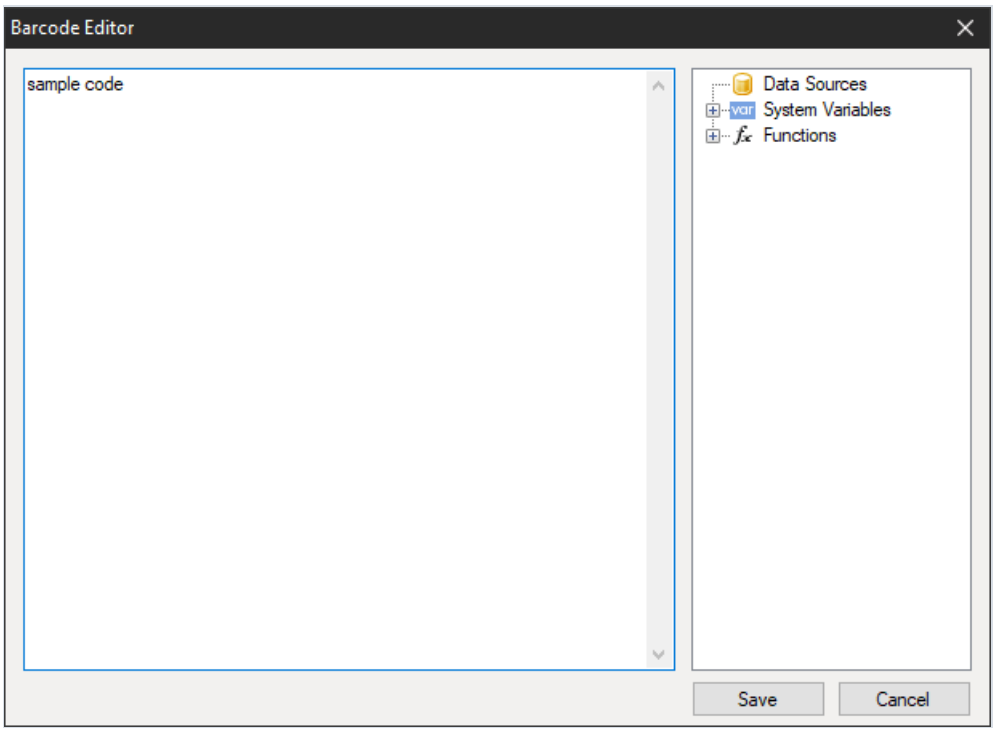
Wie bereits erwähnt, können mit dem Code 93 Extended 128 Zeichen codiert werden. Um jedoch die Standard-Code 93-Zeichentabelle zu erweitern, muss ein zusätzliches Zeichen ($, /, %, +) verwendet werden. Um Kleinbuchstaben zu codieren, müssen Sie also eine Kombination aus dem Zeichen + und dem richtigen Großbuchstaben verwenden. Die übrigen Modifizierungszeichen werden zur Erweiterung des Sonderzeichensatzes benötigt.

Um der Berichtsseite einen Code 93 Extended hinzuzufügen, klicken Sie auf das Barcode-Symbol in der seitlichen Symbolleiste des Berichtsdesigners:



Wählen Sie dann den Code 93 Extended aus und platzieren Sie das Objekt auf der Seite der Berichtsvorlage.

Um den hinzugefügten Code zu bearbeiten, doppelklicken Sie auf das Objekt oder rufen Sie mit mit einem Rechtsklick das Kontextmenü auf und wählen Sie "Bearbeiten..." aus.



Sie können die Textanzeige unter dem Code deaktivieren, indem Sie die ShowText-Eigenschaft im Eigenschaftsinspektor des hinzugefügten Barcode-Objekts auf False setzen.

Code 128

Der lineare Barcode Code 128 fand weite Verbreitung in vielen Bereichen. Sein Name weist darauf hin, dass er 128 Zeichen codieren kann. Nicht nur Ziffern, sondern auch Buchstaben und einige Sonderzeichen können codiert werden. Dies ist der Hauptvorteil von Code 128 gegenüber EAN-Codes.

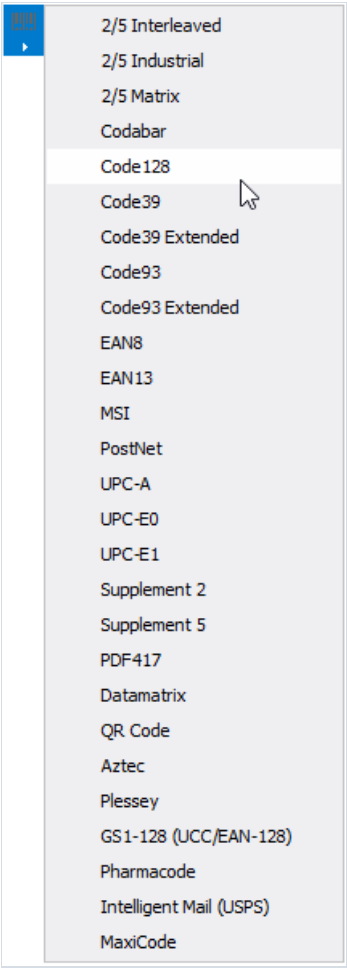
Der Code wie die meisten linearen Barcodes wird einfach aufgebaut. Auf beiden Seiten des Codes befindet sich ein Leerzeichen, so dass der Anfang des Codes eindeutig identifiziert werden kann. Dann folgt der Anfang des Codes, der durch ein Sonderzeichen, eine Folge von Strichen und Zwischenräumen gekennzeichnet wird. Nach dem Start-Zeichen folgen zu codierende Daten. Anschließend kommt das Prüfsummenzeichen, um die Code-Integrität zu überprüfen. Am Ende steht ein Stopp-Code, der das Ende der codierten Daten signalisiert.

Verschiedene Informationen werden mit unterschiedlichen Zeichensätzen für Buchstaben, Ziffern und Sonderzeichen codiert. Jedes Zeichen wird mit drei Strichen und drei Zwischenräumen codiert. Die Striche und Zwischenräume können unterschiedlich breit sein. Die Breite wird durch die Anzahl der Module bestimmt - von 1 bis 4.

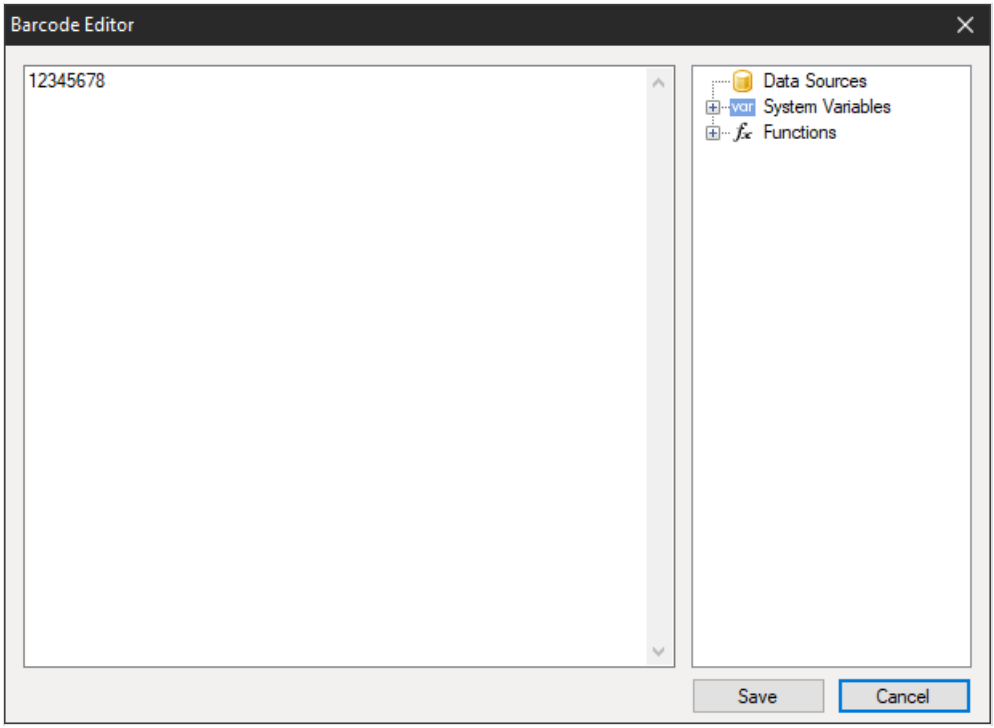
Der Code 128 hat also folgende Vorteile:

- Es können sowohl Ziffern als auch Buchstaben (Großbuchstaben) codiert werden;
- Der reiner Ziffercode ist dank der Doppelpackung sehr kompakt.

Um der Berichtsseite einen Code 128-Barcode hinzuzufügen, verwenden Sie das Barcode-Objekt in der seitlichen Symbolleiste des Berichtsdesigners und wählen Sie den Code aus:



Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt oder wählen Sie "Bearbeiten..." im Kontextmenü aus:



Sie können die Anzeige von Ziffern unter dem Code mit der ShowText-Eigenschaft im Eigenschafteninspektor des Barcode-Objekts deaktivieren:

QR-Code

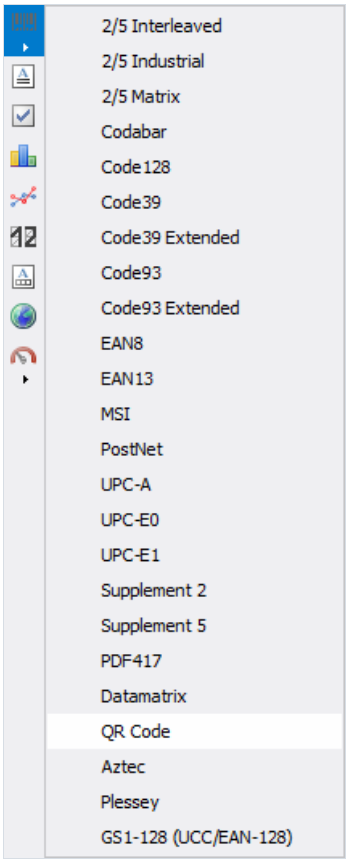
Der QR-Code ist ein zweidimensionaler Strichcode, der zur Speicherung von numerischen, alphanumerischen und binären Informationen entwickelt wurde.

Damit der QR-Code von der Kamera richtig erkannt wird, werden in diesem Code spezielle Markierungen in den Ecken und über dem Bildbereich verwendet. Auf diese Weise können Sie das Bild nach dem Lesen normalisieren und die Punktcodierung in Binärzahlen mit Prüfsummenüberprüfung umwandeln.

Ein QR-Code kann bis zu 4296 Zeichen enthalten (es wird eine alphanumerische Codierung verwendet).

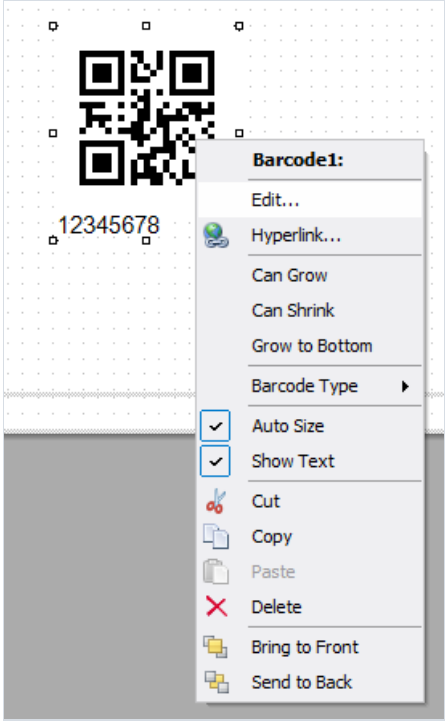
Objekt

Um der Berichtsseite einen QR-Code hinzuzufügen, klicken Sie in der Symbolleiste auf Barcode.



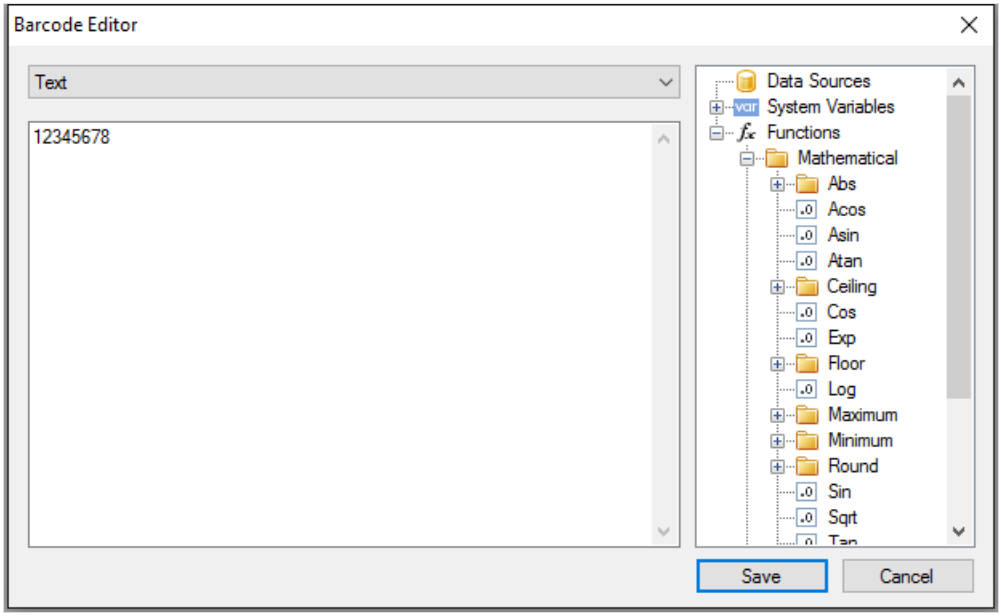
Wählen Sie dann den QR-Code in der Liste aus und klicken Sie auf die Berichtsseite.

Doppelklicken Sie auf das Barcode-Bild, um einen speziellen Editor zu öffnen. Dies kann auch über das Kontextmenü erfolgen, indem Sie `Bearbeiten` auswählen.

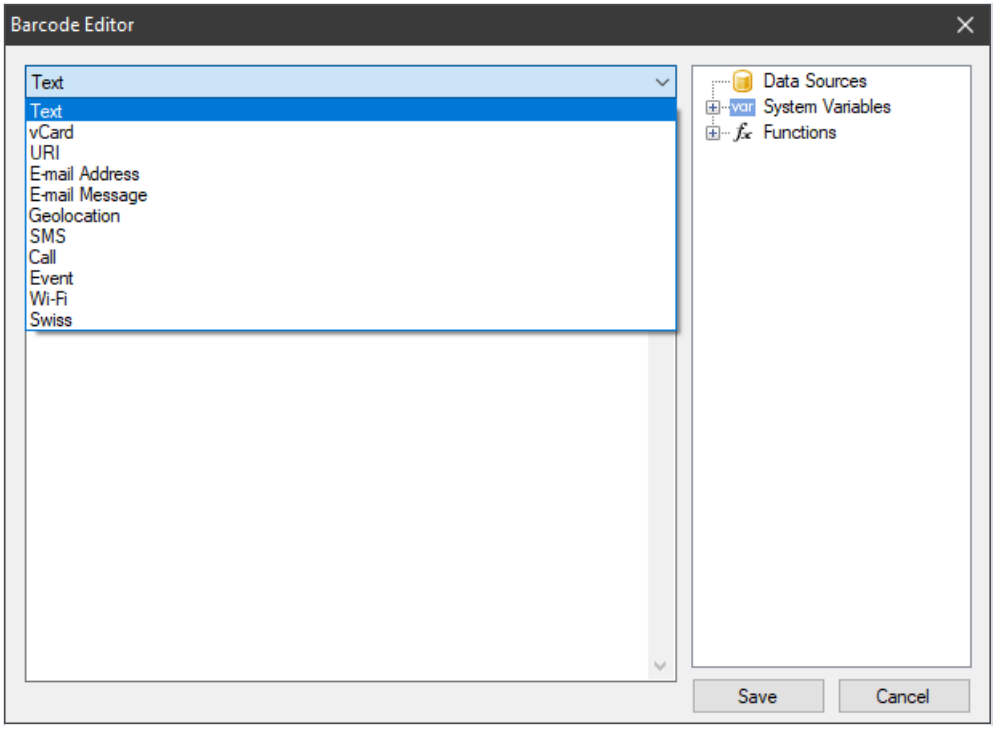


Editor

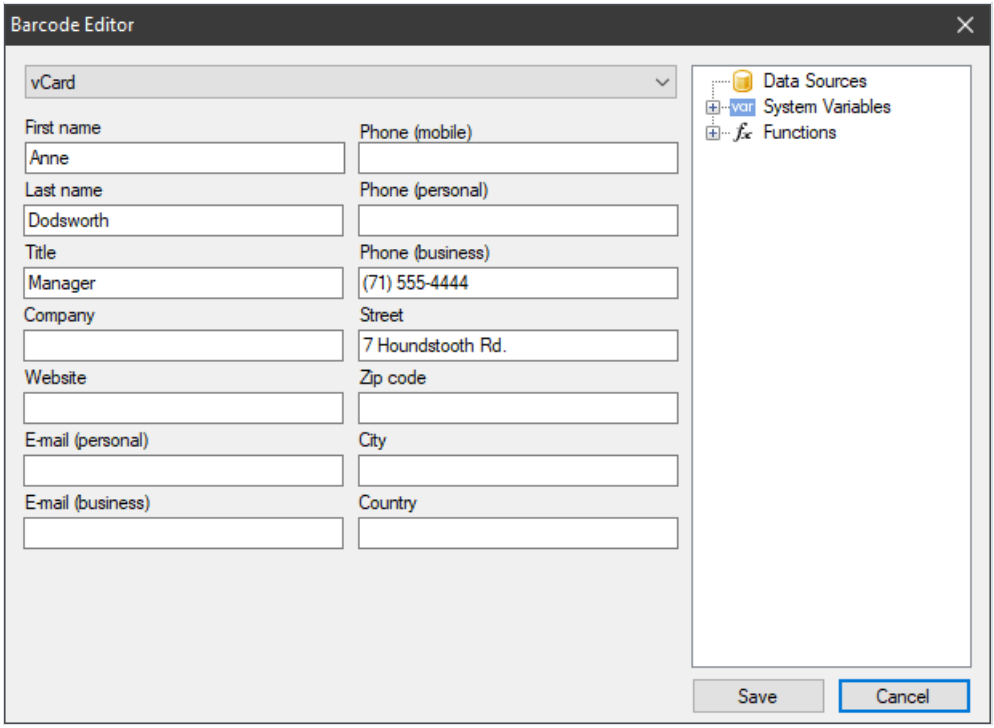
Der QR-Code-Editor ähnelt dem normalen Ausdruckseditor. Auf der rechten Seite befindet sich ein Baum von Daten, Parametern und Funktionen. Von dort aus können Sie Elemente in den Texteditor ziehen und dort ablegen.



Der Hauptunterschied zum Ausdruckseditor ist das Vorhandensein eines Dropdown-Menüs, das den Inhaltstyp des QR-Codes bestimmt.



Je nach ausgewähltem Inhaltstyp erscheint ein spezieller Satz von Feldern, die ausgefüllt werden müssen.



QR-Code-Inhaltstype

Wenn Sie mit dem Editor arbeiten, wird der Inhaltstext (die vierte Spalte der Tabelle) automatisch generiert. Die Felder des Editors können beliebige Ausdrücke enthalten, einschließlich der Felder aus der Datenquelle (sie können aus dem Baum auf der rechten Seite gezogen werden).

Inhaltstyp	Beschreibung	Beispielbild	Musterinhaltsangabe
Text	Text in alphabetischer und numerischer Darstellung		12345678

Inhaltstyp	Beschreibung	Beispielbild	Musterinhaltsangabe

vCard	Eine Versitcard ist eine „elektronische Visitenkarte“. Kann die folgenden Informationen enthalten: Nachname; Vorname; Funktion; Unternehmen; Website des Unternehmens; Persönliche E-Mail-Adresse; geschäftliche E-Mail-Adresse; Handynummer; Haustelefon; Geschäftstelefon; Anschrift; Postleitzahl; Stadt; Land. Es ist zugelassen, dass nur einige Felder ausgefüllt werden.		`BEGIN:VCARD`
			`VERSION:2.1`
			`FN:AnneDodsworth`
			`N:Dodsworth;Anne`
			`TITLE:Manager`
			`TEL;WORK;VOICE(71) 555-4444`
			`ADR:;;7 HoundstoothRd.;;;;`
			`END:VCARD`

URI	Einheitlicher Bezeichner für Ressourcen. Eine Zeichenfolge mit einem Link zu einer Datei, einem Dokument, einem Bild, einer E-Mail, einer Website usw.		`http://www.fast-report.com/en/product/fast-report-net/`
E-mail Address	E-Mail-Adresse		`support@fast-report.com`
E-mail Message	E-Mail-Text		`MATMSG:TO:support@fast-report.com;SUB:FastReport.Net question;BODY:Hello, I have a question about FastReport.Net.;;`
Geolocation	Koordinaten zur Bestimmung des tatsächlichen geografischen Standorts		`geo:-50.737563,-79.490016,120`
SMS	Textnachricht		`SMSTO:(71) 555-4444:Hello, Dolly! I'm fine!`
Call	Rufnummer		`tel:(71) 555-4444`

Event	Ereignis, das zum Kalender hinzugefügt wird. Neben der Uhrzeit und dem Datum kann es auch eine Textnachricht enthalten		`BEGIN:VEVENT`
			`SUMMARY:Future Day!`
			`DTSTART:20221021T121141Z`
			`DTEND:20221021T121141Z`
			`END:VEVENT`

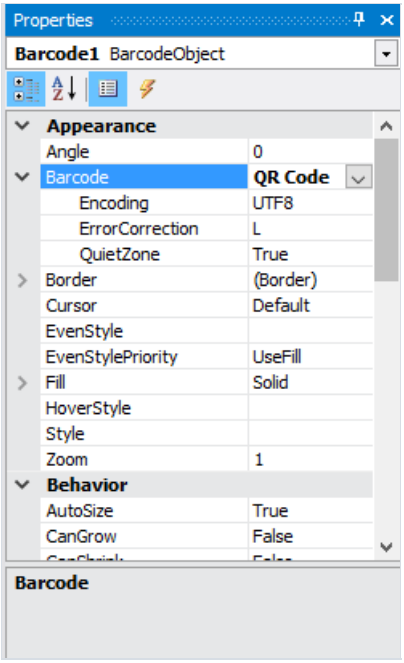
Wi-Fi	Angaben über die WLAN-Verbindung		`WIFI:T:WPA;S:Honeypot;P:youarewelcome;H:true;`

Inhaltstyp	Beschreibung	Beispielbild	Musterinhaltsangabe

Swiss	Ein spezieller QR-Code mit Zahlungsinformationen für Swiss Bill. Lesen Sie mehr über diesen Barcode in einem anderen Artikel.		`SPC`
			`0200`
			`1`
			`CH4431999123000889012`
			`S`
			`FastReports`
			`North Pitt Str`
			`901`
			`123456`
			`Alexandria`
			`US`

Barcode-Eigenschaften

Schauen wir uns nun die Eigenschaften des QR-Codes an. Diese sind im Objektinspektor unter der Eigenschaft Barcode verfügbar.



Eigenschaft	Beschreibung
"Codierung" `Encoding`	Codierung des Barcode-Inhaltstextes, z.B.: UTF8, Windows_1251, CP_866 etc. Standard: UTF8.
"Redundanz" `ErrorCorrection`	Redundanz zur Fehlerkorrektur mit einem Reed-Solomon-Code. Sie kann folgende Werte annehmen: L (low - 7%), M (medium - 15%), Q (25%), H (high - 30%). Standard - L.
"Feld" `QuietZone`	Legt fest, ob der QR-Code mit einem weißen Rand versehen werden soll. Standard: True (aktiviert).
"Farbe" `Color`	Legt die Farbe des Barcodes fest. Standard - Black (schwarz).

Die Redundanz ist erforderlich, um Daten korrekt zu lesen, wenn das Codebild teilweise beschädigt ist oder ein Bild überlagert wird.

Wenn beispielsweise die Redundanz auf H (30 %) eingestellt ist, kann der unten abgebildete Barcode problemlos gelesen werden:

Setzen Sie die Eigenschaft `ShowText` auf `False` um den Text unter dem Code auszublenden.

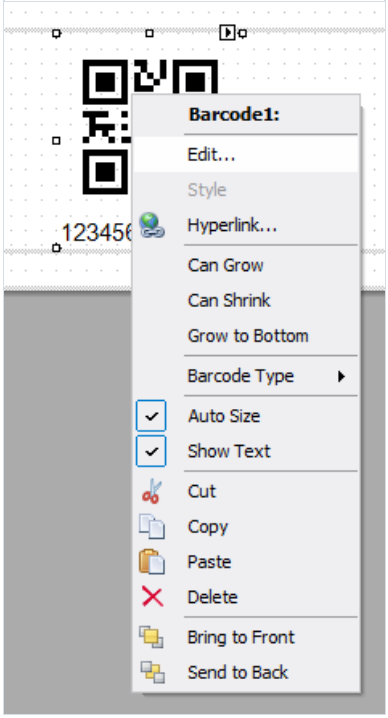
Die Eigenschaft `AutoSize` passt die Größe des Codes automatisch an die Größe des Objekts an. Wenn Sie die Größe des Barcodes manuell (mit der Maus) ändern müssen, müssen Sie diese Eigenschaft deaktivieren (auf `False` setzen). In diesem Fall müssen Sie die horizontalen und vertikalen Proportionen selbst überwachen.

Swiss QR

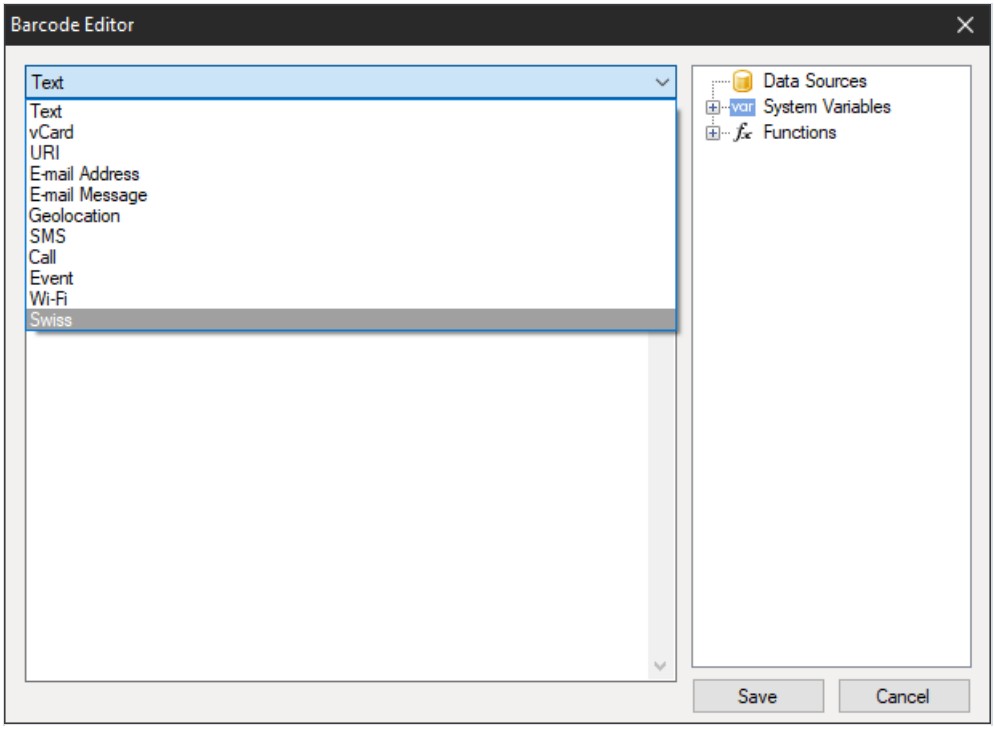
Der Swiss QR ist eine Art von QR-Codes. In der Schweiz, wie auf der ganzen Welt werden QR-Codes überall verwendet. Um diese Codes jedoch im Bereich des elektronischen Zahlungsverkehrs zu verwenden, wurde beschlossen, eine eigene QR-Variante zu entwickeln. Alle Zahlungsbelege und Rechnungen in der Schweiz müssen jetzt einen QR-Code mit einem Schweizerkreuz in der Mitte haben - das ist das Markenzeichen von Swiss QR.

FastReport .NET unterstützt den Swiss QR-Code. Der Code selbst sieht ziemlich genau so wie ein normaler QR-Code aus, der einzige Unterschied ist das Schweizer Kreuz in der Mitte:

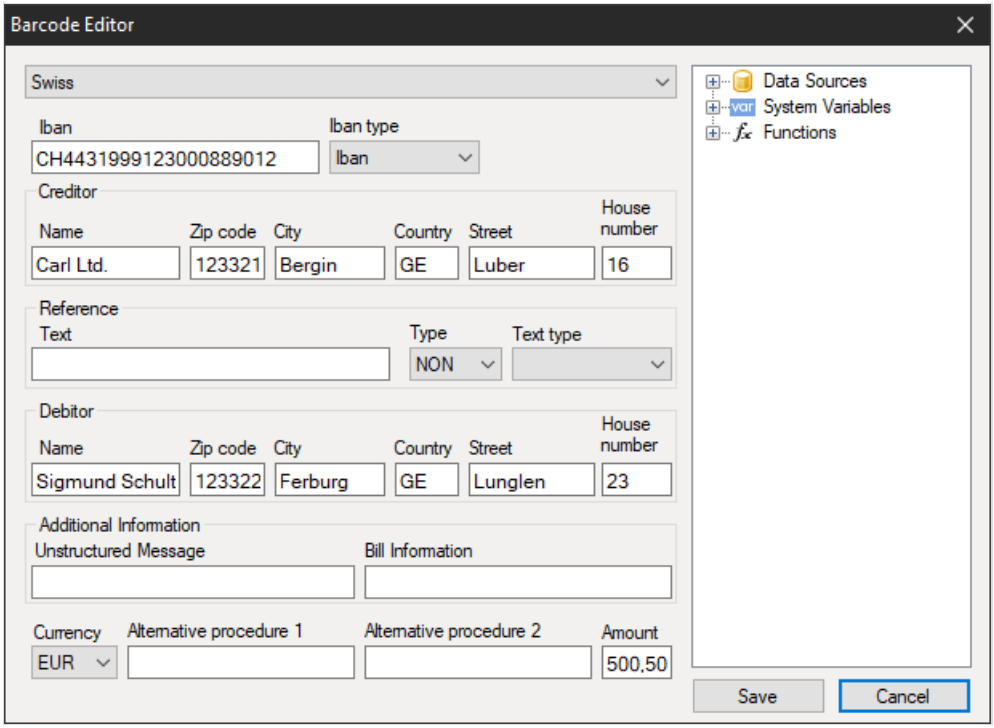
Der Schwerpunkt dieser Art von Code liegt darin, die Zahlungsdaten zu codieren. Um den Swiss QR in einem Bericht zu verwenden, müssen Sie der Berichtsseite ein Barcode-Objekt vom Typ QR-Code hinzufügen. Dann müssen Sie den QR-Code-Editor durch einen Doppelklick auf das Objekt oder über das Kontextmenü aufrufen:



Wählen Sie im QR-Code-Editor den Typ Swiss aus:



Dadurch werden die Einstellungen des Swiss-Codes angezeigt:



Schauen wir uns die Parameter näher an:

Iban

Die Schweiz verwendet den IBAN-Standard (*International Bank Account Number*) zur Darstellung der Bankkontonummer. Es handelt sich um eine internationale Norm gemäß ISO 13616.

Hier können Sie zwischen zwei Arten von `Iban` oder `QRIban` wählen.

Für Zahlungen mit QR-Referenz muss QRIban verwendet werden. Die `QRIban` entspricht ebenso den in ISO 13616 festgelegten Regeln. Jede juristische Person, die an dem System teilnimmt, erhält eine eigene Identifikation im Bereich von 30000 bis 31999. Diese Identifikation wird QR-IID genannt und ist in QR-Iban enthalten.

Creditor

Hier werden die Daten des Rechnungsstellers angegeben. Name und Anschrift des Unternehmens.

Reference

Es handelt sich um eine Referenz auf die Zahlung seitens des Zahlers, die der Zahlungsempfänger benötigt.

Type

- QRR, QR-Referenz: Schweizer Standard für Referenz mit einer Länge von 26 Zeichen (immer numerische Zeichen);
- SCOR, Kreditorenreferenz: internationaler Standard mit einer Länge von 5 bis 25 Zeichen;
- NON – die Referenz kann leer sein

Text type

- QR-Reference – wird mit einer QRR-Referenz verwendet;
- ISO 13616 – wird mit einer SCOR-Referenz verwendet.

Die QR-Referenz dient als Ersatz der derzeit verwendeten ISR-Referenz. Sie hilft Ihnen, von den bisher verwendeten roten und orangen Belegen zu den neuen QR-Rechnungen zu wechseln.

Debitor

Daten des Zahlers, vollständiger Name oder Name des Unternehmens und Anschrift.

Additional Information

Der Rechnungssteller kann beliebige zusätzliche strukturierte/unstrukturierte Informationen für den Zahlers eingeben.

Currency

Da es sich um ein Schweizer Zahlungssystem handelt, wird von zwei Arten von Währungen ausgegangen: Euro und Schweizer Franken (CHF).

Alternative Procedure 1 и Alternative Procedure 2

Es wird angenommen, dass die Rechnungssteller in Zukunft alternative Verfahren zur Banküberweisung anbieten können. In Swiss QR sind dafür zwei Felder vorgesehen.

Amount

Hier wird der Zahlungsbetrag mit einem Komma als Trennzeichen angezeigt. Normalerweise wird der Schweizer QR-Code in einem Swiss bill verwendet. Ein Beispiel für einen solchen Swiss bill finden Sie im Ordner `Demos/Reports/`

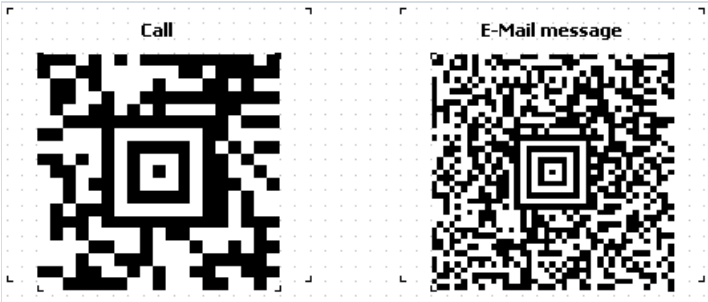
Aztec–Code

Der Aztec-Code ist eine Art von QR-Code. Der Name Aztec-Code leitet sich von der Hochkultur der Azteken in Zentralmexiko deren Stufenpyramiden ab. Wenn man eine Stufenpyramide aus der Vogelperspektive betrachtet, dann ähnelt sie stark den direkt in der Code-Mitte erkennbaren Quadraten, die um einen einzelnen Punkt zentriert sind. Dies ist ein spezielles Suchelement, mit dem der Mittelpunkt des Codes und seine Ausrichtung bestimmt werden können.

Der Aztec-Code kombiniert die besten Ideen von 2D-Barcodes: MaxiCode, SuperCode, CodeOne, DataMatrix, DotCode und PDF417. Obwohl es sich beim Aztec-Code um ein Patent handelt, ist er frei verfügbar. Er ist unter ISO/IEC 24778:2008 normiert.

Die Größe des Codes hängt von der Menge der zu codierenden Informationen ab. Die Mindestgröße von 15x15 Pixeln ermöglicht beispielsweise die Codierung von 6 Bytes, d.h. 12 Buchstaben oder 13 Ziffern. Und die maximale Größe von 151x151 Pixeln erlaubt die Codierung von 1914 Bytes, 3067 Buchstaben oder 3832 Ziffern.

Beachten Sie, dass der Code zwei Anzeigeformate hat: Compact, bei dem das Suchelement aus zwei Quadraten besteht, und Full-Range, bei dem das Suchelement aus drei Quadraten besteht. Die Wahl des Formats hängt von der Menge der zu codierenden Daten ab.



Der Vorteil dieser Art von Codierung gegenüber anderen besteht darin, dass der Code in jeder Ausrichtung gelesen werden kann. Auch wenn der Code gespiegelt ist, ist er leicht zu lesen. Dies wird durch die Verwendung von Navigationsmarkern erreicht.

Die Verwendung des Suchelements im Mittelpunkt des Codes ermöglicht es, Informationen auch aus verzerrten oder gestreckten Bildern zu lesen.

Dank des Reed-Solomon-Codierungsalgorithmus kann der Aztec-Code auch gelesen werden, wenn er teilweise

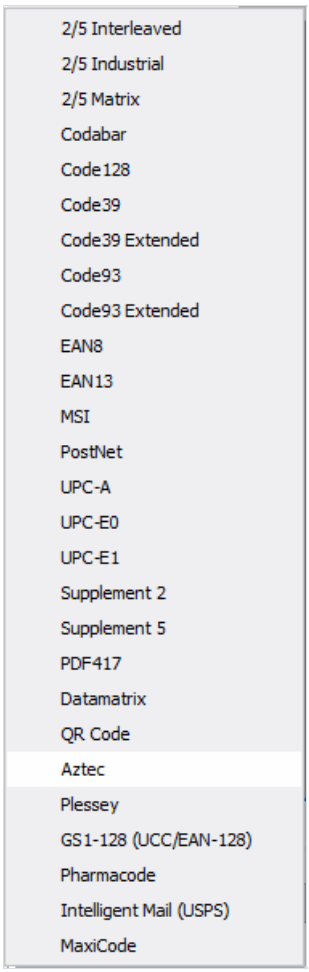
beschädigt ist. In diesem Fall ist der Code redundant. Es besteht die Möglichkeit, den Prozentsatz des redundanten Codes von 5 bis 95 einzustellen. Auf diese Weise kann eine sehr hohe Resistenz gegen Lesefehler gewährleistet werden.

Der schichtweise Codenaufbau ermöglicht es, die Menge der gespeicherten Informationen durch Vergrößerung des Codierbereichs zu erhöhen.

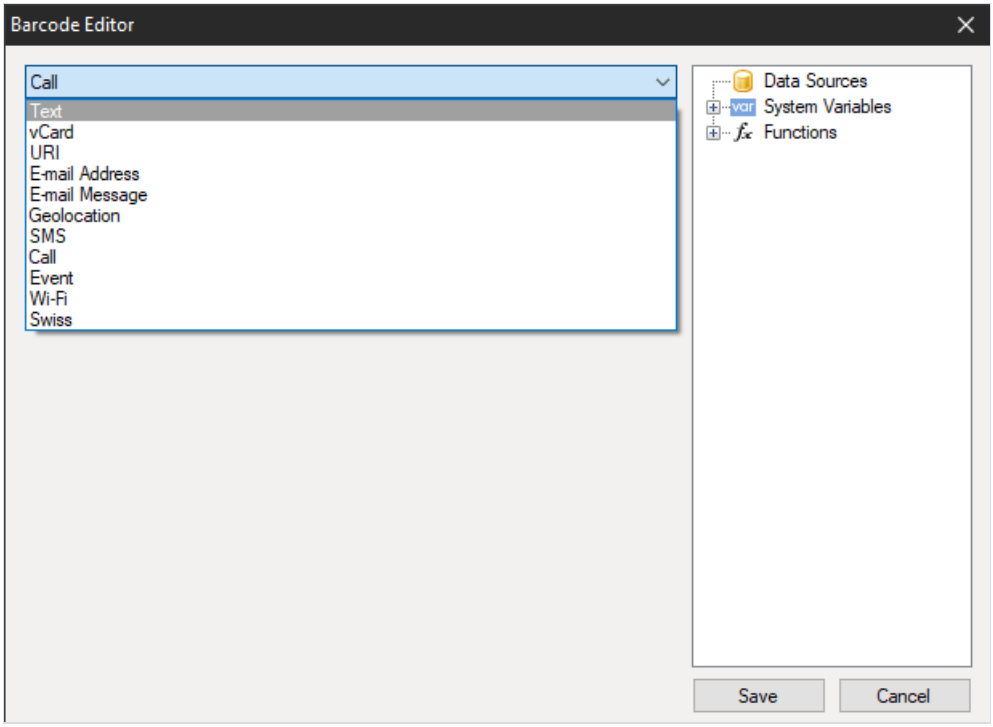
All diese Vorteile haben den Aztec-Code für den Einsatz in Verkehrsnetzen als E-Ticket, z. B. im Flug- und Bahnverkehr, sehr attraktiv gemacht. In einer Reihe von Ländern wird es auch in staatlichen Dokumenten verwendet. Wie andere High-Density-Codes sind auch Aztec-Codes im Handel, in der Logistik, in der Produktion und in der Pharmazie beliebt.

Im Vergleich zum QR-Code hat der Aztec-Code eine höhere Aufzeichnungsdichte und benötigt keine Felder um den Code herum. Außerdem beträgt die Mindestgröße von Aztec-Code 15x15 im Vergleich zu 21x21 beim QR-Code.

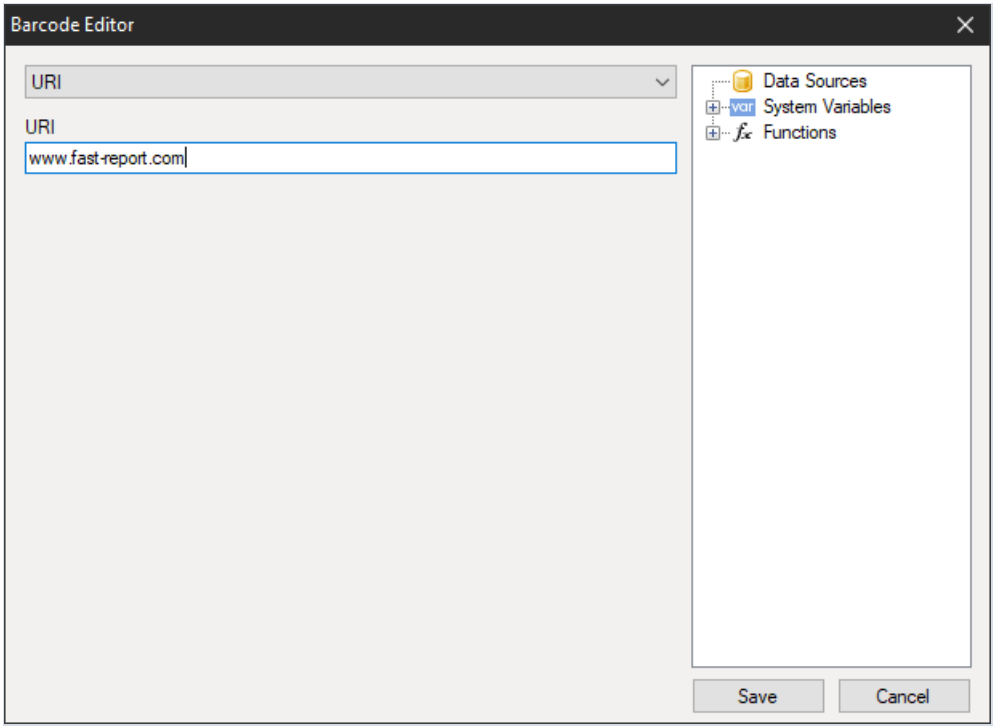
Um einen Aztec-Code in FastReport .NET zu erzeugen, wählen Sie das Barcode-Objekt im Komponentenbereich des Berichtsdesigners aus. Wählen Sie Aztec im Dropdown-Liste aus:



Doppelklicken Sie auf den hinzugefügten Barcode, um den Editor zu öffnen:



Hier können Sie eine Vorlage für die Codierung von Informationen auswählen. Alle außer Swiss können im Aztec-Code verwendet werden. Zum Beispiel die Adresse einer Website:



Der resultierende Code sieht wie folgt aus:

Pharmacode

Der Pharmacode ist ein binärer Code, der von der deutschen Firma LAETUS GMBH speziell für Pharmaverpackungen entwickelt wurde. Er ist eine Teilmenge von CODE39. Er wird in der pharmazeutischen Industrie zur Packmittelkontrolle eingesetzt.

Als Teil eines computergestützten Verpackungssystems ermöglicht Pharmacode das einfache Scannen und Erfassen von Arzneimittelsendungen mit universellen Identifikatoren. Mit den Scannern lässt sich auch leicht feststellen, ob eine Charge von Medikamenten mit einer anderen vermischt wurde.

Der Pharmacode garantiert, dass der Code trotz möglicher Druckfehler lesbar ist. Um sicherzustellen, dass der Rest der Verpackung, außer Code korrekt gedruckt wird, dürfen Teile des Pharmacodes im Gegensatz zu anderen Barcodearten, die mit Laser oder Laseremulation gelesen werden, in verschiedenen Farben (Code und Hintergrund), gedruckt sein. Dies ist möglich, weil Pharmacode mit speziellen weißen LAETUS-Scannern gescannt wird. Dies macht den Pharmacode zu einem sehr praktischen Format für den Druck auf Verpackungen oder Dokumenten, die keine schwarze Tinte enthalten.

Wie bereits erwähnt, kann der Pharmacode in verschiedenen Farben gedruckt werden. Wobei können sowohl der Code selbst als auch der Hintergrund eine andere Farbe als Weiß oder Schwarz haben. Je nach Art des Lesescanners gelten spezielle Vorgaben für Kombinationen aus Code- und Hintergrundfarbe. Standard-Schwarz-Weiß-Scanner nehmen beispielsweise nur kontrastreiche Codes und Hintergründe wahr, während spezielle Scanner, die Farbe erkennen, keine strengen Einschränkungen haben.

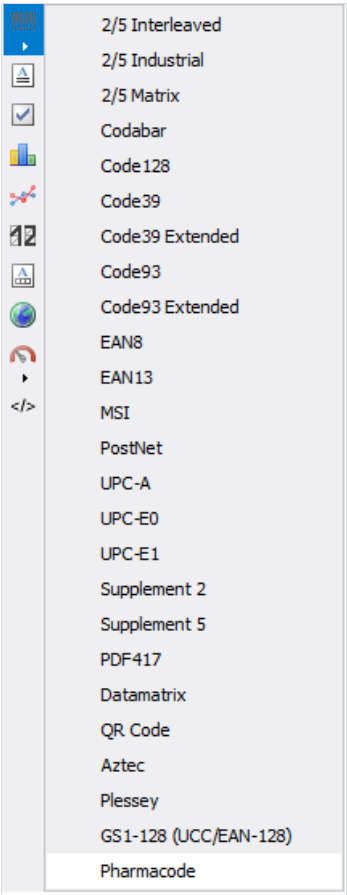
Im Gegensatz zu anderen 1D-Barcodes speichert der Pharmacode die Daten in binärer und nicht in dezimaler Form. Darüber hinaus können mit dem Pharmacode nur Ganzzahlen von 3 bis 131070 codiert werden. Die Mindestanzahl der Striche beträgt 2 für 3 und der Höchstwert beträgt 16 für 131070. Der Pharmacode wird von rechts nach links gelesen, was ihn unter anderen linearen Barcodes, die ein Start- und Stopp-Zeichen haben, einzigartig macht. Liest man den Code von links nach rechts, ergibt sich eine ganz andere Zahlenfolge.

Der Pharmacode-Standard wird durch den Entwickler LAETUS geregelt und im PharmaCode Guide beschrieben.

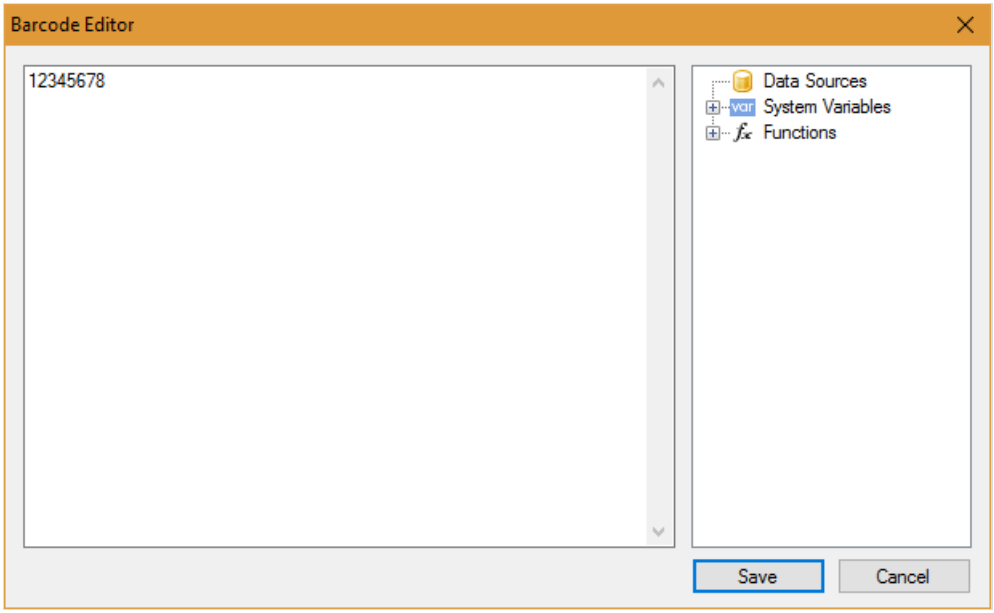
Hier ist ein Beispiel für einen Pharmacode:

Mit FastReport .NET können Sie solche Codes in Ihren Berichten erstellen. Sie können Ihre Verpackungen im Generator sofort mit Barcodes gestalten.

Um der Berichtsseite einen Code hinzuzufügen, müssen Sie das Barcode-Objekt hinzufügen. Wählen Sie den Pharmacode in der Dropdown-Liste aus:



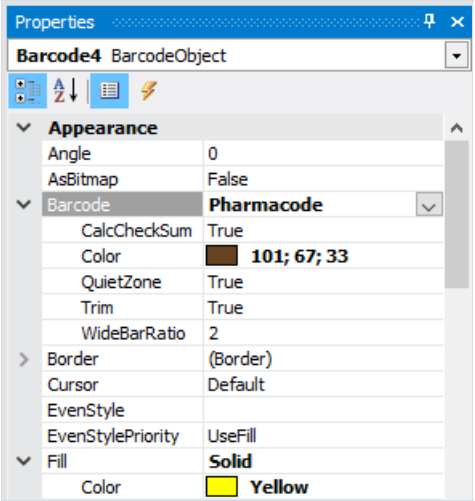
Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt, oder wählen Sie im Kontextmenü des Barcode-Objekts „Edit" aus:



Als Codewert können Sie eine Zahlenfolge eingeben, eine Funktion, eine Berichtsvariable, einen Wert aus der Datenbank festlegen.

In den Barcode-Eigenschaften können Sie den Abstand zwischen den Strichen (WideBarRatio), die Höhe des Codes (Height), die Anzeige von Ziffern unter dem Code (ShowText) ändern.

Standardmäßig ist der Barcode schwarz auf weißem Hintergrund. Sie können die Farbe des Codes in der Eigenschaft Barcode -> Color ändern. Die Hintergrundfarbe kann über die Eigenschaft Fill -> Color geändert werden:



Als Ergebnis der Farbanpassung kann der Pharmacode wie folgt aussehen:

MaxiCode

Der zweidimensionale MaxiCode Barcode hat einen sehr ungewöhnlichen Aufbau. Er kann mit keinem anderen Typ verwechselt werden. Er hat ein kreisförmiges Suchmuster zum Ausrichten des Codes beim Scannen. Die Codeelemente sind Punkte, die sich in einer bestimmten Reihenfolge angeordnet sind, wobei eine Wabenstruktur gebildet wird. Ein hexagonales (wabenförmiges) Gitter besteht aus 888 informationstragenden Zellen.

Der MaxiCode kann 93 ASCII-Zeichen oder 138 Ziffern codieren. Der Barcode hat eine feste Größe von ungefähr 1,11 x 1,054 Zoll. Es gibt eine Reed-Solomon-Fehlerkorrektur. Da die meisten dieser Barcodes auf der Verpackung angebracht werden, müssen sie resistent gegen Beschädigungen sein. Die Rekonstruktion ist noch möglich, wenn bis zu 25 % des Codes zerstört ist.

Ein MaxiCode umfasst eine primäre Nachricht und eine sekundäre Nachricht. Die primäre Nachricht enthält Postleitzahl, Länderkennzeichen und Klasse, während die sekundäre Nachricht Adressangaben enthält.

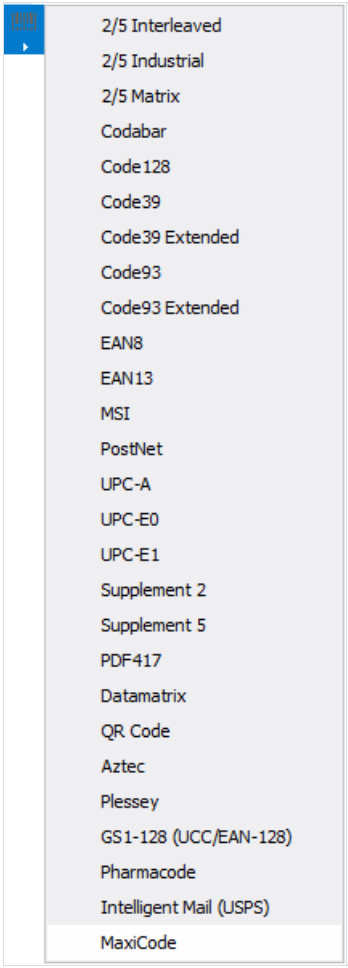
Um den Maxicode zu lesen ist ein 2D-Barcodescanner erforderlich, vorzugsweise einer, der eine Tastaturemulation durchführt und über den USB-Anschluss mit Strom versorgt wird, sodass keine externe Stromversorgung erforderlich ist.

Der MaxiCode wird hauptsächlich verwendet, um Pakete, Palette usw. zu kennzeichnen. Er stellt sicher, dass wichtige Informationen jederzeit verfügbar sind, wenn eine Sendung bearbeitet wird.

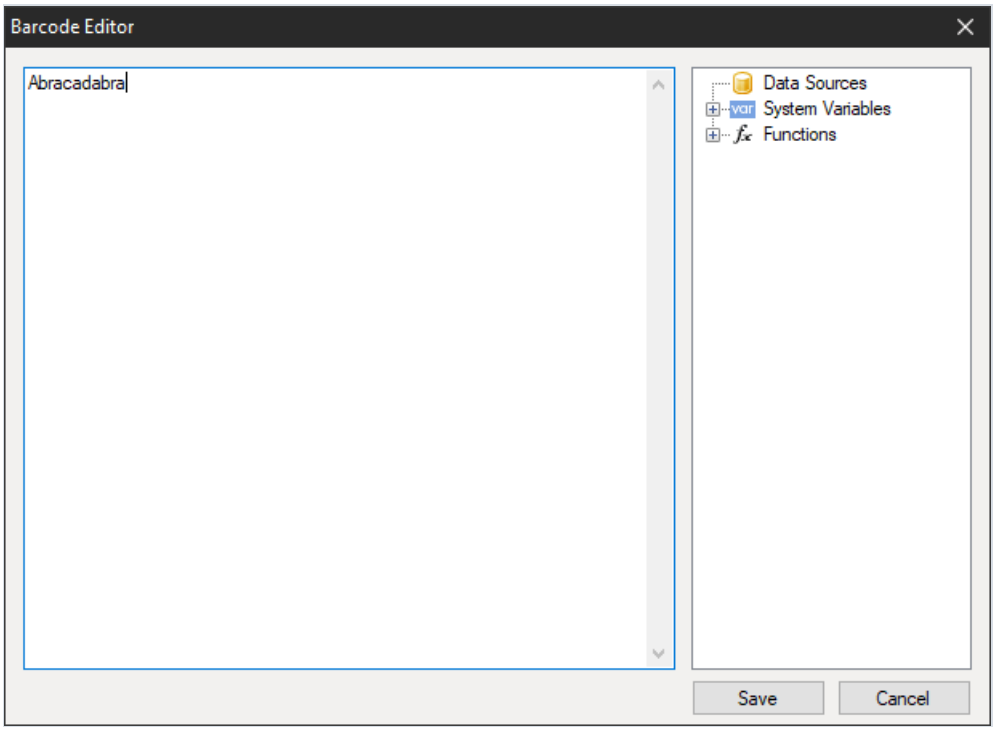
Der MaxiCode wurde so konzipiert, dass es in das bestehende Manifest des Liefersystems integriert werden kann. Aufgrund der kompakten Größe kann der MaxiCode auch verwendet werden, um Zeichen mit geringerer Dichte wie lineare Barcodes zu ersetzen.

Eines der wichtigsten Merkmale von MaxiCode ist, dass er mit hoher Geschwindigkeit in einem großen Sichtfeld gelesen werden kann. Dies bedeutet, dass er erfolgreich in automatischen Informationsverarbeitungssystemen eingesetzt werden kann.

Um der Berichtsseite einen MaxiCode hinzuzufügen, klicken Sie auf das Barcode-Objekt in der Symbolleiste des Berichtsdesigners und wählen Sie den MaxiCode in der Dropdown-Liste aus. Sie können den Code dann auf der Berichtsseite platzieren.



Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt oder rufen Sie das Kontextmenü auf und wählen Sie "Bearbeiten..." aus. Im Barcode-Editor ist es möglich, alphanumerische Informationen einzugeben:



Wie alle FastReport-Barcodes kann auch der MaxiCode mit oder ohne Textinformationen unter dem Code angezeigt werden. Dies wird durch die ShowText-Eigenschaft gesteuert.

Plessey/MSI

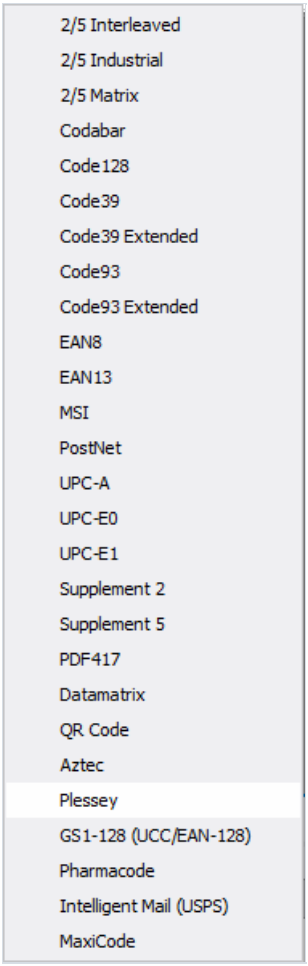
Der Plessey-Barcode wurde 1971 von der gleichnamigen Firma entwickelt. Es handelt sich um einen klassischen eindimensionalen linearen Barcode, der hauptsächlich zur Kennzeichnung von Waren an Regalen in Supermärkten und Lagerhallen eingesetzt wird. Der Hauptvorteil des Codes bestand zum Zeitpunkt seiner Entstehung darin, dass er sich leicht auf einem Nadeldrucker drucken ließ. Er gilt heute als veraltet und ist praktisch nicht mehr gebräuchlich.

Mit dem Plessey-Code können Sie Hexadezimalziffern (0-F) codieren. Jede Ziffer wird durch vier Bits, Balken, dargestellt. Die Null ist ein dünner Strich, die Eins ist ein dicker Strich. Neben den Ziffern können auch die Buchstaben A-F codiert werden. Der Barcode besteht aus einem Start-Zeichen, codierten Daten, einem Prüfziffer, einem End-Zeichen und einem Code zum Rückwärtslesen (der Code kann in beide Richtungen gelesen werden).

Eine Form vom Plessey ist MSI. Im Gegensatz zum Standard-Plessey.Barcode können nur Ziffern codiert werden und es gibt keinen Code zum Rückwärtslesen. Der MSI unterstützt mehrere Arten von Prüfziffern, zum Beispiel: Mod-10, Mod-11, Mod-1010, Mod-1110.

Sowohl der MSI als auch der Plessey schränken die Länge des Codes nicht ein, aber ein zu langer Code passt möglicherweise nicht auf die Verpackung, und der Lesescanner ist nicht für große Längen ausgelegt. So würde ein solcher langer Code aussehen:

Um einen Plessey- oder MSI-Code hinzuzufügen, klicken Sie auf das Barcode-Symbol in der Symbolleiste des Berichtsdesigners .



Nachdem Sie den Barcodetyp ausgewählt haben, können Sie das Objekt auf der Berichtsseite platzieren. Um den Barcode zu bearbeiten, doppelklicken Sie auf das Objekt, oder rufen Sie das Kontextmenü auf und wählen Sie "Bearbeiten..." aus.

GS1-128

Dieser Barcode ist auch als EAN-128, UCC-128 bekannt. Es handelt sich um einen eindimensionalen linearen Barcode, der sowohl Ziffern als auch Buchstaben codieren kann. Es wird vor allem in der Logistikbranche zur Kennzeichnung von Paketen verwendet, findet aber auch in anderen Bereichen Anwendung. Der GS1 Logistic Label Standard beschreibt ein Transportetikett, das den GS1-128 Code verwendet.

Um den Code aufzubauen werden die Datenbezeichner von Code128 genutzt. Diese bezeichnen den Informationstyp und werden in Klammern angegeben, zum Beispiel «(37)». Wenn man den Datentyp in Klammern weiß, ist es klar, wie die folgenden Daten zu interpretieren sind.

In Klammern steht zuerst der Code des Bezeichners, dann sein Wert. Anschließend ohne Leerzeichen folgt der nächste Code des Bezeichners und sein Wert. Der Code hat also einen Wert in der Form:

```
(01)12646846874672(10)ABC11(15)100420
```

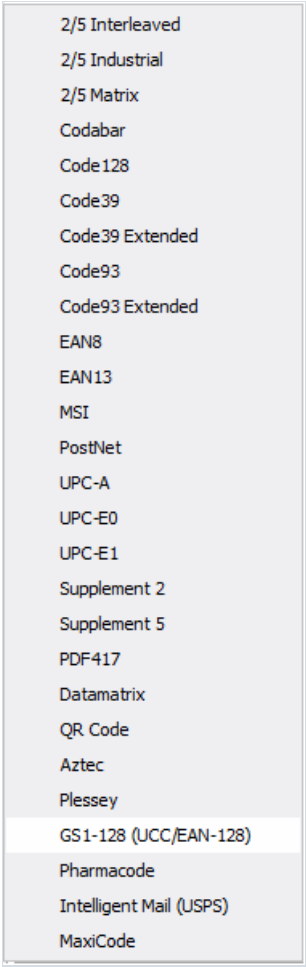
Die Länge des Codes ist auf 48 Zeichen ohne Klammern begrenzt.

Die Liste der unterstützten Datenbezeichner ist recht lang, hier ein Beispiel für einige davon:

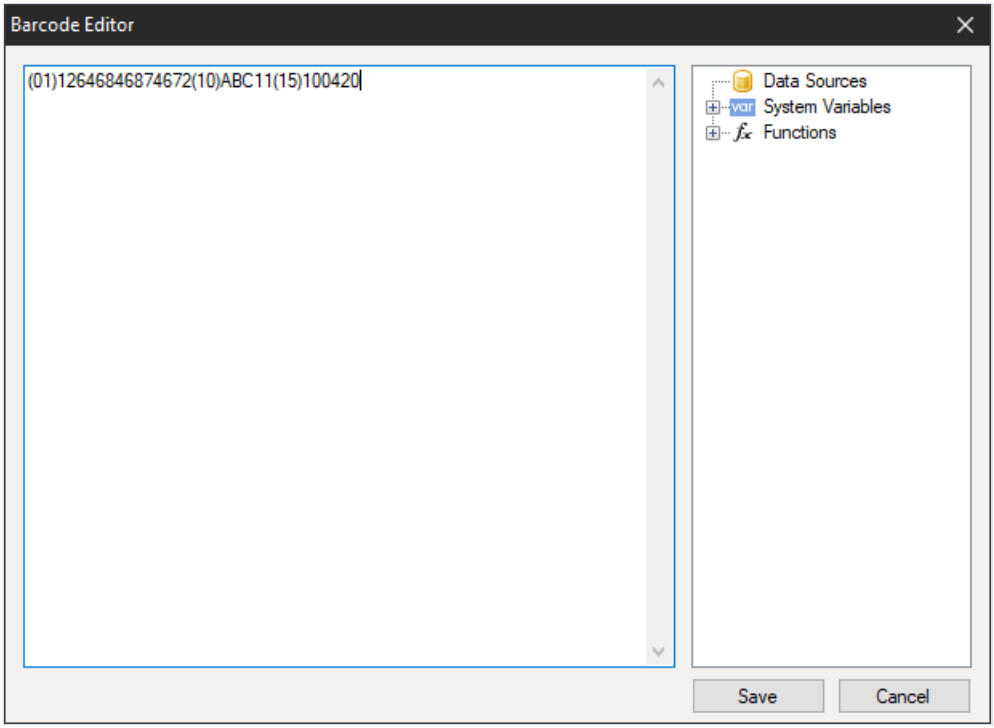
Code	Bezeichnung	Anzahl der Zeichen
02	GTIN-Nummer der Verpackungseinheit	14
11	Production Date	6
12	Due Date	6
13	Pack Date	6
15	ExpirationDate	6
10	Batch No	bis 20
00	Serial Shipping Container Code	bis 20
37	Number of Units	bis 8

Wenn Sie den obigen Beispielcode beim Generieren in FastReport .NET verwenden, erhalten Sie:

Um der Berichtsseite einen GS1-128-Code hinzuzufügen, klicken Sie auf das Barcode-Symbol in der Seitenleiste des Berichtsdesigners :



Um den hinzugefügten Barcode zu bearbeiten, doppelklicken Sie auf ihn oder rufen Sie das Kontextmenü auf und wählen Sie "Bearbeiten..." aus.



Intelligent Mail

Der IMb (Intelligent Mail barcode) (USPS OneCode) ist ein höhenmodulierter Barcode, der bis zu 31 Dezimalstellen von Poststückdaten in 65 vertikale Striche codiert. Der Code besteht aus vier verschiedenen Symbolen. Diese Symbole verwenden vier verschiedene Arten von Strichen, wodurch mehr Informationen in einem einzigen Barcode codiert werden können.

Der Intelligent-Mail-Standard basiert auf den POSTNET- und PLANET-Standards, die zuvor in den Vereinigten Staaten auf Post verwendet wurden.

Diese Standards ermöglichen die Codierung von Postleitzahlen und werden hauptsächlich für die Sortierung und Verfolgung von Postsendungen verwendet.

POSTNET codiert eine fünfstellige Postleitzahl, einen vierstelligen Plus-Code und einen zweistelligen Lieferstellencode.

POSTNET-Barcodes haben variable Längen von 32 bis 62 Strichen und PLANET-Barcodes haben 62 oder 72 Striche. Sie sind in der Höhe moduliert (vertikale Balken haben unterschiedliche Längen) und haben zwei Zustände (es gibt zwei verschiedene Arten von Strichen, kurze und hohe). Jede Ziffer der codierten Daten wird durch eine Gruppe von fünf Strichen dargestellt. POSTNET-Barcodes haben genau zwei volle Striche in jeder Fünfergruppe und PLANET-Barcodes haben drei.

Die IMb-Technologie vereint die Funktionalität von PLANET und POSTNET in einem einzigen Barcode. Er ermöglicht es den Postbenutzern, mit einem einzigen Barcode an mehreren Postdiensten gleichzeitig teilzunehmen, verbessert die Möglichkeiten der Postbenutzer, einzelne Sendungen zu verfolgen, und bietet einen besseren Einblick in den Postfluss. Die Verwendung dieses Barcodes ermöglicht es der Post, der Postbranche mehrere Dienstleistungen und zusätzliche Funktionen für die Sendungsverfolgung anzubieten sowie die Leistung der Post zu verfolgen und die Kosten zu senken. Wie POSTNET verfügt auch der IMb über eine Prüfsumme, um die Integrität des Codes und die mögliche Wiederherstellung von beschädigtem Code zu überprüfen.

Im Vergleich zu POSTNET hat der IMb eine viel höhere Datenkapazität (31 Zeichen gegenüber 11). Neben dem Routing-Code enthält der Intelligent Mail Barcode vier weitere Felder: Barcode-Kennung, Diensttypkennung (STID), Mailer-ID (MID) und Sequenznummer. Diese zusätzlichen Felder ermöglichen es den Postbenutzern, die Klasse der Post zu definieren, die gewünschten Dienste zu identifizieren (z.B.

Adressverfolgung und -korrektur) sowie den Postkunden, Sendungen eindeutig zu identifizieren. Die Nachteile sind die relativ große Codelänge und die Möglichkeit, nur Ziffern zu codieren.

Mit dem Intelligent Mail können die folgenden Informationen codiert werden:

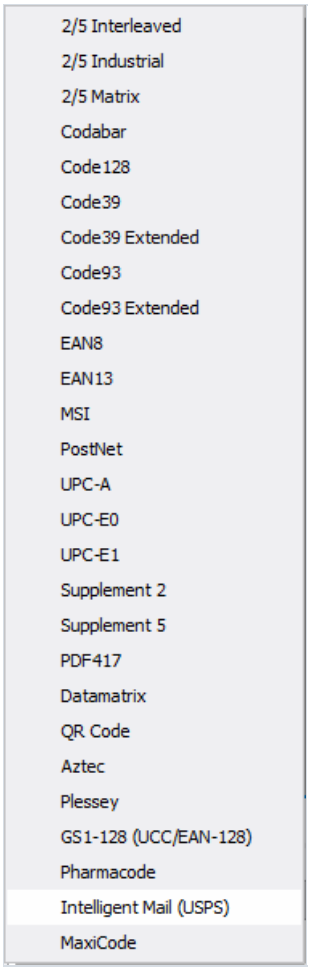
Information	Beschreibung
Barcode-Kennung (2 Zeichen)	wird vom Postdienst der Vereinigten Staaten zugewiesen
Diensttypkennung (3 Zeichen)	Klasse der Post oder andere Dienste

Information	Beschreibung
Mailer -ID (6 oder 9 Zeichen)	identifiziert das spezifische Unternehmen, das die Postsendung versendet, wird vom Postdienst der Vereinigten Staaten zugewiesen
Sequenznummer (6 oder 9 Zeichen)	Eine vom Absender zugewiesene ID, um den spezifischen Empfänger oder Haushalt zu identifizieren
Postleitzahl der Lieferstelle (11 Zeichen)	Dieser Abschnitt des Codes kann weggelassen werden

Striche haben unterschiedliche Höhe, Richtung (aufwärts, abwärts), Dicke.

Solche Barcodes können nicht nur auf Umschläge, sondern auch direkt auf Dokumente gedruckt werden, um sie durch einen Fensterumschlag anzuzeigen.

Um der Berichtsseite einen Intelligent Mail Code hinzuzufügen, klicken Sie im Berichtsdesigner auf das Barcode-Symbol in der linken Symbolleiste und wählen Sie den Intelligent Mail aus:



Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt oder rufen Sie das Kontextmenü auf und wählen Sie "Bearbeiten..." aus.

In FastReport beträgt die Mindestcodelänge 20 Zeichen. Dies liegt daran, dass die Postleitzahl der Zustellstelle mit dem optionalen 21. Zeichen beginnt.

Das Ändern einer einzigen Ziffer verändert den Barcode vollständig:

12345678933439877398 74

PDF417

Es handelt sich um einen gängigen zweidimensionalen Barcode zur Codierung großer Datenmengen. Er steht für Portable Data File, und 417 ist das Ergebnis der Addition von 4 und 17. 4 steht für vier Striche und vier Zwischenräume, und 17 steht für die Anzahl der Module im Codewort.

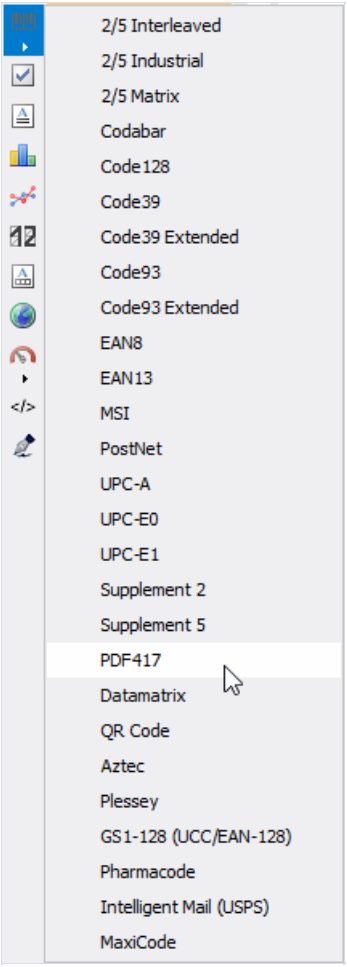
Wie bereits erwähnt, ist der PDF417 neben dem QR Code, Maxi Code und Data Matrix ein häufig verwendeter Barcode. PDF417 sieht wie eine Mischung aus einem klassischen linearen Barcode und einem Matrix-Barcode aus, ist aber in Wirklichkeit ein komplexer linearer Code. An den Rändern hat er die gleichen Striche wie ein linearer Code und in der Mitte sind die Striche in Reihen übereinander angeordnet. Diese Verdichtung des linearen Codes wird dadurch erreicht, dass die Codezeilen übereinander angeordnet werden. Dadurch können große Mengen an Informationen gespeichert werden, von 3 bis 90 Zeilen, in denen bis zu 1.859 alphabetische Zeichen oder 2.725 numerische Zeichen codiert werden können.

Der Schutz vor Codebeschädigung sorgt für die Redundanz, die bis zu 50 % des Codes abdecken kann. Dies ist ein sehr hoher Wert, allerdings wächst auch die Größe des Codes. Im Vergleich zu Matrixcodes benötigt PDF417um ein Vielfaches mehr Platz, um die gleiche Menge an Informationen zu codieren, was als Nachteil angesehen werden kann.

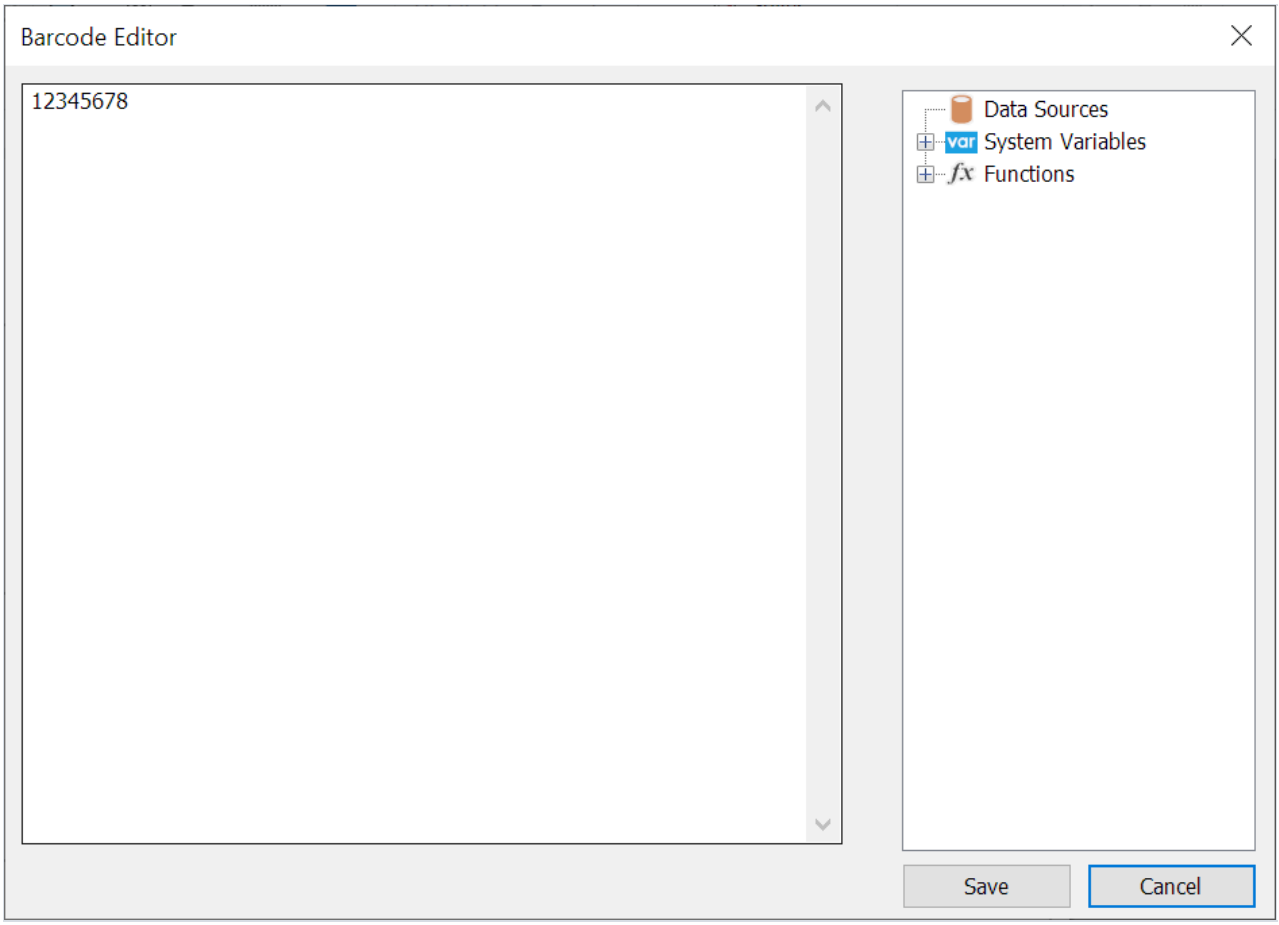
Das Anwendungsspektrum von PDF417 ist sehr umfangreich. Es wird verwendet von: Verkehrsunternehmen zum Bedrucken von Fahrkarten und Frachtsendungen, in Postsendungen, Berichtsunterlagen, verschiedenen Ausweisen, im Lager, und in vielen anderen Bereichen, in denen eine Kennzeichnung und Identifizierung erforderlich sind.

Zum Lesen dieses Codes werden Laserscanner verwendet, die sich von den herkömmlichen linearen Codescannern geringfügig unterscheiden. Wie oben erwähnt, befinden sich an den Seiten des Codes die üblichen Striche, die für lineare Barcodes charakteristisch sind. Sie werden benötigt, um den Anfang und das Ende des Codes zu identifizieren.

Um der Berichtsseite einen PDF417-Barcode hinzuzufügen, wählen Sie in der Seitenleiste das Barcode-Objekt aus. Wählen Sie in der Dropdown-Liste PDF417 aus:



Sie können nun den Barcode auf der Berichtsseite platzieren. Um ihn zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt oder rufen Sie das Kontextmenü auf und wählen Sie "Bearbeiten..." aus. Danach wird das Barcode-Bearbeitungsfenster angezeigt:



Im Eigenschafteninspektor des hinzugefügten Barcode-Objekts können Sie die Textanzeige unter dem Code mithilfe der ShowText-Eigenschaft deaktivieren.

EAN – 8

Der EAN-8-Code (European Article Number), auch als GTIN-8 bekannt, stellt einen kurzen 8-stelligen Barcode. Die Ziffern sind in 2 Blöcke zu je 4 Ziffern unterteilt. Die ersten 2 Ziffern bezeichnen das Ursprungsland der Ware, dann folgen 5 Ziffern, die Codeinformationen bezeichnen, und die letzte Ziffer ist eine Prüfsumme zur Überprüfung der Datenintegrität.

Dieser Code wurde von EAN-13 abgeleitet, indem der Code um 8 Ziffern reduziert wurde. Dadurch wurde es möglich, den Code auf kleinen Warenverpackungen anzubringen. Aufgrund der Länderkennung im Code handelt es sich um einen internationalen Code.

Der Anwendungsbereich des EAN-8-Codes ist die Kennzeichnung von Waren und Geräten.

Wie wird der Code aufgebaut? Jede Ziffer wird mit 7 Modulen (senkrechte Striche und Zwischenräume) codiert. Nehmen wir an, 0 ist ein Zwischenraum und 1 ist ein Strich. Die Codierung der Ziffern kann dann wie folgt dargestellt werden:

- 0 – 0001101;
- 1 – 0011001;
- 2 – 0010011;
- 3 – 0111101;
- 4 – 0100011;
- 5 – 0110001;
- 6 – 0101111;
- 7 – 0111011;
- 8 – 0110111;
- 9 – 0001011.

Ein Beispiel dazu:

Zur Erinnerung: Der Code beginnt mit dem Code 101. Darauf folgt der Code für die Ziffer 3 und so weiter. Wie bereits erwähnt, ist der Code in zwei Teile unterteilt. Das Trennzeichen sowie der Anfang bzw. das Ende des Codes haben eine größere Länge der Striche. Es wird durch den Code 01010 dargestellt.

Wenn Sie genau hinsehen, werden Sie feststellen, dass die Striche nach dem Trennzeichen nicht mit den Ziffern in der obigen Liste übereinstimmen. Dies liegt daran, dass für den zweiten Teil eine andere Codierung verwendet wird:

· 0 – 1110010;

· 1 – 1100110;

· 2 – 1101100;

· 3 – 1000010;

· 4 – 1011100;

· 5 – 1001110;

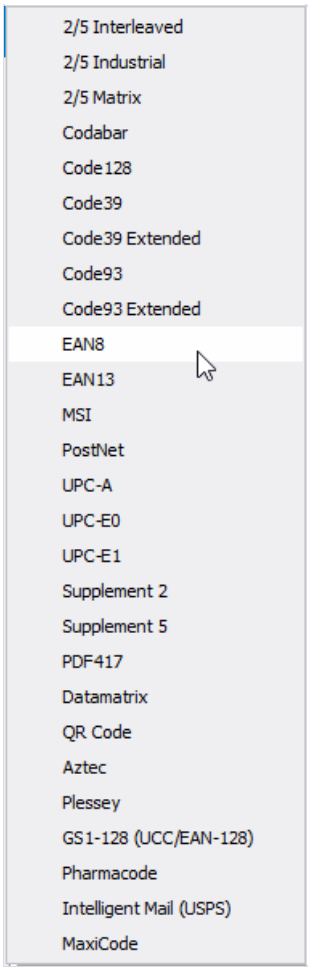
· 6 – 1010000;

· 7 – 1000100;

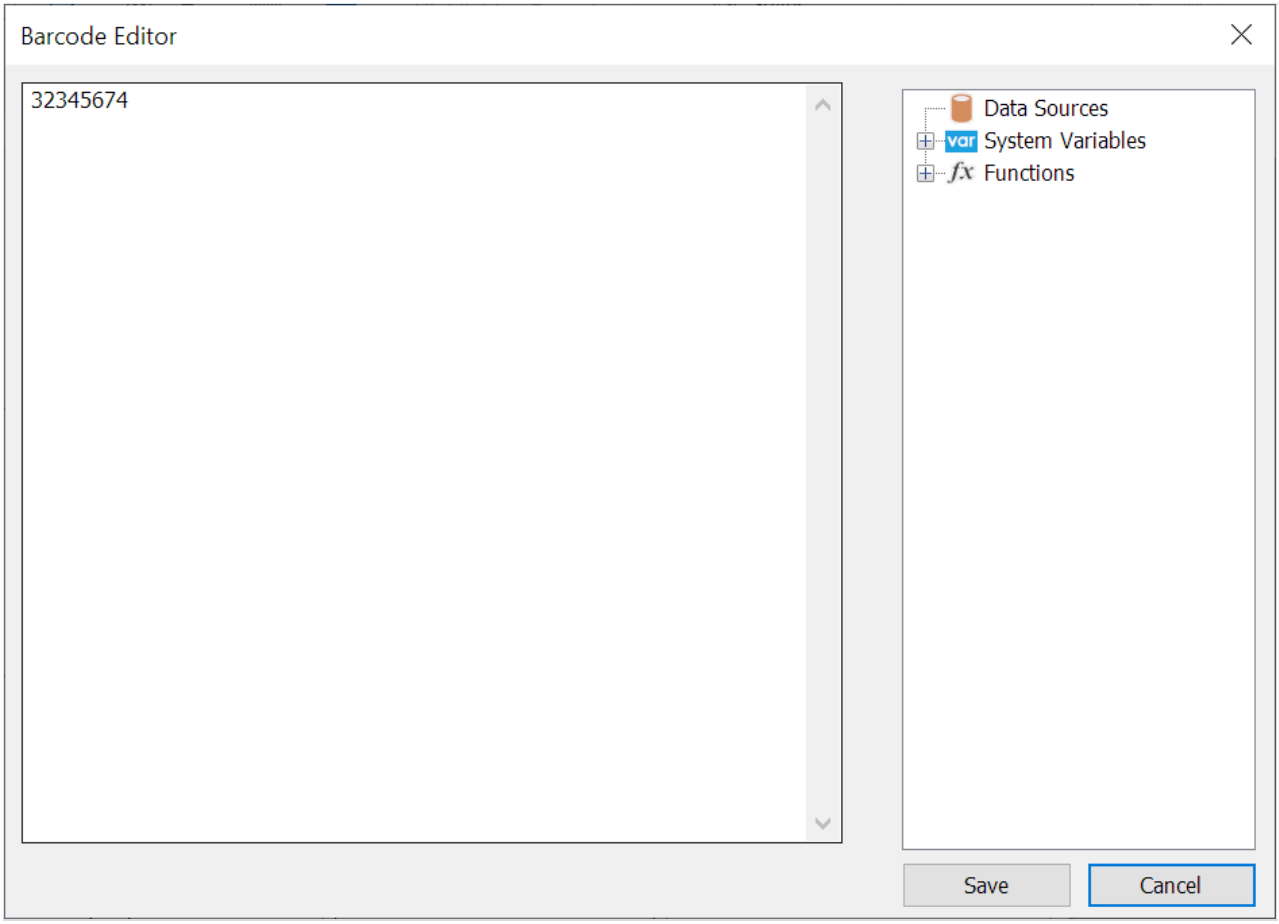
· 8 – 1001000;

· 9 – 1110100.

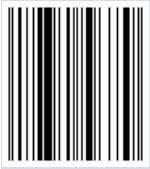
Um der Berichtsseite einen EAN-8-Barcode hinzuzufügen, verwenden Sie das Barcode-Objekt in der seitlichen Symbolleiste des Berichtsdesigners und wählen den Code aus:



Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt oder wählen Sie "Bearbeiten..." im Kontextmenü aus:



Sie können die Anzeige von Ziffern unter dem Code mit der ShowText-Eigenschaft im Eigenschafteninspektor des Barcode-Objekts deaktivieren:



EAN-13

Der EAN-13-Barcode (European Article Number) ist der am weitesten verbreitete unter den linearen Codes. Mit diesem Zeichensystem können 12 Ziffern codiert werden, die letzte 13. Ziffer des Codes ist eine Prüfsumme zur Überprüfung der Integrität des Codes. Nur Ziffern können codiert werden.

Der Code wird wie folgt aufgebaut:

- Die ersten 2-3 Ziffern des Codes sind für die Codierung des Ursprungslandes der Waren bestimmt;
- Weitere 4-5 Ziffern sind für die Codierung des Produktherstellers bestimmt;
- Die restlichen 3-5 Ziffern sind die Artikelnummer des Produktes.

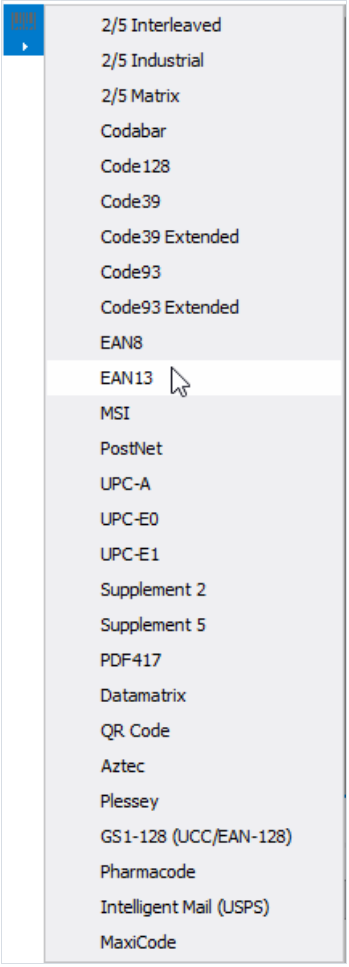
Wie bereits erwähnt, ist die 13. Ziffer eine Prüfziffer. Sie wird automatisch auf der Grundlage der vorherigen 12 Ziffern nach einem speziellen Algorithmus berechnet. Diese Ziffer wird verwendet, um festzustellen, ob der Code richtig gelesen wird.

Wenn Sie die Abbildung des Codes betrachten, werden Sie feststellen, dass am Anfang und am Ende des Codes zwei Striche stehen. Stellt man den Code als Binärfolge dar, wobei der Strich 1 und das Zwischenraum 0 ist, dann sieht diese Strichfolge wie 101 aus. Außerdem wird der Barcode genau in der Mitte durch dieselbe Folge in zwei Teile geteilt.

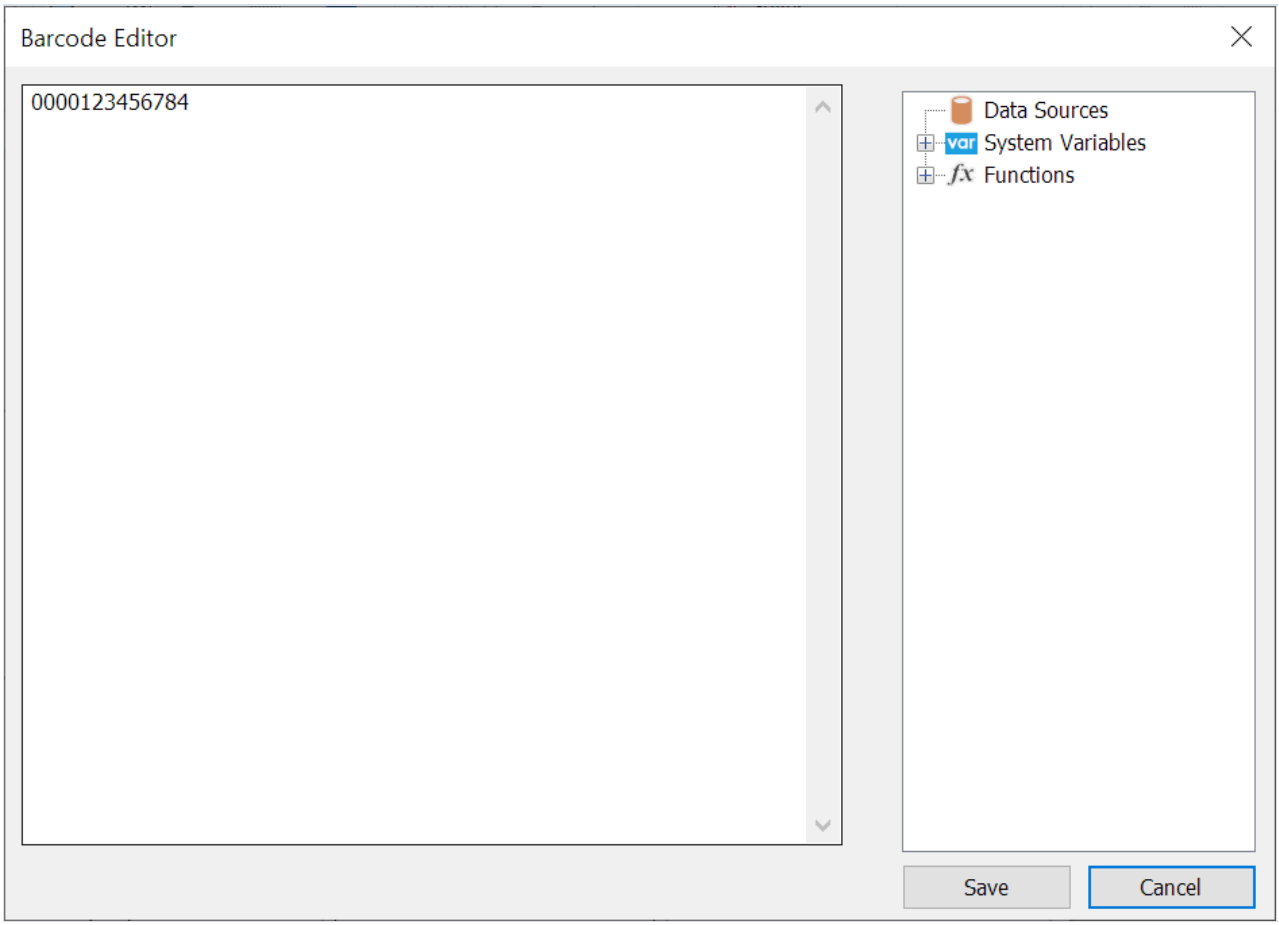
Jede Ziffer wird in 7 Modulen (Striche und Zwischenräume) codiert. Es gibt drei spezielle Tabellen mit Codefolgen für Ziffern, wobei in der ersten und zweiten Hälfte des Codes unterschiedliche Versionen dieser Tabellen verwendet werden.

Dieser Code kann in beliebiger Reihenfolge von links nach rechts oder von rechts nach links gelesen werden. Die Einfachheit des Codes und seine leichte Lesbarkeit machten ihn bei Einzelhändlern schnell zu einem beliebten Mittel zur Kennzeichnung von Waren.

Zu den Nachteilen dieses Codes gehört häufig seine geringe Kapazität, die Länge der codierten Folge. Um der Berichtsseite einen EAN-8-Barcode hinzuzufügen, verwenden Sie das Barcode-Objekt in der seitlichen Symbolleiste des Berichtsdesigners und wählen Sie den Code aus:

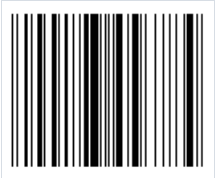


Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt oder wählen Sie "Bearbeiten..." im Kontextmenü aus:



Sie können die Anzeige von Ziffern unter dem Code mit der ShowText-Eigenschaft im Eigenschafteninspektor des

Barcode-Objekts deaktivieren:



UPC-A

Der Universal Product Code ist ein linearer Barcode, der die Codierung von 12 Zeichen ermöglicht. Er wurde 1973 gemeinsam vom Uniform Grocery Product Code Council und IBM entwickelt. Er ähnelt vom Aufbau und Zweck her dem bekannten EAN-13-Code. Die UPC-Codes wurden für Nordamerika und die EAN-Codes für Europa entwickelt.

Aufbau des Codes:

- Start-Zeichen, das den Anfang des Codes bezeichnet;
- Präfix, das den Produkttyp bezeichnet, 1 Zeichen;
- Herstellercode - 5 Zeichen;
- Produktcode - 5 Zeichen;
- Prüffziffer - 1 Zeichen, wird aus den vorangegangenen 11 Ziffern nach dem Modulo-10-Berechnungsverfahren berechnet;
- Stopp- Zeichen, das das Ende des Codes bezeichnet.

Vor und nach dem Code muss ein Zwischenraum von etwa 9 Modulen vorhanden sein. Dies soll die Erkennung des Codes durch den Scanner gewährleisten.

Die Codelänge beträgt 12 Zeichen. Der Code ist 12 Zeichen lang, von denen nur 11 codierte Daten sind und ein weiteres eine Prüfziffer ist.

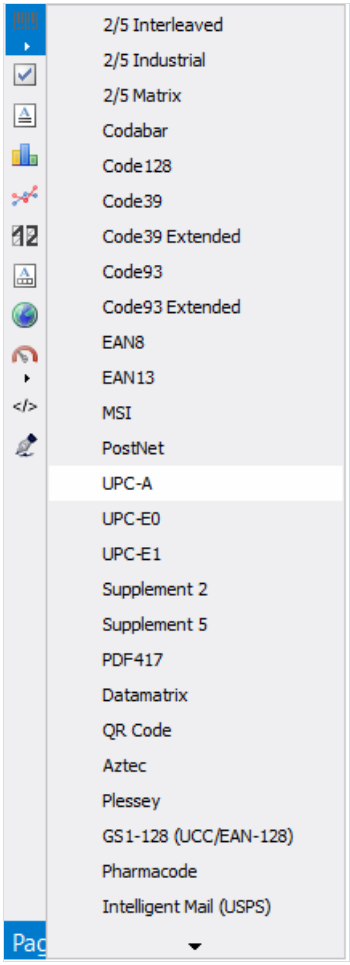
Jedes Zeichen wird mit zwei Strichen und zwei Zwischenräumen codiert. Ein Strich oder Zwischenraum kann 1, 2, 3 oder 4 Module breit sein (ein Modul entspricht 0,33 mm).

Anwendung. Der UPC-A-Barcode ist in den USA und Kanada weit verbreitet. Er wird in Supermärkten zur Kennzeichnung von Produkten verwendet.

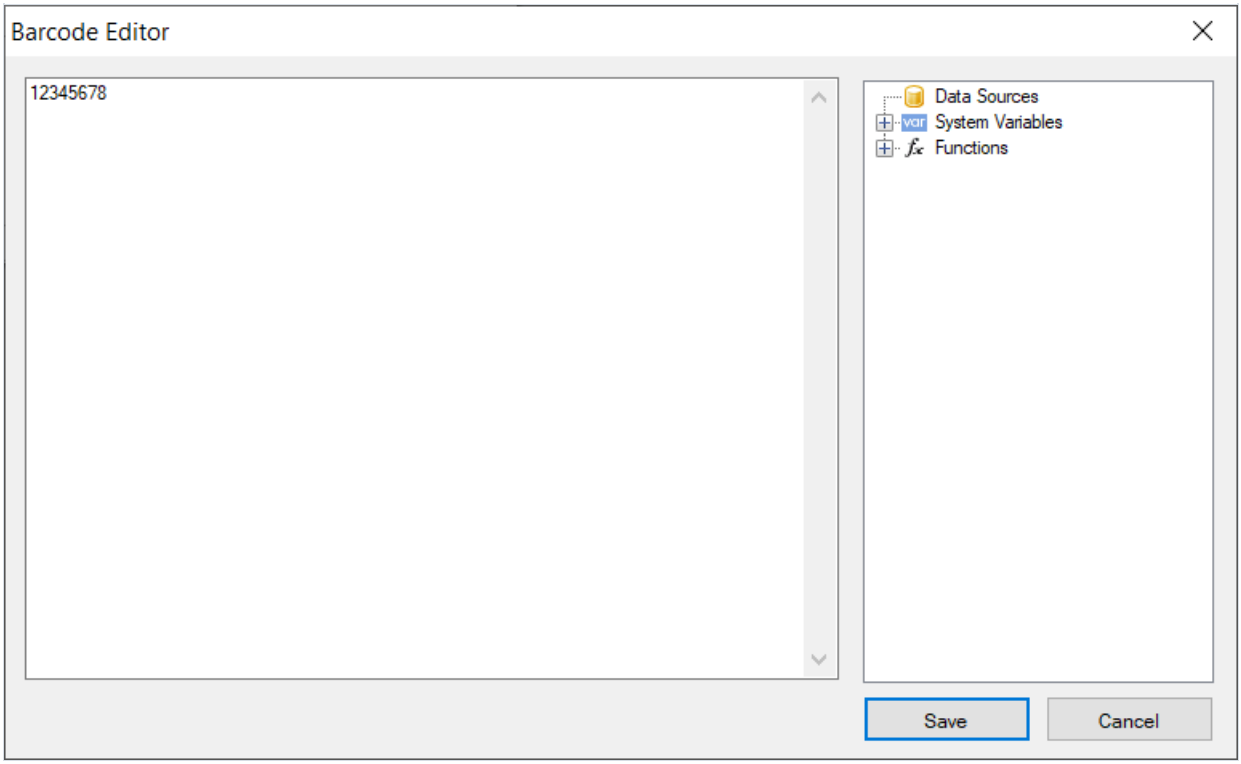
Dieser Barcode hat aufgrund seiner kompakten Größe, der einfachen Lesbarkeit durch Scanner und der Prüfziffer zum Schutz vor Lesefehlern an Beliebtheit gewonnen.

Zu den Nachteilen des Codes gehören seine Fähigkeit, nur Ziffern zu codieren, und seine geringe Größe, die seinen Anwendungsbereich begrenzen.

Um der Berichtsseite einen UPC-A-Barcode hinzuzufügen, klicken Sie auf das Barcode-Symbol in der seitlichen Symbolleiste des Berichtsdesigners und wählen Sie UPC-A aus:



Um einen hinzugefügten Barcode zu bearbeiten, doppelklicken Sie darauf oder rufen Sie das Kontextmenü mit einem Rechtsklick auf und wählen Sie Bearbeiten... aus:



Alle Barcode-Objekte haben eine Reihe von Eigenschaften. Hier sind die gebräuchlichsten:

● Angle – Mit dieser Eigenschaft können Sie die Drehung eines Objekts auf einen der festen Winkel - 0, 90, 180, 270 Grad - einstellen.

● Zoom – Die Eigenschaft legt die Skalierung des Barcodes fest. Diese Eigenschaft wird nur in Verbindung mit der AutoSize-Eigenschaft verwendet.

● AutoSize – Wenn diese Eigenschaft aktiviert ist, wird das Objekt so gestreckt, dass der gesamte Barcode angezeigt wird. Wenn sie deaktiviert ist, wird der Barcode auf die Größe des Objekts gestreckt.

● ShowText – Die Eigenschaft bestimmt, ob der Text unter dem Barcode angezeigt wird.

● DataColumn – Datenfeld, aus dem der Objekttext geladen werden soll.

● Expression – Ausdruck, der den Objekttext zurückgibt.

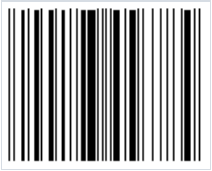
● Text – Objekttext.

● Padding – Mit dieser Eigenschaft können Sie Einzüge von den Rändern des Objekts in Pixel festlegen.

● WideBarRatio – Diese Eigenschaft ist bei allen linearen Barcodes vorhanden. Sie bestimmt die relative Größe der breiten Balken des Barcodes.

● CalcCheckSum – Viele lineare Barcodes verfügen über diese Eigenschaft. Sie bestimmt, ob die Prüfsumme automatisch berechnet werden soll. Wenn diese Eigenschaft deaktiviert ist, muss die Prüfsumme im Objekttext vorhanden sein.

Wenn Sie die ShowText-Eigenschaft deaktivieren, sieht der Code wie folgt aus:



UPC-E

UPC-E ist ein linearer digitaler Barcode, der für die Codierung von Produktinformationen im Einzelhandel entwickelt wurde. Wie Sie wissen, basiert der UPC-Standard auf der EAN. Der amerikanische UPC-A ist nämlich ein Analog zur europäischen EAN-13. Der Unterschied besteht nur in der Länge des Codes - 12 Zeichen gegenüber 13. Die EAN-13 ist eine verkürzte Version von EAN-8. Der Code wurde mit dem Ziel entwickelt, den Code klein genug zu machen, damit er auf kleine Verpackungen passt. Der UPC-A hat auch eine "Lite"-Version von UPC-E mit 6 Zeichen. Der in UPC-A verwendete Produkt- bzw. Warencode hat in der Regel viele Nullen. Durch die Entfernung von Nullen aus diesen Codes konnte die Länge des Barcodes von 12 auf 6 Zeichen gekürzt werden, ohne dass Informationen verloren gingen.

Der Code wird wie folgt aufgebaut:

- Das erste Präfix-Zeichen gibt ein Zahlensystem 0 oder 1 an, das bestimmt, wie der Code interpretiert wird;
- Danach folgen die 6 Zeichen des Hersteller- und Produktcodes (getrennt in UPC-A);
- Das letzte Zeichen ist eine Prüfziffer, um die Integrität des Codes zu überprüfen.

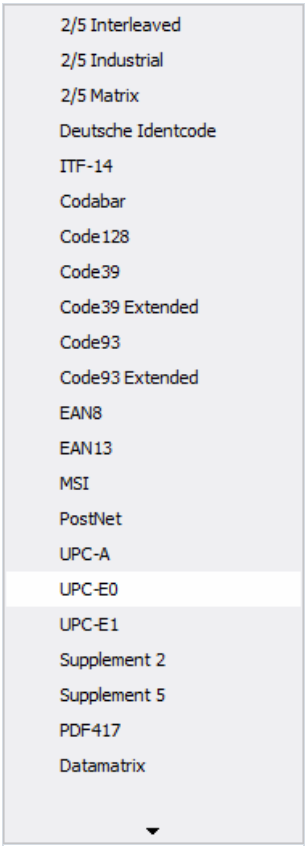
Mit dem UPC-E-Barcode können 6 bis 8 Zeichen codiert werden. Bei der Codierung von 6 Zeichen werden das erste Zeichen, das das Zahlensystem darstellt, und das letzte Zeichen, das die Prüfziffer darstellt, ausgeschlossen. Bei der Codierung von 7 Zeichen wird nur die Prüfziffer ausgeschlossen. Und der 8-stellige Code enthält den gesamten oben beschriebenen Aufbau. Wie Sie sehen können, enthält die verkürzte UPC Version kein Start- und Stopp-Zeichen, dadurch wird der kompakte Aufbau erreicht.

Je nach gewähltem Zahlensystem wird die Prüfziffer automatisch berechnet. Somit wird der UPC-E-Standard in UPC-E0 und UPC-E1 unterteilt.

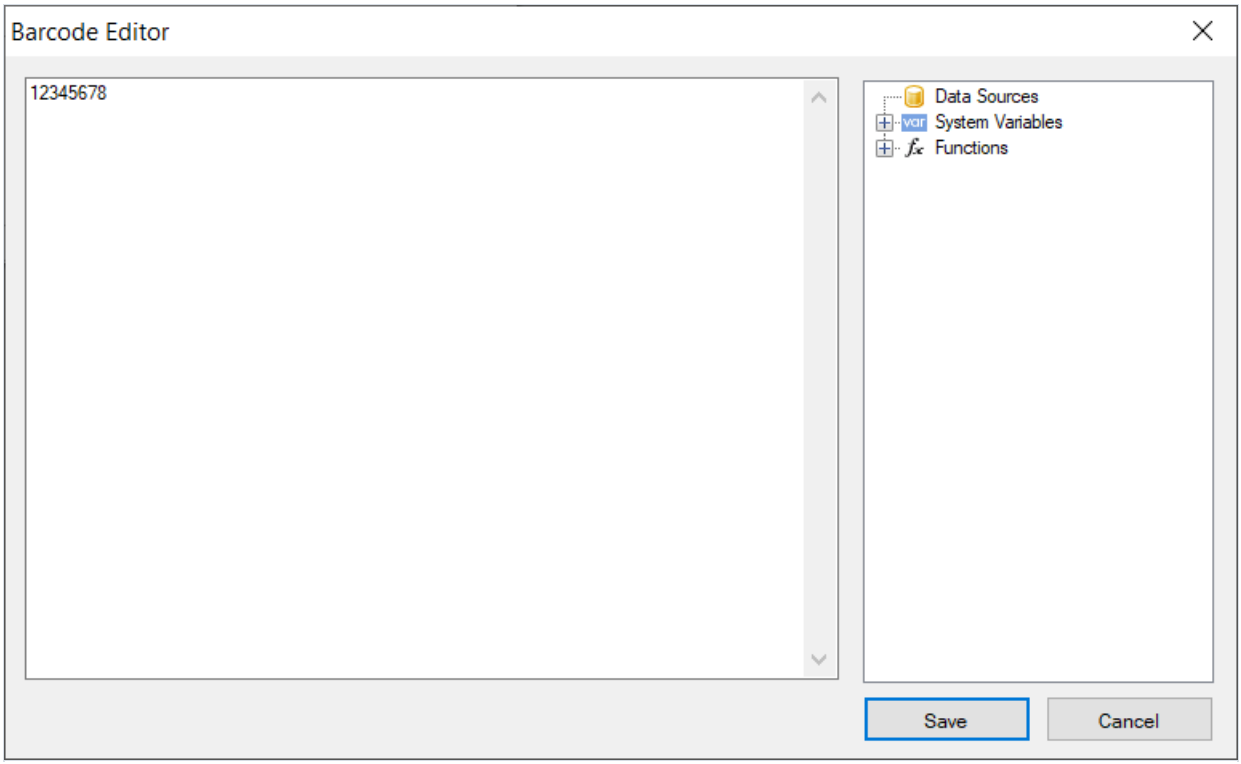
Es ist wichtig zu beachten, dass mit dem UPC-E nur GTIN-12-Strukturen mit einem Start-Zeichen 0 und einer Folge von Nullen im Code codiert werden können. Diese Folge von Nullen wird im UPC-E-Code schließlich abgebaut.

Wie beim UPC-A wird jedes Datenzeichen beim UPC-E mit zwei Strichen und zwei Zwischenräumen codiert. Der Strich oder Zwischenraum kann 1, 2, 3 oder 4 Module breit sein (ein Modul entspricht 0,33 mm).

Um der Berichtsseite einen UPC-E0- oder UPC-E1-Barcode hinzuzufügen, klicken Sie auf das Barcode-Symbol in der seitlichen Symbolleiste des Berichtsdesigners und wählen Sie den gewünschten Barcode aus:



Um den hinzugefügten Barcode zu bearbeiten, doppelklicken Sie darauf oder rufen Sie das Kontextmenü mit einem Rechtsklick auf und wählen Sie Bearbeiten... aus:



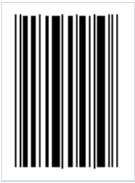
Alle Barcode-Objekte haben eine Reihe von Eigenschaften. Hier sind die gebräuchlichsten:

- Angle – Mit dieser Eigenschaft können Sie die Drehung eines Objekts auf einen der festen Winkel - 0, 90, 180, 270 Grad - einstellen.
- Zoom – Die Eigenschaft legt die Skalierung des Barcodes fest. Diese Eigenschaft wird nur in Verbindung mit der AutoSize-Eigenschaft verwendet.
- AutoSize – Wenn diese Eigenschaft aktiviert ist, wird das Objekt so gestreckt, dass der gesamte Barcode

angezeigt wird. Wenn sie deaktiviert ist, wird der Barcode auf die Größe des Objekts gestreckt.

- ShowText – Die Eigenschaft bestimmt, ob der Text unter dem Barcode angezeigt wird.
- DataColumn – Datenfeld, aus dem der Objekttext geladen werden soll.
- Expression – Ausdruck, der den Objekttext zurückgibt.
- Text – Objekttext.
- Padding – Mit dieser Eigenschaft können Sie Einzüge von den Rändern des Objekts in Pixel festlegen.
- WideBarRatio – Diese Eigenschaft ist bei allen linearen Barcodes vorhanden. Sie bestimmt die relative Größe der breiten Balken des Barcodes.
- CalcCheckSum – Viele lineare Barcodes verfügen über diese Eigenschaft. Sie bestimmt, ob die Prüfsumme automatisch berechnet werden soll. Wenn diese Eigenschaft deaktiviert ist, muss die Prüfsumme im Objekttext vorhanden sein.

Wenn Sie die ShowText-Eigenschaft deaktivieren, sieht der Code wie folgt aus:



MSI

Der lineare MSI Barcode wurde 1971 auf der Grundlage des Original-Plessey-Code entwickelt und ist im Wesentlichen eine verbesserte Version davon. Der Code kann nur die Ziffern 0-9 darstellen. Der Code wird wie folgt aufgebaut:

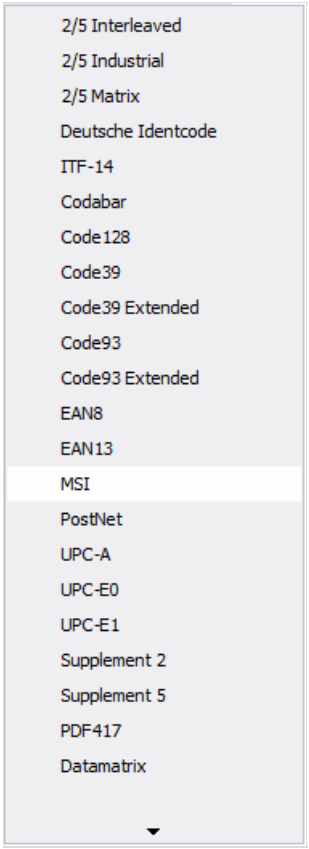
- Start-Zeichen, das den Beginn der Datenlesung bezeichnet (110);
- Daten;
- Optionales Prüfzeichen;
- Stopp- Zeichen, das das Ende der Datenlesung bezeichnet (1001).

Jedes Zeichen des Codes wird in binärer Form durch einen Strich und einen Zwischenraum dargestellt. Ein Strich ist 1 und ein Zwischenraum ist 0.

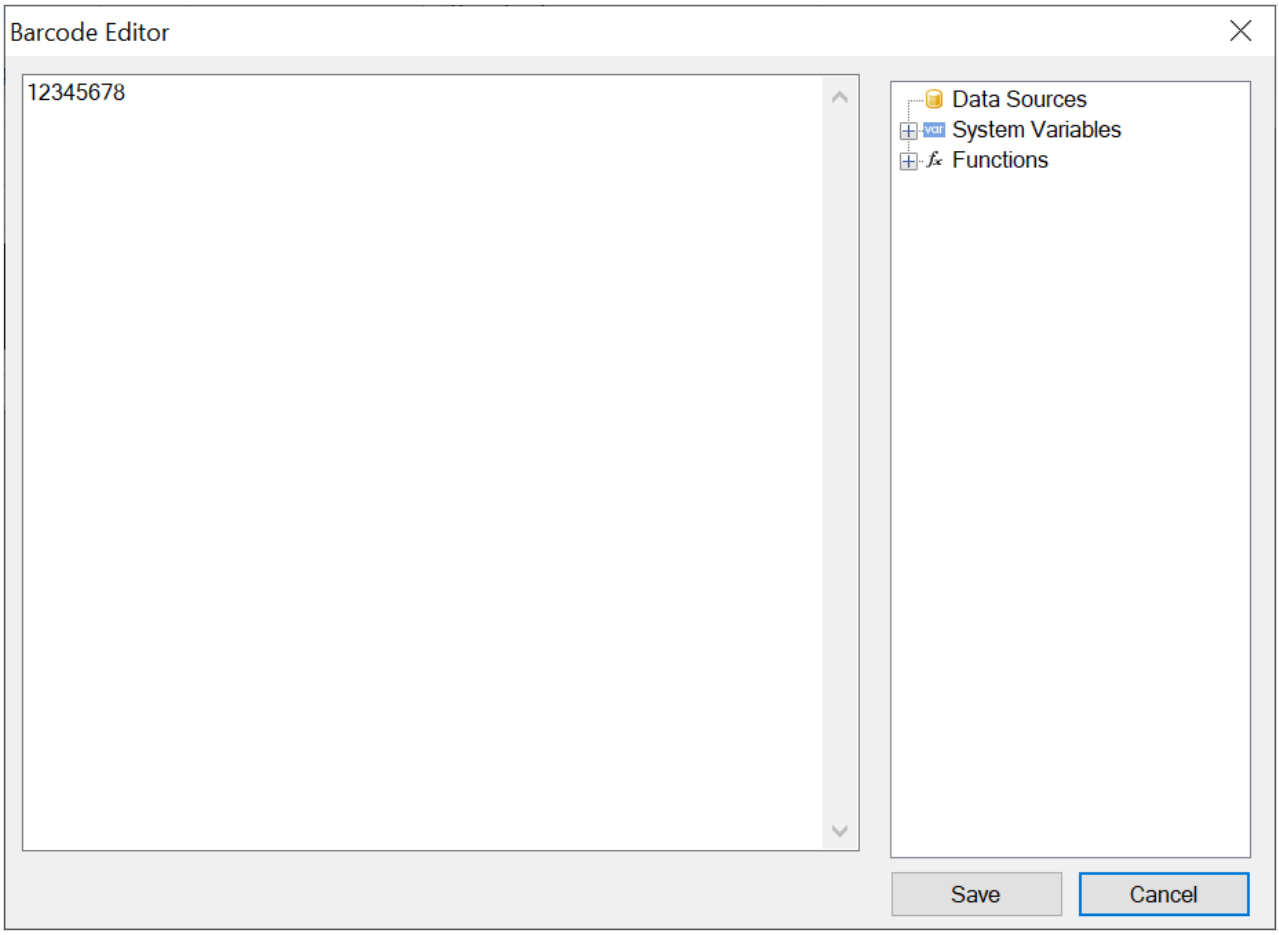
Das Prüfzeichen kann nach einem von vier Algorithmen berechnet werden: Modulo 10, Modulo 11, Modulo 1010, Modulo 1110. Der häufigste Algorithmus ist Modulo 10. Ein Prüfzeichen wird nach einem bestimmten Algorithmus berechnet. Die vom Scanner gelesenen Daten werden gemäß dem Algorithmus addiert und das Ergebnis wird mit dem Prüfzeichen verglichen. Ist das Ergebnis positiv, gilt der Code als korrekt gelesen.

Der MSI-Code hat keine feste Länge. Sie ist nur durch die Möglichkeiten des Scanners begrenzt. Heute ist dieser Barcodetyp veraltet und wird kaum noch verwendet. Früher wurde er zur Kennzeichnung von Waren in Lagerhäusern und Supermärkten verwendet.

Um der Berichtsseite einen MSI-Code hinzuzufügen, klicken Sie auf das Barcode-Symbol in der Seitenleiste des Berichtsdesigners :



Wählen Sie den MSI in der Dropdown-Liste aus. Fügen Sie den ausgewählten Barcode der Berichtsseite hinzu. Um den Barcode zu bearbeiten, doppelklicken Sie auf den hinzugefügten Barcode oder rufen Sie das Kontextmenü auf und wählen Sie Edit... aus:



Um den Text unter dem Code auszublenden, wählen Sie den hinzugefügten Barcode aus und setzen Sie die ShowText Eigenschaft auf False:

Supplement

Die in Europa etablierten eindimensionalen EAN-Barcodes und die UPC-Barcodes in den USA können um einen zusätzlichen Code erweitert werden. Er befindet sich rechts neben dem Hauptcode.

Mit diesem zusätzlichen Code können Sie 2 oder 5 Zahlzeichen codieren.

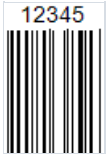
UPC-A, UPC-E, EAN-13 und EAN-8 können rechts neben dem Hauptbarcode einen zusätzlichen Barcode enthalten. Dieser zweite Barcode, der normalerweise nicht so hoch wie der Hauptbarcode ist, wird verwendet, um zusätzliche Informationen für Zeitungen, Bücher und andere Zeitschriften zu codieren.

Der zusätzliche Code kann 2- oder 5-stellige Informationen codieren.

Der zweistellige zusätzliche Code wird in der Regel zur Codierung von Informationen über die Heftnummer in Zeitschriften (Magazine, Zeitungen) verwendet.

Mit dem zusätzlichen Code kann die Heftnummer im Hauptbarcode weggelassen werden, so dass sie für alle Hefte unverändert bleibt. Es besteht also keine Notwendigkeit, den umfangreichen Code zu lesen, wenn nur Informationen zum Heft benötigt werden.

Der fünfstellige Zusatzcode wird häufig in den Büchern verwendet und enthält Informationen über die unverbindliche Preisempfehlung eines Buches. Von den fünf Zeichen steht das erste für den Währungscode und die restlichen 4 für den Preis.



Es gibt bestimmte Codes für die Kennzeichnung ohne Preis:

● Code 90000 - Das Buch hat keinen Einzelpreis;

● Code 99991 - Das Buch wird kostenlos verbreitet.

Der zusätzliche Code kann aber auch andere Informationen zur internen Verwendung im Verlag enthalten.

Jede Ziffer wird mit 7 Modulen (Striche und Zwischenräume) codiert.

Aufbau des Codes:

1. Start-Zeichen (1011, wenn wir eine Analogie zwischen Strichen und Ziffern ziehen);
2. Das erste Zeichen der Daten;
3. Trennzeichen (01);
4. Das zweite Zeichen der Daten oder 4 Zeichen im Falle eines 5-stelligen Codes.

Es gibt kein ausdrückliches Prüf- und Stoppzeichen. Nach dem Trennzeichen wird die zulässige Anzahl von Zeichen gelesen.

Die Daten werden unter Verwendung der in EAN verwendeten Sätze "links gerade" und "links ungerade" codiert.

Je nach Größe des zusätzlichen Barcodes werden unterschiedliche Paritätsmuster von Zeichen verwendet. Die Parität des Zeichens bestimmt die Prüfsumme.

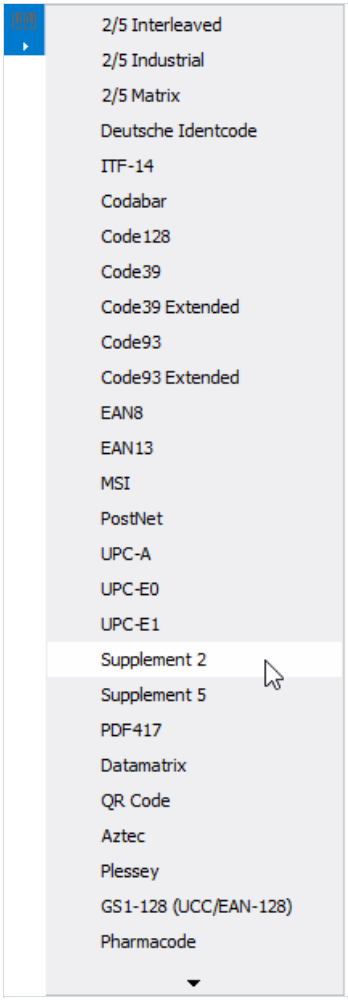
Algorithmus zur Paritätskodierung eines zweistelligen Codes.

Wenn der zusätzliche Barcode zweistellig ist, muss die zweistellige Ziffer, die sich aus der Addition der ersten und zweiten Ziffer ergibt, durch 4 geteilt werden. Wenn der Wert der Division mit Rest, eine gerade Zahl ist, wird das erste Zeichen gerade und das zweite ungerade codiert. Das bedeutet, dass ein gerader Wertesatz zur Codierung der ersten Ziffer und ein ungerader Wertesatz zur Codierung der zweiten Ziffer verwendet wird.

Beim Scannen wird nun die Parität des gelesenen Wertes ermittelt. Stimmt sie nicht mit der berechneten und erwarteten Parität überein, wird also der Barcode nicht korrekt gelesen.

Im Falle eines fünfstelligen Codes ist die Berechnung der Prüfsumme komplizierter. Es wird davon ausgegangen, dass die letzte Ziffer des Codes an einer ungeraden Stelle steht. Von der letzten Ziffer bis zur ersten Ziffer werden nacheinander die geraden und ungeraden Stellen vergeben. Dann wird die Summe aller ungeraden Ziffern genommen und mit 3 multipliziert. Die Summe aller geraden Ziffern wird mit 9 multipliziert. Dann wird das Ergebnis der Summe von beiden vorherigen Berechnungen genommen, d.h. die Ziffer ganz rechts. Dies ist die Prüfziffer, mit der das Paritätsmuster in der speziellen Tabelle bestimmt wird.

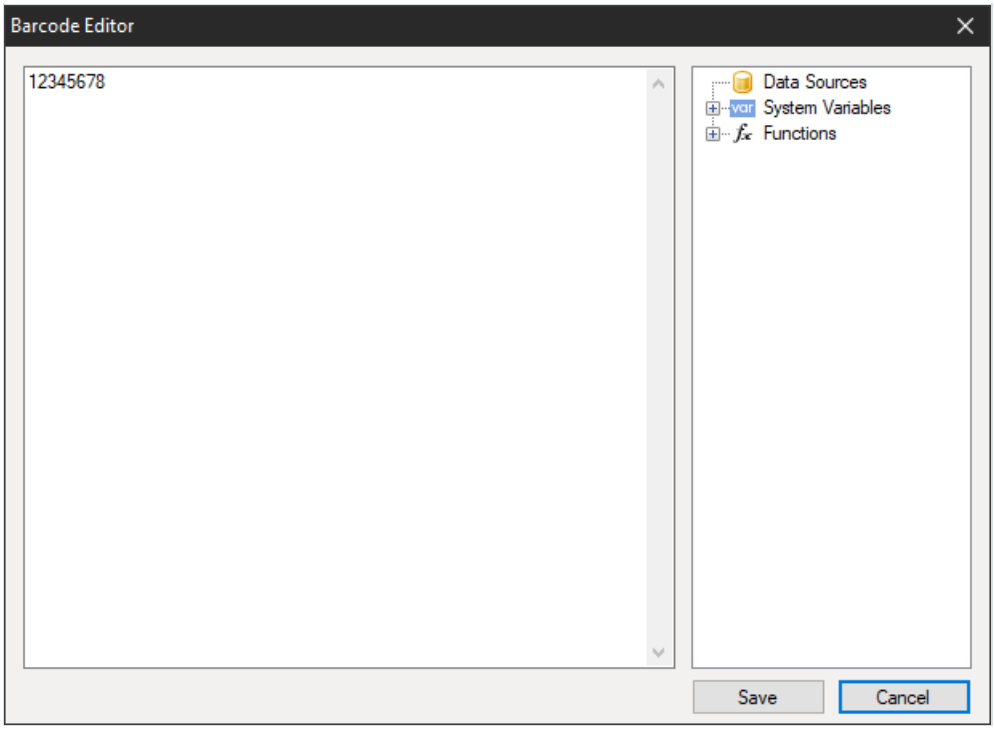
In FastReport .NET wird der Supplement-Code getrennt von EAN oder UPC hinzugefügt. Um der Berichtsseite einen Code hinzuzufügen, wählen Sie den Barcode in der seitlichen Symbolleiste des Berichtsdesigners aus:



Wählen Sie Supplement 2 für einen zweistelligen Code oder Supplement 5 für einen fünfstelligen Code aus und platzieren Sie ihn auf der Berichtsseite.

Platzieren Sie den zusätzlichen Code neben dem Hauptcode:

Um den Code zu bearbeiten, doppelklicken Sie darauf:



Wie alle Barcodes hat Supplement in FastReport.Net eine Reihe von Eigenschaften, die Sie im Eigenschafteninspektor des Objekts bearbeiten können:

● Angle – Angle – Mit dieser Eigenschaft können Sie die Drehung eines Objekts auf einen der festen Winkel - 0, 90, 180, 270 Grad - einstellen; ● Zoom – Die Eigenschaft legt die Skalierung des Barcodes fest. Diese Eigenschaft wird nur in Verbindung mit der AutoSize-Eigenschaft verwendet; ● AutoSize – Wenn diese Eigenschaft aktiviert ist, wird das Objekt so gestreckt, dass der gesamte Barcode angezeigt wird. Wenn sie deaktiviert ist, wird der Barcode auf die Größe des Objekts gestreckt; ● ShowText – Die Eigenschaft bestimmt, ob der Text unter dem Barcode angezeigt wird; ● DataColumn – Datenfeld, aus dem der Objekttext geladen werden soll; ● Expression – Ausdruck, der den Objekttext zurückgibt; ● Text – Objekttext; ● Padding – Mit dieser Eigenschaft können Sie Einzüge von den Rändern des Objekts in Pixel festlegen; ● WideBarRatio – Diese Eigenschaft ist bei allen linearen Barcodes vorhanden. Sie bestimmt die relative Größe der breiten Balken des Barcodes; ● CalcCheckSum – Viele lineare Barcodes verfügen über diese Eigenschaft. Sie bestimmt, ob die Prüfsumme automatisch berechnet werden soll. Wenn diese Eigenschaft deaktiviert ist, muss die Prüfsumme im Objekttext vorhanden sein.

PostNet

Dieser lineare Zahlen-Barcode wurde speziell für den United States Postal Service entwickelt. Er wird für die Maschinensortierung von Post verwendet. Das Aussehen dieses Barcodes unterscheidet sich stark von den bekannten UPC- und EAN-Codes, die im Einzelhandel zur Kennzeichnung von Produkten verwendet werden. Die Striche sind unterschiedlich hoch, und im Allgemeinen ist der Code auch von geringer Höhe, damit er leicht auf einem Umschlag angebracht werden kann.

Der ZIP-Code in den USA war ursprünglich 5 Zeichen lang, wurde aber 1983 aus Kapazitätsgründen auf 9 Zeichen erweitert. Aber dann wurde der Code um 2 weitere Zeichen erweitert. Der PostNet hat also drei Größen:

- 5 Zeichen;
- 9 Zeichen;
- 11 Zeichen.

Der Code wird wie folgt aufgebaut:

- Start-Zeichen;
- Daten;
- Prüffziffer;
- Stopp-Zeichen.

Jedes Zeichen ist mit 5 Strichen codiert. Die Start- und Stopp-Zeichen werden mit einem langen Strich angezeigt.

Im Folgenden finden Sie eine Liste mit übereinstimmenden Nummern und Barcodes:

1 - ;

2 - ;

3 - ;

4 - ;

5 - ;

6 - ;

7 - ;

8 - ;

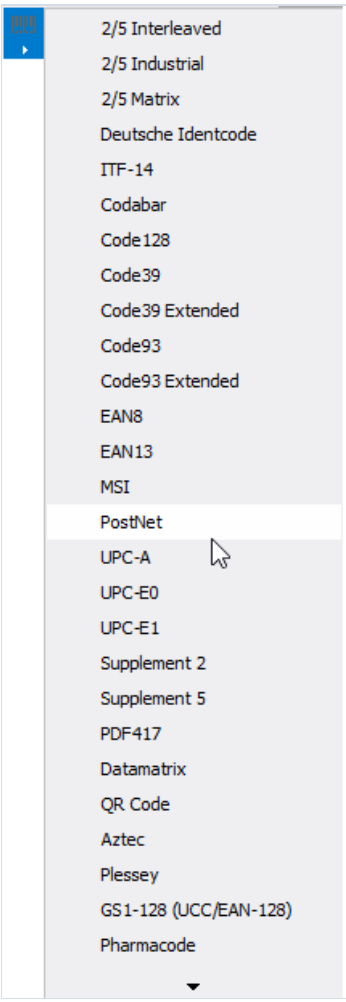
9 - ;

0 - .

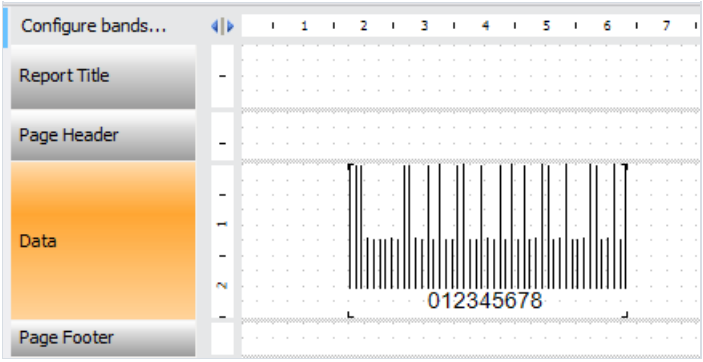
Aufgrund der relativ großen Anzahl von Strichen pro Zeichencodierung ist der PostNet-Barcode recht groß. Es ist auch möglich, nur Zahlen zu codieren. Diese Nachteile haben die Popularität des Codes verringert, und er wird jetzt durch den neueren Intelligent Mail ersetzt.

In FastReport.Net hat der PostNet keine begrenzte Zeichenanzahl, aber Sie sollten bedenken, dass der Standard die 3 oben erwähnten Codegrößenoptionen vorsieht.

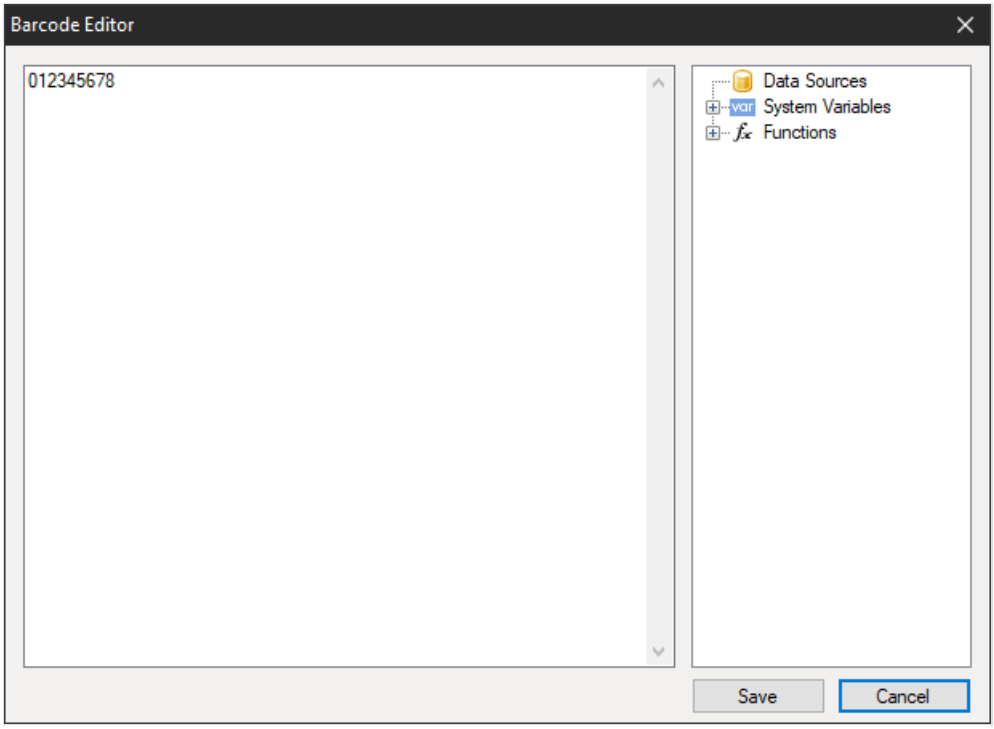
Um einen PostNet-Barcode der Berichtsseite hinzuzufügen, müssen Sie auf das Barcode-Symbol in der Seitenleiste im Berichtsdesigner klicken :



Wählen Sie den PostNet aus und fügen Sie den Code der Berichtsseite hinzu:



Um einen Barcode zu bearbeiten, doppelklicken Sie darauf, oder öffnen Sie mit einem Rechtsklick das Kontextmenü und wählen Sie „Bearbeiten...“ aus.



Wie alle Barcodes der PostNet-Code in FastReport.Net verfügt über eine Reihe von Eigenschaften, die Sie im Eigenschafteninspektor des Objekts bearbeiten können:

- Angle – Mit dieser Eigenschaft können Sie die Drehung eines Objekts auf einen der festen Winkel - 0, 90, 180, 270 Grad - einstellen;
- Zoom – Die Eigenschaft legt die Skalierung des Barcodes fest. Diese Eigenschaft wird nur in Verbindung mit der AutoSize-Eigenschaft verwendet;
- AutoSize – Wenn diese Eigenschaft aktiviert ist, wird das Objekt so gestreckt, dass der gesamte Barcode angezeigt wird. Wenn sie deaktiviert ist, wird der Barcode auf die Größe des Objekts gestreckt;
- ShowText – Die Eigenschaft bestimmt, ob der Text unter dem Barcode angezeigt wird;
- DataColumn – Datenfeld, aus dem der Objekttext geladen werden soll;
- Expression – Ausdruck, der den Objekttext zurückgibt;
- Text – Objekttext;
- Padding – Mit dieser Eigenschaft können Sie Einzüge von den Rändern des Objekts in Pixel festlegen;
- WideBarRatio – Diese Eigenschaft ist bei allen linearen Barcodes vorhanden. Sie bestimmt die relative Größe der breiten Balken des Barcodes;
- CalcCheckSum – Viele lineare Barcodes verfügen über diese Eigenschaft. Sie bestimmt, ob die Prüfsumme automatisch gelesen werden soll. Wenn diese Eigenschaft deaktiviert ist, muss die Prüfsumme im Objekttext vorhanden sein.

Deutsche Post Identcode

Der Deutsche Post Identcode basiert auf dem beliebten 2 of 5 Interleaved Barcode und verwendet dessen Zeichensatz. Der Unterschied liegt im Verwendungszweck des Codes. Der Identcode wurde speziell für die Deutsche Post entwickelt, um den Absender zu identifizieren. Basierend auf der erforderlichen Größe der Absenderdaten wird die Länge des Codes auf 12 Ziffern reduziert, im Vergleich zu 14 Ziffern bei Interleaved. Die tatsächliche Länge der zu codierenden Daten beträgt jedoch 13 Zeichen, da das 14. Zeichen für die Prüfsumme verwendet wird, die erforderlich ist, um zu überprüfen, ob der Code korrekt gelesen werden kann.

Der Identcode wird wie folgt aufgebaut:

● Start-Zeichen; ● 2 Ziffern: Identifizierung des Abgangsfrachtpostzentrums; ● 3 Ziffern: Kundenkennung; ● 6 Ziffern: Einlieferungsnummer; ● Prüfziffer; ● Stopp- Zeichen.

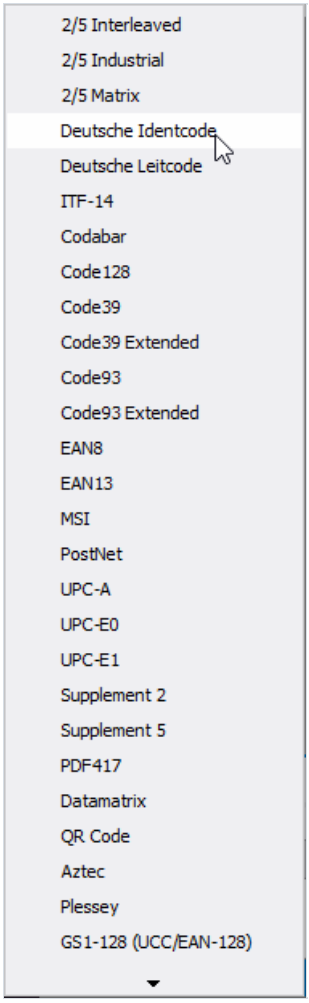
Wenn Sie die Striche und Zwischenräume als binäre Zeichen darstellen, wird das Start-Zeichen wie folgt bezeichnet: 1010. Dieses Zeichen hilft, die Leserichtung zu bestimmen.

Die 2 of 5 Interleaved Codetabelle wird zur Codierung der Datenziffern verwendet. Jedes Datenzeichen besteht aus 5 Strichen oder 5 Zwischenräumen. Zwei davon müssen breit und die übrigen schmal sein. An dieser Stelle sollte klargestellt werden, dass alle 2 of 5 Codes sowohl Striche als auch Zwischenräume zur Codierung verwenden. Daher ist die Länge des Codes immer gerade, da ein Satz von 5 Strichen und 5 Zwischenräumen jeweils zwei Ziffern codiert. Dies führt zu einer hohen Dichte und damit zu einem kompakten Code.

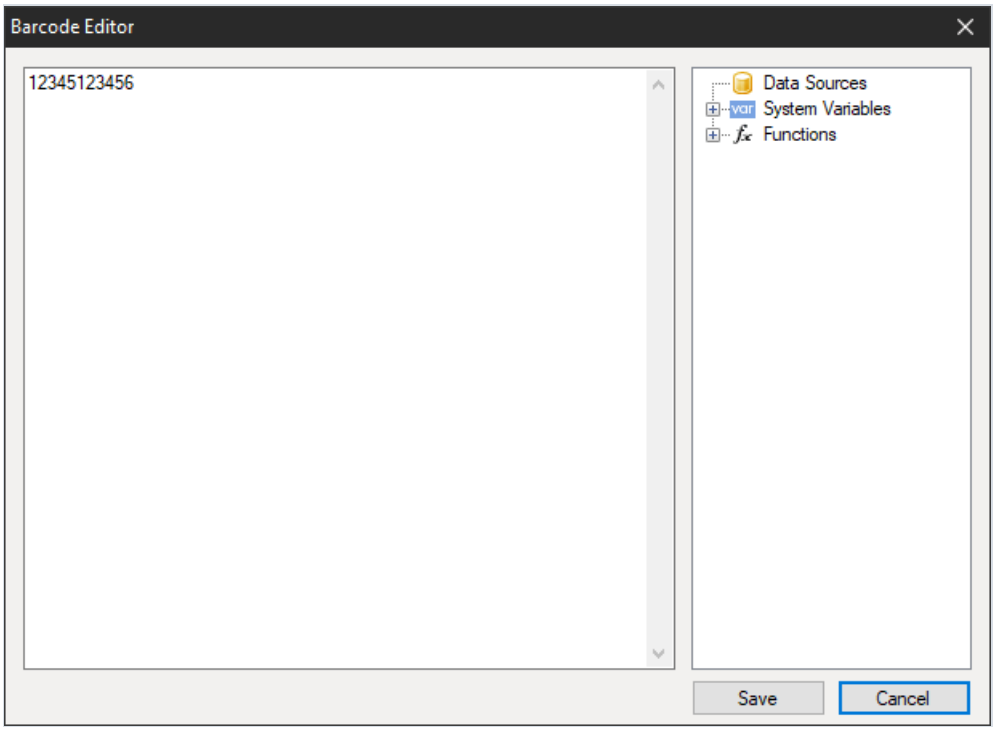
Nach den 11 Datenziffern folgt eine Prüfziffer, die mit dem Modulo-10-Algorithmus berechnet wird.

Der Code endet mit einem Stopp-Zeichen, das als 101 codiert ist.

Um der Berichtsseite einen Deutsche Post Identcode hinzuzufügen, klicken Sie auf das Barcode-Symbol :



Und wählen Sie den Deutsche Post Identcode aus. Platzieren Sie dann das Objekt auf der Berichtsseite. Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt. Alternativ kann der Code-Editor auch über das Kontextmenü des hinzugefügten Objekts aufgerufen werden, und zwar mit der Option Edit:



Sie können die Anzeige von Ziffern unter dem Code deaktivieren, indem Sie die ShowText-Eigenschaft im Eigenschaftsinspektor des hinzugefügten Barcode-Objekts auf False setzen.

Deutsche Post Leitcode

Der Deutsche Post Leitcode, oder einfach LeitCode, basiert auf dem weitbekannten 2 of 5 Interleaved Barcode. Mit diesem Code können 13 Ziffernzeichen (0...9) codiert werden. Da der Leitcode von der Deutschen Post (DHL) verwendet wird, hat er eine klar definierte Datenstruktur. Dadurch unterscheidet er sich vom 2 of 5 Interleaved.

Aufbau des Codes:

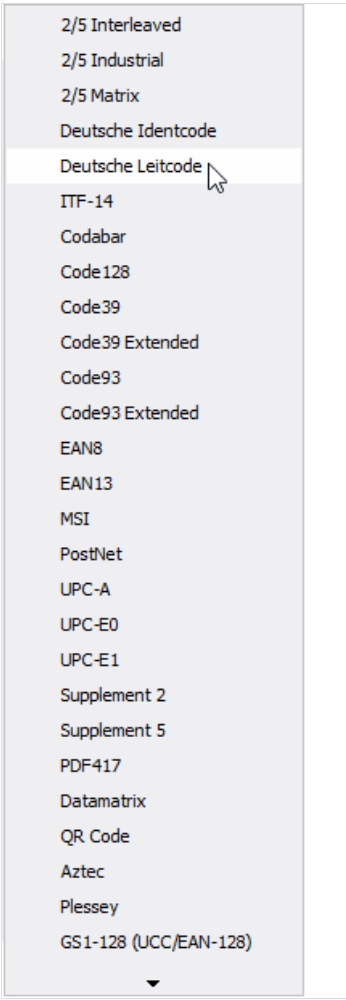
● Start-Zeichen; ● 1-5 – ZIP-Code; ● 6-8 - Straßenkennzahl; ● 9-11 - Hausnummer; ● 12-13 - Produktcode; ● 14 - Prüfziffer; ● Stopp- Zeichen.

Ein weiterer Unterschied zwischen dem Code und dem 2 of 5 Interleaved besteht in der Verwendung des Modulo-10-Algorithmus, um die bereits obligatorische Prüfsumme zu berechnen.

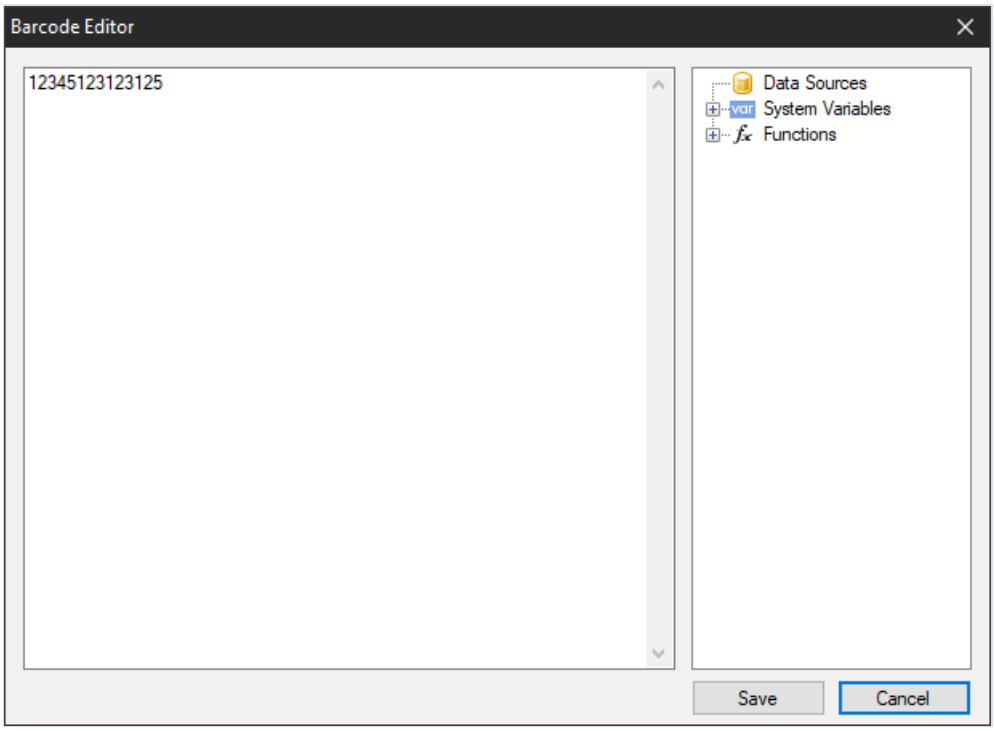
Der Leitcode wird von der Deutschen Post und DHL verwendet, um die Sortierung von Postsendungen zu automatisieren.

Der Barcode 2 of 5 und damit der Leicode verwenden sowohl Striche als auch Zwischenräume, um Daten zu codieren. Durch diesen Ansatz konnte die Größe des Codes reduziert werden. Das Leitcode-Datenzeichen wird in fünf Modulen (Striche/Zwischenräume) codiert. Zwei Striche sind breit und drei sind schmal. Das Gleiche gilt für Zwischenräume. Der Name des ursprünglichen 2 of 5 Barcodes ist von dieser Kombination abgeleitet.

Um der Berichtsseite einen Deutsche Post Leitcode hinzuzufügen, klicken Sie auf das Barcode-Symbol :



Und wählen Sie den Deutsche Post Leitcode aus. Platzieren Sie dann das Objekt auf der Berichtsseite. Um den Code zu bearbeiten, doppelklicken Sie auf das hinzugefügte Objekt. Alternativ kann der Code-Editor auch über das Kontextmenü des hinzugefügten Objekts aufgerufen werden, und zwar mit der Option Edit:



Sie können die Anzeige von Ziffern unter dem Code deaktivieren, indem Sie die ShowText-Eigenschaft im Eigenschaftsinspektor des hinzugefügten Barcode-Objekts auf False setzen.

Das "Kontrollkästchen"-Objekt

Dieses Objekt zeigt das Kontrollkästchen im Bericht an. Es sieht folgendermaßen aus:

Das Objekt kann zwei Zustände annehmen: "Aktiviert" und "Deaktiviert". Die folgenden Zustandsarten des Objektes werden genutzt:

· Festlegung des Zustandes in der Eigenschaft "Aktiviert"

· Anbinden des Objektes an die Datenspalte mit Hilfe der Eigenschaft "Datenspalte";

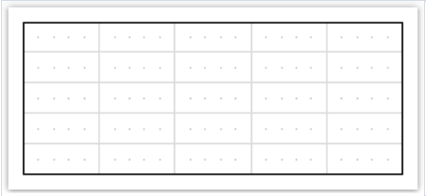
· Festlegung des Ausdrucks, der true oder false in der Eigenschaft "Ausdruck" zurückführt.

Das "Kontrollkästchen"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
CheckedSymbol, UncheckedSymbol	Diese Eigenschaften bestimmen das Symbol das, je nach Objektstatus, im Objekt angezeigt wird.
CheckColor	Diese Eigenschaft legt die Farbe des Kontrollkästchens fest.
CheckWidthRatio	Diese Eigenschaft wird zur Festlegung des Breitenverhältnisses des Kontrollkästchens genutzt. Die Breite des Prüfsymbols ist von der Größe des Objektes abhängig. Dieses kann Werte zwischen 0.2 - 2 annehmen.
HideIfUnchecked	Diese Eigenschaft ermöglicht das Verbergen des deaktivierten Symbols.
Aktiviert	Diese Eigenschaft steuert den Zustand des Objektes.
DataColumn	Die Datenspalte an die dieses Objekt gebunden ist. Der Spaltentyp ist entweder bool oder int.
Ausdruck	Der Ausdruck der true oder falsezurück gibt.

Das "Tabellen"-Objekt

Das "Tabellen"-Objekt setzt sich aus Zeilen, Spalten und Zellen zusammen. Es ist eine vereinfachte Form der Microsoft Exceltabelle. Dieses sieht folgendermaßen aus:



Weitere Informationen zu diesem Objekt werden im Kapitel "Berichterstellung" erläutert.

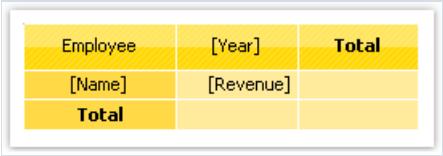
Das "Tabellen"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
ColumnCount	Diese Eigenschaft wird zur schnellen Festlegung der Spaltenanzahl genutzt. Ist die Spaltenanzahl der Tabelle geringer, werden diese hinzugefügt; ist sie größer, werden diese gelöscht.
RowCount	Diese Eigenschaft wird zur schnellen Festlegung der Zeilenanzahl genutzt. Ist die Zeilenanzahl der Tabelle geringer, werden diese hinzugefügt; ist sie größer, werden diese gelöscht.
FixedColumns	Diese Eigenschaft bestimmt wie viele Spalten der Tabelle fest sind. Feste Spalten bilden die Kopfzeile der Tabelle. Der Druck der Kopfzeile wird über die Eigenschaft "RepeatHeaders" gesteuert.
FixedRows	Diese Eigenschaft bestimmt wie viele Zeilen der Tabelle fest sind. Feste Zeilen bilden die Kopfzeile der Tabelle. Der Druck der Kopfzeile wird über die Eigenschaft "RepeatHeaders" gesteuert.
RepeatHeaders	Diese Eigenschaft erlaubt den Druck der Kopfzeile auf jeder neuen Seite. Sie funktioniert lediglich bei dynamisch gebildeten Tabellen.

Das "Matrix"-Objekt

Das "Matrix"-Objekt, wie das "Tabellen"-Objekt auch, setzt sich aus Zeilen, Spalten und Zellen zusammen. Die Anzahl der Zeilen und Spalten der Matrix sind bereits im Voraus bekannt, da diese von den Daten mit denen sie verbunden sind, abhängen.

Das Objekt sieht folgendermaßen aus:



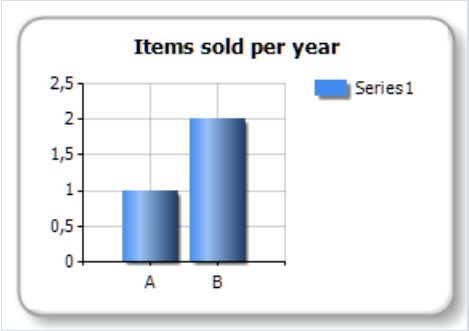
Weitere Informationen zu diesem Objekt werden im Kapitel "Berichterstellung" erläutert.

Das "Matrix"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
RepeatHeaders	Bei einer, über mehrere Seiten verteilen, Matrix ermöglicht diese Eigenschaft den Druck der Kopfzeile auf jeder neuen Seite.
CellsSideBySide	Diese Eigenschaft legt die Verortung der Matrixzellen fest, wenn diese mehrere Datenzellenniveaus aufweisen. Mögliche Anzeigearten:· Anzeige der Zellen nebeneinander;· Anzeige der Zellen untereinander.
Format	Durch Nutzung dieser Eigenschaft wird der Stil der Gesamtmatrix festgelegt. Es kann ein bereits vordefinierter Stil ausgewählt werden.
AutoSize	Diese Eigenschaft erlaubt die automatische Berechnung der Matrixgröße. Wenn die Objektgröße manuell eingestellt werden soll, kann diese Eigenschaft deaktiviert werden.
DataSource	Die Eigenschaft ermöglicht die Verbindung der Matrix mit der Datenquelle. Diese Eigenschaft wird automatisch festgelegt, wenn die Datenspalte in die Matrix gezogen wird. Werden jedoch Ausdrücke in Zellen verwendet, sollte überprüft werden, dass diese Eigenschaft korrekt eingestellt wurde.
Filter	Diese Eigenschaft beinhaltet Ausdrücke zur Datenfilterung, die auf die Datenquelle der Matrix (siehe Eigenschaft "DatenQuelle") angewendet werden können.

Das "Diagramm"-Objekt

Das "MS Diagramm"-Objekt ermöglicht die Anzeige von Diagrammen. Es stehen mehr als 30 verschiedene Serientypen zur Verfügung: Balken, Spalten, Bereiche, Linien, Blasen, Kuchen, kreisförmig, Finanzen, Pyramide, Bereiche etc. Das Objekt sieht folgendermaßen aus:



Weitere Informationen zu diesem Objekt werden im Kapitel "Berichterstellung" erläutert.

Das "Diagramm"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
Diagramm	Verweis auf das Microsoft Diagrammobjekt.
AlignXValues	Diese Eigenschaft ermöglicht einer Ausrichtung von X Werten in verschiedenen Diagrammreihen (durch Eingabe leerer Werte). Sie wird bei einem Diagramm mit mindestens zwei Reihen eingesetzt.
AutoSeriesColumn,AutoSeriesColor,AutoSeriesSortOrder	Diese Eigenschaft ermöglicht ein automatisches Erstellen von Serien. Weitere Informationen hierzu werden im Kapitel "Berichterstellung" erläutert.
DataSource	Die Eigenschaft ermöglicht die Verbindung des Diagramms mit der Datenquelle.
Filter	Diese Eigenschaft beinhaltet Ausdrücke zur Datenfilterung, die auf die Datenquelle des Diagramms (siehe Eigenschaft "DataSource" beendet werden.

Das "Postleitzahl"-Objekt

Das "Postleitzahl"-Objekt ermöglicht den Druck einer Postleitzahl auf einem Briefumschlag. Es kann numerische Zeichen (0-9) beinhalten.

Das Objekt sieht folgendermaßen aus:

Man kann die Objekte mit den Daten, durch eine der folgenden Methoden, verbinden:

· Festlegen der Postleitzahldaten in den "Text"-Eigenschaften;

· Anbinden des Objektes an die Datenspalte mit Hilfe der Eigenschaft "Datenspalte";

· Festlegung der Ausdrücke welche die Postleitzahldaten in die Eigenschaft "Ausdrücke" zurückführen.

Das "Postleitzahl"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
SegmentCount	Die Anzahl der Segmente in einem Code. Diese Eigenschaft ist standardmäßig auf 6 eingestellt.
SegmentWidth, SegmentHeight	Die Größe eines Kennzahlsegmentes. Die Standardgröße beträgt 0,5x1 cm.
Spacing	Diese Eigenschaft bestimmt die Distanz zwischen den zwei Segmentursprüngen. Der Standardwert beträgt 0,9 cm.
ShowGrid	Bestimmt ob die Anzeige eines Gitters notwendig ist.
ShowMarkers	Bestimmt ob die Anzeige der Marker (fette, horizontale Linien über der Postleitzahl) notwendig ist.
DataColumn	Die Datenspalte an die dieses Objekt gebunden ist.
Expression	Der Ausdruck der die Postleitzahldaten zurückgibt.
Text	Der Text der die Postleitzahl enthält.

Objekt "Erweiterte Matrix"

Mit diesem Objekt wie mit dem "Matrix" Objekt können Sie Kreuztabellenberichte erstellen.

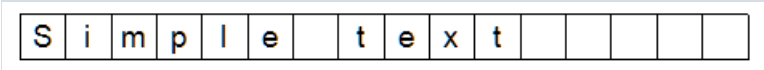
	Top 5 customers		Total
	Total	Others (84)	
Beverages (12)	$144 007,88	$192 684,30	**$336 692,18**
Condiments (12)	$123 407,20	$105 665,89	**$229 073,09**
Confections (13)	$54 823,13	$112 615,09	**$167 438,23**
Dairy Products (10)	$103 074,28	$166 433,01	**$269 507,29**
Grains/Cereals (7)	$31 368,01	$64 376,58	**$95 744,59**
Meat/Poultry (6)	$269 495,38	$113 526,98	**$383 022,36**
1. Alice Mutton	$13 428,48	$19 269,90	**$32 698,38**
2. Mishi Kobe Niku	$1 319,20	$5 907,30	**$7 226,50**
3. Pâté chinois	$7 137,60	$10 288,80	**$17 426,40**
4. Perth Pasties	$6 916,56	$13 657,61	**$20 574,17**
5. Thüringer Rostbratwurst	$239 795,40	$60 573,28	**$300 368,67**
6. Tourtière	$898,14	$3 830,10	**$4 728,24**
Produce (5)	$67 154,22	$71 955,36	**$139 109,58**
Seafood (12)	$31 732,55	$99 591,19	**$131 323,74**
Total	**$825 062,63**	**$926 848,41**	**$1 751 911,04**

Weitere Einzelheiten zur Verwendung dieses Objektes finden Sie im Kapitel Objekt "Erweiterte Matrix".

Das "Zellentext"-Objekt

Dieses Objekt kann jedes Textzeichen in seiner individuellen Zelle anzeigen. Es wird häufig zum Druck von Formularen in Finanzanwendungen genutzt.

Das Objekt sieht folgendermaßen aus:



Das Objekt wird vielmehr direkt aus dem "Text"-Objekt vererbt. Es kann auf die gleiche Art mit den Daten verbunden werden. Zum Beispiel: Objekt Editor aufrufen und den folgenden Text eingeben:

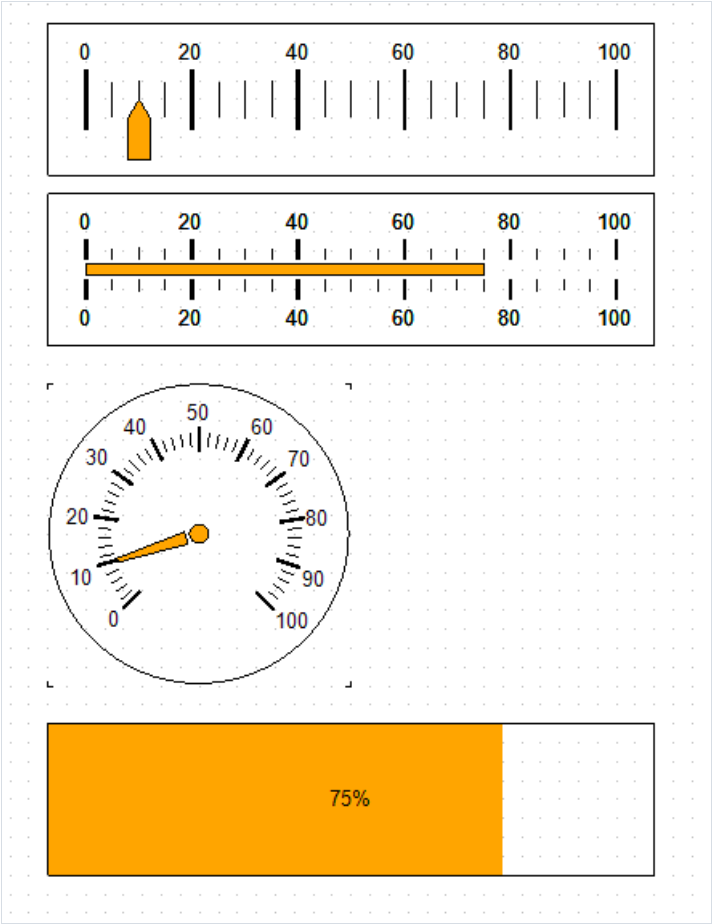
```
[Employees.FirstName]
```

Das "Zellentext"-Objekt weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
CellWidth, CellHeight	Diese Eigenschaft legt die Größe einer individuellen Zelle fest. Sind beide Wert auf 0 (Standard) eingestellt, wird die Zellengröße automatisch, je nach genutzter Schriftart, berechnet.
HorzSpacing, VertSpacing	Diese Eigenschaft legt den horizontalen und vertikalen Abstand zwischen zwei benachbarten Zellen fest.

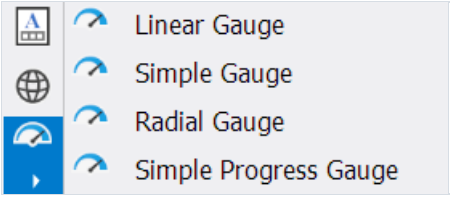
Objektgruppe "Geber"

Diese Objekte sind dafür ausgelegt, einen gewissen Wert visuell darzustellen. Die vier verschiedenen Arten von Gebern, die derzeit unterstützt werden, sehen wie folgt aus:

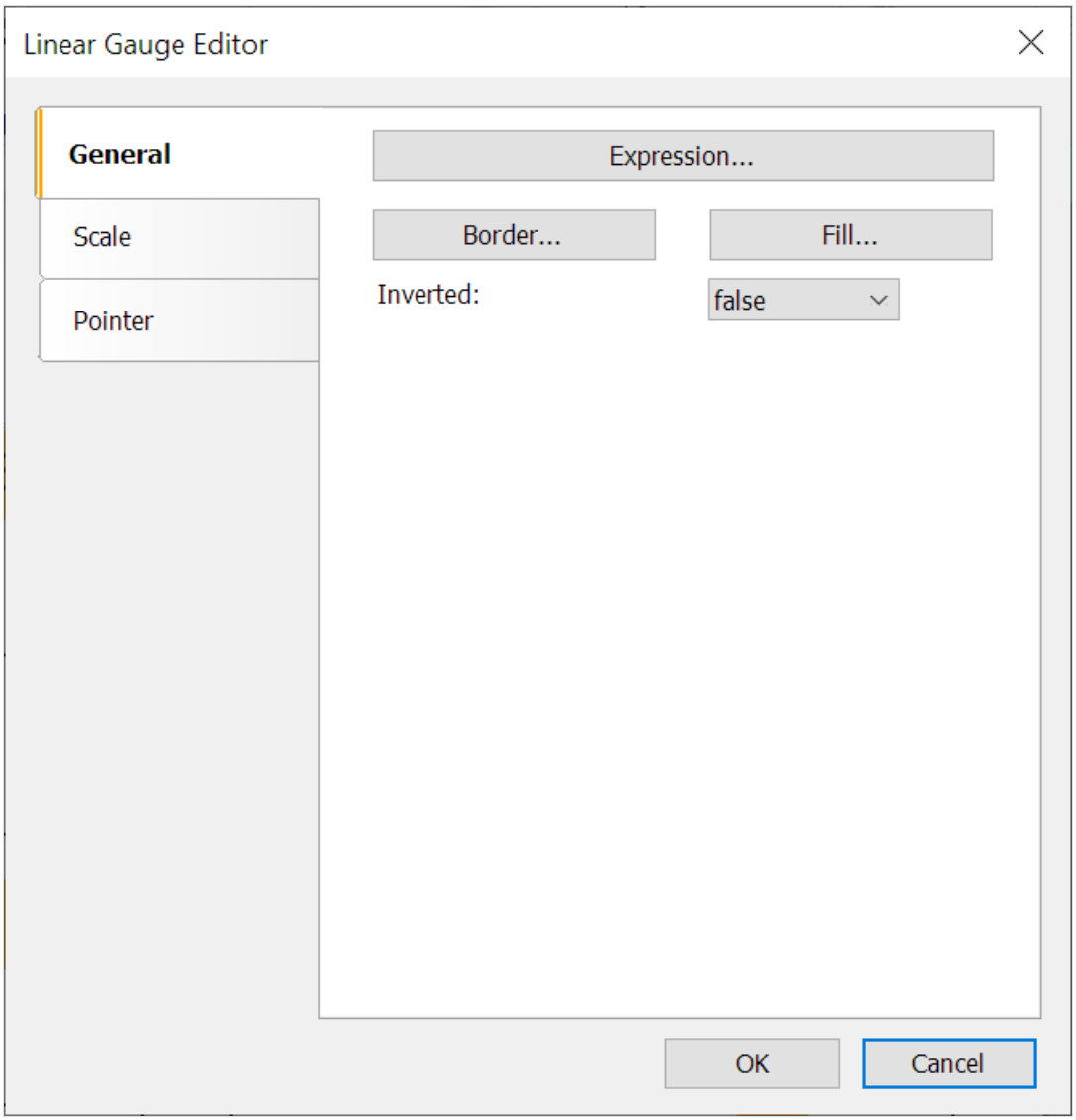


- Lineargeber
- Einfacher Geber
- Rotatorischer Geber
- Linearer Fortschrittsgeber

Um Ihrem Bericht ein „Geber"-Objekt hinzuzufügen, wählen Sie eine der Optionen in dem Untermenü, das sich öffnet, wenn Sie auf die Schaltfläche klicken :



Sie können einige Aspekte des Erscheinungsbildes des Gebers ändern, z. B. die Farbe des Zeigers. Um die Geberparameter zu bearbeiten, doppelklicken Sie mit der linken Maustaste darauf oder wählen Sie im Kontextmenü „Bearbeiten..." aus.



Die Schaltfläche "Expression" öffnet einen Texteditor, in dem Sie den Geberwert selbst eingeben oder einen Ausdruck dafür schreiben und den Geber mit den Daten verbinden können.

"Digitale Signatur" Objekt

Bei der digitalen Signatur handelt es sich um ein Verschlüsselungsverfahren, das die Einzigartigkeit und Unverwechselbarkeit sichert, mit dem die Urheberschaft eindeutig nachgewiesen wird und ein Dokument vor Änderungen geschützt wird. Dank zuverlässiger Verschlüsselungsalgorithmen sind solche Signaturen nicht schlechter als handschriftliche Unterschriften, sondern sogar besser und zuverlässiger.

In der aktuellen Version gibt es zwei Arten von Signaturen:

1. **Das Unterschriftsfeld ** (signature field) bedeutet, dass es im Dokument ein spezielles Feld gibt, auf das der Benutzer klicken kann, um sein Zertifikat anzuhängen.

Ein Objekt, das diese Funktion implementiert, wird durch Drücken dieser Taste hinzugefügt:

Es wird als digitale Signatur (Digital Signature) bezeichnet. Wenn dieses Objekt auf der Berichtsseite platziert wird, sieht es wie folgt aus:

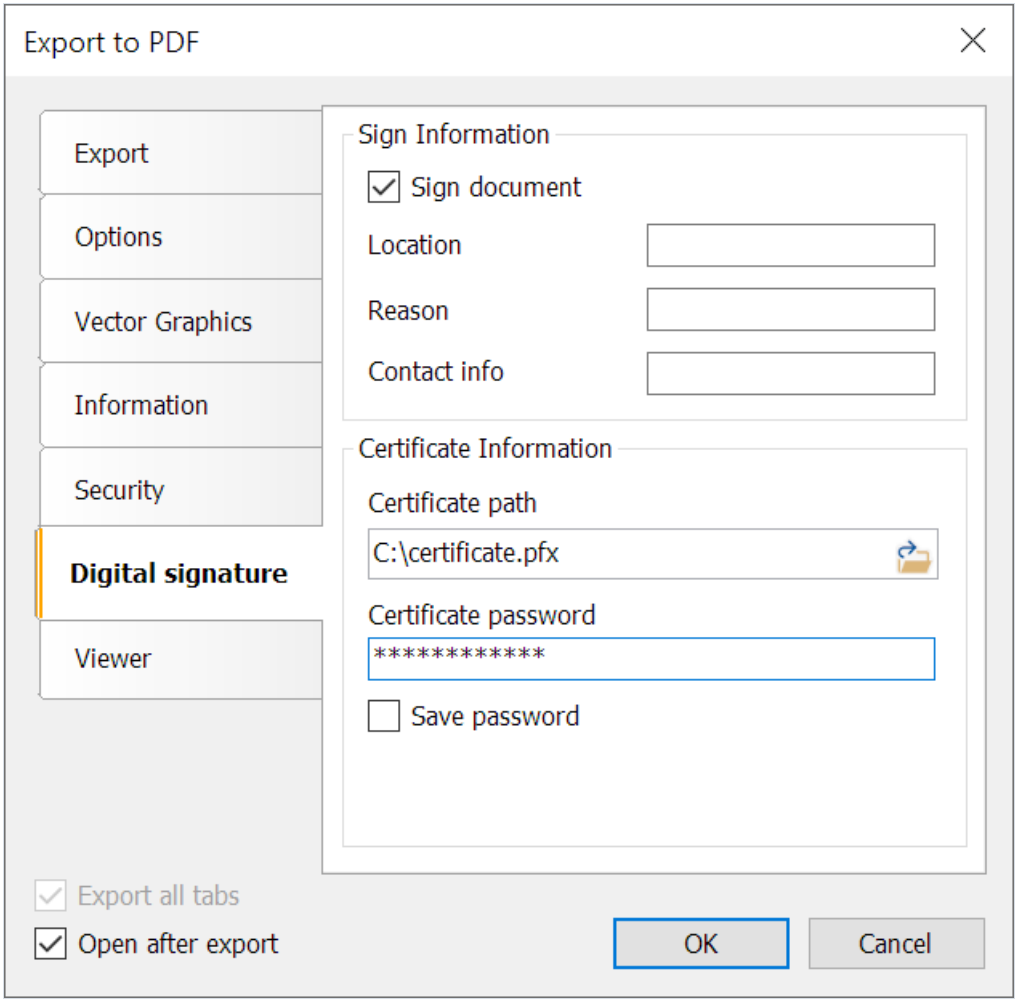
Im Ansichtsmodus ist es im erstellten Bericht unsichtbar. Seine Funktionalität ist ausschließlich auf den PDF-Export beschränkt, d.h. Sie sehen dieses Feld, wenn Sie die PDF-Datei in Acrobat Reader oder einem anderen PDF-Viewer aufrufen.

Nach dem Exportieren sieht das Feld wie folgt aus:

Beim Anklicken öffnet sich ein Auswahldialog für Zertifikate und nach Eingabe des Passworts wird das Feld signiert.

2. **Eine unsichtbare Signatur ** (invisible signature) ist eine Zertifizierung. Optisch wird sie nicht dargestellt, aber in den Dokumenteigenschaften können Sie Informationen über den Unterzeichner, die Echtheit der Unterschrift, die Version des Dokuments zum Zeitpunkt seiner Unterzeichnung und andere Informationen erhalten.

Um eine exportierte PDF-Datei mit einer unsichtbaren Signatur zu signieren, müssen Sie das Objekt „Digitale Signatur" nicht zur Berichtsseite hinzufügen. Sie müssen die Signaturoption in den Exporteinstellungen aktivieren und das Zertifikat dort auswählen:


Export to PDF
Export
Options
Vector Graphics
Information
Security
Digital signature
Viewer
Sign Information
Sign document
Location
Reason
Contact info
Certificate Information
Certificate path
C:\certificate.pfx
Certificate password
****
Save password
Export all tabs
Open after export
OK
Cancel

Der erste FastReport Bericht

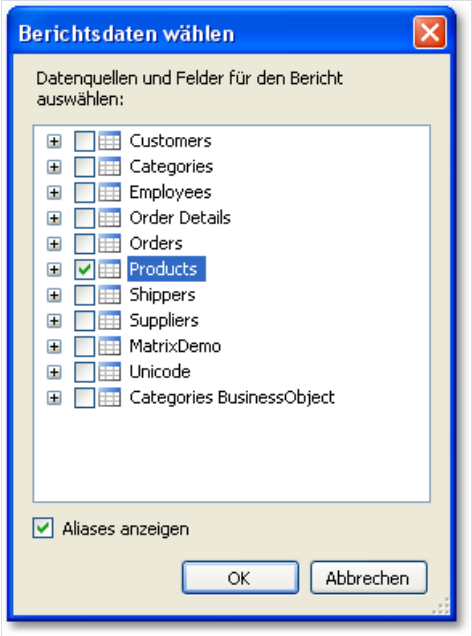
Hier wird die Erstellung eines einfachen Berichts (Produktliste) erläutert. Als Datenquelle werden die Produkttabellen der Demodatenbank genutzt.

Bitte befolgen Sie die weiter unten beschriebenen Schritte des Demonstrationsprogramms: Demo.exe. Dies startet den Designer.

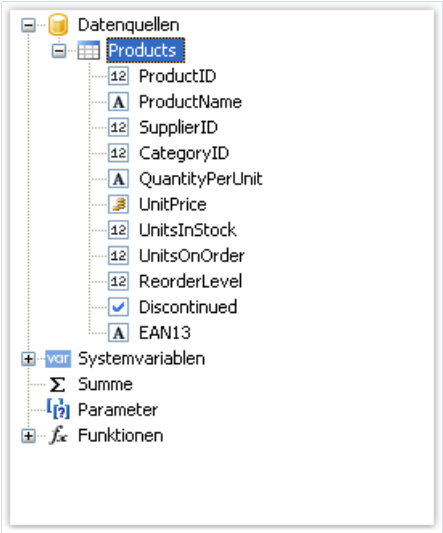
Beispiel 1: Händisches Erstellen eines Berichts

Hier lernen Sie, wie man händisch einen Bericht erstellt. Um dies zu tun, müssen die folgenden Anweisungen beachtet werden:

- Drücken Sie die Schaltfläche auf der Symbolleiste und wählen Sie "Leerer Bericht" aus dem Fenster "Neues Item hinzufügen";
- Wählen Sie aus dem Menü "Daten" den "Berichtsdaten auswählen..." und aktivieren Sie die Datenquelle "Produkte":

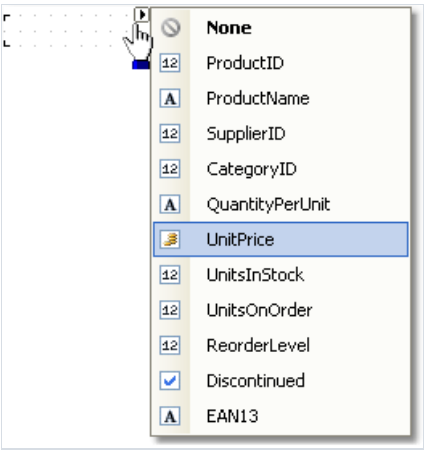


- Wechseln Sie zum Dienstfenster "Daten" (sollte sich dies nicht auf dem Bildschirm befinden, nutzen Sie das Menüelement "Daten|Datenfenster" anzeigen). Erweitern Sie das Element "Data Sources", dann das Element "Products":

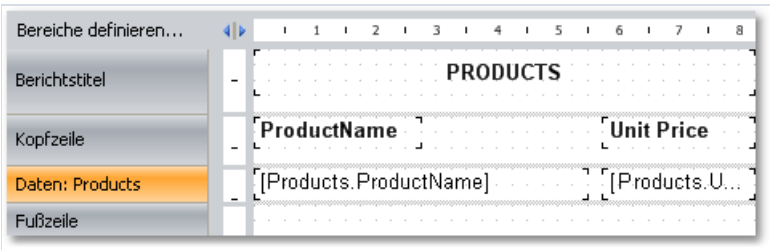


- Ziehen Sie die Datenspalte "ProduktName" auf die Datenleiste. FastReport erstellt ein Textobjekt, das mit der Spalte und seiner entsprechenden Kopfzeile verbunden ist;
- Die Datenspalte "UnitPrice" wird mit einer anderen Methode erstellt. Hierzu drücken Sie die Schaltfläche "Text" auf der "Objekt"-Symbolleiste:

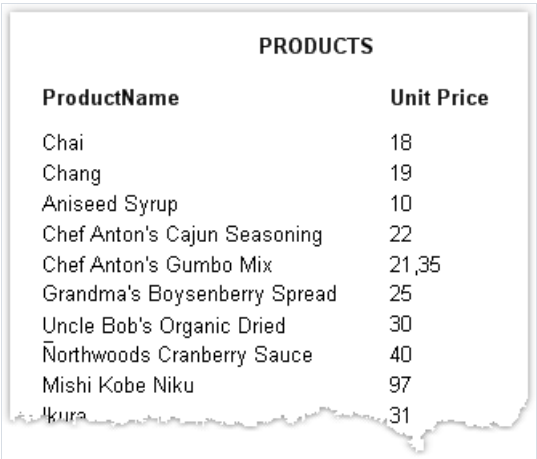
- Lassen Sie die Maus los und ziehen Sie den Mauszeiger auf die "Daten"-Leiste. FastReport fordert Sie auf ein Objekt einzufügen. Wählen Sie die benötigte Position und klicken Sie die Maus um das Objekt einzufügen.
- Positionieren Sie den Mauszeiger auf das Objekt und klicken Sie die kleine Schaltfläche in der rechten Objektecke. Hier wird Ihnen eine Liste mit Datenspalten angezeigt. Wählen Sie das Element "UnitPrice" aus der Liste:



- Erstellen Sie den "Text"-Objektkopf für die Spalte "UnitPrice". Positionieren Sie diesen auf der Leiste "Seitenkopf". Doppelklicken Sie auf das Objekt und tippen Sie den Text "UnitPrice" ein.
- Erstellen Sie das "Text"-Objekt: Berichtstitel. Platzieren Sie dieses auf die Leiste "Berichtstitel" und geben Sie den Text "PRODUCTS" ein;
- Legen Sie "Fett" als Schriftart-Stil für alle Objekte, die auf der Leiste "Seitenkopf" und "Berichtstitel" positioniert sind, fest. Wählen Sie hierzu Objekte durch Drücken von "Alt" und Drücken der Schaltfläche B auf der "Text"-Symbolleiste aus. Der Bericht sollte folgendes Aussehen haben:



Um den Bericht zu erstellen, klicken Sie auf der Symbolleiste auf die Schaltfläche . Der Bericht wird erstellt und im Vorschaufenster angezeigt:



PRODUCTS

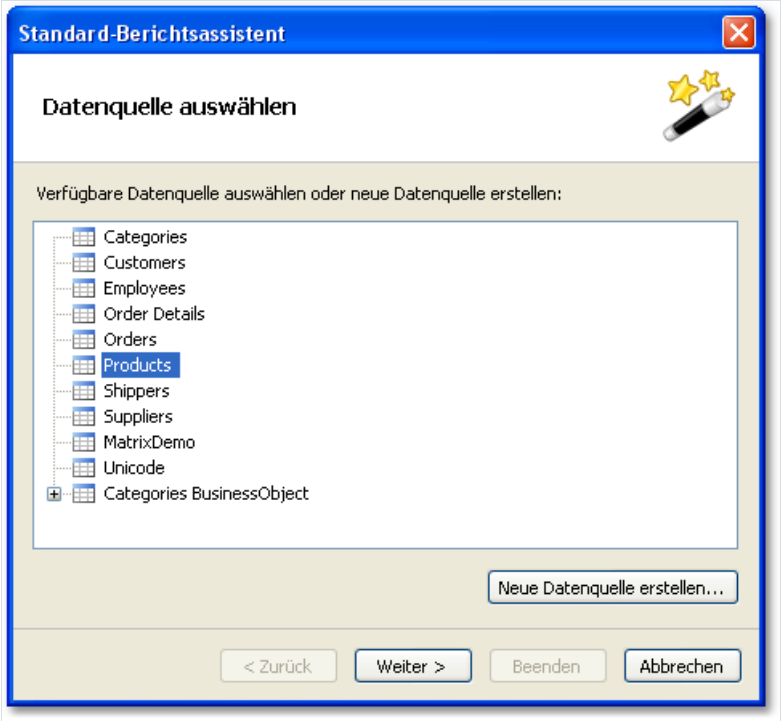
ProductName	Unit Price
Chai	18
Chang	19
Aniseed Syrup	10
Chef Anton's Cajun Seasoning	22
Chef Anton's Gumbo Mix	21,35
Grandma's Boysenberry Spread	25
Uncle Bob's Organic Dried	30
Northwoods Cranberry Sauce	40
Mishi Kobe Niku	97
Ikura	31

Beispiel 2: Erstellen eines Berichts mit Hilfe des Assistenten

In diesem Beispiel wird Ihnen erläutert, wie man einen Bericht mit Hilfe des "Standard-Berichtsassistent" erstellt. Um dies zu tun, befolgen Sie bitte die folgenden Anweisungen:

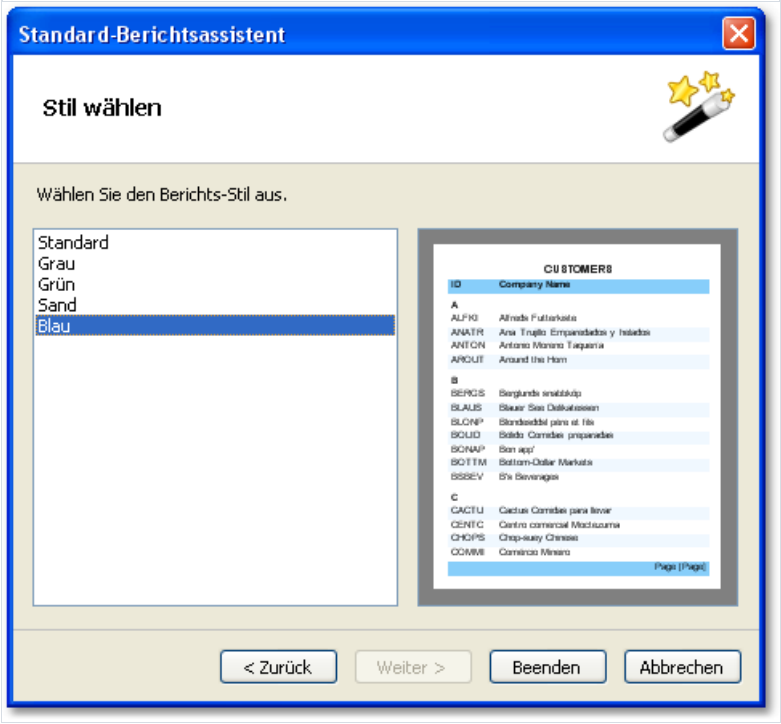
- Drücken Sie auf der Symbolleiste die Schaltfläche und wählen Sie "Standard-Berichtsassistent" aus dem Fenster "Neuen Eintrag hinzufügen";

- Der erste Schritt des Assistentenprogramms fordert Sie auf, die "Produkt"-Tabelle auszuwählen und dann die Schaltfläche "Weiter" zu klicken:

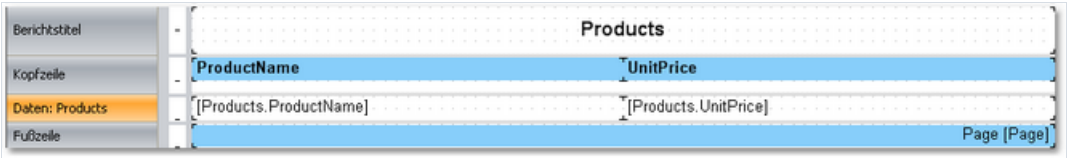


- In einem zweiten Schritt des Assistenten werden Sie aufgefordert die Datenspalten "ProductName" und "UnitPrice" festzulegen:

- Die restlichen Schritte können übersprungen werden. Klicken Sie hierzu auf "Weiter";
- Im letzten Schritt des Assistenten legen Sie den Stil als "Blau" fest und klicken die Schaltfläche "Beenden":



FastReport erstellt den folgenden Bericht:



Um den Bericht zu erstellen, klicken Sie auf der Symbolleiste auf die Schaltfläche . Der Bericht wird erstellt und im Vorschaufenster angezeigt:

Products

ProductName	UnitPrice
Chai	18
Chang	19
Aniseed Syrup	10
Chef Anton's Cajun Seasoning	22
Chef Anton's Gumbo Mix	21,35
Grandma's Boysenberry Spread	25
Uncle Bob's Organic Dried Pears	30
Northwoods Cranberry Sauce	40
Mishi Kobe Niku	97
Ikura	31
Queso Cabrales	21

Berichterstellung

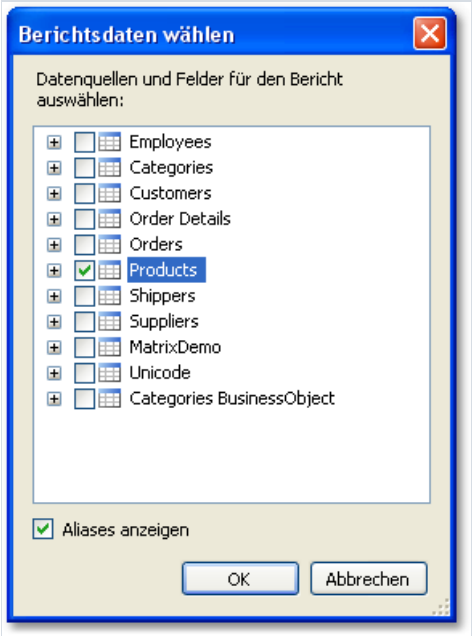
In diesem Kapitel werden die Methoden zur Erstellung allgemeiner Berichtstypen erläutert. Grundsätzlich muss bei der Erstellung eines Berichts das Folgende beachtet werden:

1. Auswahl und Erstellen von Daten, die im Bericht genutzt werden.
2. Erstellen der Berichtstruktur durch Hinzufügen der benötigten Berichtleisten.
3. Verbinden der Leisten mit einer Datenquelle.
4. Zum Druck der Daten wird das "Text"-Objekt auf die Leiste platziert.
5. Festlegen des Aussehens und der Formatierung.

Auswahl der Berichtsdaten

Vor der Erstellung eines Berichts müssen die Daten, die im Bericht gedruckt werden sollen, ausgewählt werden. Dies kann auf zwei Arten geschehen:

· Auswahl einer der Datenquellen, die im Bericht von einem Programmierverfahren registriert wurde. Dies kann über das Menü "Daten|Berichtsdaten wählen" geschehen, indem die benötigte Datenquelle aktiviert wird:



- Erstellen einer neuen Datenquelle im Menü "Daten|Neue Datenquelle".

Weitere Informationen zu dieser Eigenschaft werden im Kapitel "Datenquellen" erläutert.

Nachdem die Datenquelle ausgewählt wurde, erscheint diese im Fenster "Daten". Nun kann diese Quelle im Bericht genutzt werden. Die Mehrzahl der Berichte nutzt lediglich eine Datenquelle. Für Berichte des Typs "Master-Detail", werden zwei zueinander in Beziehung stehende Datenquellen benötigt (weitere Informationen hierzu werden im Kapitel "Datenquellen" erläutert). Es können mehrere Datenquellen, die Daten aus verwandten Quellen drucken, in einem Bericht benötigt werden.

Dynamisches Layout

Bei der Erstellung eines Berichts ist es oftmals notwendig einen Text zu drucken, dessen Größe nicht bekannt ist. Beispielweise bei der Beschreibung von Gütern. In einem solchen Fall muss die folgende Aufgabe gelöst werden:

· Berechnung der Objekthöhe zum Einschluss des gesamten Textes;

· Berechnung der Leistenhöhe zum Einschluss eines Objektes mit einer variablen Anzahl an Text;

· Verschiebung oder Änderung anderer Objekte, die sich auf der Leiste befinden, damit diese nicht das grundsätzliche Design des Berichts stören.

Diese Aufgaben lassen sich mit den gleichen Objekt-, und Leisteneigenschaften einfach lösen:

· "Cangrow" und "CanShrink" Eigenschaften erlauben die automatische Berechnung der Objekthöhe;

· Die Eigenschaft "ShiftMode" ermöglicht das Verschieben von Objekten, die sich unter den erweiterbaren Objekten befinden;

· Die Eigenschaft "GrowToBottom" ermöglicht die Größenänderung eines Objektes zum unteren Rand der Leiste;

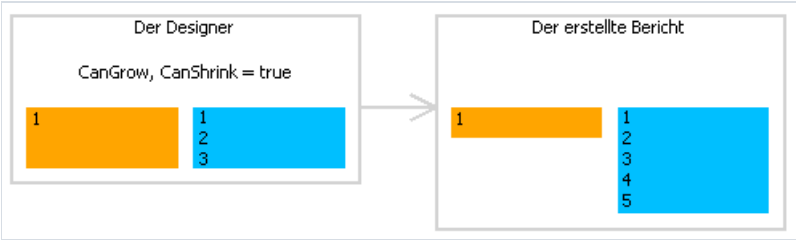
· Die Eigenschaften "Anker" und "Dock" erlauben die Steuerung der Objektgröße abhängig von der Leistengröße.

All diese Eigenschaften werden weiter unten genauer betrachtet.

Eigenschaften CanGrow, CanShrink

Alle Leisten und Berichtsobjekte weisen diese Eigenschaften auf. Diese bestimmt ob sich ein Objekt, je nach Größe des Inhaltes, vergrößern oder verkleinern kann. Sind beide Eigenschaften deaktiviert weist die Leiste stets, die im Designer festgelegte Größe auf.

Diese Eigenschaften sind zum Druck eines Textes, dessen Größe beim Design unbekannt ist, sehr nützlich. Damit ein Objekt den gesamten Text umfassen beinhalten kann, müssen die Eigenschaften "CanGrow" und "CanShrink" aktiviert sein:



Die folgenden Eigenschaften können die Leistenhöhe beeinflussen:

· "Text";

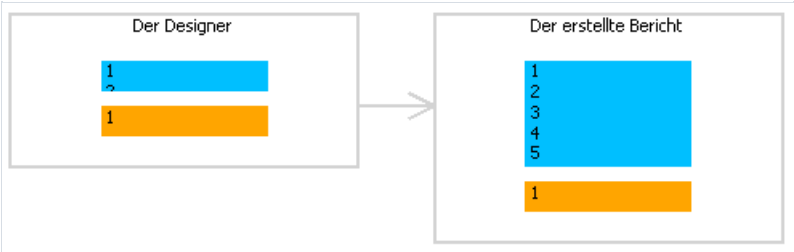
· "Rich Text";

· "Bild" (mit "AutoSize" deaktiviert);

· "Tabelle".

Eigenschaft ShiftMode

Alle Berichtsobjekte weisen diese Eigenschaft auf. Diese Eigenschaft kann nur über das Fenster "Eigenschaft" angesteuert werden. Ein Objekt, dessen "ShiftMode"-Eigenschaft aktiviert ist, bewegt sich entsprechend der Vergrößerung oder Verkleinerung des darüber liegenden Objekts, hoch und runter.



Die Eigenschaft "ShiftMode" kann die folgenden Werte aufweisen:

· Always (standardmäßig). Das Objekt muss immer wechseln.

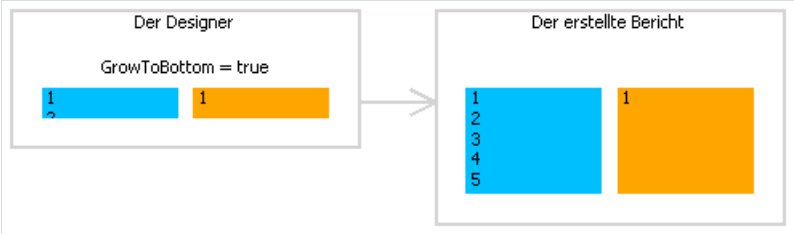
· Never. Gibt an, dass das Objekt nicht wechseln muss.

· WhenOverlapping. Das Objekt muss verschoben werden, wenn das erweiternde Objekt direkt darüber platziert ist (d.h. beide Objekte überlappen sich horizontal).

Diese Eigenschaft vereinfacht den Druck von Informationen in Tabellenform, wenn mehrere Zellen der Tabelle übereinander platziert sind und kann eine unterschiedliche Anzahl an Text aufweisen.

GrowToBottom Eigenschaft

Alle Berichtsobjekte weisen diese Eigenschaft auf. Beim Druck eines Objektes mit dieser Eigenschaft, erstreckt sich dieses zum unteren Rand der Leiste:



Dies wird zum Druck von Informationen in Tabellenform benötigt. In einer Tabellenzeile können sich mehrere Objekte befinden, die sich strecken können. Diese Eigenschaft ermöglicht alle Objekthöhen auf die maximale Leistenhöhe einzustellen.

Anker Eigenschaft

Alle Berichtsobjekte weisen diese Eigenschaft auf. Diese Eigenschaft bestimmt, wie das Objekt seine Position und/oder seine Größe ändert, wenn der Container auf dem es liegt, vergrößert oder verkleinert wird. Die Nutzung eines Ankers erlaubt es dem Objekt zu expandieren oder sich synchron mit dem Container zu verschieben.

In den meisten Fällen handelt es sich bei dem hier erwähnten Container um eine Leiste. Dies ist jedoch nicht zwingend notwendig; er kann auch eine "Tabelle" oder eine "Matrix" sein.

Die "Anker" Eigenschaft kann einen der folgenden Werte oder eine Kombination derselbigen aufweisen:

Wert	Beschreibung
Links	Verankert die linke Kante des Objektes. Ändert sich die Containergröße wird sich das Objekt nicht nach links/rechts bewegen.
Oben	Verankert die obere Objektkante. Wenn sich die Containerhöhe ändert wird sich das Objekt nicht nach oben/unten bewegen.
Rechts	Verankert die rechte Containerkante. Wenn sich die Containerbreite ändert, bleibt der Abstand zwischen der rechten Objektkante und dem Container unverändert. Ist zusätzlich die linke Containerkante verankert, vergrößert und verkleinert sich das Objekt synchron mit dem Container.
Unten	Verankert die untere Objektkante. Ändert sich die Containerbreite, bleibt der Abstand zwischen der unteren Objektkante und dem Container unverändert. Ist zusätzlich die obere Containerkante verankert, vergrößert und verkleinert sich das Objekt synchron zum Container.

Standardmäßig ist der Wert dieser Eigenschaft: Links, Oben. Ändert sich die Containergröße wird sich das Objekt nicht ändern. Die folgende Tabelle stellt einige Kombinationen häufig genutzter Werte dar:

Wert	Beschreibung
Left, Top	Standardwert. Das Objekt ändert sich nicht, wenn sich die Containergröße verändert.
Left, Bottom	Das Objekt bewegt sich hoch/runter wenn sich die Containerhöhe ändert. Die Position des Objektes im Verhältnis zur Containerunterkante ändert sich nicht.
Left, Top, Bottom	Ändert sich die Containerhöhe, ändert sich die Objekthöhe synchron.
Left, Top, Right, Bottom	Ändern sich die Containerhöhe und -breite, vergrößert oder verkleinert sich das Objekt synchron.

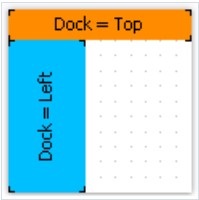
Eigenschaft Dock

Alle Berichtsobjekte weisen diese Eigenschaft auf. Diese Eigenschaft legt fest an welcher Seite des Containers das Objekt angedockt wird.

Die Eigenschaft "Andocken" kann die folgenden Werte aufweisen:

Wert	Beschreibung
Keine	Standardwert. Das Objekt ist nicht angedockt.
Links	Das Bild wird in der linken Objektkante angezeigt. Die Objekthöhe entspricht der Containerhöhe*.
Oben	Das Bild wird an der linken Containerkante angedockt. Die Objektbreite entspricht der Containerbreite*.
Rechts	Das Objekt wird an der rechten Containerkante angedockt. Die Objekthöhe entspricht der Containerhöhe*.
Unten	Das Objekt wird an der unteren Containerkante angedockt. Die Objektbreite entspricht der Containerbreite*.
Ausfüllen	Das Objekt füllt den gesamten freien Platz des Containers aus.

- Dem ist nicht so, wenn mehrere Objekte zur gleichen Zeit angedockt sind. Die folgende Abbildung zeigt zwei Objekte; das erste wurde an die obere Containerkante angedockt und das zweite an die linke Kante:



Wie man sehen kann entspricht die Höhe des zweiten Objektes der Höhe des freien Raumes, der nach Andockung des ersten Objektes verbleibt.

Das Andockverhalten ist abhängig von der Reihenfolge der Objekterstellung. Die Reihenfolge des Kontextmenüs des Objektes kann geändert werden. Hierzu kann entweder das Menüelement "In den Vordergrund" oder "In den Hintergrund" genutzt werden.

Formatierung

In diesem Abschnitt werden die folgenden Aspekte näher betrachtet:

· Änderung des Aussehens eines Objektes;

· Änderung des Formats des Druckwertes;

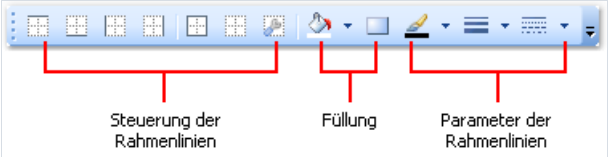
· Automatische Änderung des Aussehens eines Objektes bei der Erfüllung irgendeiner Art von Bedingung;

· Verbergen unnötiger Werte;

· Hervorheben gerade Datenzeilen in unterschiedlichen Farben.

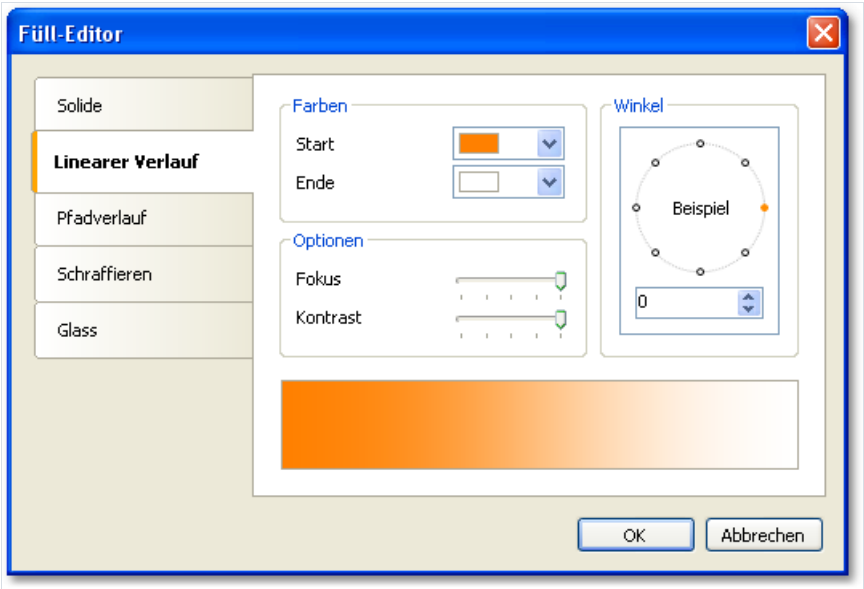
Rahmen und Füllung

Nahezu alle Berichtsobjekte haben einen Rahmen und eine Füllung. Um mit diesen Eigenschaften zu arbeiten wird die Symbolleiste "Rahmen und Füllung" genutzt



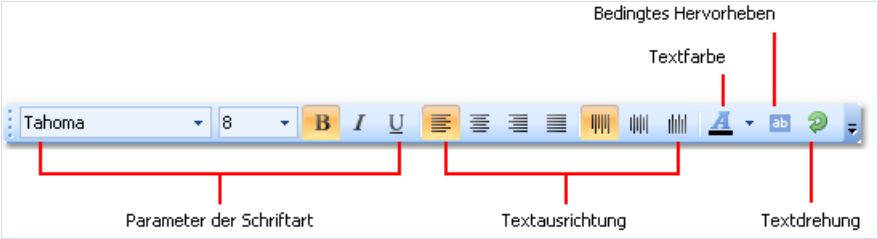
Der Objektrahmen setzt sich aus vier Linien zusammen. Jede Linie kann eine unterschiedliche Breite, Farbe oder einen unterschiedlichen Stil annehmen. Die Schaltfläche der Symbolleiste beeinflusst alle Linien des Rahmens. Die Schaltfläche zeigt einen Dialog an, der die Festlegung jeder einzelnen Linie ermöglicht:

Zur Arbeit mit Füllung werden auf der Symbolleiste zwei Schaltflächen angeboten. Die Schaltfläche ermöglicht die Auswahl einer Farbe für die einfarbige Füllart. Die Schaltfläche zeigt einen Dialog an, der eine Auswahl zwischen verschiedenen Füllarten ermöglicht:



Textformatierung

Zur Änderung des Aussehens der "Text"-Eigenschaft wird die Symbolleiste "Text" genutzt:



Stile

Zur Festlegung des Aussehens des Objektes können Stile genutzt werden. Ein Stil setzt sich aus den folgenden Eigenschaften zusammen:

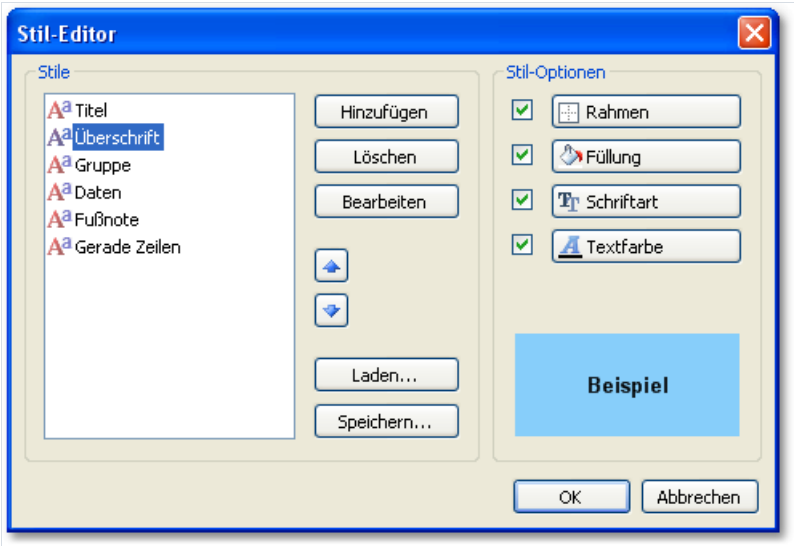
· Rahmen;

· Füllung;

· Schriftart;

· Textfarbe

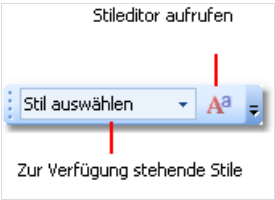
Die Liste der Stile wird im Bericht gespeichert. Diese können entweder über das Menü "Bericht|Stile" oder mittels der Schaltfläche in der Symbolleiste "Stil" gesteuert werden:



Die Eigenschaften eines Stils können folgendermaßen festgelegt werden:

· Festlegen der "Stil" Eigenschaften im Fenster "Eigenschaften";

· Nutzung der "Stil" Symbolleiste:



Sollte die Symbolleiste nicht auf dem Bildschirm sichtbar sein, kann diese über das Menü "Ansicht|Symbolleiste" aktiviert werden.

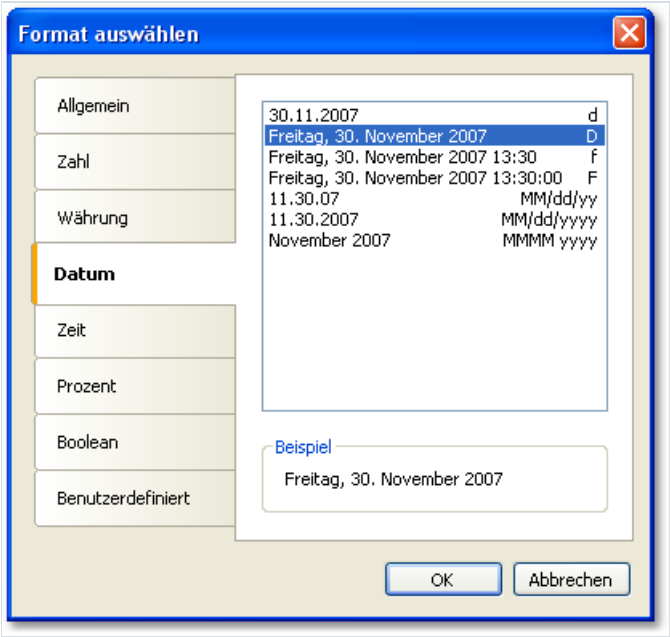
Bei der Einstellung des Objektstils, ändert sich das Aussehens des Objektes entsprechend der Einstellungen des Stils. Eine Änderung der Einstellungen des Stils ändert automatisch das Objekt das diesen Stil aufweist.

Datenformatierung

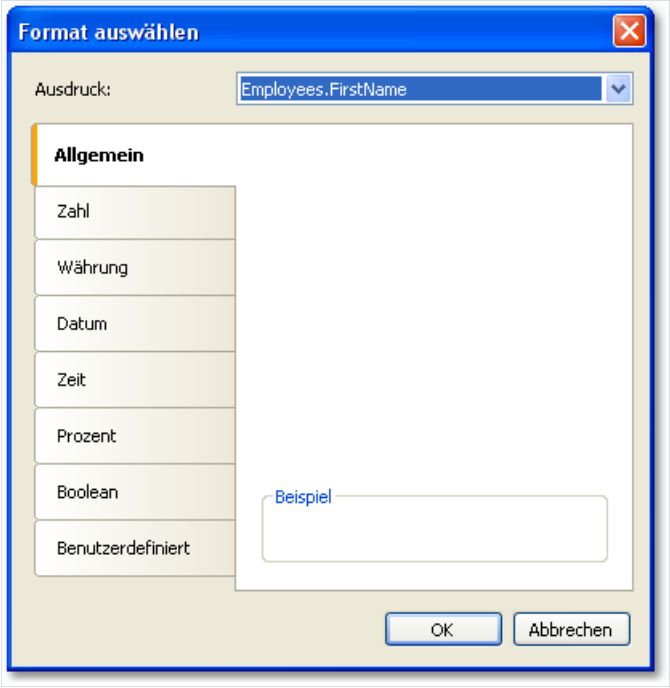
Zum Druck der Textdaten im Bericht wird das Objekt "Text" genutzt. Die Standardformatierung wird auf alle Daten, die aus den Datenquellen kommen, angewendet. Die Datenquellspalte des Typs "DateTime" (diese hängt von den Ländereinstellungen im System ab) wird beispielsweise folgendermaßen gedruckt:

```
11.10.2008 18:04:52
```

Soll nur das Datum gedruckt werden, muss die Datenformatierung entsprechend eingestellt werden. Zur Anzeige des Kontextmenüs wird mit der rechten Maustaste auf das "Text" Objekt geklickt. Dann wird das "Format"-Elements im Menü ausgewählt. Das Editorfenster zur Editierung wird angezeigt:



Hier wird eine der verfügbaren Formatierungstypen ausgewählt; es können auch eigene Formatierungszeichenfolgen festgelegt werden. Hierzu wird die "benutzerdefinierte" Formatierung ausgewählt. Enthält das "Text"-Objekt mehrere Datenspalten oder Ausdrücke, sollten geeignete Formate für jede/jeden ausgewählt werden. Hierzu wird der Ausdruck im oberen Fenster ausgewählt und dann das Format ausgewählt:

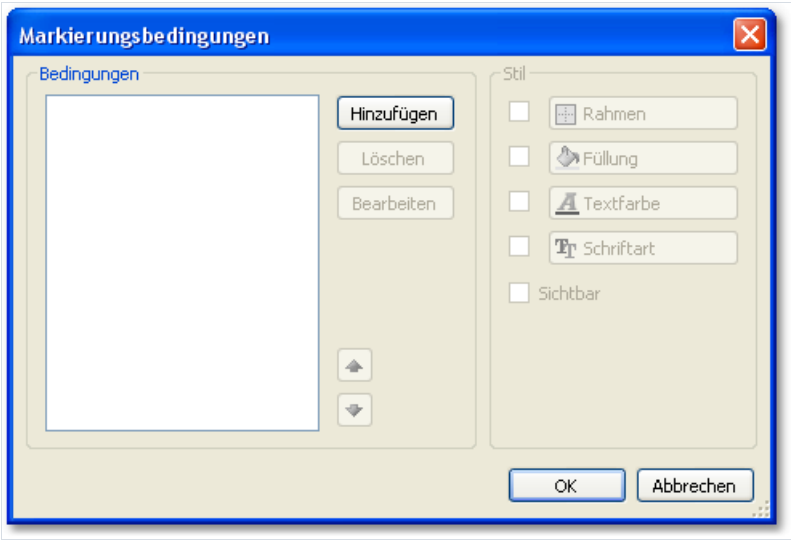


Die Daten können auch mittels der Zeichenfolgeformat-Methode formatiert werden. Für weitere Erläuterungen dieser Methode lesen Sie das MSDN.

```
Heute ist [Zeichenfolgeformat("{0:d}", [Datum])]
```

Bedingte Markierung

Es besteht die Möglichkeit das Aussehen der "Text"-Objekte, je nach gegebenen Bedingungen, zu ändern. Beispiel: Hervorheben eines Objektes, wenn dieses einen negativen Wert aufweist. Diese Eigenschaft wird "bedingtes markieren" genannt. Um dieses festzulegen wird das "Text" Objekt ausgewählt und auf die Schaltfläche auf der Symbolleiste "Text" gedrückt. Das folgende Dialogfenster wird angezeigt:



Es ist möglich eine oder mehrere Bedingungen, sowie den Stil für jede Bedingung, festzulegen. Der Stil kann eine oder mehrere Eigenschaften umfassen.

· Füllung;

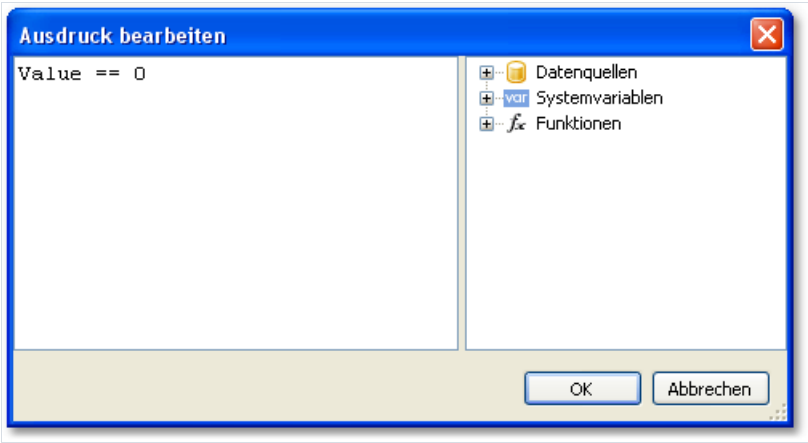
· Textfarbe;

· Schriftart;

· Sichtbarkeit des Objekts.

Es wird angezeigt welche Einstellungen zur Erfüllung der Bedingung verändert werden müssen. Aktivieren des Kontrollkästchens ändert die benötigte Einstellung. Standardmäßig umfasst ein neuer Stil eine Einstellung: die Textfarbe.

Die Schaltfläche "Hinzufügen" wird zur Erstellung einer neuen Bedingung geklickt. Ein Expression Editor wird eingeblendet.



Es ist möglich einen Ausdruck zu schreiben, der ein booleanisches Ergebnis zurückgibt. In den meisten Fällen wird

die "Wert"-Variable, die den aktuellen Duckwert aufweist, genutzt.

Beispiel: Wir haben ein "Text"-Objekt, in dem wird die Anzahl der Produkte auf Lager drucken:

```
[Products.UnitsInStock]
```

Ist der Anzahl der Produkte = 0 soll das Objekt rot dargestellt werden. Hierzu wird die folgende Bedingung erstellt:

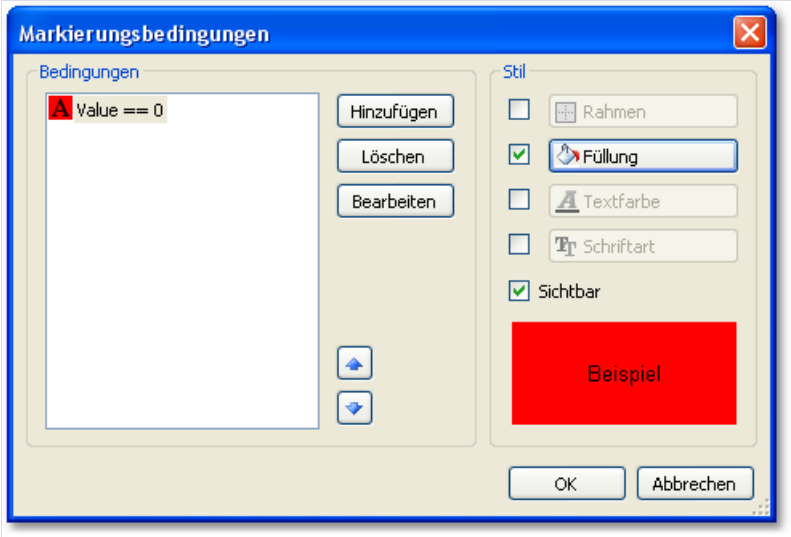
```
Value == 0
```

Hier wurde die "Wert"-Variable genutzt, die einen Druckwert aufweist. Hat das Objekt mehrere Ausdrücke, wird diese Variable den Wert des letzten Ausdrucks annehmen. Anstatt des "Wertes" kann auch die Datenspalte genutzt werden:

```
[Products.UnitsInStock] == 0
```

Der Ausdruck wird im Stil C# geschrieben. Dann ist die gewählte Berichtssprache C#. Für VisualBasic.Net muss das einfache "="-Symbol genutzt werden. Die Berichtssprache kann über das Menü "Bericht|Optionen" geändert werden.

Konfigurieren Sie den Stil für die gegebene Bedingung, so dass nur Füllung genutzt werden kann und wählen Sie die Farbe rot.



Wird ein Objekt mit dem Nullwert gedruckt, hat dieses die Farbe rot. Hier ein komplexeres Beispiel: Hinzufügen einer weiteren Bedingung. Erreichen die sich auf Lager befindlichen Einheiten einen Wert unter 10, sollen diese in gelb gedruckt werden. Hierzu wird der Editor für die Bedingung geöffnet und die Schaltfläche "Hinzufügen" geklickt. Die zweite Bedingung schaut folgendermaßen aus:

```
Value < 10
```

Sind mehrere Bedingungen, wie hier, angezeigt, überprüft FastReport alle Bedingungen und startet mit der ersten Bedingung. FastReport wendet die Einstellungen des Stil auf das Objekt an und der Prozess wird angehalten, wenn eine bestimmte Bedingung erfüllt ist. Es ist wichtig, die Bedingungen in die korrekte Reihenfolge zu bringen. In

diesem Beispiel ist die Reihenfolge korrekt:

```
1. Value == 0
2. Value < 10
```

Wird die Reihenfolge der Bedingungen geändert, wird das Falsche hervorgehoben.

```
1. Value < 10
2. Value == 0
```

In diesem Fall wird der Value==0 nicht ausgeführt, da die erste Bedingung erfüllt ist, wenn der Wert Null annimmt. Um die Reihenfolge der Bedingungen zu ändern, werden die Schaltflächen und genutzt.

Verbergen von Nullwerten

Das "Text"-Objekt weist die Eigenschaft "HideZeros" auf. Diese kann genutzt werden um Nullwerte zu verbergen. Hier ein Beispiel für ein Objekt mit den folgenden Inhalten:

```
Total elements: [CountOfElements]
```

Entspricht der Wert der Variablen AnzahlElemente 0 und die Eigenschaft HideZeros ist true, wird das Objekt wie folgt gedruckt:

```
Total elements:
```

Das "Text"-Objekt weist auch die Eigenschaft "HideValue" auf. Diese kann genutzt werden um den Wert eines Ausdrucks, der dem gegebenen Wert entspricht, zu verbergen. Ist beispielsweise der Wert der Eigenschaft 0, dann werden alle Nullfelder verborgen. Diese Eigenschaft kann auch zum Verbergen von Nulldaten genutzt werden. In der Regel handelt es sich hierbei um ein Datum, wie "1.1.0001" oder "1.1.1900". In einem solchen Fall sollte der Wert von "HideValue" wie folgt sein:

```
1.1.1900 0:00:00
```

Wie ersichtlich, muss neben dem Datum auch die Zeit angegeben werden. Dies ist notwendig, da der Wert des Datums in .Net auch die Zeit beinhaltet.

Wichtiger Hinweis: Dieser Mechanismus hängt von den Ländereinstellungen des Systems ab und kann in der Systemsteuerung eingestellt werden. Dies beruht auf das Tatsache, dass FastReport zwei Zeichenreihen vergleicht: "ToString()". Diese Methode konvertiert den Wert eines Ausdrucks in eine Zeichenfolge. Daher sollte man bei der Erstellung eines Berichts darauf achten, ob dieser auf einem Computer mit zwei unterschiedlichen Ländereinstellungen gestartet wird.

Schlussendlich erlaubt die Eigenschaft "NullValue" des "Text"-Objektes den Druck von Text anstelle eines Nullwertes. Diese wird häufig genutzt um einen Strich anstelle eines Nullwertes zu drucken. Hier ein Beispiel für ein Objekt mit den folgenden Inhalten:

```
Total elements: [CountOfElements]
```

Entspricht der Wert der Variable AnzahlElemente 0 und die Eigenschaft Nullwerte wurde als -- festgelegt, wird das Objekt wie folgt gedruckt:

```
Total elements: --
```

Verbergen duplizierter Werte

Das "Text"-Objekt weist die Eigenschaft "Duplicates" auf, die die Steuerung des Drucks duplizierter Werte ermöglicht. Diese Eigenschaft kann genutzt werden, wenn sich das "Text"-Objekt auf der "Daten"-Leiste befindet. Die Werte werden als Duplikate betrachtet, wenn sie in den nahe gelegenen Datenreihen gedruckt werden.

Die Eigenschaft "Duplikate" kann die folgenden Werte aufweisen:

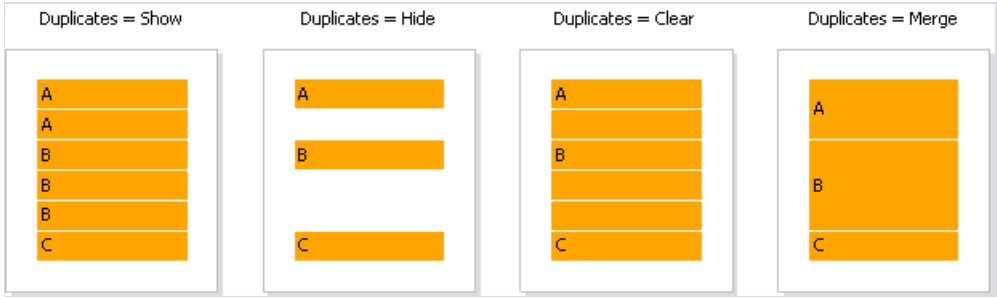
· Show - Anzeigen der Duplikate (standardmäßig).

· Hide - Verbergen der Objekte mit doppeltem Wert.

· Clear- Löschen des Objekttextes aber Anzeige des Objektes.

· Merge - Zusammenführen mehrerer Objekte mit gleichen Werten.

Der Unterschied zwischen den Modi ist in der folgenden Abbildung dargestellt:



Hervorheben ungerader/gerader Zeilen

Um das Aussehen eines Berichtes zu verbessern, können die geraden Datenzeilen in verschiedenen Farben hervorgehoben werden. Hierzu wird die Eigenschaft "EvenStyle" auf der Leiste oder das entsprechende Objekt genutzt. Die Eigenschaft beinhaltet einen Stilnamen, der zum Hervorheben gerader Leistenreihen genutzt wird.

Die Nutzung der Eigenschaft "EvenStyle" ist der Nutzung der Leiste vorzuziehen. Dies vermeidet mögliche Probleme beim Export des Berichts.

Zur Konfigurierung des Hervorgehobenen sind die folgenden Schritte zu beachten:

1. Festlegen des Stils zum Hervorheben der Zeilen. Dazu wird das Menü "Bericht|Stil" genutzt:
2. Anzeige des Namens des neuen Stils in der Eigenschaft "EvenStyle" der Leiste oder des entsprechenden Objektes.

Standardmäßig nutzen Objekte lediglich eine Fülleigenschaft des Stils der Eigenschaft "EvenStyle". Dieses Verhalten ist in der Eigenschaft "EvenStylePriority" festgelegt: der Standard lautet "UseFill". Zur Nutzung der übrigen Stil-Parameter stellt man diese Eigenschaft auf "UseAll" ein.

Ein fertiger Bericht, der sich diese Technik zu Eigen macht, kann folgendermaßen aussehen:

Product name	Unit price
Chai	18,00
Chang	19,00
Chartreuse verte	18,00
Côte de Blaye	263,50
Guaraná Fantástica	4,50
Ipoh Coffee	46,00

Bericht mit einer "Daten"-Leiste

Dieser Berichtstyp ist häufiger erforderlich. Er ermöglicht den Druck einer Liste an Zeilen aus der Datenquelle. Hierbei kann es sich z.B. um eine Kundenliste handeln.

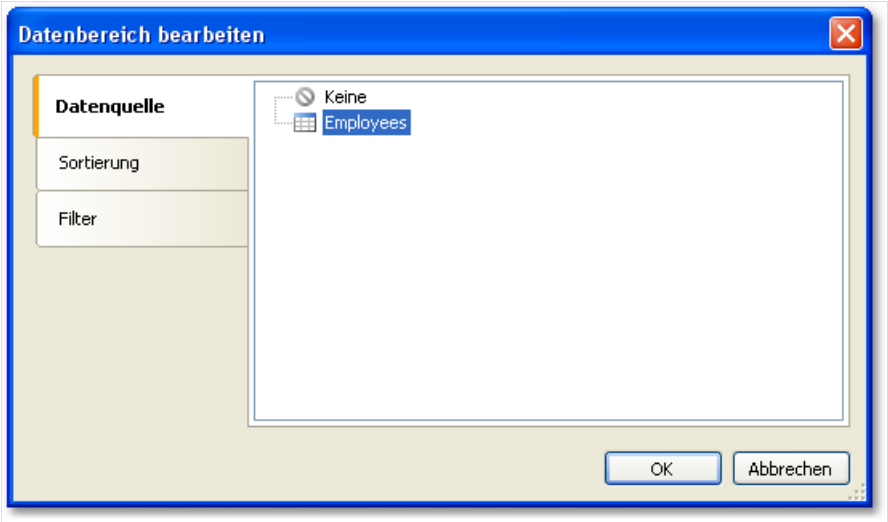
Verbinden einer Leiste mit einer Datenquelle

Zum Druck einer Leiste aus einer Datenquelle wird die "Daten"-Leiste genutzt, die mit der Datenquelle verbunden sein sollte. Die Leiste wird entsprechend der Anzahl der Zeilen in der Datenquelle gedruckt.

Sie wird einmal gedruckt, wenn die "Daten"-Leiste nicht mit der Quelle verbunden ist.

Ein neuer Bericht beinhaltet dieses bereits einige leere Leisten, einschließlich der "Daten"-Leiste. Diese Leiste kann auch über das Fenster "Bereich konfigurieren" in den Bericht eingefügt werden; Hierzu wählt man das Menüelement "Bericht|Bereich konfigurieren".

Durch Doppelklicken kann die Leiste mit den Daten verbunden werden. Dann erfolgt die Auswahl der Datenquelle im Editor-Fenster und das Klicken auf "OK".



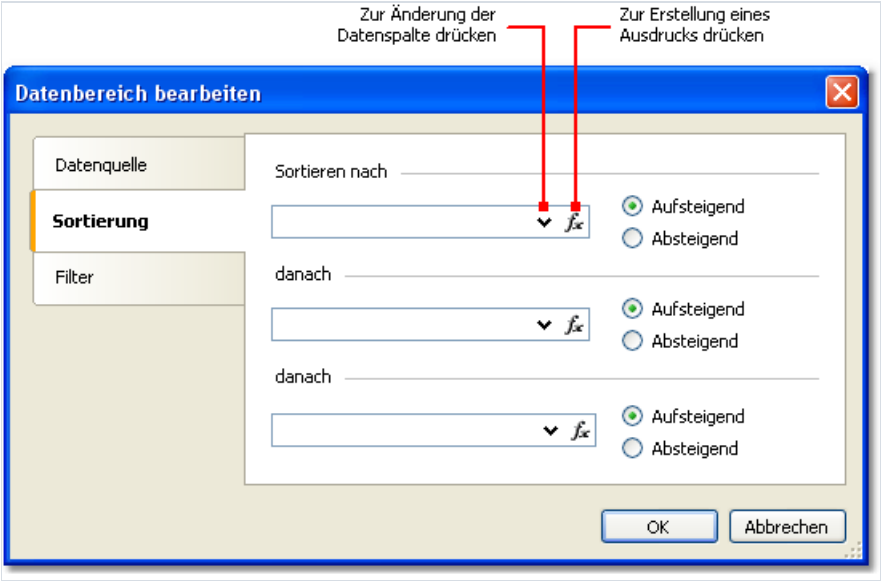
Druck des Textes

Nachdem die Leiste mit der Datenquelle verbunden wurde, kann die "Text"-Eigenschaft auf die Leiste platziert werden und dann die Informationen aus einer Datenspalte anzeigen. Am schnellsten lässt sich dies mittels des Ziehens und Fallenlassens (drag&drop) einer Datenspalte aus dem "Daten"-Fenster auf eine Leiste handhaben. Weitere Informationen zu dem "Text"-Objekt werden im Kapitel "Das Text-Objekt" erläutert.

Sortierung der Daten

Standardmäßig druckt die "Daten"-Leiste die Daten in natürlicher Reihenfolge. Es ist häufig notwendig, die Daten vor dem Druck zu sortieren. Eine Kundenliste kann z.B. praktisch in alphabetischer Reihenfolge präsentiert werden.

Die Sortierung wird über den "Daten"-Leisten Editor gesteuert. Um den Editor aufzurufen wird auf eine freie Stelle auf der Leiste geklickt:

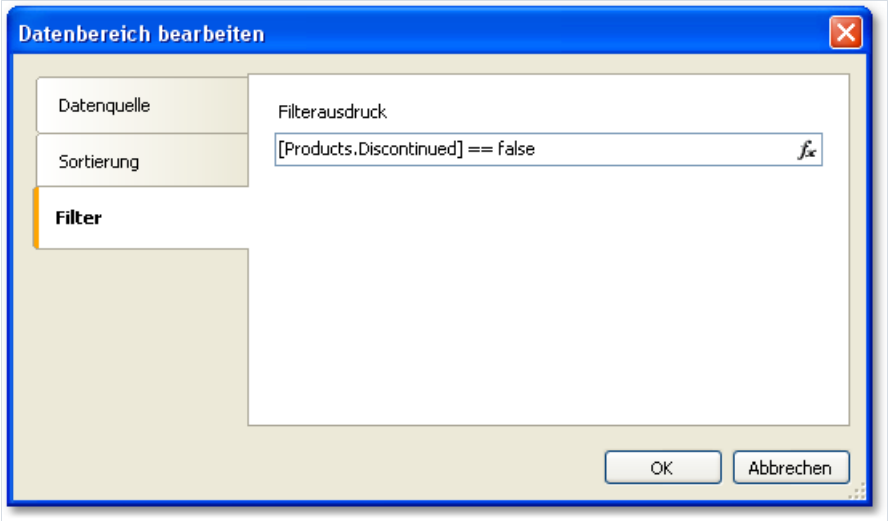


Als Sortierkriterium kann sowohl eine Datenspalte, als auch ein Ausdruck genutzt werden. Es können mehrere (höchstens drei) Bedingungen zur Sortierung angezeigt werden. Dies kann beispielsweise genutzt werden, wenn man die Kundenliste zuerst nach Stadt und danach nach Kundennamen zu sortieren wünscht. Für jede Bedingung kann eine Sortierrichtung (aufsteigend oder absteigend) gewählt werden.

Eine weitere Methode zur Datensortierung: Nutzung der SQL-Abfrage als Datenquelle. Die Abfrage wird auf dem Datenserver ausgeführt und gibt sortierte Zeilen zurück.

Filterung der Daten

Zur Filterung einer Zeile, die in der "Daten"-Leiste gedruckt ist, wird der Editor aufgerufen und zur Registerkarte "Filter" gewechselt:



Als ein Filterausdruck kann jeder korrekte Ausdruck aufgerufen werden. Detaillierte Informationen über Ausdrücke sind im Kapitel "Ausdrücke" erläutert.

Im obigen Beispiel wird der folgende Filter genutzt:

```
[Products.Discontinued] == false
```

Dies bedeutet, dass alle Datenreihen, die als Markierung gleich falsch sind, ausgewählt werden.

Es können komplexe Filterbedingungen genutzt werden:

```
[Products.Discontinued] == false && [Products.UnitPrice] < 10
```

Dies bedeutet, dass alle Datenreihen, die als Markierung gleich falsch sind und dessen Preis kleiner als 10 ist, ausgewählt werden.

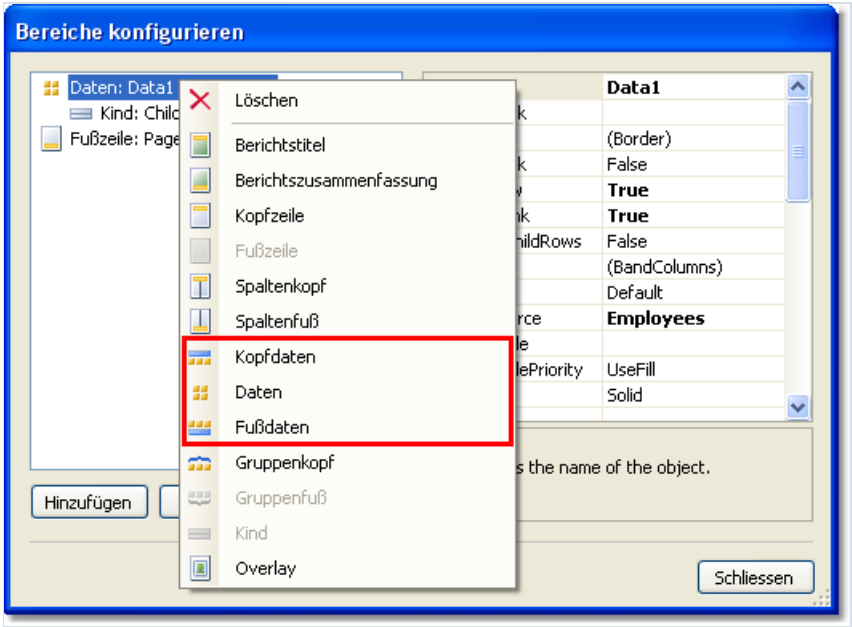
Diese Filtermethode setzt voraus, dass die Datenquelle alle Zeilen beinhaltet, die teilweise gefiltert werden. Sollte die Datenquelle eine große Zeilenanzahl aufweisen, kann dies den Bericht erheblich verlangsamen. In einem solchen Fall kann die SQL-Abfrage als Datenquelle genutzt werden, um die benötigte Filterung durchzuführen. Die Abfrage wird auf dem Datenserver ausgeführt und wird nur diejenigen Zeilen zurückführen, dessen Zeilen auch benötigt werden.

Es können auch die Dialogformulare zur Durchführung einer Datenfilterung genutzt werden. Weitere Details hierzu werden im Kapitel "Dialogformulare" erläutert.

Datenkopf und Datenfuß

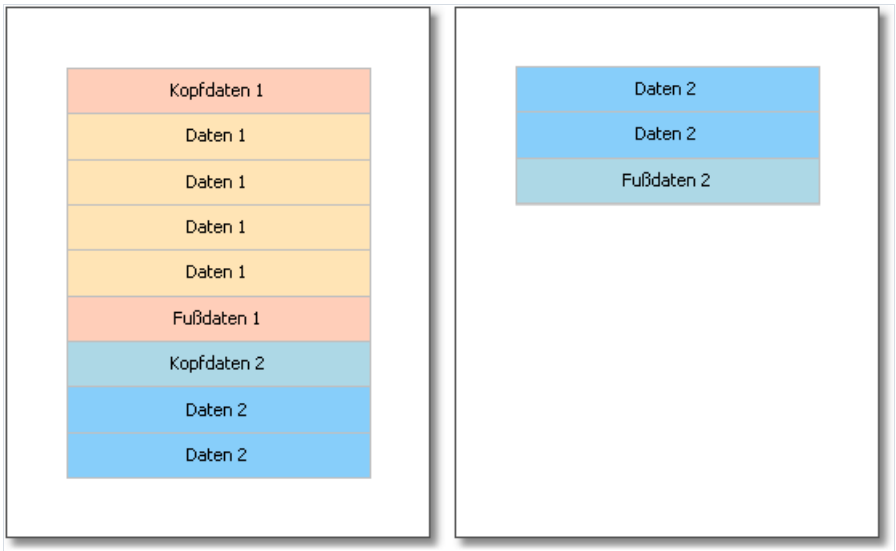
Eine "Daten"-Leiste kann eine Kopf-, und eine Fußzeile beinhalten. Die Kopfzeile wird einmal vor den Daten gedruckt; die Fußzeile wird nach dem Output aller Daten gedruckt.

Um eine Kopf-, und Fußzeile auf eine "Daten"-Leiste hinzuzufügen, wird das Menüelement "Bericht|Bereich konfigurieren..." gewählt. Die "Daten"-Leiste wird im Fenster gewählt und mit der rechten Maustaste geklickt. Im Kontextmenü wird das Element "Kopfzeile" und/oder "Fußzeile" gewählt:

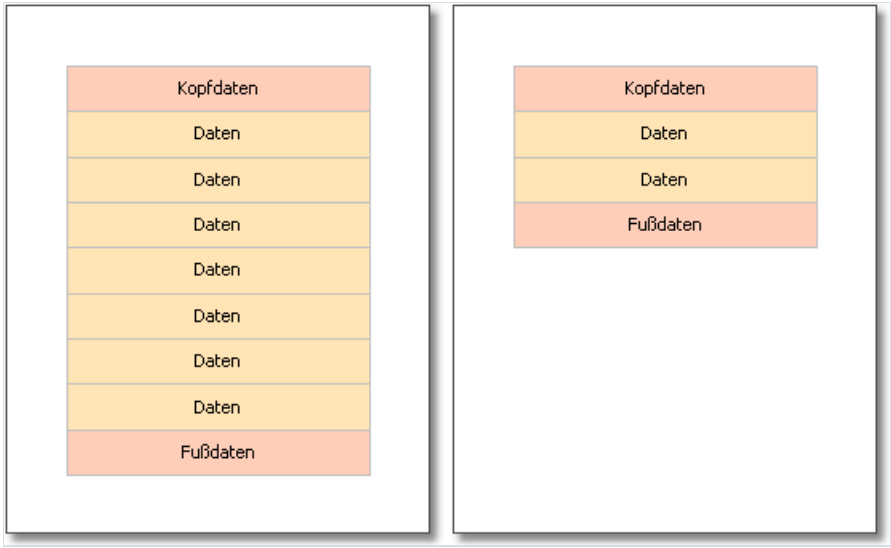


Diese Leisten können in den folgenden Situationen zweckdienlich sein:

- Beim Druck von mehreren Listen auf einer Seite ("Master-Master" Berichte). In diesem Fall hat jede Datenleiste ihre eigene Kopf-, und Fußzeile.



- Beim Druck einer Liste (falls diese Liste nicht auf ein Seite des erstellen Berichts passt). Durch Nutzung der Eigenschaft der Kopf-, und Fußzeile "Auf jeder Seite wiederholen" kann diese Leiste auf jeder Berichtsseite gedruckt werden.


Kopfdaten
Daten
Daten
Daten
Daten
Daten
Daten
Daten
Fußdaten
Kopfdaten
Daten
Daten
Fußdaten

Umbruch und Zusammenhalten von Daten

In diesem Abschnitt werden zwei Modi des Datendrucks näher erläutert: "Break" und "Keep".

In einem regulären Modus fpr den Leistendruck, überprüft FastReport ob ausreichend Platz zum Druck einer neuen Leiste zur Verfügung steht. Sollte dem nicht so sein, wird diese in ihrer Gesamtheit auf der nächsten Seite gedruckt. Ist die "CanBreak" Eigenschaft aktiviert, wird FastReport versuchen einen Teil der Leiste auf dem verfügbaren Platz zu drucken, d.h. diese "umbrechen".

Ein Umbruchversuch kann erfolgreich oder nicht erfolgreich sein. Dies hängt von dem Objekttyp, der auf die Leiste platziert wurde, und dessen Eigenschaften ab. Die folgenden Objekte können umgebrochen werden:

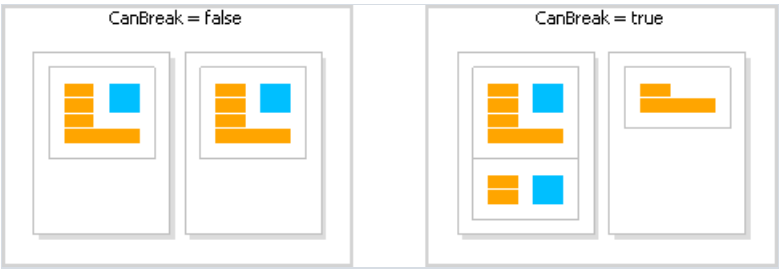
· "Text"

· "Rich Text"

· "Tabelle"

Auch diese Objekte weisen eine "CanBreak" Eigenschaft auf. Das Objekt kann umgebrochen werden, wenn diese aktiviert ist. Nicht-umbrechbare Objekte werden stets dort in ihrer Gesamtheit angezeigt, wo ausreichend Platz zur Verfügung steht.

Die folgende Tabelle stellt dar, wie eine Leiste umgebrochen werden kann.



> Umbruchalgorithmen funktionieren nicht immer einwandfrei. Es können Artefakte auftreten, wenn sich mehrere Objekte mit unterschiedlichen Schriftgrößen auf einer Leiste befinden.

Der Leistenumbruch dient der Platzersparnis auf der gedruckten Seite. Die Datenhaltung verfolgt das gegenteilige Ziel: Anzeige einer Reihe an Leisten, die sich alle auf einer Seite befinden. Dies resultiert in ungenutztem Platz auf einer Seite, die Daten sind jedoch so gedruckt, dass sie einfach wahrzunehmen sind.

Der Mechanismus "Zusammenhalten" ermöglicht das Zusammenhalten einer Reihe an Leisten auf einer Seite (oder, falls der Bericht diese aufweist, Spalte). Falls beim Druck die Daten das Ende der Seite erreichen, verschiebt FastReport alle Daten die bereits auf eine neue Seite gedruckt wurden.

"Zusammenhalten" kann in den folgenden Fällen genutzt werden:

· Druck aller Zeilen einer "Daten"-Leiste ;

· Druck aller Elemente einer Gruppe (Kopfzeile, Daten, Fußzeile);

· Druck der Zeile der Master-Datenquelle zusammen mit allen Detailzeilen (in dem "Master-Detail"-Bericht);

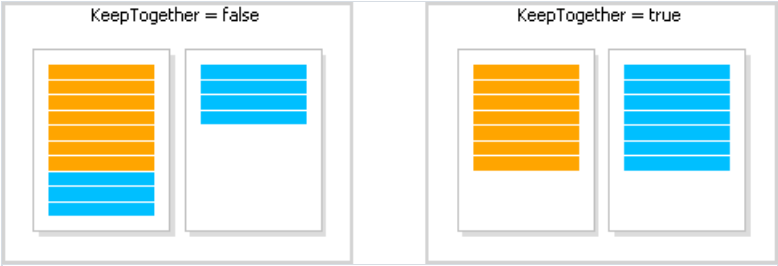
· Druck des Berichtstitels oder des Datenkopfes, zusammen mit mindestens einer Datenzeile;

· Druck der Fußzeile des Berichts zusammen mit mindestens einer Datenzeile;

· Druck der über-, und untergeordneten Leisten.

Im Folgenden wird der Mechanismus "Zusammenhalten" erläutert.

Zum Zusammenhalten aller Datenzeilen und Gruppenelemente (Kopfzeile, Daten, Fußzeile) wird die Eigenschaft "Zusammenhalten" aktiviert. Diese Eigenschaft wird in den Leisten "Daten" und "Gruppenkopf" genutzt. Die folgende Abbildung zeigt den Druck von Daten mit und ohne "Zusammenhalten":



Zum Zusammenhalten der Masterdatenzeile wird die Eigenschaft "Zusammenhalten" der Leiste "Daten" aktiviert. Diese Eigenschaft wird im Berichttyp "Master-Detail" genutzt:

Zur Vermeidung "hängender" Kopf-, oder Fußzeilen wird die Eigenschaft "KeepWithData" genutzt. Die folgenden Leisten haben diese Eigenschaft:

· Berichtstitel;

· Berichtszusammenfassung;

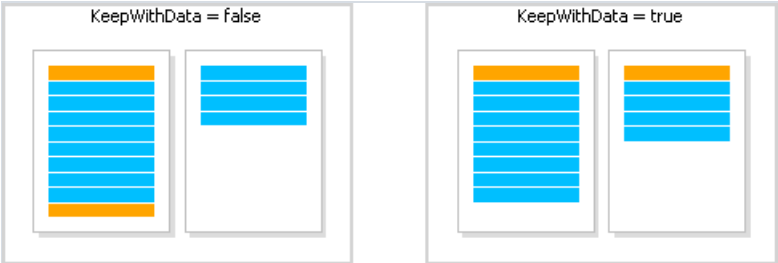
· Datenkopf;

· Datenfuß;

· Gruppenkopf;

· Gruppenfuß.

Diese Eigenschaft erlaubt das Beibehalten der Kopf-, Fußzeile mit mindestens einer Datenzeile:

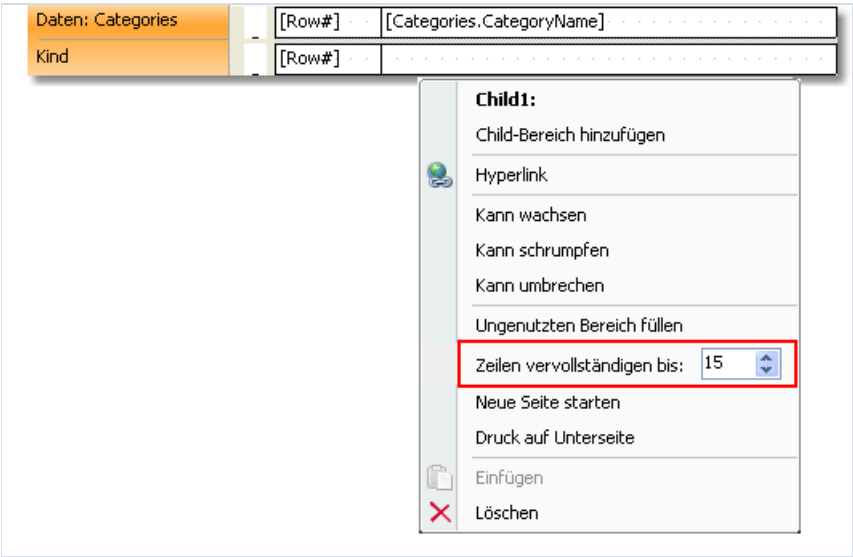


Um eine Leiste mit seinem entsprechenden untergeordneten Element beizubehalten, wird die Eigenschaft "KeepChild" genutzt.

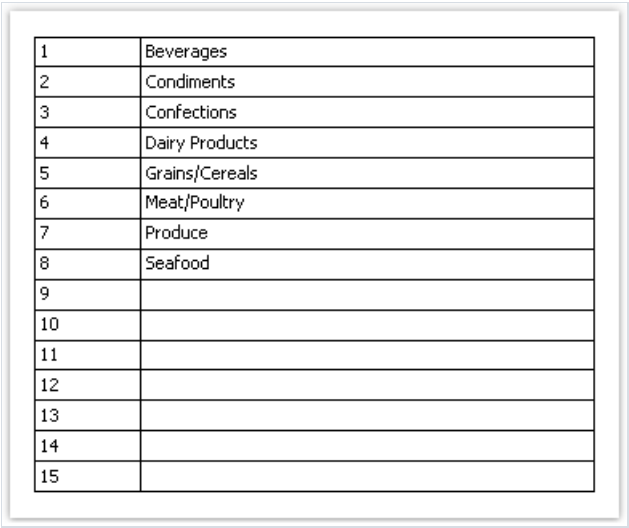
Drucken leerer Datenzeilen

Beim Druck vorgedruckter Formulare wird oftmals erwartet, dass eine bestimmte Anzahl an Datenzeilen gedruckt wird. Übersteigt die benötigte Zeilenanzahl die tatsächliche Datenmenge muss eine leere Zeile gedruckt werden. Dies kann mittels der Leiste "Kind" durchgeführt werden, indem diese an die Leiste "Daten" angehängt wird.

Die Leiste "Kind" hat eine Eigenschaft namens "CompleteToNRows". Ist die Eigenschaft auf einen Wert größer als 0 eingestellt, wird die Leiste für zusätzliche Datenzeilen bis zur anzeigten Höhe genutzt. Beispiel: Es sollen 15 Zeilen gedruckt werden, es existieren jedoch lediglich 8 Zeilen in der Datenquelle. In diesem Fall wird die Leiste "Kind" 7-Mal gedruckt.



Der erstellte Bericht schaut folgendermaßen aus:



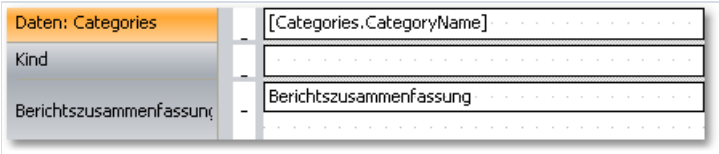
1	Beverages
2	Condiments
3	Confections
4	Dairy Products
5	Grains/Cereals
6	Meat/Poultry
7	Produce
8	Seafood
9	
10	
11	
12	
13	
14	
15	

Hat die Datenquelle mehr Zeilen als in der Eigenschaft "CompleteToNRows" angezeigt, wird die leere Zeile nicht gedruckt.

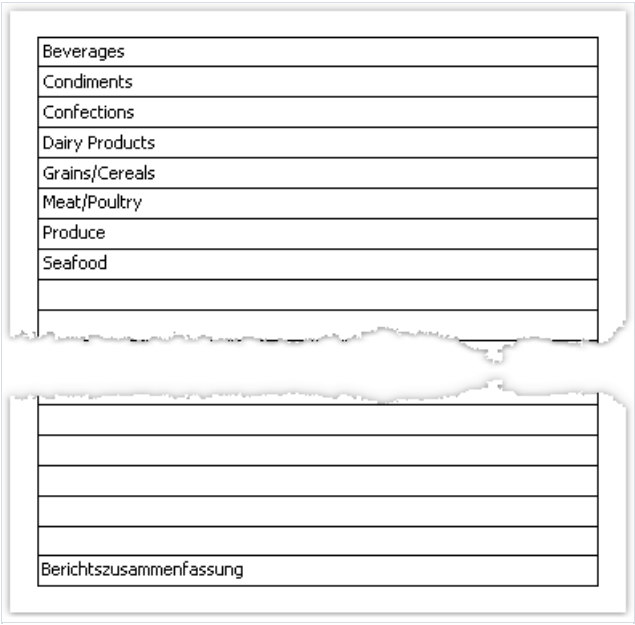
Eine weitere Möglichkeit eine leere Zeile zu drucken besteht darin, den freien Raum auf der Seite auszufüllen. In einem solchen Fall ist die Leiste "Untergeordnetes Element", entweder an die Leistentypen "Datenfuß", oder "Gruppenfuß" angehängt, und füllt den freien Raum auf der Seite aus. Die Fußzeilenleiste wird am unteren Ende der Seite gedruckt.

Um auf diese Art eine leere Zeile zu drucken, muss die Leiste "Untergeordnetes Element" an die Fußzeilenleiste

angehängt werden und die Eigenschaft "FillUnusedSpace" aktiviert werden. Man kann erkennen, dass die Leiste Kind nun über der Leiste, an die sie angehängt ist, angezeigt wird. In der folgenden Abbildung ist die Leiste "Kind" an die Leiste "Berichtszusammenfassung" angehängt:



Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:



Drucken des Textes "Keine Daten"

Ist die Datenleiste mit einer leeren Datenquelle verbunden wird diese nicht gedruckt. Es kann erforderlich werden einen Text, wie beispielsweise "Keine Daten", anstelle einer leeren Seite zu drucken. Um dies zu tun:

· Hinzufügen einer Kindleiste zu einer Datenleiste;

· Festlegen der Eigenschaft PrintIfDatabandEmpty der Kindleiste auf true (dies findet im Fenster "Eigenschaften" statt);

· Legen des "Text"-Objektes auf die Kindleiste und Eingabe von "Keine Daten vorhanden".

Der Text wird wie folgt gedruckt:

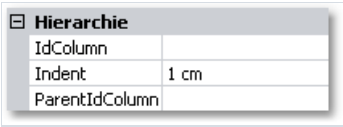
- Hat die Datenquelle einige Datenzeilen, wird die Datenleiste zusammen mit allen verwandten Leisten (Datenkopf, Datenfuß) gedruckt;
- Bei leere Datenquelle wird lediglich die Kindleiste mit dem Text "Keine Daten vorhanden" gedruckt.

Druckhierarchie

Die Leiste "Daten" ermöglicht den Druck einer hierarchischen Liste. Hierzu werden eine Datenleiste und eine Datenquelle benötigt. Die Hierarchie muss in der Datenquelle mittels zweier Datenspalten festgelegt sein:

1. Schlüsselspalte. Dies ist der Bezeichner der Datenzeile.
2. Spalte die den Schlüssel des übergeordneten Elementes enthält.

Zum Druck einer solchen Quelle in hierarchischer Form, müssen die folgenden Eigenschaften der "Daten"-Leiste festgelegt werden. Dies kann im Fenster "Eigenschaften" geschehen:



- Anzeige der Schlüsselspalte in der Eigenschaft "IdColumn";

· Anzeige der Spalte, die den Wert des übergeordneten Elementes beinhaltet, in der Eigenschaft "ParentIdColumn";

· Anzeige des Einzugs der Hierarchie in der Eigenschaft "Einzug";

Lernen Sie hier, wie Sie eine Mitarbeiterhierarchie aus der Demotabelle "Employees" drucken. Die Tabelle weist zwei benötigte Spalten auf:

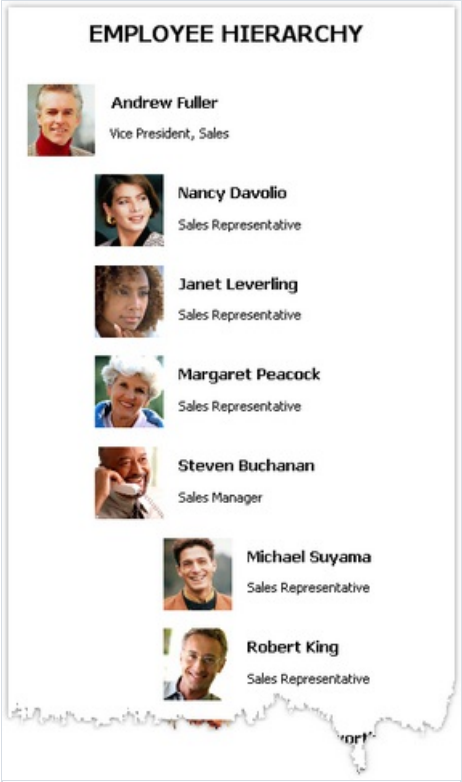
· Die Spalte EmployeeID ist der Schlüssel und beinhaltet die Mitarbeiter ID;

· Die Spalte ReportsTo beinhaltet die ID des "übergeordneten Elementes" Mitarbeiter.

Erstellen Sie einen Bericht, der folgendermaßen aussieht:

Festlegen der Eigenschaft "Daten"-Leiste, die für die Hierarchie zuständig ist, auf folgende Art:

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:



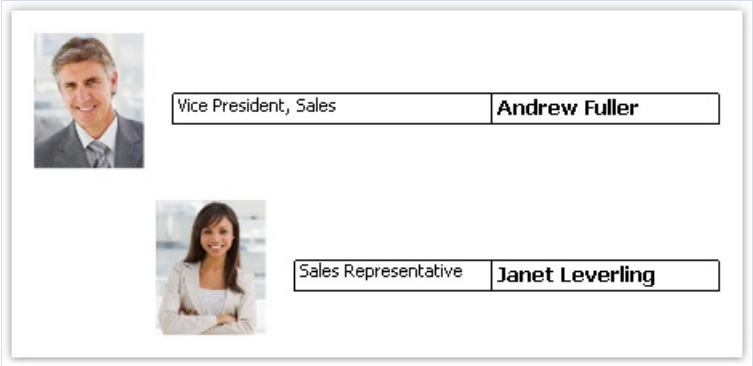
Beim Druck der Hierarchie verschiebt FastReport die Leiste nach rechts (um einen in der Eigenschaft Einzug festgelegten Wert) und verringert gleichzeitig die Leistenbreite um den gleichen Wert. Dies ermöglicht die Nutzung der Eigenschaft Anker des Leistenobjektes. Hier die möglichen Werte dieser Eigenschaft, die in einem solchen Fall genutzt werden können:

· Left, Top (standardmäßig): Das Objekt wird mit der Leiste verschoben;

· Right, Top: Das Objekt verbleibt ihn seiner ursprünglichen Position;

· Left, Right, Top: Die rechte Seite des Objektes verbleibt in der Originalposition, die linke Seite wird mit der Leiste verschoben.

Es ermöglicht einige nützliche Effekte:



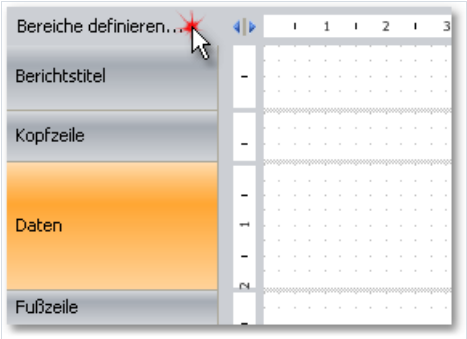
In diesem Beispiel weist das Bildobjekt die Ankereigenschaft links, oben auf; Das Objekt mit der Berufsbezeichnung ist links, rechts, oben verankert. Das Objekt mit dem Namen ist rechts, oben verankert.

Master-Detail Bericht

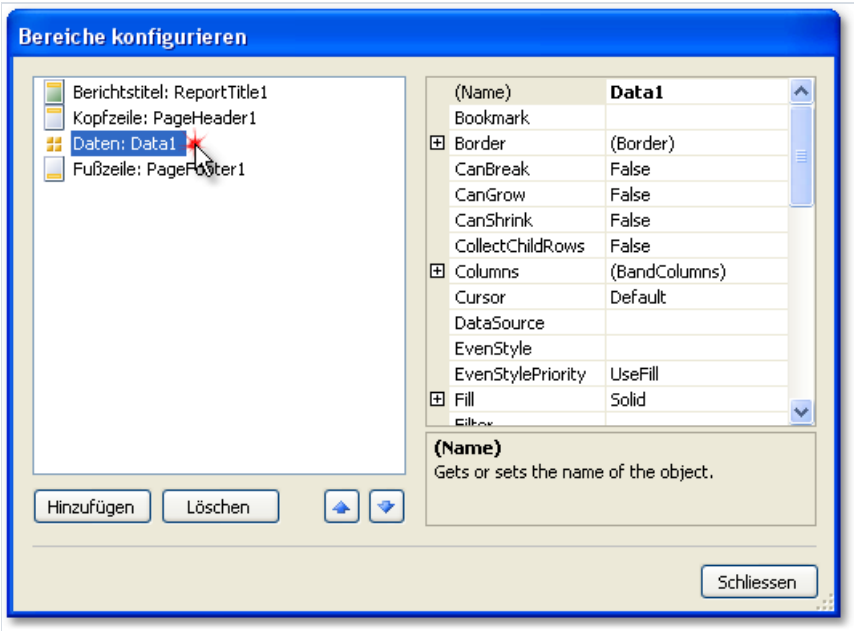
Die Nutzung zweier "Daten"-Leisten vereinfacht die Erstellung eines Berichts des "Master-Detail" Typen. In diesem Bericht werden zwei Datenquellen, zwischen denen eine Beziehung besteht, genutzt. Eine Zeile der Masterquelle kann mit mehreren Zeilen der Detailquelle korrespondieren. Detaillierte Informationen über Beziehungen sind im Kapitel "Daten" erläutert.

Die Leiste muss so in den Bericht eingefügt werden, dass die Masterleiste die Detailleiste beinhaltet. Dies kann im Fenster "Bereiche konfigurieren", die über das Menü "Bericht|Bereiche konfigurieren..." aufgerufen wird, eingestellt werden.

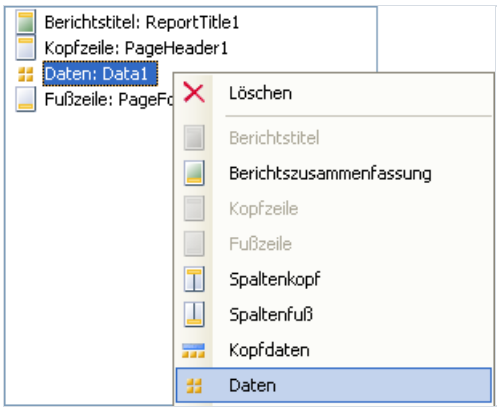
Hier eine Erläuterung der Erstellung eines "Master-Detail" Berichts von Grund auf. Hierzu wird der Bericht-Designer genutzt und ein leerer Bericht erstellt. Dieser beinhaltet bereits eine "Daten"-Leiste:



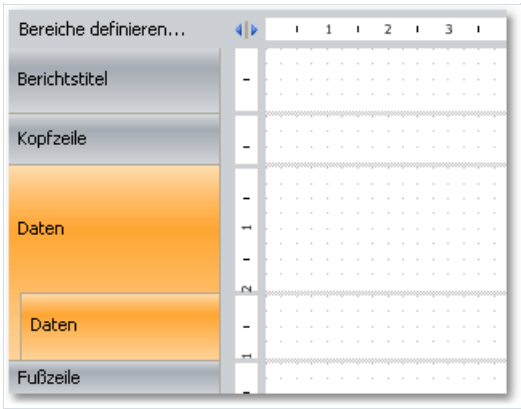
Um eine Detailleiste hinzuzufügen, wird das Fenster "Bereiche konfigurieren" aufgerufen. Dies kann durch Drücken der Schaltfläche "Leiste konfigurieren..." (siehe Abbildung), oder durch die Auswahl des Menüelements "Bericht|Bereiche konfigurieren..." erfolgen. Die Leistenstruktur wird im Konfigurationsfenster angezeigt:



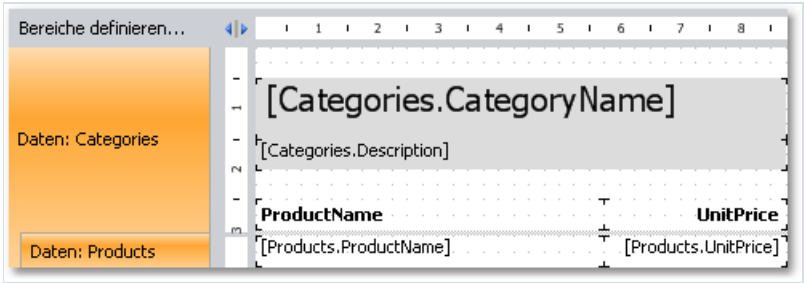
Wie in der Abbildung gezeigt, wird die "Daten"-Leiste ausgewählt und mit der rechten Maustaste geklickt um das Kontextmenü anzuzeigen (oder durch Drücken der Schaltfläche "Hinzufügen" im unteren Teil des Fensters). Auswahl von "Daten"-Leiste im sich öffnenden Fenster:



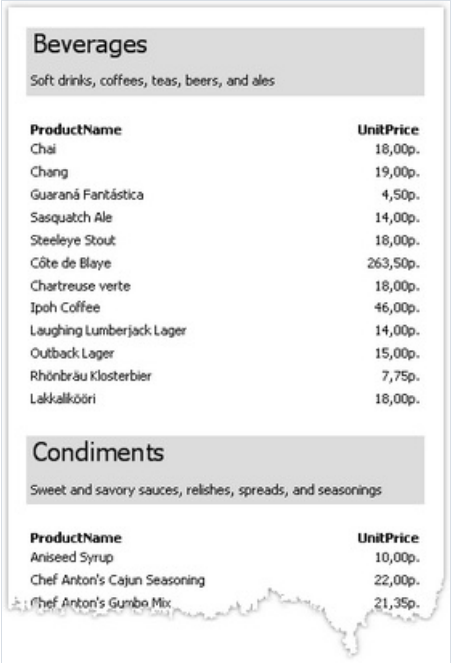
Hiernach werden geschachtelte "Daten" zur ausgewählten Leiste hinzugefügt. Das Fenster wird durch Klicken der Schaltfläche "Schließen" geschlossen. Hier ist ersichtlich, dass sich die Berichtvorlage auf folgende Weise ändert:



In der linken Seite des Fensters lassen sich deutlich geschachtelte Leistenstrukturen erkennen. Hiernach muss die Leiste mit der korrespondierenden Datenquelle verbunden und die Datenspalten auf die Leiste positioniert werden. Wir werden zwei Datenquellen aus der Demodatenbank, die mit FastReport geliefert wird, nutzen: Kategorien und Produkte.



Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:



Beverages

Soft drinks, coffees, teas, beers, and ales

ProductName	UnitPrice
Chai	18,00p.
Chang	19,00p.
Guaraná Fantástica	4,50p.
Sasquatch Ale	14,00p.
Steeleye Stout	18,00p.
Côte de Blaye	263,50p.
Chartreuse verte	18,00p.
Ipoh Coffee	46,00p.
Laughing Lumberjack Lager	14,00p.
Outback Lager	15,00p.
Rhönbräu Klosterbier	7,75p.
Lakkalikööri	18,00p.

Condiments

Sweet and savory sauces, relishes, spreads, and seasonings

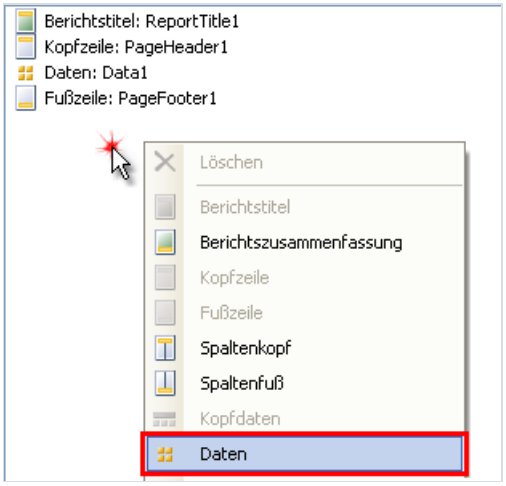
ProductName	UnitPrice
Aniseed Syrup	10,00p.
Chef Anton's Cajun Seasoning	22,00p.
Chef Anton's Gumbo Mix	21,35p.

Auf diese Art lässt sich ein Master-Detail-Bericht mit einer uneingeschränkten Anzahl geschachtelter Daten erstellen , z.B. Master-Detail-Teildetail. Eine weitere Methode zur Erstellung eines Master-Detail Berichtstypen verbunden mit einem geschachtelten Bericht: Geschachtelte Bericht werden im Abschnitt "Teilberichte" näher betrachtet.

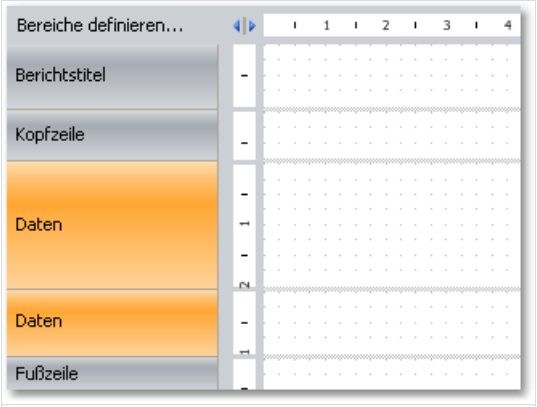
Master-Master-Bericht

Auf der Berichtsseite können mehrere einfache Listen gedruckt werden. Dies kann durch Positionierung einer oder mehrerer "Daten"-Leisten auf die Seite erreicht werden. Im Gegensatz zum Master-Detail-Bericht, in dem Leisten ineinander verschachtelt sind und Daten aus verwandten Quellen gedruckt werden, hängen in einem solchen Berichtstyp sowohl Leisten als auch Datenquellen nicht voneinander ab.

An einem Beispiel soll gezeigt werden, wie ein Bericht, der zwei Listen (Kategorientabelle und Kundentabelle) auf einer Seite druckt, erstellt werden kann. Es wird ein neuer Bericht erstellt und die benötigten Datenquellen hinzugefügt. Um eine zweite "Daten"-Leiste hinzuzufügen, wird das Fenster "Leiste konfigurieren" aufgerufen.



Mit der rechten Maustaste auf einen leeren Raum auf der Liste klicken (siehe Abbildung) und Auswahl der Leiste "Daten" im Kontextmenü. Dies erstellt eine neue unabhängige "Daten"-Leiste. Die Berichtsvorlage schaut folgendermaßen aus:



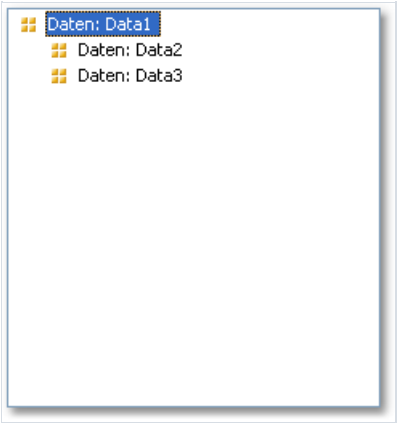
Nun wird die Leiste mit der Datenquelle verbunden und mehrere Datenspalten auf ihr positioniert:

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

Beverages
Condiments
Confections
Dairy Products
Grains/Cereals
Meat/Poultry
Produce
Seafood
Alfreds Futterkiste
Ana Trujillo Emparedados y helados
Antonio Moreno Taquería
Around the Horn

Master-Detail-Detail-Bericht

Eine "Daten"-Leiste kann eine oder mehrere geschachtelte "Daten"-Leisten beinhalten. Dies ermöglicht die Erstellung eines Berichtes des Typen Master-Detail-Detail. Hierzu wird das Fenster "Leisten konfigurieren" aufgerufen, mit der rechten Maustaste auf die Master "Daten"-Leiste geklickt und eine Detail "Daten"-Leiste hinzugefügt. Dieser Vorgang wird wiederholt, um eine zweite Detailleiste hinzuzufügen:



Somit ist es möglich eine unbeschränkte Anzahl an Detailleisten zu der Master "Daten"-Leiste hinzuzufügen. Eine Beispielberichtsstruktur kann folgendermaßen aussehen (dies ist lediglich als Beispiel und soll verdeutlicht die Möglichkeiten, die FastReport bietet, verdeutlichen):

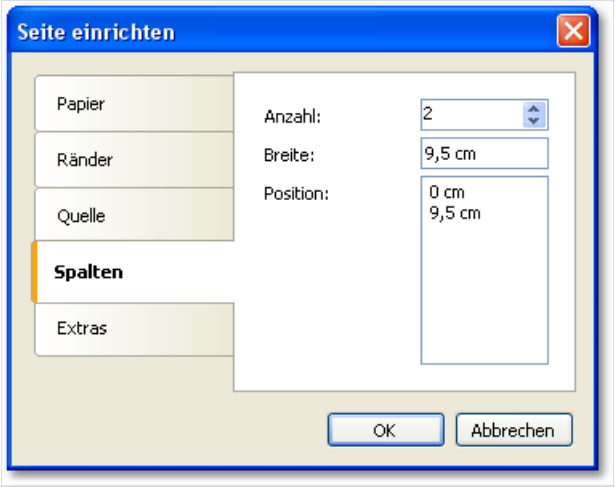
Mehrspaltige Berichte

Ein regulärer Bericht druckt die Daten solange diese nicht das Seitenende erreicht haben. Hiernach wird eine neue Seite gebildet und auf dieser gedruckt. Ein Bericht mit Spalten druckt Daten in mehreren Spalten. Ist das Seitenende erreicht, setzt sich der Druck in einer neuen Spalte auf derselben Seite fort. In diesem Sinne kann ein regulärer Bericht mit einer Spalte betrachtet werden.

Zum Druck von Spalten bietet FastReport zwei Methoden.

Seitenspalten

Die erste Methode basiert auf der Festlegung der Anzahl der Spalten des Berichts. Dies geschieht mittels des Fensters "Seite einrichten" auf der Leiste "Spalten":



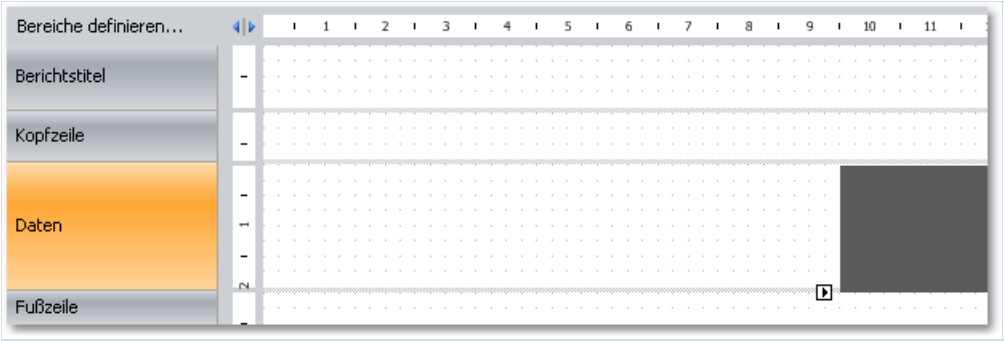
Wie bereits erwähnt können die folgenden Spaltenparameter festgelegt werden:

· Spaltenanzahl;

· Spaltenbreite;

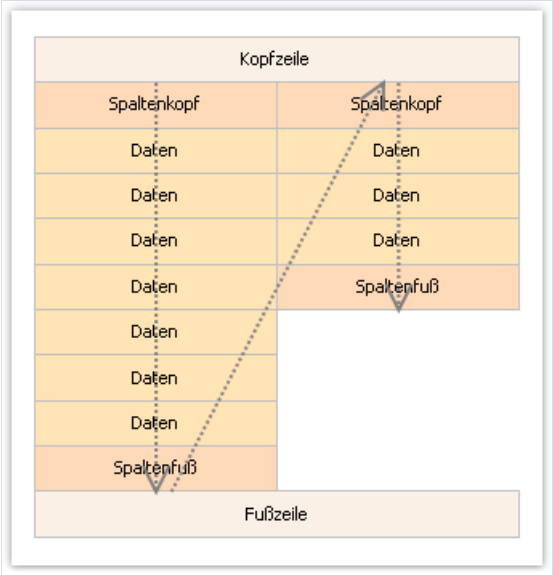
· die Position jeder Spalte.

Um einen regulären Bericht in einen Bericht mit Spalten umzuwandeln, muss lediglich die Anzahl der Spalten auf der Seite festgelegt werden. Die restlichen Parameter werden von FastReport automatisch berechnet. Bei der Aktivierung von Spalten ändert sich der Modus im Designer:



Der in grau angezeigte Bereich sollte niemals zur Positionierung von Objekten genutzt werden. Dieser wird zum Druck des nächsten Spaltenobjektes genutzt.

Zur Arbeit mit Spalten werden die Leisten "Spaltenkopf" und "Spaltenfuß" genutzt. Wie der Name vermuten lässt, werden diese am oberen und unteren Teil jeder Spalte gedruckt. Die folgende Abbildung stellt die Druckreihenfolge der Leisten, in einem Bericht mit Spalten, dar:


Kopfzeile
Spaltenkopf
Spaltenkopf
Daten
Daten
Daten
Daten
Daten
Daten
Daten
Daten
Daten
Daten
Spaltenfuß
Spaltenfuß
Fußzeile

Datenleiste Spalten

Auch die Leistenspalte "Daten" kann zum Druck eines mehrspaltigen Berichts genutzt werden. Die übrigen Leisten werden weiterhin in einer Spalte gedruckt.

Die Spaltenparameter können mittels der Eigenschaft "Spalten", die im Fenster "Eigenschaften" geändert werden kann, konfiguriert werden. Die folgenden Parameter können festgelegt werden:

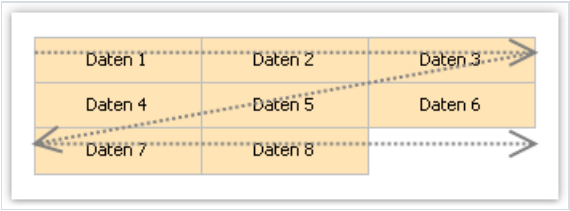
· Spaltenanzahl;

· Spaltenbreite;

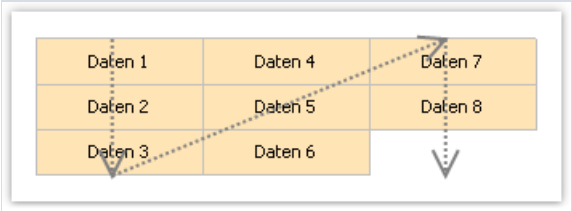
· Druckmodus der Spalten. Es kann einer der beiden folgenden Modi ausgewählt werden: "AcrossThenDown" und "AcrossThenDown";

· die Mindestanzahl der Zeilen in einer Spalte bei Auswahl des Modus "AcrossThenDown".

Die Spaltenleiste kann in beiden Modi gedruckt werden. Im Modus "AcrossThenDown" (Standardmodus) werden Spalten auf die folgende Art gedruckt:



Im Modus "DownThenAcross" werden die Spalten folgendermaßen gedruckt:



In diesem Modus berechnet FastReport die Anzahl der Datenzeilen in einer Spalte, so dass diese gleichermaßen gefüllt werden. Die Mindestanzahl an Zeilen in einer Spalte kann mittels der Eigenschaft "Columns.MinRowCount" eingestellt werden.

Berichte in Broschürenart

Der Druck eines Berichts im Broschürenformat stellt die folgenden Herausforderungen:

· Getrennte Berichtsseiten: Titelblatt, Inhaltsverzeichnis, Inhalt des Berichts, Rückseite;

· Unterschiedliche Seitenränder für gerade und ungerade Seiten;

· Unterschiedliche Kopf-, und Fußzeilen für gerade und ungerade Seiten.

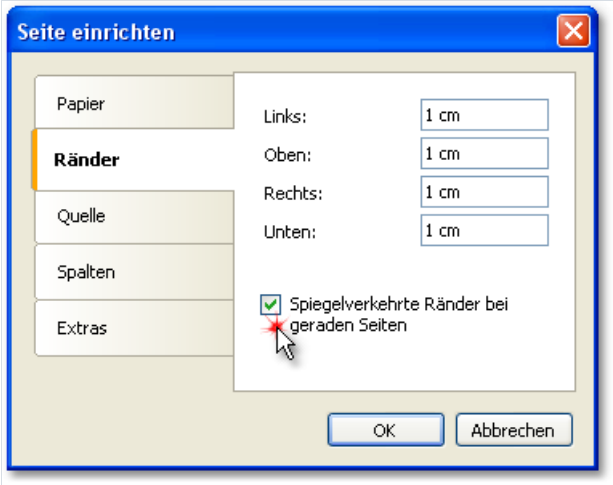
Hinzufügen von Berichtsseiten

Es kann eine beliebige Seitenanzahl in die Berichtsvorlage integriert werden. Auf jeder Seite kann ein eigener Bericht positioniert werden. Um eine neue Seite hinzuzufügen, wird auf der Symbolleiste die Schaltfläche gedrückt. Auch durch Klicken der Schaltfläche "Neue Seite" und Auswahl des Elements "Seite" in dem Fenster, kann eine neue Seite hinzugefügt werden.

> Zum Erstellen des Abschnitts "Inhaltsverzeichnis" kann die im Kapitel "Interaktive Berichte" erläuterte Methode genutzt werden.

Seiteneinstellungen

Im Fenster "Seite einrichten" kann festgelegt werden, dass FastReport den rechten und linken Seitenrand aller geraden Seiten spiegelt.



Beginnt die Seite mit einer ungeraden Zahl, stellt man die Eigenschaft "StartOnOddPage" auf true. Bei Bedarf druckt FastReport die leere Seite vor der angegebenen Seite.

Druck auf geraden/ungeraden Seiten

Alle Berichtsobjekte haben die Eigenschaft "PrintOn". Mit Hilfe dieser Eigenschaft können unterschiedliche Objekte auf gerade und ungerade Seiten gedruckt werden.

> Diese Eigenschaft kann im Fenster "Eigenschaften" festgelegt werden.

Diese Eigenschaft legt fest auf welchen Seiten das Objekt gedruckt werden kann. Die Eigenschaft kann einen der folgenden Werte, oder eine Kombination derselbigen, annehmen:

· FirstPage;

· LastPage;

· OddPages;

· EvenPages;

· RepeatedBand. Dieser Wert bezieht sich auf eine Leiste mit der Eigenschaft "RepeatOnEveryPage" in true.

Standardmäßig ist der Wert dieser Eigenschaft gleich "FirstPage, LastPage, OddPages, EvenPages, RepeatedBand". Dies bedeutet, dass das Objekt auf alle Seiten des Berichts gedruckt wird. Hier einige typische Nutzungsbeispiele dieser Eigenschaft:

Wert der Eigenschaft	Druckposition des Objektes
FirstPage	Lediglich auf der ersten Seite.
LastPage, OddPages, EvenPages, RepeatedBand	Auf allen Seiten mit Ausnahme der ersten Seite.
FirstPage, OddPages, EvenPages, RepeatedBand	Auf allen Seiten mit Ausnahme der letzten Seite.
RepeatedBand	Nur auf Leisten deren Eigenschaft "RepeatOnEveryPage" auf wahr eingestellt ist.
FirstPage, LastPage, OddPages, EvenPages	Auf allen Leisten außer der wiederholten Leiste.
FirstPage, LastPage, OddPages, RepeatedBand	Nur auf ungeraden Seiten.
FirstPage, LastPage, EvenPages, RepeatedBand	Nur auf geraden Seiten.

Zum Druck von verschiedenen Texten, auf beispielsweise geraden und ungeraden Seiten, werden die beiden "Text"-Objekte auf die Leiste gelegt und folgendermaßen eingestellt:

· Das erste Objekt wird auf ungerade Seiten gedruckt. Die Eigenschaft "PrintOn" wird auf "FirstPage, LastPage, OddPages, RepeatedBand" (z.B. alle Werte außer "EvenPages") gesetzt.

· Das zweite Objekt wird auf gerade Seiten gedruckt. Die Eigenschaft "PrintOn" wird auf "FirstPage, LastPage, EvenPages, RepeatedBand" (z.B. alle Werte außer "OddPages") gesetzt.

Diese Objekte werden niemals gleichzeitig gedruckt. Sie können übereinander positioniert werden.

Alle Leisten haben die gleiche Eigenschaft. Zum Druck unterschiedlicher Leisten auf gerade und ungerade Seiten wird die Leiste "Kind" genutzt. Diese kann, im Fenster "Bereiche konfigurieren", an jede Leiste angehängt werden. Die Hauptleiste und das dazugehörige, untergeordnete Element wird folgendermaßen eingestellt:

· Die Hauptleiste wird auf ungeraden Seiten gedruckt. Die Eigenschaft "PrintOn" wird auf "FirstPage, LastPage, OddPages, RepeatedBand" (z.B. alle Werte außer "EvenPages") gesetzt.

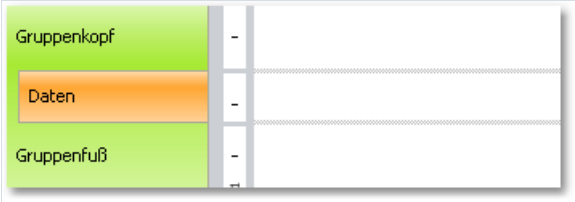
· Die Leiste des untergeordneten Elements wird auf gerade Seiten gedruckt. Die Eigenschaft "PrintOn" wird auf "FirstPage, LastPage, EvenPages, RepeatedBand" (z.B. alle Werte außer "OddPages") gesetzt.

Leisten können unterschiedliche Höhen, ein unterschiedliches Aussehen oder Inhalte aufweisen. Das folgende Beispiel illustriert den Druck unterschiedlicher Seitenköpfe:

Gruppen und Gesamt

Weiter oben wurde der Berichtstyp "Master-Detail" erläutert. Dieser druckt Daten aus zwei verbundenen Quellen. FastReport erlaubt das Erstellen eines derartig aussehenden Berichts, benutzt jedoch nur eine Datenquelle. Hierzu werden Gruppen genutzt.

Eine Gruppe besteht aus einer Serie aus drei Leisten: "Gruppenkopf", "Daten" und "Gruppenfuß". Dies wird im Designer, wie folgt dargestellt:



Eine Gruppe beinhaltet stets eine Kopfzeile und Daten. Der Gruppenfuß ist optional und kann gelöscht werden.

Um eine Gruppe nutzen zu können, sollte die Gruppenbedingung für den Gruppenkopf eingestellt und die Datenquelle mit der "Daten"-Leiste verbunden werden. Bei der Bedingung kann es sich um einen Ausdruck handeln, aber in der Regel handelt es sich um eine der Datenspalten. Die Gruppe wird folgendermaßen gedruckt:

1. Druck der Gruppenkopfs
2. Druck der Datenzeile
3. Überprüfung einer Änderung der Gruppenbedingung
4. Hat sich die Bedingung nicht verändert, wird die nächste Datenreihe gedruckt (S. 2)
5. Hat sich die Bedingung verändert, wird der Gruppenfuß gedruckt und beginnt den Druck einer neuen Gruppe (S.1).

Angenommen es existiert eine Produkttabelle mit den folgenden Daten:

CategoryName	ProductName
Beverages	Côte de Blaye
Beverages	Chartreuse verte
Beverages	Steeleye Stout
Beverages	Guaraná Fantástica
Beverages	Sasquatch Ale
Beverages	Rhönbräu Klosterbier
Beverages	Lakkalikööri
Beverages	Outback Lager
Beverages	Ipoh Coffee
Beverages	Laughing Lumberjack Lager
Beverages	Chang
Beverages	Chai
Condiments	Original Frankfurter grüne Soße
Condiments	Sirop d'érable
Condiments	Chef Anton's Gumbo Mix
Condiments	Northwoods Cranberry Sauce
Condiments	Grandma's Boysenberry Spread
Condiments	Chef Anton's Cajun Seasoning
Condiments	Aniseed Syrup
Condiments	Louisiana Hot Spiced Okra
Condiments	Vegie-spread
Condiments	Louisiana Fiery Hot Pepper Sauce
Condiments	Gula Malacca
Condiments	Genen Shouyu

Die Daten können auf der Spalte CategoryName gedruckt werden. Diese Spalte wird im Gruppenkopf gedruckt: Die Daten selbst werden im Feld ProductName präsentiert. Der Bericht wird folgendermaßen aussehen:

Nach einem Durchlauf des Berichtes sieht dieser folgendermaßen aus:

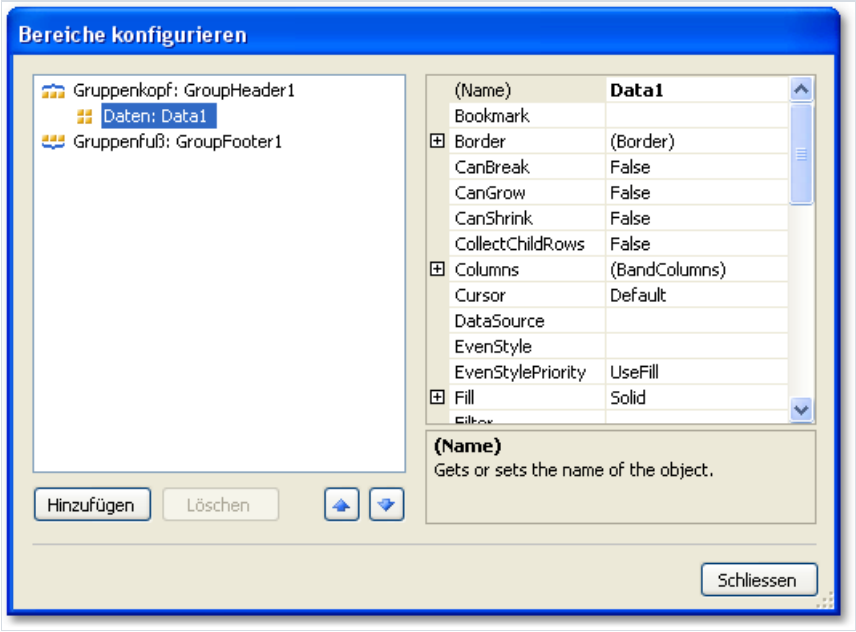
Beverages
Côte de Blaye
Chartreuse verte
Steeleye Stout
Guaraná Fantástica
Sasquatch Ale
Rhönbräu Klosterbier
Lakkalikööri
Outback Lager
Ipoh Coffee
Laughing Lumberjack Lager
Chang
Chai

Condiments
Original Frankfurter grüne Soße
Sirop d'érable
Chef Anton's Gumbo Mix
Northwoods Cranberry Sauce
Grandma's Boysenberry Spread
Chef Anton's Cajun Seasoning
Aniseed Syrup
Louisiana Hot Spiced Okra
Vegie-spread
Louisiana Fiery Hot Pepper Sauce
Gula Malacca
Genen Shouyu

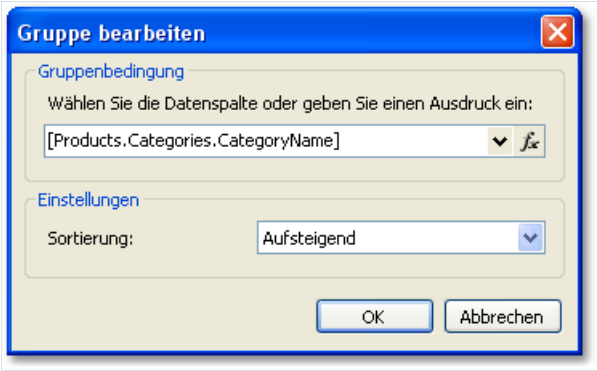
Erstellen von Gruppen

Eine Gruppe kann mittels zwei Methoden zu einen Bericht hinzugefügt werden:

Die erste Methode: Hinzufügen der Leiste "Gruppenkopf" in das Fenster "Bereiche konfigurieren". Hierzu wird die Schaltfläche "Hinzufügen" gedrückt und die Leiste "Gruppenkopf" ausgewählt. FastReport fügt die Gruppe zur verfügbaren "Daten"-Leiste hinzu oder, wenn diese nicht im Bericht vorhanden ist, erstellt eine ganze Gruppe:

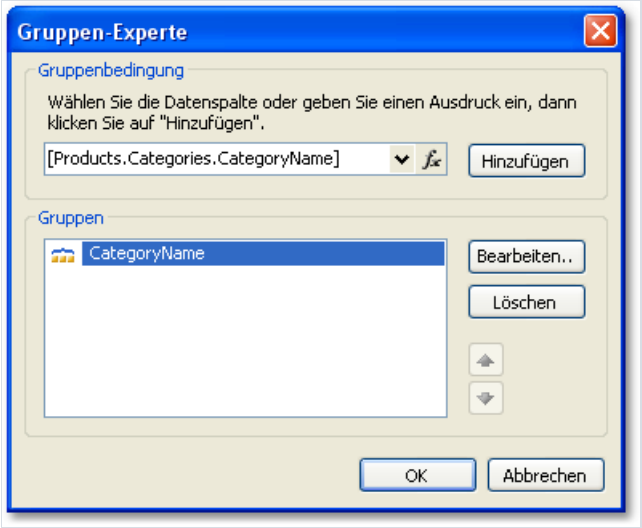


Zur Konfiguration einer Gruppe wird die "Gruppenkopf"-Leiste doppelgeklickt. Der Editor für Gruppenköpfe wird angezeigt:

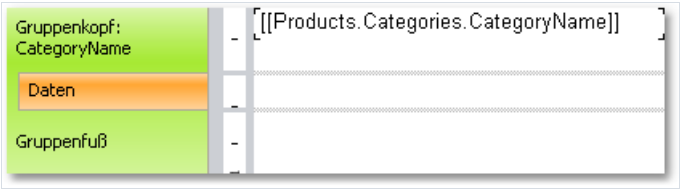


Die Gruppenbedingung muss festgelegt werden. Dabei kann es sich um einen beliebigen Ausdruck oder Datenquellenspalte handeln. Auch die Sortierreihenfolge muss ausgewählt werden. Standardmäßig werden die Daten in absteigender Reihenfolge geordnet.

Die zweite Methode: Nutzung des Assistenten, der über das Menü "Bericht|Gruppen Experte..." aufgerufen werden kann. Zur Erstellung einer Gruppe, wird die Gruppenbedingung eingegeben und die Schaltfläche "Hinzufügen" gedrückt:



Der Assistent fügt alle Element der Gruppe in den Bericht ein. Es erstellt auch ein "Text"-Objekt in der Kopfzeile, in der die Gruppenbedingung gedruckt wird:



Sortierung der Daten

Zum ordnungsgemäßen Funktionieren der Gruppe ist es notwendig die folgenden Bedingungen zu erfüllen: Datenquelle muss in der Spalte sortiert sein, die in der Gruppenbedingung genutzt wird. Ist diese Bedingung nicht erfüllt, sieht man eine Menge an Gruppen, die 1-2 Datenzeilen umfassen:

Glücklicherweise besteht die Möglichkeit die Datenquelle auf zwei Arten zu sortieren.

· Die Datensortierung kann im Gruppeneditor festgelegt werden. Die Datenquelle wird automatisch entsprechend der Gruppenbedingung sortiert;

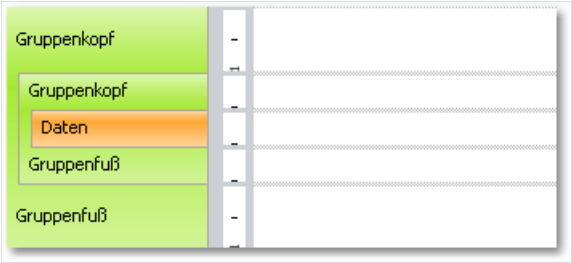
· Die Datensortierung kann im Editor "Daten"-Leiste festgelegt werden.

Auch wenn beide Methoden gleichwertig sind, ist es bequemer die erstere zu nutzen. Bei der Erstellung von Gruppen wird die Datengruppierung und -sortierung in einem Dialog festgelegt.

In bestimmten Situationen sollte jedoch die erste Methode nicht genutzt werden. Angenommen, die Gruppierung wird für die ersten Buchstaben des Produktnamens festgelegt. In einem solchen Fall, wird das Produkt nur entsprechend des ersten Buchstabens sortiert; dies ist nicht akzeptabel. Sie sollten die zweite Methode nutzen und die Sortierung des vollständigen Namens des Produktes anzeigen.

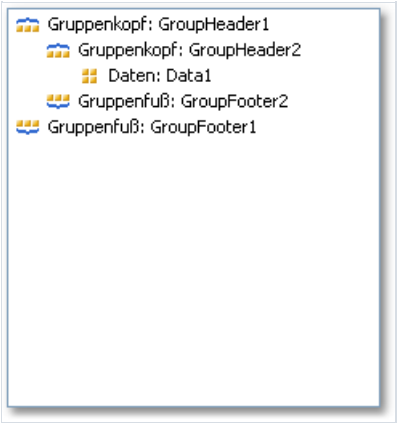
Geschachtelte Gruppen

Eine geschachtelte Gruppe weist mehrere "Gruppenkopf"-Leisten auf. Die letzte Leiste umfasst die "Daten"-Leiste:



Jeder Gruppenkopf kommt mit einer individuellen Gruppenbedingung.

Eine geschachtelte Gruppe kann auf dieselbe Art, wie eine reguläre Gruppe, erstellt werden. Im ersten Fall wird eine einfache Gruppe erstellt und in die geschachtelte Gruppe im Fenster "Bereiche konfigurieren" hinzugefügt. Hierzu wird die existierende Leiste "Gruppenkopf" ausgewählt, die Schaltfläche "Hinzufügen" gedrückt und eine weitere Leiste "Gruppenkopf" hinzugefügt:



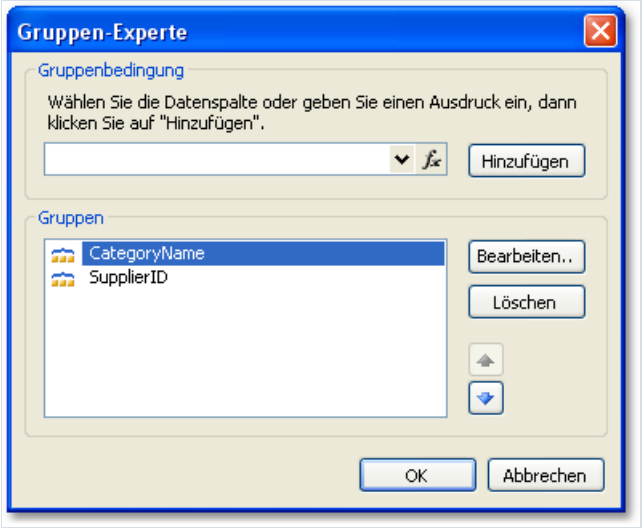
Hiernach wird der Editor der hinzugefügten Gruppe aufgerufen und die Gruppenbedingung festgelegt.

Im zweiten Fall, nutzt man den Gruppen Experten, der bereits betrachtet wurde. Die Gruppenbedingung wird festgelegt und die Schaltfläche "Hinzufügen" geklickt. Der Assistent fügt die neue, der bereits bestehenden Gruppe, hinzu.

Der Druck der geschachtelten Gruppe unterscheidet sich nicht sonderlich vom Druck einer regulären Gruppe. Beim Druck von Daten wird FastReport alle Gruppenbedingungen aller Gruppen überprüfen. Wenn sich die Bedingungen ändern, wird der Druck der entsprechenden Gruppe beendet und der Druck einer neuen Gruppe beginnt.

Gruppenverwaltung

Zur Verwaltung von Gruppen kann der Gruppen Experte genutzt werden. Dieser kann über das Menü "Bericht|Gruppen Experte" aufgerufen werden:



Der Assistent unterstützt das Löschen oder Hinzufügen einer Gruppe und ändert auch die Gruppierreihenfolge. Zur Änderung der Gruppierreihenfolge werden die Schaltflächen und genutzt. Mittels der Schaltfläche "Bearbeiten" kann die Gruppenbedingung der ausgewählten Gruppe geändert werden.

Druck der Gesamtwerte

Eine Gruppierung wird oftmals genutzt um den Gesamtwert einer Gruppe zu drucken. Dies kann beispielsweise die Anzahl der Zeilen in einer Gruppe betreffen, oder die Summe einer der Datenspalten. Zum Druck solcher Werte wird der Gesamtwert genutzt. Die Nutzung des Gesamts wird im Kapitel "Daten" erläutert.

Zum Druck des Gesamtwertes in der Gruppe, muss das Folgende durchgeführt werden:

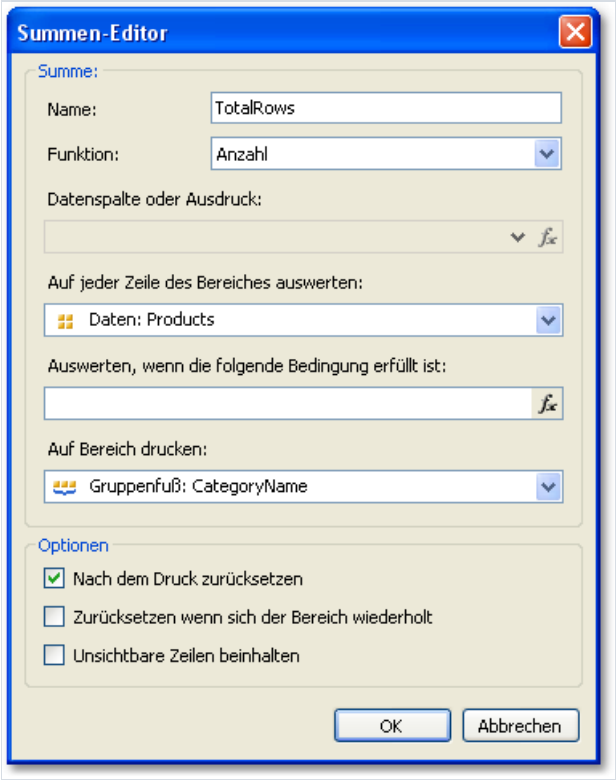
· Erstellen eines Gesamts durch Auswahl des Elements "Aktionen|Neues Gesamt..." im Fenster "Daten";

· Auswahl der Datenleiste der Gruppe in der Kombobox "Jede Zeile der Leiste evaluieren";

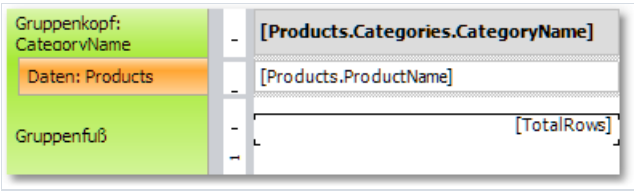
· Auswahl der Gruppenfußleiste in der Kombobox "Auf der Leiste drucken";

· Positionierung des "Text"-Objektes, das den Gesamtwert druckt, auf der Gruppenfußleiste.

Zum Druck der Anzahl der Zeilen in jeder Gruppe, wird das Gesamt beispielsweise folgendermaßen konfiguriert:



Um den Gesamtwert anzuzeigen wird dieses auf die Gruppenfuß gezogen:



Der erstellte Bericht schaut folgendermaßen aus:

Meat/Poultry
Alice Mutton
Perth Pasties
Thüringer Rostbratwurst
Pâté chinois
Tourtière
Mishi Kobe Niku

6

Produce
Rössle Sauerkraut
Uncle Bob's Organic Dried Pears
Manjimup Dried Apples
Longlife Tofu
Tofu

5

Wiederholen der Kopf-, und Fußzeile

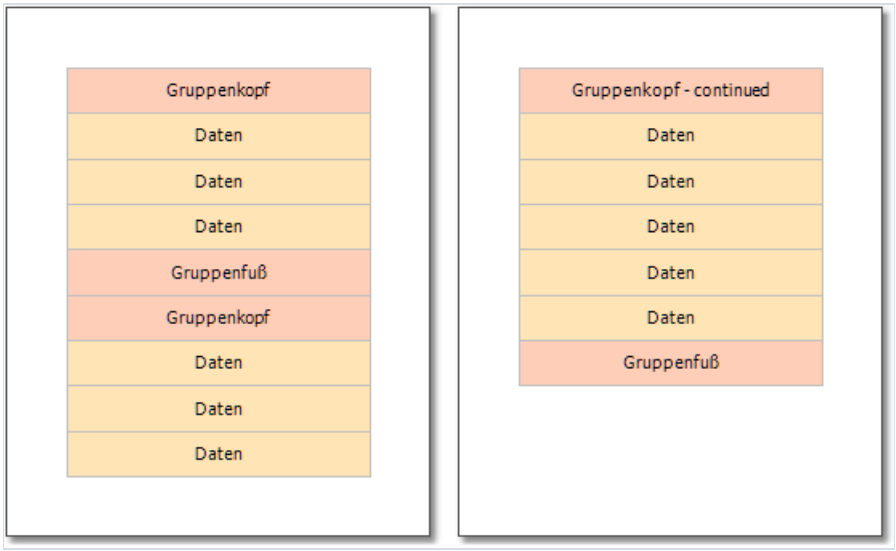
Gruppenkopf und -fuß weisen die gleiche "RepeatOnEveryPage" Eigenschaft auf. Dies kann nützlich sein, wenn die Gruppe nicht auf eine Berichtsseite passt. Die Nutzung dieser Eigenschaft ermöglicht den Druck der Kopf-, und Fußzeile auf jeder Seite auf der die Gruppe gedruckt ist. Beim Druck solcher Kopf-, und Fußzeilen, markiert FastReport diese als "Wiederholt". Dies kann zum Druck verschiedener Objekte auf einen regulären Gruppenkopf genutzt werden und zum Druck, beispielsweise, des Textes "Fortsetzen" auf einer neuen Seite. Hierzu wird die Eigenschaft "PrintOn" des "Text"-Objektes genutzt (weitere Details werden im Abschnitt "Berichtstyp "Broschüre" erläutert).

Zum Druck unterschiedlicher Texte, werden zwei Objekte auf den Gruppenkopf positioniert; einer über dem anderen:

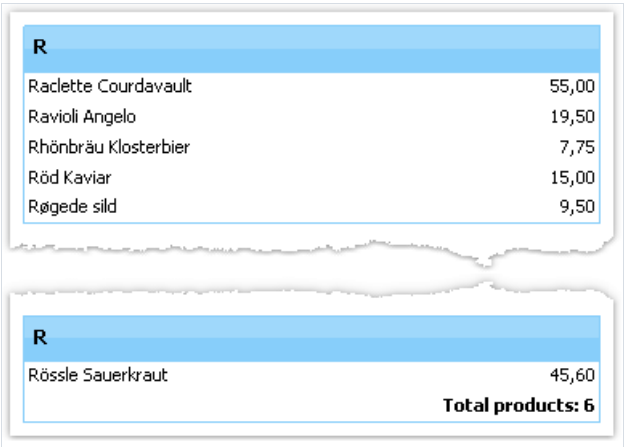
· Das erste Objekt wird auf den regulären Kopf gedruckt. Die Eigenschaft "PrintOn" wird auf "FirstPage, LastPage, OddPages, EvenPages" gesetzt (z.B. alle Werte außer "RepeatedBand") eingestellt.

· Das zweite Objekt wird lediglich auf den wiederholten Köpfen wiederholt. Die Einstellung die Eigenschaft "PrintOn" bis "RepeatedBand". Das "Text"-Objekt wird über den Text "Fortsetzen" auf der Kopfzeile hinzugefügt.

Der Bericht wird folgendermaßen gedruckt:



Auch der Gruppenkopf kann auf jeder Seite wiederholt werden:



In diesem Bericht hat der Gruppenkopf zwei Objekte, einer über dem anderen platziert:

Total products: [TotalProducts]
PrintOn = FirstPage, LastPage, OddPages, EvenPages
PrintOn = RepeatedBand

Gruppeneigenschaften

Die Leiste "Gruppenkopf" weist eine Reihe nützlicher Eigenschaften auf.

Die Eigenschaft "StartNewPage" ermöglicht eine neue Seite zu drucken bevor die Gruppe gedruckt wird. So wird jede Gruppe vor einer neuen Seite positioniert.

> Diese neue Seite wird nicht vor der ersten Gruppe hinzugefügt. Dies verhindert eine erste leere Seite.

Die Eigenschaft "ResetPageNumber" erlaubt das Zurücksetzen der Seitenzahl beim Druck einer neuen Gruppe. Üblicherweise wird diese in Kombination mit der Eigenschaft "StartNewPage" genutzt. Sind beide Eigenschaften aktiviert wird jede Gruppe, mit deren eigenen Seitennummerierung, auf einer neue Seite gedruckt.

Teilberichte

Beizeiten müssen an einer bestimmten Position des Hauptberichtes, zusätzliche Daten angezeigt werden; dies kann ein separater Bericht mit einer sehr komplizierten Struktur sein. Man kann diese Schwierigkeit lösen, indem man die reichhaltige Auswahl an Leisten in FastReport nutzt. Trotzdem ist es manchmal vorzuziehen, das Objekt "Teilbericht" zu nutzen.

Das Objekt "Teilbericht" ist ein reguläres Berichtsobjekt, das auf einem der Leisten positioniert werden kann. Wird dies ausgewählt, fügt FastReport eine Sonderseite in den Bericht ein und verbindet diese mit dem Teilbericht. Diese Seite ermöglicht das Erstellen eines zusätzlichen Berichts mit einer beliebigen Struktur.

Beim Druck des Berichts mit einem Objekt "Teilbericht" wird das folgende durchgeführt:

1. Der Hauptbericht wird gedruckt, wenn das "Teilbericht"-Objekt nicht erfüllt ist;
2. Die Teilberichtleiste wird gedruckt;
3. Der Druck des Hauptberichtes wird fortgesetzt.

> Da der Teilbericht auf den Seiten des Hauptberichtes erstellt ist, kann es die folgenden Leisten nicht umfassen: "Berichtkopf/-fuß", "Seitenkopf/-fuß", "Spaltenkopf/-fuß", "Overlay".

Druckmodi

Der Teilbericht kann in zwei Modi gedruckt werden.

Im ersten Druckmodus werden die Objekte und Leisten des Teilberichts auf einer Seite des Hauptberichtes gedruckt. Objekte die auf der Leiste positioniert werden können unterliegen Beschränkungen:

- Das Objekt "Teilbericht"muss am unteren Rahmen der Leiste positioniert werden;
- Andere Objekte können niemals unter das Objekt "Teilbericht" positioniert werden. Diese Objekte würden sonst mit den Objekten des Teilberichts überlappen:

Zur Platzierung anderer Objekte unter den Teilbericht muss die Leiste "Untergeordnete Elemente" genutzt werden. Die Objekte müssen folgendermaßen positioniert werden:

Der zweite Druckmodus unterscheidet sich darin, dass das Teilberichtsobjekt auf der Leiste, die das Objekt "Teilbericht" beinhaltetn, gedruckt wird. Dieser Modus kann über das Kontextmenü des "Teilbericht"-Objektes aktiviert werden. Um dies zu tun muss das Element "Auf Eltern drucken" ausgewählt werden. Diese Modus beschränkt die Positionierung des Objektes nicht. In diesem Modus kann, darüber hinaus. die Leiste des übergeordneten Elements sich, je nach Datenumfang des Teilberichtes, entweder vergrößern oder verkleinern.

Im zweiten Modus treten jedoch die Schwierigkeiten auf, wenn der Teilbericht viele Daten beinhaltet. Beim Druck wird die Leiste des übergeordneten Elements sehr hoch. Um ein derartige Leiste korrekt zu drucken, muss deren Inhalt umgebrochen werden (Eigenschaft "CanBreak"). Der Unbruchalgorithmus liefert keine 100% Qualität und führt in einigen Fällen zur Deplatzierung von Objekten.

Teilberichte nebeneinander

Durch die Positionierung zweier "Teilbericht"-Objekte nebeneinander auf derselben Leiste können zwei unabhängige Datenleisten gedruckt werden. Beim Druck eines solchen Berichts verhält sich FastReport wie folgt:

- Druck des Hauptberichtes bis das Objekt "Teilbericht" nicht erfüllt ist;

· Druck des Teilberichtes;

· Wechsel zu der Seite an der der Teilbericht begonnen hat und Druck des folgenden Teilberichts;

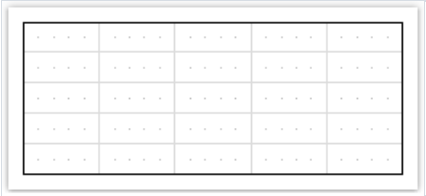
- nach dem Druck aller Teilberichte wird der Druck des Hauptberichtes fortgesetzt und zwar dort, wo der längste Teilbericht geendet hat.

Geschachtelte Teilberichte

Au der Seite Teilbericht kann ein weiteres "Teilbericht"-Objekt platziert werden; dies ändert den Bericht in einen geschachtelten Teilbericht. Formal ist die Anzahl der Schachtelungsniveaus nicht beschränkt; ihr Einsatz sollte jedoch nicht übertrieben werden. Eine starke Verschachtelung ist schwer verständlich. Die Nutzung von Datenleisten ist beim Druck geschachtelter Daten vorzuziehen. Eine "Daten"-Leiste kann eine oder mehrere geschachtelte "Daten"-Leisten beinhalten. Besteht die Notwendigkeit einen Bericht des Master-Detailtyps oder Master-Detail-Teilberichts zu drucken, muss der Teilbericht nicht verwendet werden.

Berichtstyp Tabelle

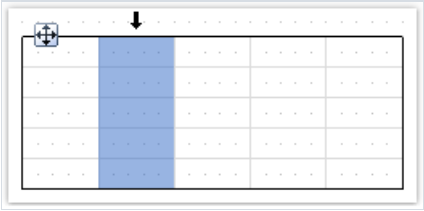
Das "Tabellen"-Objekt setzt sich aus Zeilen, Spalten und Zellen zusammen. Es ist eine vereinfachte Form der Microsoft Exceltabelle. Diese schaut wie folgt aus:



Konfiguration von Spalten

Man kann Spalten mit Hilfe des Kontextmenüs hinzufügen oder löschen. Hierzu:

- wählt man die Tabelle, oder ein beliebiges Element der Tabelle, und platziert den Zeiger auf der benötigten Spalte. Die Form des Zeigers ändert sich zu einem kleinen, schwarzen Pfeil.



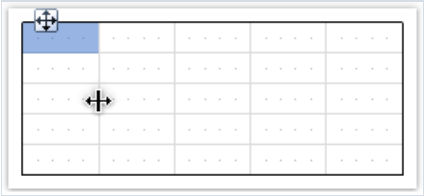
- Linker Mausklick zur Auswahl der Spalte;
- Rechter Mausklick zur Anzeige des Kontextmenüs der Spalte;
- Sollen mehrere benachbarte Spalten ausgewählt werden, hält man die linke Maustaste gedrückt und bewegt die Maus nach rechts um die benachbarten Spalten auszuwählen.

Das Kontextmenü der Spalte kann auch im Fenster "Bericht Baum" aufgerufen werden. Das Fenster wird geöffnet, die benötigte Spalte ausgewählt und mit der rechten Maustaste angeklickt.

Festlegen der Spaltengröße

Die Spaltenbreite kann durch eine der folgenden Methoden festgelegt werden:

- Wahl der Tabelle oder eines beliebigen Elementes der Tabelle und Platzierung des Zeigers zwischen zwei Spalten. Die Form des Zeiger ändert sich in einen horizontalen "Teiler":



- Auswahl einer Spalte und Anzeige der benötigten Weite in der Eigenschaft "Weite". Diese Eigenschaft kann über das "Eigenschafts"-Fenster angesteuert werden.

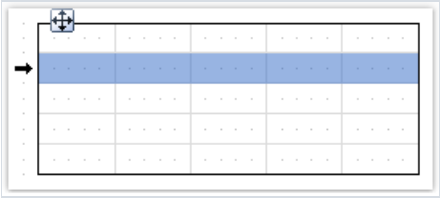
Auch die Spalteneigenschaft "AutoSize" kann aktiviert werden. Beim Durchlauf des Berichts wird die Spaltenbreite automatisch berechnet. Zur Begrenzung der Spaltenbreite lassen sich die Eigenschaften "MinWidth" und "MaxWidth" einstellen.

> Die Spaltenbreite sollte die Seitenbreite niemals überschreiten.

Konfiguration der Zeilen

Zeilen werden auf die gleiche Art konfiguriert. Zur Auswahl einer Zeile sind die folgenden Schritte zu beachten:

- Auswahl der Tabelle oder irgendeines ihrer Elemente und Platzierung des Zeigers zur linken der benötigten Zeile. Die Form des Zeigers ändert sich in einen kleinen, schwarzen Pfeil:



- Linker Mausklick zur Auswahl der Zeile;

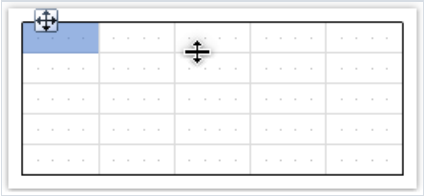
· Rechter Mausklick zur Anzeige des Kontextmenüs der Zeile;

Sollen mehrere benachbarte Zeilen ausgewählt werden, hält man die linke Maustaste gedrückt und bewegt die Maus nach rechts um die benachbarten Spalten auszuwählen.

Festlegen der Zeilengröße

Die Zeilenhöhe kann durch eine der folgenden Methoden festgelegt werden:

- Auswahl der Tabelle oder eines beliebigen Elementes der Tabelle und Platzierung des Zeigers zwischen zwei Zeilen. Der Zeiger nimmt die Form eines vertikalen Teilers an:



Zur Änderung der Zeilengröße wird die linke Maustaste geklickt.

· Auswahl einer Zeile und Anzeige der benötigten Höhe in der Eigenschaft "Height". Diese Eigenschaft kann über das "Eigenschafts"-Fenster angesteuert werden.

Auch die Zeileneigenschaft "AutoSize" kann aktiviert werden. Beim Durchlauf des Berichts wird die Zeilenhöhe automatisch berechnet. Zur Begrenzung der Zeilenhöhe können die Eigenschaften minimale Höhe ("MinHeight") und maximale Höhe ("Maxheight") genutzt werden:

> Die Zeilenhöhe sollte die Seitenhöhe niemals überschreiten.

Konfiguration von Zellen

Zellen sind Textobjekte. Im Wesentlichen wird die Zellenklasse aus dem "Text"-Objekt vererbt. Alles bisher über "Text"-Objekte besprochene findet auch auf die Textzellen Anwendung.

Die Bearbeitung des Textes einer Zelle entspricht der Bearbeitung eines "Text"-Objektes. Ein Element aus dem Datenfenster kann mittels drag&drop in eine Zelle gezogen werden.

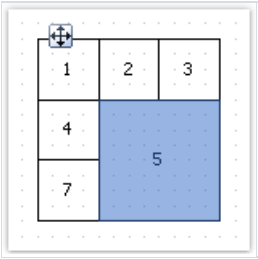
Der Rahmen und die Füllung einer Zelle wird mittels der Symbolleiste "Rahmen und Füllung" konfiguriert.

Zum Aufruf des Kontextmenüs einer Zelle muss diese mit der rechten Maustaste geklickt werden.

Verbinden und Teilen von Zellen

Benachbarte Zellen einer Tabelle können verbunden werden. Das Ergebnis: Eine große Zelle. Um dies zu tun:

- Auswahl der ersten Zelle mit der Maus;
- Linker Mausklick und, mit angehaltener Maustaste, wird zur Auswahl einer Gruppe von Zellen ,die Maus bewegt;
- Rechter Mausklick in dem ausgewählten Bereich zur Anzeige des Kontextmenüs der Zellen;
- Auswahl des Items "Zellen verbinden" aus dem Kontextmenü.



Auswahl des Items "Zellen teilen" zur Teilung einer Zelle.

Einfügen von Objekten in Zellen

Man kann auch andere Objekte, beispielsweise Bilder, in Zellen einfügen. Einige Objekte können nicht in Zellen eingefügt werden. Diese sind:

- "Tabelle";
- "Matrix";
- "Teilbericht".

Um der Zelle ein Objekte hinzuzufügen, muss dieses lediglich in die Zelle gezogen werden. Objekte können frei zwischen Zellen hin-, und her bewegt und auch auf die Begrenzungen der Tabellen zurückgezogen werden.

Eine Zelle dient als Container für die darin platzierten Objekte. Dies bedeutet, dass die Eigenschaften "Dock" und "Anker" eines Objektes auch innerhalb einer Zelle genutzt werden können. Dies erlaubt eine Größenänderung des Objektes entsprechend der Änderung der Zellengröße.

Druck einer Tabelle

Eine Tabelle kann in zwei Modi gedruckt werden:

Im ersten Modus wird die Tabelle innerhalb der zugehörigen Leiste gedruckt und hat das gleiche Aussehen wie im Designer. In diesem Modus erstreckt sich die Tabelle nicht über mehrere Seiten, wenn ihre Breite breiter, als die der Berichtsseite ist. Dies ist der Standarddruckmodus.

Der zweite Modus ist dynamisch. In diesem Modus wird die Tabelle mittels eines Skriptes erzeugt. Während dieses Prozesses kann sich die Ergebnistabelle von der Anfangstabelle unterscheiden; ähnlich wie sich der erstelle Bericht von der Berichtsvorlage unterscheidet. Im dynamischen Modus kann sich die Tabelle über mehrere Seiten aufteilen, wenn sie nicht auf eine Berichtsseite passt.

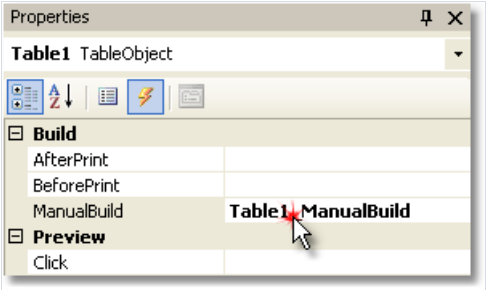
Im dynamischen Modus wird die Tabelle nicht auf die Leiste gedruckt auf der sie platziert wurde. Stattdessen generiert die Tabelle selbst eine Reihe an Leisten, die Teile der Ergebnistabelle beinhalten. Dieser Arbeitsmodus birgt die folgenden Restriktionen:

- Andere Objekte können nicht unterhalb der Tabelle oder nahe der Tabelle platziert werden. Stattdessen sollte hier die Kindleiste genutzt werden;
- Zwei "Tabellen"-Objekte können niemals auf eine Leiste platziert werden.

Detailbetrachtung des dynamischen Modus.

> Dieser Modus erfordert Programmierkenntnisse. Es bedarf daher erweiterter Fertigkeiten vom Berichtsentwickler.

Die Erstellung der Tabelle erfolgt mittels eines Skripts. Zur Erstellung eines Skripts wählt man das "Tabellen"-Objekt. Im "Eigenschaften"-Fenster klickt man die Schaltfläche "Event" und doppelklickt auf das Event ManualBuild:



Hierdurch wird ein leerer Ereignishandler in den Berichtcode eingefügt:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
}
```

In diesem Modus wird die Quelltabelle als Vorlage genutzt. Im Eventcode lassen sich, sooft wie notwendig, Zeilen und Spalten aus der Quelltabelle drucken. Währenddessen wird die verbleibende Tabelle, die eine unbeschränkte Anzahl an Zeilen und Spalten beinhalten kann, erstellt. Eine derartige Tabelle kann sich über mehrere Seiten erstrecken, wenn sie nicht auf eine Berichtsseite passt.

Zum Druck einer Tabelle wird die folgende Methode des "Tabellen"-Objektes genutzt:

Methode	Parameter	Beschreibung
PrintRow	int index	Druckt die Zeile mit dem angegebenen Index. Die Zeilennummerierung beginnt mit 0.
PrintColumn	int index	Druckt die Spalte mit dem angegebenen Index. Die Spaltennummerierung beginnt mit 0.
PrintRows	int[] indices	Druckt mehrere Zeilen der Tabelle.
PrintRows	-	Druckt alle Zeilen der Tabelle.
PrintColumns	int[] indices	Druckt mehrere Spalten der Tabelle.
PrintColumns	-	Druckt alle Spalten der Tabelle.
PageBreak	-	Fügt einen Seitenumbruch vor dem Druck einer neuen Spalte oder Zeile ein.

Die Tabelle kann mittels einer der folgenden Methoden gedruckt werden:

Die erste Methode: Druck von oben nach unten, dann von links nach rechts. Diese Methode eignet sich am besten für eine Tabelle mit einer unterschiedlichen Anzahl an Zeilen. Die Methoden sollten in der folgenden Reihenfolge aufgerufen werden:

- PrintRow(row index);
- eine oder mehrere Aufrufe der PrintColumn(column index) oder PrintColumns(columns indices) Methode zum Druck der anzeigten Spalten;
- oder ein Aufruf der Methode PrintColumns() zum Druck aller Spalten;
- Wiederholung dieser Abfolge zum Druck aller benötigten Tabellenzeilen.

> Jede Tabellenzeile muss die gleiche Anzahl an Spalten beinhalten. Dies sollte bei der der Nutzung der Methoden PrintColum(int index) und PrintColumns(int [] indices) im Auge behalten werden.

Die zweite Methode: Druck von links nach rechts, dann von oben nach unten. Diese Methode ist besser für eine Tabelle mit einer variablen Anzahl an Zeilen geeignet. Die Methoden sollten in der folgenden Reihenfolge aufgerufen werden:

- PrintColumn(column index);
- eine oder mehrere Aufrufe der PrintRow(row number) oder PrintRows(rows indices) Methode zum Druck der anzeigten Zeilen;
- oder ein Aufruf der Methode PrintRows() zum Druck aller Zeilen;
- Wiederholung dieser Abfolge zum Druck aller benötigten Tabellenspalten.

Jede Tabellenzeile muss die gleiche Anzahl an Spalten beinhalten. Dies sollte bei der der Nutzung der Methoden PrintRow(int index) und PrintRows(int [] indices) im Auge behalten werden.

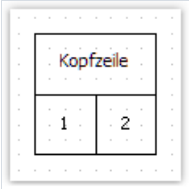
Ein fehlerhaftes Aufrufen der Reihenfolge der Druckmethoden führt zu Fehlern bei der Durchführung des Berichts. Ein möglicher Fehler: Versuch die Tabelle mittels des folgenden Codes zu drucken

```
Table1.PrintRows();
Table1.PrintColumns();
```

Die Abfolge der Methoden ist inkorrekt. Der Druck muss entweder mit der PrintRow-, oder der PrintColumn-Methode starten.

Druck komplexer Kopfzeilen

Hier werden Kopfzeilen, die umspannende Zellen beinhalten, betrachtet. Beim Druck von Tabellenzeilen oder -Spalten, die umspannte Zellen beinhalten, vergrößert sich die Größe automatisch. Dies wird im nächsten Beispiel erläutert:

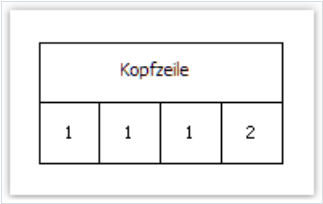


Hier wird manuell ein Ereignishandler erstellt, der die ersten drei Spalten 3-Mal und die zweiter Spalte 1-Mal druckt:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // printing row 1 and columns 0, 0, 0, 1
  Table1.PrintRow(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(1);
  // printing row 1 and columns 0, 0, 0, 1
  Table1.PrintRow(1);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(1);
}
```

> Hier sollte beachtet werden, dass die gleiche Anzahl an Spalten in jeder Reihe gedruckt wurde. Wird diese Regel nicht beachtet, erhält man ein nicht vorhersagbares Ergebnis.

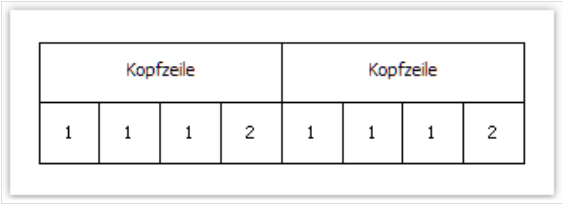
Als Ergebnis der Ausführung dieses Codes erhält man das Folgende:



Wie man hier sehen kann, wird die Kopfzeile automatisch erstreckt. Hier ein Beispiel mit einem komplexeren Code, bei dem zwei Gruppen von Spalten gedruckt werden:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // print 0 row and two groups of 0, 0, 0, 1 columns
  Table1.PrintRow(0);
  // group 1
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(1);
  // group 2
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(1);

  // print 1 row and two groups of 0, 0, 0, 1 columns
  Table1.PrintRow(1);
  // group 1
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(1);
  // group 2
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(0);
  Table1.PrintColumn(1);
}
```

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:



Beim Druck der zweiten Spalte mit dem folgenden Code:

```
Table1.PrintColumn(1);
```

Die Kopfzeile ist abgeschlossen und der weitere Druck der ersten Spalte beginnt eine neue Kopfzeile:

```
// group 2
Table1.PrintColumn(0);
```

Summen nutzen

Der dynamische Modus des "Tabellen"-Objektes unterstützt die folgenden Summenfunktionen:

Funktion	Parameter	Beschreibung
Sum	TableCell cell	Gibt die Summe der in der Zelle enthaltenen Werte zurück.
Min	TableCell cell	Gibt den Miminalwert der in der Zelle enthaltenen Werte zurück.
Max	TableCell cell	Gibt den Maximalwert der in der Zelle enthaltenen Werte zurück.
Avg	TableCell cell	Gibt denn Durchschnitt der in der Werte enthaltenen Zelle zurück.
Count	TableCell cell	Gibt die Anzahl der Zeilen in der bestimmten Zelle zurück.

> Diese Funktionen können nicht im regulären (nicht dynamischen) Druckmodus genutzt werden.

Um die Funktion Summe nutzen zu können, muss diese in die Tabellenzelle platziert werden. Beispielweise berechnet die folgende Funktion die Summe der Werte, die sich in einer Zelle mit dem Namen "Zelle1" befindet

```
[Sum(Cell1)]
```

> Während dieser Berechnung werden alle Zellen, die sich über und zur linken der aktuellen Zelle (in der die Summe berechnet wird) befindet, analysiert.

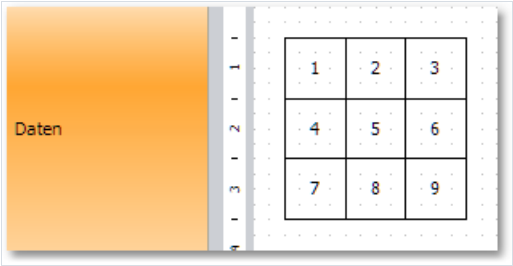
Layout der Tabelle

Eine dynamisch erstellte Tabelle kann sich automatisch über mehrere Seite erstrecken. Dieses Verhalten wird über die Eigenschaft "Layout" der Tabelle gesteuert. Es stehen die folgenden Werte zur Verfügung:

Wert	Beschreibung
AcrossThenDown	Die Tabelle wird quer und dann abwärts wiedergegeben.
DownThenAcross	Die Tabelle wird abwärts und dann quer wiedergegeben.
Wrapped	Eine breite Tabelle wird umgebrochen und auf derselben Seite wiedergegeben.

Beispiele

Das folgende Beispiel soll dies erläutern. Der folgende Bericht wird als Vorlage genutzt:

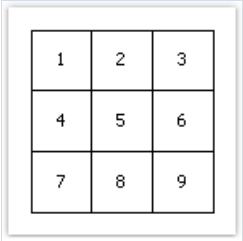


Zuerst wird eine Tabelle ausgewählt und ein Ereignishandler für das Ereignis "ManualBuild" erstellt.

Beispiel 1: Druck der Gesamttabelle von oben nach unten

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // printing row 0 and all of its columns
  Table1.PrintRow(0);
  Table1.PrintColumns();
  // printing row 1 and all of its columns
  Table1.PrintRow(1);
  Table1.PrintColumns();
  // Printing row 2 and all of its columns
  Table1.PrintRow(2);
  Table1.PrintColumns();
}
```

Das Ergebnis: Die folgende Tabelle, die sich nicht von der Vorlage unterscheidet, wird gedruckt:

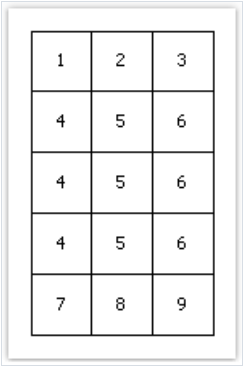


1	2	3
4	5	6
7	8	9

Beispiel 2: Druck der Tabelle von oben nach unten mit einer sich wiederholenden Zeile

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // Printing row 0 and all of its columns
  Table1.PrintRow(0);
  Table1.PrintColumns();
  // Printing 3 copies of row 1 and all of its columns
  for (int i = 0; i < 3; i++)
  {
    Table1.PrintRow(1);
    Table1.PrintColumns();
  }
  // printing row 2 and all of its columns
  Table1.PrintRow(2);
  Table1.PrintColumns();
}
```

In diesem Beispiel wird die mittlere Zeile 3-Mal gedruckt. Das Ergebnis sieht folgendermaßen aus:

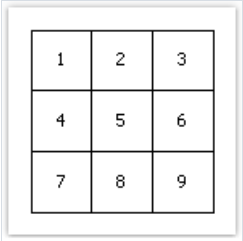


1	2	3
4	5	6
4	5	6
4	5	6
7	8	9

Beispiel 3: Druck der Gesamttabelle von links nach rechts

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // printing column 0 and all of its rows
  Table1.PrintColumn(0);
  Table1.PrintRows();
  // printing column 1 and all of its rows
  Table1.PrintColumn(1);
  Table1.PrintRows();
  // printing column 2 and all of its rows
  Table1.PrintColumn(2);
  Table1.PrintRows();
}
```

Das Ergebnis: Die folgende Tabelle, die sich nicht von der Vorlage unterscheidet, wird gedruckt:



1	2	3
4	5	6
7	8	9

Beispiel 4: Druck der Tabelle von links nach rechts mit einer sich wiederholenden Spalte

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // printing column 0 and all of its rows
  Table1.PrintColumn(0);
  Table1.PrintRows();
  // printing 3 copies of column 1 and all of its rows
  for (int i = 0; i < 3; i++)
  {
    Table1.PrintColumn(1);
    Table1.PrintRows();
  }
  // printing column 2 and all of its rows
  Table1.PrintColumn(2);
  Table1.PrintRows();
}
```

In diesem Beispiel wird die mittlere Spalte 3-Mal gedruckt. Das Ergebnis sieht folgendermaßen aus:

1	2	2	2	3
4	5	5	5	6
7	8	8	8	9

Beispiel 5: Druck einer Tabelle mit sich wiederholenden Zeilen und Spalten

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // --------- printing row 0
  Table1.PrintRow(0);
  // printing column 0
  Table1.PrintColumn(0);
  // printing 3 copies of column 1
  for (int i = 0; i < 3; i++)
    Table1.PrintColumn(1);
  // printing column 2
  Table1.PrintColumn(2);

  // --------- printing 3 copies of row 1
  for (int j = 0; j < 3; j++)
  {
    Table1.PrintRow(1);
    // printing column 0
    Table1.PrintColumn(0);
    // printing 3 copies of column 1
    for (int i = 0; i < 3; i++)
      Table1.PrintColumn(1);
    // printing column 2
    Table1.PrintColumn(2);
  }
  // --------- printing row 2
  Table1.PrintRow(2);
  // printing column 0
  Table1.PrintColumn(0);
  // printing 3 copies of column 1
  for (int i = 0; i < 3; i++)
    Table1.PrintColumn(1);
  // printing column 2
  Table1.PrintColumn(2);
}
```

> Hier sollte beachtet werden, dass die gleiche Anzahl an Spalten in jeder Zeile gedruckt wurde. Wird diese Regel nicht beachtet, erhält man ein nicht vorhersagbares Ergebnis.

In diesem Beispiel wird die mittlere Spalte und mittlere Zeile 3-Mal gedruckt. Das Ergebnis sieht folgendermaßen aus:

1	2	2	2	3
4	5	5	5	6
4	5	5	5	6
4	5	5	5	6
7	8	8	8	9

Beispiel 6: Nutzung der Datenquelle

All diese Beispiele betrachten eine Tabelle die regulären Text beinhaltet. Anhand dieses Beispiels wird erläutert wie eine Tabelle mittels einer Datenquelle erstellt wird. Hierzu wird eine Tabelle der folgenden Form erstellt:

Product Name	Unit Price	Units In Stock
[Products.ProductName]	[Products.UnitPrice]	[Products.UnitsInStock]

Zuerst wird der Ereignishandler für den ManualBuild erstellt. Dieser übernimmt die folgenden Aufgaben:

- Holen der Datenquelle, die im Bericht definiert wurde;
- Initialisierung (Auffüllen mit Daten);
- Druck der Tabellenzeilen so oft wie Zeilen in der Datenquelle vorhanden sind.

Hier der Code des Handlers:

```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // get the data source by its name
  DataSourceBase rowData = Report.GetDataSource("Products");
  // initialize it
  rowData.Init();

  // printing the table header
  Table1.PrintRow(0);
  Table1.PrintColumns();

  // loop through the data source rows
  while (rowData.HasMoreRows)
  {
    // printing the table row
    Table1.PrintRow(1);
    Table1.PrintColumns();

    // select the next data row
    rowData.Next();
  }

  // printing the table footer
  Table1.PrintRow(2);
  Table1.PrintColumns();
}
```

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

Product Name	Unit Price	Units In Stock
Chai	18,00	39
Chang	19,00	17
Aniseed Syrup	10,00	13
Chef Anton's Cajun Seasoning	22,00	53
Chef Anton's Gumbo Mix	21,35	0
Grandma's Boysenberry Spread	25,00	120
Uncle Bob's Organic Dried Pears	30,00	15
Northwoods Cranberry Sauce	40,00	6

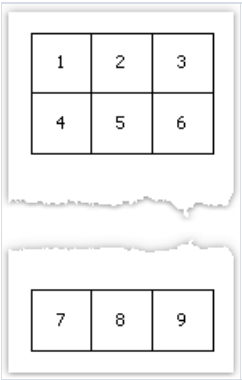
Beispiel 7: Einfügen von Seitenumbrüchen

Durch Nutzung der Methode "PageBreak" des "Tabellen"-Objektes lässt sich beim Druck einer Tabelle ein Seitenumbruch einfügen. Dieser wird vor dem Druck einer Zeile/Spalte aufgerufen.

Zur Erläuterung des "PageBreak"-Modus wird das Beispiel Nr. 1 herangezogen. Die dritte Zeile soll auf eine neue Seite gedruckt werden.

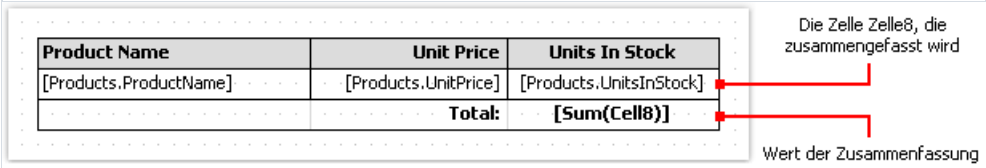
```
private void Table1_ManualBuild(object sender, EventArgs e)
{
  // print the row 0 with all its columns
  Table1.PrintRow(0);
  Table1.PrintColumns();
  // print the row 1 with all its columns
  Table1.PrintRow(1);
  Table1.PrintColumns();
  // insert page break before the row 2
  Table1.PageBreak();
  // print the row 2 with all its columns
  Table1.PrintRow(2);
  Table1.PrintColumns();
}
```

Das Ergebnis sieht folgendermaßen aus:

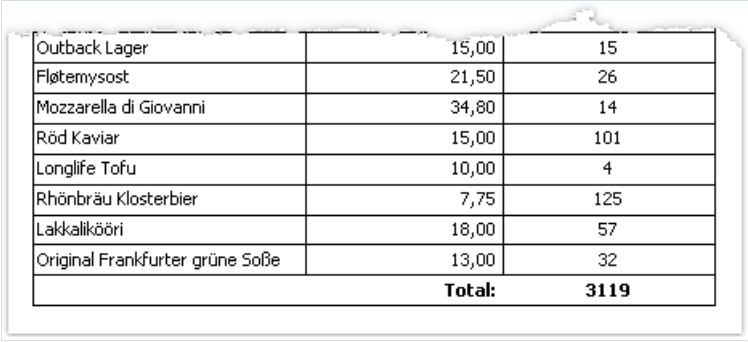


Beispiel 8: Druck von Summen

In Beispiel 6 wurde die Nutzung der Summenfunktion erläutert. Diese wird hier folgendermaßen verändert:



Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:



Outback Lager	15,00	15
Fløtemysost	21,50	26
Mozzarella di Giovanni	34,80	14
Röd Kaviar	15,00	101
Longlife Tofu	10,00	4
Rhönbräu Klosterbier	7,75	125
Lakkalikööri	18,00	57
Original Frankfurter grüne Soße	13,00	32
	Total:	**3119**

Berichte vom Typ Matrix

Das "Matrix"-Objekt ist ein variantenreiches Objekt und setzt sich, wie das "Tabellen"-Objekt, aus Zeilen, Spalten und Zellen zusammen. Die Anzahl der Zeilen und Spalten der Matrix sind hier nicht im Voraus bekannt, da diese von den mit ihnen verbundenen Daten abhängen.

Das Objekt sieht folgendermaßen aus:

Employee	[Year]	**Total**
[Name]	[Revenue]	
Total		

Beim Druck, füllt die Matrix die Werte und wächst aufwärts und abwärts. Der Bericht wird folgendermaßen aussehen:

Employee	1999	2000	2001	2002	**Total**
Andrew Fuller	$3,900.00	$2,100.00		$1,800.00	$7,800.00
Janet Leverling	$6,100.00	$3,200.00			$9,300.00
Nancy Davolio	$3,300.00	$2,700.00	$3,100.00	$1,700.00	$10,800.00
Steven Buchanan		$3,999.00	$8,100.00		$12,099.00
Total	$13,300.00	$11,999.00	$11,200.00	$3,500.00	**$39,999.00**

Ein wenig Theorie

Detailbetrachtung der Matrixelemente.

	1	2	3	4
a	a1	a2	a3	a4
b	b1	b2	b3	b4

Die Abbildung zeigt eine Matrix mit 2 Zeilen und 4 Spalten. Hier a,b - Zeilenkopf 1, 2, 3, 4 - Spaltenkopf, a1...a4, b1...b4 - Zellen. Zur Erstellung eines solchen Berichts wird lediglich eine Datenquelle benötigt, die 3 Spalten und die folgenden Daten umfasst:

a 1 a1

a 2 a2

a 3 a3

a 4 a4

b 1 b1

b 2 b2

b 3 b3

b 4 b4

Wie gezeigt, repräsentiert die erste Spalte die Matrixzeile, die zweite Spalte die Matrixspalte und die dritte die Inhalte der Zellen an der Kreuzung zwischen den Zeilen und Spalten mit den anzeigten Zahlen. Zur Erstellung eines Berichts erstellt FastReport eine Matrix im Speicher und füllt diese mit Daten. Wenn die Zeile oder Spalte mit der angegebenen Zahl noch nicht existiert, wächst die Matrix dynamisch.

Eine Kopfzeile kann mehr als eine Ebene aufweisen. Das folgende Beispiel soll dies erläutern:

	10		20	
	1	2	1	2
a	a10.1	a10.2	a20.1	a20.2
b	b10.1	b10.2	b20.1	b20.2

In diesem Beispiel, ist die Spalte ein Verbund, d.h. sie hat zwei Werte. Der Bericht erfordert die folgenden Daten:

a 10 1 a10.1

a 10 2 a10.2

a 20 1 a20.1

a 20 2 a20.2

b 10 1 b10.1

b 10 2 b10.2

b 20 1 b20.1

b 20 2 b20.2

Hier repräsentiert die ersten Spalte die Zeile und die zweite und die dritte Spalte repräsentieren die Matrixspalten. Die letzte Datenspalte umfasst die Werte der Zellen.

Das neue Matrixelement - Zwischen-, und Gesamtsumme - wird in der nächsten Abbildung dargestellt:

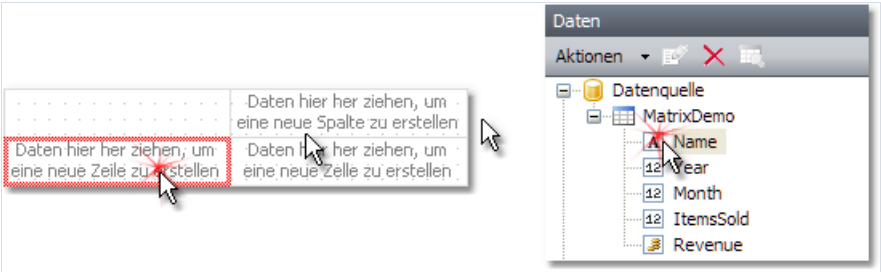
	10			20			Total
	1	2	Total	1	2	Total	
a	a10.1	a10.2	a10.1+a10.2	a20.1	a20.2	a20.1+a20.2	sum(a)
b	b10.1	b10.2	b10.1+b10.2	b20.1	b20.2	b20.1+b20.2	sum(b)
Total	a10.1+ b10.1	a10.2+ b10.2	a10.1+b10.1+ a10.2+b10.2	a20.1+ b20.1	a20.2+ b20.2	a20.1+b20.1+ a20.2+b20.2	sum(a)+ sum(b)

Dieser Bericht stützt sich auf die gleichen Daten wie im vorangegangenen Beispiel. Die in grau dargestellten Zeilen werden automatisch berechnet.

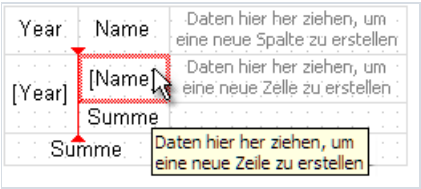
Konfiguration der Matrix

Nach der Positionierung eines neuen "Matrix"-Objektes auf der Tabelle, schaut diese folgendermaßen aus:

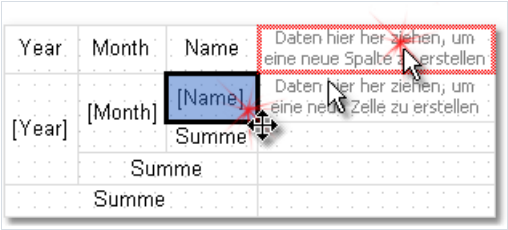
Die Matrix kann mittels der Maus konfiguriert werden. Zur Erstellung von Zeilen, Spalten und Zellen, werden die Spalten der Datenquellen aus dem Datenfenster mittels drag&drop in die Matrix gezogen. Die Matrix zeigt an dem Ort einen roten Rahmen an, an dem die neuen Daten angezeigt werden:



Enthält die Matrix bereits einige Elemente, wird bei der Platzierung eines neuen Elementes, ein Indikator angezeigt. Im vorliegenden Fall werden die neuen Daten zwischen das Element "Year" und "Name" platziert:



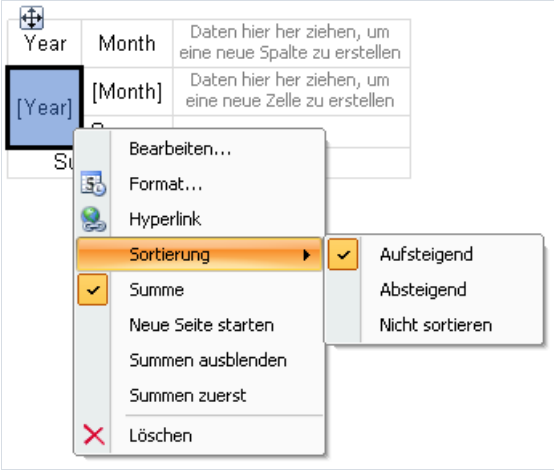
Auch die Reihenfolge der existierenden Elemente kann geändert werden. Um dies zu tun wird auf den Rahmen des Elements (In Schwarz angezeigt) geklickt und an den erforderlichen Platz gezogen:



Zum Löschen eines Elementes, wird es mit der Maus ausgewählt und die Taste "Löschen" gedrückt.

Konfiguration der Kopfzeilen

Zur Konfiguration des Kopfzeilenelementes, wird dieses mit der rechten Maustaste angeklickt und sein Kontextmenü angezeigt:



Standardmäßig werden die Daten in der Matrixkopfzeile in absteigender Reihenfolge sortiert. Die Sortierrichtung kann durch die Auswahl des Elementes "Sortierung" geändert werden.

Üblicherweise hat jedes Element in der Matrixkopfzeile eine Summe (die Zelle mit dem Text "Summe"). Die Summe kann durch drücken der Taste "Löschen" entfernt werden. Zur Aktivierung der Summe wird das zugehörige Element ausgewählt und das Element "Summe" in diesem Kontextmenü ausgewählt.

Das Menüelement "Neue Seite starten" weist einen neuen Seitenumbruch, nach dem Druck eines jeden Kopfzeilenwertes, an. Sind beispielsweise Seitenumbrüche für das Element "Year" (wie in der obigen Abbildung dargestellt) aktiviert, wird jeder Jahreswert auf einer eigenen Seite gedruckt.

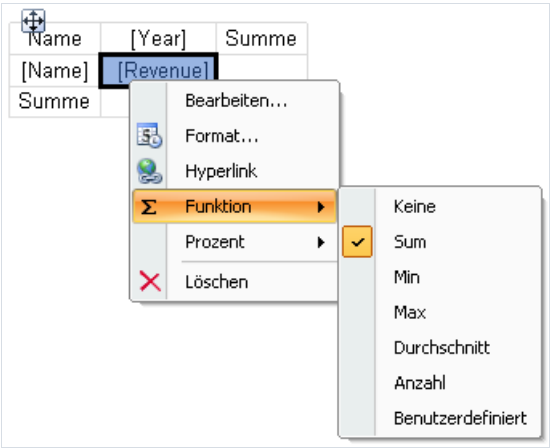
Das Element "Summen ausblenden" erlaubt das Unterdrücken von Summen. Dies gilt für den Fall, dass die Gruppe (für die der Gesamtwert berechnet wird) nur einen Wert enthält.

Konfiguration von Zellen

Für eine Matrixzelle kann, zur Berechnung der Summe, eine Funktion ausgewählt werden. Eine Liste der möglichen Funktionen ist im folgenden dargestellt:

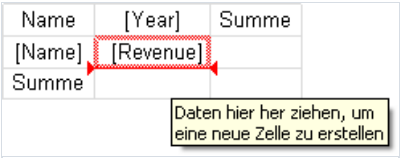
Funktion	Beschreibung
Keine	Der Zellenwert wird nicht verarbeitet.
Sum	Gibt die Summe, der in der Matrixzelle enthaltenen Werte, zurück.
Min	Gibt den Minimalwert zurück.
Max	Gibt den Maximalwert zurück.
Average	Gibt den Durchschnittswert zurück.
Count	Gibt eine Anzahl nicht-leerer Werte zurück.
Anzahl der verschiedenen	Gibt die Anzahl der verschiedenen (einmaligen) Werte zurück.

Die Summenfunktion wird als Standard genutzt. Dies kann geändert werden, indem zuerst eine Zelle und danach das Element "Funktion", aus seinem Kontextmenü, ausgewählt wird.



Soll die Summe nicht in der Zelle gedruckt werden, wird die Funktion "Keine" ausgewählt.

Eine Matrix kann eine oder mehrere Datenzellen umfassen. Hat die Matrix mehrere Zellen, können diese sowohl nebeneinander als auch übereinander angeordnet werden. Die Eigenschaft "CellsSideBySide" der Matrix steuert die Anordnung der Zellen. Dieser Modus kann über das Kontextmenü des "Matrix"-Objektes aktiviert werden. Die Anordnung kann auch geändert werden wenn eine zweite Zelle in die Matrix eingefügt wird. Hierbei muss die rote Anzeige, die die Platzierung des zweiten Werter indiziert, beachtet werden.

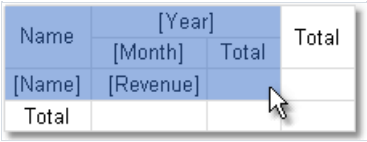


Nachdem der zweite Wert eingefügt wurde, werden die übrigen Werte entsprechend der ausgewählten

Reihenfolge hinzugefügt.

Gestaltung der Matrix

Zur Änderung des Aussehens einer Matrixzelle muss diese angeklickt werden. Über die Symbolleiste werden die Schriftart, der Rahmen und die Füllung festgelegt. Zur gleichzeitigen Änderung des Aussehens mehrerer Zellen, wird eine Zellengruppe ausgewählt. Hierzu wird die linke, obere Zelle und danach, mit angehaltener Maustaste, die weiteren Zellen dieser Gruppe ausgewählt.



Stile helfen das Aussehen der Matrix als Ganzes zu ändern. Hierzu wird das Kontextmenü des "Matrix"-Objektes aufgerufen und der Stil ausgewählt

Steuerung der Zeilen-, und Spaltengröße

Da das "Matrix"-Objekt dem "Text"-Objekt ähnelt, können die Zeilen-, und Spaltengrößen auf die gleiche Art festgelegt werden.

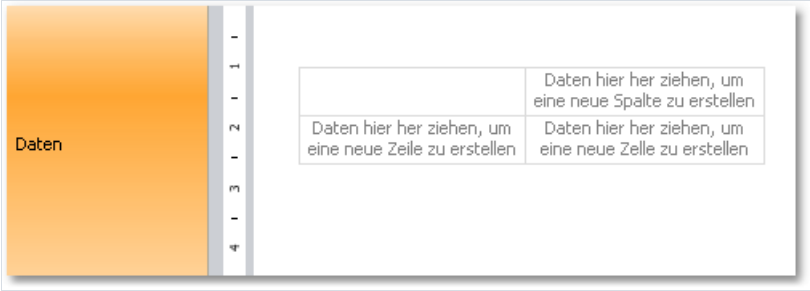
Der "AutoSize"-Modus der Matrix ist standardmäßig aktiviert. In diesem Modus werden die Spalten-/Zellengrößen automatisch kalkuliert. Die Größe der Objekte kann auch manuell gesteuert werden. Hierzu muss die Eigenschaft "AutoSize" deaktiviert werden. Zeilen und Spalten weisen die gleiche Eigenschaft auf. Diese kann genutzt werden, wenn AutoSize der Matrix deaktiviert ist.

Zur Begrenzung der maximalen und minimalen Spaltenbreite, wird eine Spalte ausgewählt und seine "MinWidth" und "Maxwidth" Eigenschaften festgelegt.

Zur Begrenzung der maximalen und minimalen Zeilenhöhe, wird eine Zeile ausgewählt und seine "MinHeight"und "MaxHeight" Eigenschaften festgelegt.

Beispiele

Hier einige Beispiele zur Nutzung des "Matrix"-Objektes. Ein neuer Bericht wird erstellt und das "Matrix"-Objekt auf die Leiste "Berichtstitel" gelegt. Ebenso kann die Datenleiste genutzt werden. In diesem Fall muss die Leiste nicht mit einer Datenquelle verbunden werden. Hier spielt es keine weitere Rolle auf welche der beiden Leisten man die Matrix platziert, da beide beim Start des Berichts gedruckt werden. Der Bericht wird folgendermaßen aussehen:



"Matrix"-Objekte sollten nicht auf Leisten platziert werden, die auf jede neue Seite gedruckt werden, z.B. "Kopfzeile", "Fußzeile", etc. In einem solchen Fall wird die Matrix jedes Mal erstellt, wenn die Leiste gedruckt wird, was wiederum zu einem Stapelüberfluss führen würde.

Für die meisten Beispiele wird die Tabelle "MatrixDemo" genutzt, die mit dem FastReport Paket geliefert wird. Der Bericht umfasst die folgenden Daten:

Name	Year	Month	ItemsSold	Revenue
Nancy Davolio	1999	2	1	1000
Nancy Davolio	1999	11	1	1100
Nancy Davolio	1999	12	1	1200
Nancy Davolio	2000	1	1	1300
Nancy Davolio	2000	2	2	1400
Nancy Davolio	2001	2	2	1500
Nancy Davolio	2001	3	2	1600
Nancy Davolio	2002	1	2	1700
Andrew Fuller	2002	1	2	1800
Andrew Fuller	1999	10	2	1900
Andrew Fuller	1999	11	2	2000
Andrew Fuller	2000	2	2	2100
Janet Leverling	1999	10	3	3000

Name	Year	Month	ItemsSold	Revenue
Janet Leverling	1999	11	3	3100
Janet Leverling	2000	3	3	3200
Steven Buchanan	2001	1	3	4000
Steven Buchanan	2001	2	4	4100
Steven Buchanan	2000	1	4	3999

Beispiel 1 Einfache Matrix

Diese Matrix umfasst einen Wert in einer Zeile und einer Spalte sowie eine Datenzelle. Zur Erstellung einer Matrix muss die Datenspalte "MatrixDemo" folgendermaßen hinzugefügt werden:

- Hinzufügen der Datenspalte "Jahr" zu dem Zeilenkopf;
- Hinzufügen der Datenspalte "Name" zu dem Spaltenkopf;
- Hinzufügen der Datenspalte "Umsatz" zu der Matrixzelle;

Hiernach sieht die Matrix wie folgt aus:

Year	[Name]	Summe
[Year]	[Revenue]	
Summe		

Nun wird das Aussehen der Matrix verbessert:

· Auswahl des Stils "Orange" für die Matrix;

· Auswahl der Schriftart "Tahoma, 8" für alle Matrixzellen;

· Auswahl des Wortes "Summe" in Fettdruck;

· Auswahl der Füllart "Glas" für die Zellen in der oberen Zeile;

· Deaktivierung von "Autosize" und Vergrößerung der Zeilen und Spalten der Matrix.

Hiernach sieht die Matrix wie folgt aus:

Year	[Name]	**Summe**
[Year]	[Revenue]	
Summe		

Nach Durchlauf des Berichts ist das folgende Ergebnis zu erkennen:

Year	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	**Summe**
1999	3900	6100	3300		13300
2000	2100	3200	2700	3999	11999
2001			3100	8100	11200
2002	1800		1700		3500
Summe	7800	9300	10800	12099	39999

Die folgenden Details fehlen in der Matrix:

· Die Datenspalte "Name" trägt keinen Titel;

· Summen werden nicht als Währung gedruckt;

Der Titel der Datenspalte "Name" kann folgendermaßen hinzugefügt werden:

· Der Text "Jahr/Mitarbeiter" kann in die obere rechte Ecke der Matrix hinzugefügt werden;

· Diagonale Linien und das zweite "Text"-Objekt können hier platziert werden. Dies wird oben gezeigt:

Year \ Employee	[Name]	Summe
[Year]	[Revenue]	
Summe		

- Aktivierung des Matrixtitels. Hierzu wird das Element "Show Title" aus dem Kontextmenü des "Matrix"-Objektes ausgewählt. Jeder Text kann in den Titel eingefügt werden:

	Employee	
Year	[Name]	**Summe**
[Year]	[Revenue]	
Summe		

Zur Festlegung der Datenformatierung, wird der Zellenbereich ausgewählt (siehe Abbildung unten) und das Format durch Auswahl des Items "Format" aus dem Kontextmenü festgelegt:

	Employee	
Year	[Name]	**Summe**
[Year]	[Revenue]	
Summe		

Hiernach sieht ein vorbereiteter Bericht wie folgt aus:

	Employee				
Year	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	**Summe**
1999	$3 900,00	$6 100,00	$3 300,00		$13 300,00
2000	$2 100,00	$3 200,00	$2 700,00	$3 999,00	$11 999,00
2001			$3 100,00	$8 100,00	$11 200,00
2002	$1 800,00		$1 700,00		$3 500,00
Summe	$7 800,00	$9 300,00	$10 800,00	$12 099,00	$39 999,00

Beispiel 2: Mehrstufige Kopfzeilen

Diese Matrix umfasst einen Wert in einer Zeile, zwei Werte in einer Spalte, sowie eine Datenzelle. Das vorherige Beispiel dient als Ausgangsbasis und es wird ein neues Element hinzugefügt:

· die Datenspalte "Month" wird, auf der rechten Seite neben dem Element "Year", zum Zeilenkopf hinzugefügt.

Nach dem Hinzufügen eines neuen Elements, wird das Aussehen der Matrix verbessert. Auch die Festlegung der Formatierung der Zellen ist notwendig. Hiernach sieht die Matrix wie folgt aus:

		Employee	
Year	Month	[Name]	**Summe**
[Year]	[Month]	[Revenue]	
	Summe		
Summe			

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	**Summe**
1999	2			$1 000,00		$1 000,00
	10	$1 900,00	$3 000,00			$4 900,00
	11	$2 000,00	$3 100,00	$1 100,00		$6 200,00
	12			$1 200,00		$1 200,00
	Summe	$3 900,00	$6 100,00	$3 300,00		$13 300,00
2000	1			$1 300,00	$3 999,00	$5 299,00
	2	$2 100,00		$1 400,00		$3 500,00
	3		$3 200,00			$3 200,00
	Summe	$2 100,00	$3 200,00	$2 700,00	$3 999,00	$11 999,00
2001	1				$4 000,00	$4 000,00
	2			$1 500,00	$4 100,00	$5 600,00
	3			$1 600,00		$1 600,00
	Summe			$3 100,00	$8 100,00	$11 200,00
2002	1	$1 800,00		$1 700,00		$3 500,00
	Summe	$1 800,00		$1 700,00		$3 500,00
Summe		$7 800,00	$9 300,00	$10 800,00	$12 099,00	$39 999,00

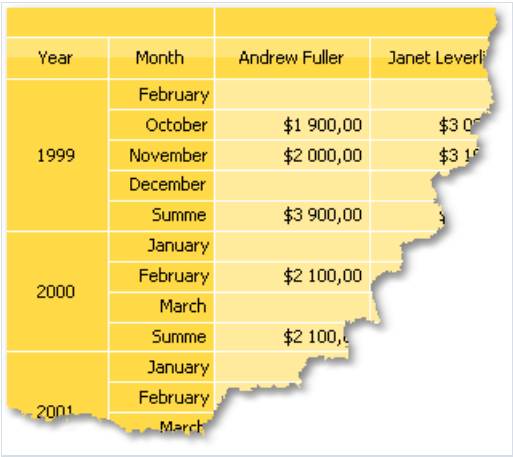
Beispiel 3: Druck des Namen des Monats

Im vorherigen Beispiel wurden die Nummern der Monate in eine Matrix gedruckt. Dies liegt darin begründet, dass die Datenspalte "Month" die Nummer des Monats und nicht ihren Namen aufwies. Um den Namen des Monats zu drucken, muss das Folgende durchgeführt werden:

- Auswahl der Zelle in der die Nummer des Monats gedruckt wird. In diesem Beispiel handelt es sich um die Zelle "Zelle8";
- Klicken der Schaltfläche im Fenster "Eigenschaften" und doppelklicken des Ereignisses "BeforePrint";
- FastReport fügt einen leeren Ereignishandler in das Skript des Berichts ein. Eingabe des folgenden Codes:

```
private void Cell8_BeforePrint(object sender, EventArgs e)
{
  string[] month Names = new string[] {
    "January", "February", "March", "April",
    "May", "June", "July", "August",
    "September", "October", "November", "December" };
  // Cell8 is a cell that prints the month number.
  // Cell8.Value is a value printed in the cell (i.e. the month number).
  // This value is of System.Object type, so we need to cast it to int
  Cell8.Text = monthNames[(int)Cell8.Value - 1];
}
```

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:



Year	Month	Andrew Fuller	Janet Leverl
1999	February		
	October	$1 900,00	$3 0
	November	$2 000,00	$3 1
	December		
	Summe	$3 900,00	$
2000	January		
	February	$2 100,00	
	March		
	Summe	$2 100,	
2001	January		
	February		
	March		

Beispiel 4: Bedingte Markierung

Für Matrixzellen kann, wie für "Text"-Objekte auch, eine bedingte Markierung festgelegt werden. Detaillierte Informationen hierzu werden im Kapitel "Conditional highlighting" erläutert.

In Beispiel 2 wird erläutert, wie ein Betrag höher als 3000, in rot hervorgehoben werden kann. Wählen Sie hierzu die Zelle mit dem Text "Revenue" und drücken Sie die Schaltfläche auf der "Text"-Symbolleiste. Fügen Sie die folgende Bedingung zum Bedingung Editor hinzu:

```
Value > 3000
```

Wählen Sie die rote Textfarbe für die Bedingung aus. Hiernach sieht ein vorbereiteter Bericht wie folgt aus:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	**Summe**
1999	February			$1 000,00		$1 000,00
	October	$1 900,00	$3 000,00			$4 900,00
	November	$2 000,00	$3 100,00	$1 100,00		$6 200,00
	December			$1 200,00		$1 200,00
	Summe	$3 900,00	$6 100,00	$3 300,00		$13 300,00
2000	January			$1 300,00	$3 999,00	$5 299,00
	February	$2 100,00		$1 400,00		$3 500,00
	March		$3 200,00			$3 200,00
	Summe	$2 100,00	$3 200,00	$2 700,00	$3 999,00	$11 999,00
2001	January				$4 000,00	$4 000,00
	February			$1 500,00	$4 100,00	$5 600,00
	March			$1 600,00		$1 600,00
	Summe			$3 100,00	$8 100,00	$11 200,00
2002	January	$1 800,00		$1 700,00		$3 500,00
	Summe	$1 800,00		$1 700,00		$3 500,00
Summe		$7 800,00	$9 300,00	$10 800,00	$12 099,00	$39 999,00

Wie sich erkennen lässt, sind Gesamtwerte nicht hervorgehoben. Dies liegt darin begründet, dass lediglich für eine Zelle das bedingte Hervorheben ausgewählt wurde. Zur Hervorhebung der restlichen Werte, ist es notwendig das Hervorheben für alle Matrixzellen festzulegen.

In diesem Beispiel wurde das bedingte Hervorheben genutzt, das vom Zellwert selbst abhängt. Eine Zelle kann entsprechend des Wertes des Matrixkopfes hervorgehoben werden. Am folgenden Beispiel soll gezeigt werden. wie eine Zelle, die zum Jahr 2000 gehört, in rot hervorgehoben wird. Hierzu werden die Matrixzellen, wie in der folgenden Abbildung dargestellt, angezeigt:

		Employee	
Year	Month	[Name]	**Summe**
[Year]	[Month]	[Revenue]	
	Summe		
Summe			

Die folgende Bedingung zum Hervorheben wird festgelegt:

```
(int)Matrix1.RowValues[0] == 2000
```

In diesem Fall lautet der Name der Matrix "Matrix1". Die Eigenschaft "RowValues" der Matrix hat ein "object[]" und

beinhaltet ein Feld von Werten aus dem Zeilenkopf der aktuell gedruckten Zeile. Die Anzahl an Werten in den Feldern entspricht der Anzahl der Ebenen in einer Kopfzeile. Das Beispiel hat zwei Werte; der erste ist "Year" und der zweite ist "Month".

> Die letzte Zeile wird nicht hervorgehoben. Die Eigenschaft "RowValues" hat einen unbestimmten Wert und wird beim Erstellen des Berichtes einen Fehler verursachen.

Beim Durchlauf eines solchen Berichtes lässt sich Folgendes erkennen:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	**Summe**
1999	February			$1 000,00		$1 000,00
	October	$1 900,00	$3 000,00			$4 900,00
	November	$2 000,00	$3 100,00	$1 100,00		$6 200,00
	December			$1 200,00		$1 200,00
	Summe	$3 900,00	$6 100,00	$3 300,00		$13 300,00
2000	January			$1 300,00	$3 999,00	$5 299,00
	February	$2 100,00		$1 400,00		$3 500,00
	March		$3 200,00			$3 200,00
	Summe	$2 100,00	$3 200,00	$2 700,00	$3 999,00	$11 999,00
2001	January				$4 000,00	$4 000,00
	February			$1 500,00	$4 100,00	$5 600,00
	March			$1 600,00		$1 600,00
	Summe			$3 100,00	$8 100,00	$11 200,00
2002	January	$1 800,00		$1 700,00		$3 500,00
	Summe	$1 800,00		$1 700,00		$3 500,00
Summe		$7 800,00	$9 300,00	$10 800,00	$12 099,00	$39 999,00

> Man kann die Matrix-Eigenschaft "ColumnValues" zur Referenzierung der Spaltenwerte nutzen.

Beispiel 5: Hervorheben gerader Zeilen

Zur Verbesserung des Aussehens einer Matrix können gerade oder ungerade Zeilen hervorgehoben werden. Zur Erläuterung wird Beispiel Nr. 2 herangezogen.

Wählen Sie den gesamten Bereich der Matrixdaten, wie in der Abbildung dargestellt:

		Employee	
Year	Month	[Name]	**Summe**
[Year]	[Month]	[Revenue]	
	Summe		
Summe			

Rufen Sie den Markierungs Editor aus. Fügen Sie die folgende Bedingung hinzu:

```
Matrix1.RowIndex % 2 != 0
```

und wählen Sie eine Hintergrundfarbe die ein wenig dunkler als die vorherige ist. In diesem Fall lautet der Name der Matrix "Matrix1". Die Eigenschaft "RowIndex" der Matrix gibt die Anzahl der aktuell gedruckten Zeilen an.

> Auf die gleiche Art wird die Eigenschaft "ColumnIndex" der Matrix zum Hervorheben der Spalten genutzt.

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

		Employee				
Year	Month	Andrew Fuller	Janet Leverling	Nancy Davolio	Steven Buchanan	**Summe**
1999	February			$1 000,00		$1 000,00
	October	$1 900,00	$3 000,00			$4 900,00
	November	$2 000,00	$3 100,00	$1 100,00		$6 200,00
	December			$1 200,00		$1 200,00
	Summe	$3 900,00	$6 100,00	$3 300,00		$13 300,00
2000	January			$1 300,00	$3 999,00	$5 299,00
	February	$2 100,00		$1 400,00		$3 500,00
	March		$3 200,00			$3 200,00
	Summe	$2 100,00	$3 200,00	$2 700,00	$3 999,00	$11 999,00
2001	January				$4 000,00	$4 000,00
	February			$1 500,00	$4 100,00	$5 600,00
	March			$1 600,00		$1 600,00
	Summe			$3 100,00	$8 100,00	$11 200,00
2002	January	$1 800,00		$1 700,00		$3 500,00
	Summe	$1 800,00		$1 700,00		$3 500,00
Summe		$7 800,00	$9 300,00	$10 800,00	$12 099,00	$39 999,00

Beispiel 6: Nutzung von Ausdrücken

In vorherigen Beispielen wurde eine Matrix durch Ziehen von Spalten aus dem "Daten"-Fenster erstellt. Zu diesem Zweck können auch Ausdrücke verwendet werden. Zum Einfügen eines Ausdrucks in eine Matrix wird das Folgende getan:

- Hinzufügen eines beliebigen Elementes aus dem "Daten"-Fenster in die Matrix. Dies kann ein beliebiges Element sein, beispielsweise, eine Systemvariable "Daten". Sie wird lediglich zur Erstellung eines Matrixelementes genutzt;
- Doppelklicken des Elements und Auswahl der benötigten Ausdrücke im Fenster des Expressions Editor.

> Sollte die Matrix Ausdrücke anstelle von Datenfeldern aufweisen, muss überprüft werden, ob die Eigenschaft "Datenquelle" ordnungsgemäß eingestellt wurde. Bei der Arbeit mit Datenspalten, wird diese Eigenschaften automatisch durch Ziehen einer Spalte in die Matrix, gefüllt.

Hier ein Beispiel zur Nutzung von Ausdrücken. Hierzu nutzen wir die Tabelle "OderDetails". Sie enthält als Datenquelle eine Liste verkaufter Produkte, die nach Mitarbeiter/in sortiert ist. Diese Tabelle beinhaltet mehrere Relationen, die Zugang zu dem Namen eines Mitarbeiters, einem Produktnamen und seiner Kategorie gewähren.

Die Matrix zeigt die Verkäufe eines jeden Mitarbeiters kategorisiert nach Produkten, an. Zur Erstellung der Matrix sind die folgenden Schritte notwendig:

- Hinzufügen der Datenspalte "Order Details.Products.Categories.CategoryName" zu dem Spaltenkopf;
- Hinzufügen eines beliebigen Elements zu einem Zeilenkopf zur Erstellung eines Matrixelements. Dann Festlegung der folgenden Ausdrücke für das Kopfzeilenelement:

```
[Order Details.Orders.Employees.FirstName] + " " + [Order Details.Orders.Employees.LastName]
```

- Hinzufügen eines beliebigen Elements zu Datenzellen zur Erstellung eines Matrixelements. Dann Festlegung der folgenden Ausdrücke für die Zelle:

```
[Order Details.UnitPrice] * [Order Details.Quantity] * (decimal)(1 - [Order Details.Discount])
```

> Warum wurde eine solch lange Datenspalte als Name für eine/n Mitarbeiter/In angezeigt, obwohl wir den Namen von der "Employees.FirstName" erhalten könnten? Dies geschah aufgrund der Verknüpfung der Matrix mit der Datenquelle "Order Details". Die Nutzung von Relationen zwischen diesen Datenquellen und anderen Tabellen vereinfacht den Bezug auf deren Spalten (weitere Details zu Relationen können im Kapitel "Daten" nachgelesen werden). Bezieht man sich direkt auf die Datenspalte "Employees.FirstName" erhält man den Namen des ersten Mitarbeiters/der ersten Mitarbeiterin in der Tabelle.

Festlegen des Aussehens der Matrix. Hiernach schaut sie wie folgt aus:

Employee	[CategoryName]	Summe
[Order Details.Orders.Employees.FirstName] + " " + [Order Details.Orders.Employees.LastName]	[[Order Details.UnitPrice] * [Order Details.Quantity] * (decimal)(1 - [Order Details.Discount])]	
Summe		

Wenn der Bericht durchläuft, sieht man eine recht große Matrix, die sich über 2 Seiten erstreckt:

Employee	Beverages	Condiments	Confections	Dairy Products	Grains/Ce
Andrew Fuller	40 248,25	14 850,67	21 455,69	23 812,55	11
Anne Dodsworth	19 642,56	10 125,55	8 053,16	21 101,13	1
Janet Leverling	44 757,41	13 381,64	33 622,40	32 320,84	2
Laura Callahan	17 897,85	14 637,66	21 699,91	21 269,47	
Margaret Peacock	50 308,21	23 314,87	27 768,73	35	
Michael Suyama	9 450,20	4 648,47	6 859,63		
Nancy Davolio	46 599,36	13 561,56	28 568,2		
Robert King	27 963,83	8 851,38			
Steven Buchanan	11 000,53				

Beispiel 7: Bilder in Zellen

Matrixzellen werden aus dem "Text"-Objekt vererbt und können Textdaten anzeigen. Ist dies nicht ausreichend, können auch Objekte in Zellen platziert werden. Hier soll näher erläutert werden, wie ein Bild in einer Matrix angezeigt wird.

Hierzu wird Beispiel 6 als Ausgangspunkt herangezogen. Es wird ein Foto eines Mitarbeiters/einer Mitarbeiterin (Datenspalte "Employees.Photo") herangezogen und das Kategorienbild (Datenspalte "Categories.Picture") hinzugefügt. Folgende Schritte sollten befolgt werden:

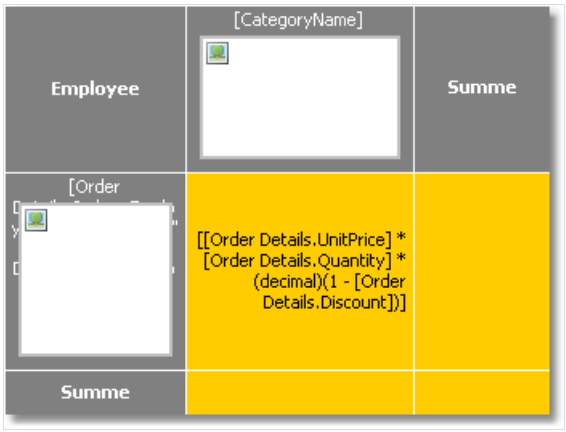
- Auswahl der Zelle, die den Namen des/der Mitarbeiters/Mitarbeiterin beinhaltet und Vergrößerung;
- Hinzufügen des "Bild"-Objektes zur Zelle;
- Zur Anzeige des Fotos eines Mitarbeiters/einer Mitarbeiterin, wird das "Bild"-Objekt mit der folgenden Datenspalte verbunden (dies findet über den Objekt Editor statt):s

```
Order Details.Orders.Employees.Photo
```

- Auswahl der Zelle, die den Kategorienamen beinhaltet und Vergrößerung;
- Hinzufügen des "Bild"-Objektes zur Zelle;
- Zur Anzeige eines Kategoriebildes, wird das "Bild"-Objekt mit der folgen Datenspalte verbunden (dies findet über den Objekt Editor statt):

```
Order Details.Products.Categories.Picture
```

Hiernach sieht die Matrix wie folgt aus:



Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

Employee	Beverages	Condiments	Confections
Andrew Fuller	40 248,25	14 850,67	
Anne Dodsworth	19 642,56	10 125,	
Janet Leverling	44		

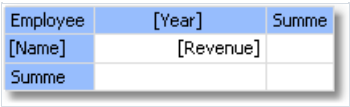
Beispiel 8: Objekte in Zellen

Durch Einfügen von Objekte in Matrixzellen, können mehrere visuelle Effekte erzielt werden. Im folgenden Beispiel wird erläutert, wie eine einfache Skala, die die Verkaufsniveaus der Mitarbeiter/innen anzeigt, gezeichnet werden kann.

Die Matrix nutzt die Datenquelle "MatrixDemo". Zu Erstellung einer Matrix, sollten Datenspalten auf folgende Art eingefügt werden:

- Hinzufügen der Datenspalte "Year" zu dem Zeilenkopf;
- Hinzufügen der Datenspalte "Name" zu dem Spaltenkopf;
- Hinzufügen der Datenspalte "Revenue" zu der Matrixzelle.

Das Aussehen der Matrix wird folgendermaßen festgelegt:



Nun werden drei "Form"-Objekte in die Zelle, mit dem Wert "Revenue", eingefügt. Diese Objekte dienen als Indikatoren:

- Liegt der Wert einer Zelle unter 100, wird lediglich ein rotes Objekt angezeigt;
- Liegt der Wert einer Zelle unter 3000, werden zwei gelbe Objekte angezeigt;
- Liegt der Wert einer Zelle über oder gleich 3000, werden drei grüne Objekte angezeigt.

Die Matrix sieht folgendermaßen aus:

Zur Steuerung der Objekte nutzen wir ein Ereignishandler für eine Matrixzelle. Hierzu wird die Zelle "Revenue" ausgewählt und der Ereignishandler "BeforePrint" mittels des "Eigenschafts"-Fensters erstellt. Der folgende Code wird in den Handler eingegeben:

```
private void Cell4_BeforePrint(object sender, EventArgs e)
{
  // In our example, a cell has the Cell4 name.
  // Get cell value which is in the Cell4.Value property.
  // Some cells in our matrix will be empty. We'll take it into account (null check).
  // The value should be cast to decimal type, because data source column
  // [MatrixDemo.Revenue] is of System.Decimal type.
  decimal value = Cell4.Value == null ? 0 : (decimal)Cell4.Value;

  // Switch shape objects on or off depending on the value:
  // value < 100 - one object is visible;
  // value < 3000 - two objects are visible;
  // value >= 3000 - all objects are visible
  Shape1.Visible = true;
  Shape2.Visible = value >= 100;
  Shape3.Visible = value >= 3000;

  // Choose the color of objects:
  // value < 100 - red color;
  // value < 3000 - yellow color;
  // value >= 3000 - green color
  Color color = Color.Red;
  if (value >= 100)
    color = Color.Yellow;
  if (value >= 3000)
    color = Color.GreenYellow;

  // Set the objects' color
  Shape1.Fill = new SolidFill(color);
  Shape2.Fill = new SolidFill(color);
  Shape3.Fill = new SolidFill(color);
}
```

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

Employee	1999	2000	2001	2002	Summe
Andrew Fuller	3 900,00	2 100,00		1 800,00	7 800,00
Janet Leverling	6 100,00	3 200,00			9 300,00
Nancy Davolio	3 300,00	2 700,00	3 100,00	1 700,00	10 800,00
Steven Buchanan		3 999,00	8 100,00		12 099,00
Summe	13 300,00	11 999,00	11 200,00	3 500,00	39 999,00

Beispiel 9: Manuelles Füllen einer Matrix

In allen bisher betrachteten Beispielen wurde die Matrix automatisch mit Daten gefüllt, da sie mit einer Datenquelle verbunden war. Die Datenquelle der Matrix ist in der Eigenschaft "DatenSource" angezeigt. Auch wenn der Wert dieser Eigenschaft nicht manuell eingestellt wurde, erschien er jedoch implizit, wenn Datenspalten zu der Matrix hinzugefügt werden.

Mittels Skript kann die Matrix auch manuell gefüllt werden. Zu Erstellung wird der Ereignishandler "ManualBuild" der Matrix genutzt. Aufruf der Methode "AddValue" im Ereigniscode um einen Wert hinzuzufügen. Hier wird dargestellt, wie eine Matrix erstellt werden kann, die eine 10x10-Tabelle der folgenden Art druckt:

	1	2	3	...
1	1			
2		2		
3			3	
...				...

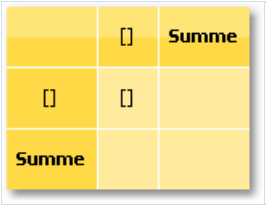
Folgende Schritte müssen hierzu befolgt werden:

· Hinzufügen einer leeren Matrix zum Bericht;

· Legen eines jeden Elementes aus dem "Daten"-Fenster in die Zeile, Spalte oder Zelle der Matrix. Aufruf des Expression Editors durch doppelklicken des Matrixelementes und bereinigen eines Ausdrucks;

· Bereinigen der Eigenschaft "Datenquelle" der Matrix.

Diese Schritte sind zur Erstellung einer "Dummy"-Matrix, mit einer Zeile, Spalte und Zelle, notwendig. Hiernach sieht die Matrix wie folgt aus:



Nun wird der Ereigsnishandler "ManualBuild" erstellt. Hierzu wählen Sie die Matrix, gehen Sie zum Fenster "Eigenschaften" und klicken Sie die Schaltfläche . Durch Doppelklicken auf das Ereignis "ManualBuild" erstellt FastReport einen leeren Ereignishandler. Der folgende Code wird in den Handler eingegeben:

```
private void Matrix1_ManualBuild(object sender, EventArgs e)
{
  // Our matrix has one level in row, column and cell.
  // Create 3 arrays of object[] type, each with one element
  // (per number of levels).
  object[] columnValues = new object[1];
  object[] rowValues = new object[1];
  object[] cellValues = new object[1];

  for (int i = 1; i <= 10; i++)
  {
    // Filling arrays
    columnValues[0] = i;
    rowValues[0] = i;
    cellValues[0] = i;

    // Adding data into the matrix
    Matrix1.AddValue(columnValues, rowValues, cellValues);
  }
}
```

Um die Matrix mit Daten zu füllen, sollte in einem Handler die Methode "AddValue" des "Matrix"-Objektes genutzt werden. Diese Methode weist drei Parameter auf, von der jeder ein Array des Typus System.Object ist. Der erste Parameter ist ein Spaltenwert, der Zweite ein Zeilenwert und der Dritte ist ein Zellenwert. Hinweis: Die Anzahl der Werte in jedem Array sollte mit den Einstellungen des Objektes übereinstimmen. In diesem Beispiel hat das Objekt eine Ebene in Spalte, Zeile und Zelle; entsprechend geben wir einen Wert für Spalte, einen für Zeile und einen für Zelle ein.

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

	1	2	3	4	5	6	7	8	9	10	**Summe**
1	1										1
2		2									2
3			3								3
4				4							4
5					5						5
6						6					6
7							7				7
8								8			8
9									9		9
10										10	10
Summe	1	2	3	4	5	6	7	8	9	10	55

Hier ein Beispiel, wie der Wert "21" an der Schnittmenge zwischen Spalte 7 und Zeile 3, in die Matrix eingefügt werden kann. Hierzu wird der Code auf folgende Weise geändert:

```
private void Matrix1_ManualBuild(object sender, EventArgs e)
{
  object[] columnValues = new object[1];
  object[] rowValues = new object[1];
  object[] cellValues = new object[1];

  for (int i = 1; i <= 10; i++)
  {
    columnValues[0] = i;
    rowValues[0] = i;
    cellValues[0] = i;

    Matrix1.AddValue(columnValues, rowValues, cellValues);
  }
  columnValues[0] = 7;
  rowValues[0] = 3;
  cellValues[0] = 21;
  Matrix1.AddValue(columnValues, rowValues, cellValues);
}
```

Das Ergebnis sieht folgendermaßen aus:

	1	2	3	4	5	6	7	8	9	10	Summe
1	1										1
2		2									2
3			3				21				24
4				4							4
5					5						5
6						6					6
7							7				7
8								8			8
9									9		9
10										10	10
Summe	1	2	3	4	5	6	28	8	9	10	76

Wie sich ablesen lässt, berechnet die Matrix die Summe automatisch.

Man kann den Ereignishandler für die Matrix "ManualBuild", der mit den Daten verbunden ist, nutzen. In diesem Fall wird der Ereignishandler zuerst aufgerufen, dann die Matrix mit Daten aus der Datenquelle gefüllt.

Objekt "Erweiterte Matrix"

Mit diesem Objekt wie mit dem "Matrix" Objekt können Sie Kreuztabellenberichte erstellen.

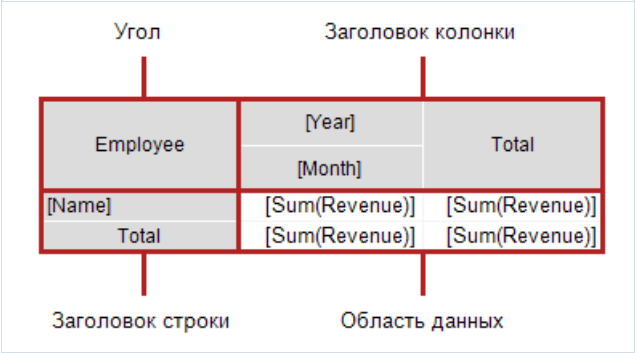
	Top 5 customers		Total
	Total	Others (84)	
Beverages (12)	$144 007,88	$192 684,30	**$336 692,18**
Condiments (12)	$123 407,20	$105 665,89	**$229 073,09**
Confections (13)	$54 823,13	$112 615,09	**$167 438,23**
Dairy Products (10)	$103 074,28	$166 433,01	**$269 507,29**
Grains/Cereals (7)	$31 368,01	$64 376,58	**$95 744,59**
Meat/Poultry (6)	$269 495,38	$113 526,98	**$383 022,36**
1. Alice Mutton	$13 428,48	$19 269,90	**$32 698,38**
2. Mishi Kobe Niku	$1 319,20	$5 907,30	**$7 226,50**
3. Pâté chinois	$7 137,60	$10 288,80	**$17 426,40**
4. Perth Pasties	$6 916,56	$13 657,61	**$20 574,17**
5. Thüringer Rostbratwurst	$239 795,40	$60 573,28	**$300 368,67**
6. Tourtière	$898,14	$3 830,10	**$4 728,24**
Produce (5)	$67 154,22	$71 955,36	**$139 109,58**
Seafood (12)	$31 732,55	$99 591,19	**$131 323,74**
Total	**$825 062,63**	**$926 848,41**	**$1 751 911,04**

Hier ist die Liste der Schlüsselbesonderheiten:

- Zeilen- und Spaltenüberschriften können Gruppen und einfache Elemente in beliebiger Reihenfolge enthalten. Dadurch können asymmetrische Berichte erstellt werden;
- über die Ausblenden-Schaltflächen können Sie die Sichtbarkeit einzelner Elemente interaktiv steuern;
- Mit den Sortierschaltflächen können Sie die Matrix interaktiv nach ausgewählten Werten, einschließlich der Gesamtwerten sortieren;
- durch das Top N-Gruppieren können Sie N-Werte in der Überschrift anzeigen und die übrigen Werte in einem separaten Element mit der Erweiterungsmöglichkeit gruppieren;
- stufenweise Anzeige der Matrix- Überschriften;
- Überschriftensortierung nach den Gesamtwerten;
- ein breites Spektrum an Aggregatfunktionen;
- Unterstützung der benutzerdefinierten Aggregatfunktionen;
- ein breites Spektrum an Sonderfunktionen, mit denen Sie Gesamtwerte, Werte der benachbarten Zellen abrufen können, sowie Funktionen für Prozentrechnung;
- Unterstützung von Objekten "Sparkline" und "Fortschrittsanzeige" in den Datenzellen.

Struktur der Matrix

Das Objekt "Erweiterte Матрix" besteht aus folgenden Elementen:



Ecke

Die Zellen in der Ecke der Matrix können beliebige Informationen enthalten. Sie können sie auch nach Belieben aufteilen/verbinden.

Überschrift

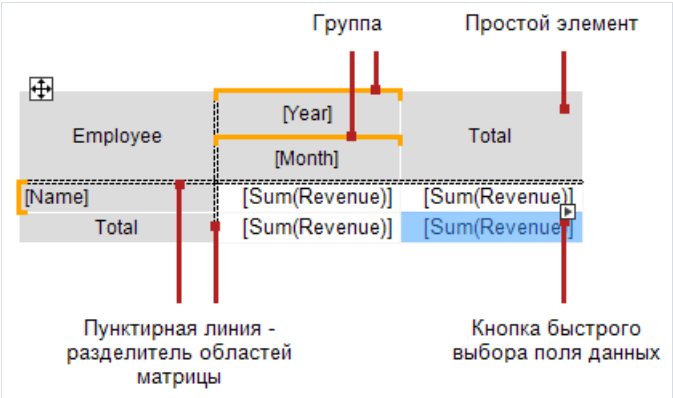
Eine Matrixüberschrift kann zwei Arten von Elementen enthalten:

- einfaches Element: zeigt statische Informationen an, z. B. den Text "Gesamt".
- Gruppe: zeigt eine Liste von Werten an, die nach einem bestimmten Kriterium gruppiert sind.

Die Überschrift hat eine Baumstruktur. Das Wurzelelement ist unsichtbar, es enthält sichtbare Elemente der ersten Ebene.

Es ist zugelassen, die Elemente frei anzuordnen; so kann eine Überschrift beispielsweise keine oder mehrere Gruppen nebeneinander enthalten. Gesamtwerte können auch frei angeordnet werden.

Im Entwurfsmodus zeigt die Matrix visuelle Hinweise im Überschriftenbereich an:



In diesem Fall sieht die Struktur wie folgt aus:

```
Zeilenüberschrift
- Gruppe "Name"
- Element "Total"

Spaltenüberschrift
- Gruppe "Year"
  - Gruppe "Month"
- Element "Total"
```

Siehe Abschnitt Konfigurieren der Überschrift für mehr Informationen.

Datenbereich

Die Zellen im Datenbereich enthalten in der Regel eine Aggregatfunktion. Siehe Abschnitt Konfigurieren des Datenbereichs.

Matrixkonfiguration

Struktureinstellung

Folgen Sie den nachstehenden Schritten, um die Matrix zu konfigurieren:

1. Überschriften konfigurieren (siehe Abschnitt Konfigurieren der Überschrift).
2. Datenzellen konfigurieren (siehe Abschnitt Konfigurieren des Datenbereichs).
3. Gesamtwerte hinzufügen (siehe Abschnitt Konfigurieren der Überschrift). Dieser Schritt wird am besten zuletzt ausgeführt, um Zeit beim Konfigurieren neuer Datenzellen zu sparen.

> Die Matrix muss mit der Datenquelle verbunden sein. Dafür ist die Eigenschaft DataSource zuständig. Diese Eigenschaft wird in der Regel automatisch während der Überschriften- und Zellenkonfigurationsphase gesetzt.

Kontextmenü

Um das Kontextmenü aufzurufen, wählen Sie ein beliebiges Matrixelement und klicken Sie dann mit der rechten Maustaste auf den Bereich in der linken oberen Ecke der Matrix:

Die folgenden Befehle sind im Menü verfügbar:

"Stil" - einen der voreingestellten Stile auswählen;

"Spalten und Zeilen tauschen" – ermöglicht, schnell Spalten und Zeilen in einer Matrix zu tauschen;

"Überschriften auf einer neuen Seite wiederholen" - wenn die Matrix aus mehreren Seiten besteht, werden die Spalten- und Zeilenüberschriften auf jeder Seite gedruckt.

Im Eigenschaftenfenster verfügbaren Einstellungen

Im Eigenschaftenfenster sind folgende Eigenschaften verfügbar, die für das Objekt "Erweiterte Matrix" spezifisch sind:

Eigenschaft	Wert	Beschreibung
DataRowPriority	Rows	Priorität der Überschriften beim Zugriff auf Datenbankfelder aus Datenzellen. Siehe Abschnitt Eigenschaften, auf die über Datenzellen zugegriffen werden kann.
DataSource		Datenquelle.

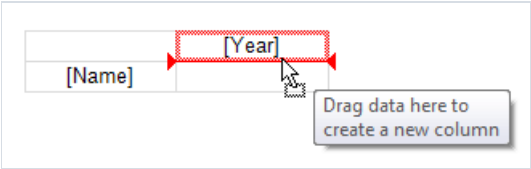
Eigenschaft	Wert	Beschreibung
EvenStylePriority	Rows	Priorisierung von Zeilen- oder Spaltennummern für das Funktionieren der EvenStyle-Eigenschaft.
Filter		Ausdruck zum Datenfiltern. Siehe Abschnitt Datenfiltern.
Layout	AcrossThenDown	Siehe Abschnitt Tabellenlayout.
PrintIfEmpty	True	Matrix drucken, wenn sie leer ist.
RepeatHeaders	True	Überschriften auf einer neuen Seite wiederholen.
ResetDataOnRun	False	Daten jedes Mal zurücksetzen, wenn der Bericht ausgeführt wird. Standardmäßig wird die Matrix bei interaktiven Operationen nicht umgebaut (siehe Abschnitt Interaktive Funktionen).
Style		Matrix-Stil.
WrappedGap	0	Lücke zwischen den Matrix-Teilen im Einblenden/Ausblenden Modus Layout = Wrapped.

Konfiguration der Überschrift

Hinzufügen eines Elements

Es gibt zwei Optionen ein Element zu einer Überschrift hinzuzufügen:

- durch Ziehen des Feldes aus dem Datenfenster. Beim Ziehen wird angezeigt, in welchem Teil der Überschrift das neue Element hinzugefügt wird:



- über das Kontextmenü der Überschrift. Wählen Sie das Element aus, neben dem Sie ein neues Element hinzufügen möchten, und wählen Sie aus dem Kontextmenü den Befehl "Neues Element hinzufügen". Ein leeres neues Element wird hinzugefügt;
- Sie können auch über das Kontextmenü ein Element "Gesamtwert" (vor oder nach dem ausgewählten Element) hinzufügen. Es wird ein neues Element mit dem Text "Gesamt" hinzugefügt (der Text hängt von der aktuellen Position ab).

> Das Hinzufügen eines Gesamtwertes entspricht dem Hinzufügen eines leeren Elements und dem Bearbeiten seines Textes.

> Wenn ein Datenbankfeld aus dem Datenfenster zu einer leeren Matrix hinzugefügt wird, wird ihre DataSource-Eigenschaft automatisch konfiguriert.

Löschen eines Elements

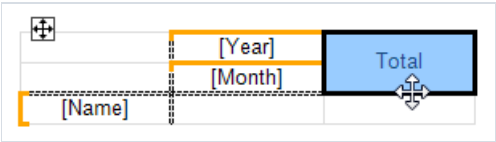
Sie können ein Element löschen, indem Sie im Kontextmenü "Löschen" wählen. Sie können nur das ausgewählte Element oder den Elementenbaum (das ausgewählte Element und alle seine untergeordneten Elemente) löschen.

Sie können ein Element mit der Delete-Taste auch löschen. In diesem Fall wird nur das ausgewählte Element gelöscht.

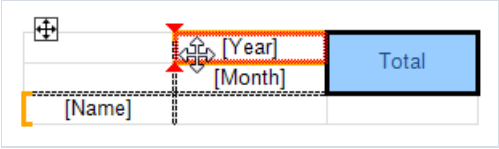
> Das Element mit dem Schloss-Symbol kann nicht wie beschrieben gelöscht werden. Siehe Abschnitt TopN Gruppieren.

Verschieben eines Elements

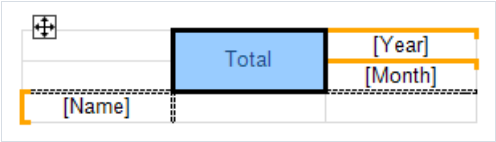
Um ein Element an eine neue Stelle zu verschieben, wählen Sie es mit der linken Maustaste aus. Das Element wird mit einem dicken schwarzen Rahmen hervorgehoben:



Fassen Sie das Element an seinem Rahmen und ziehen Sie es an eine neue Stelle. Beim Ziehen wird angezeigt, in welchen Teil der Überschrift das Element verschoben wird:



Lassen Sie die Maustaste los und das Element wird an seine neue Stelle verschoben:



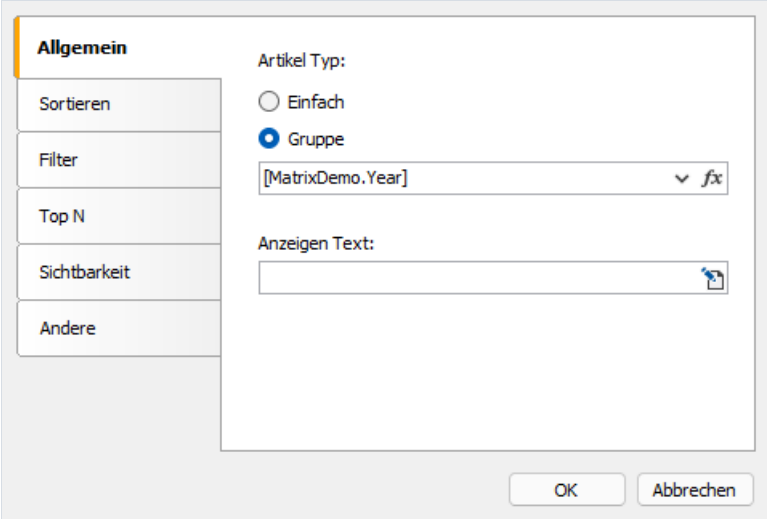
> Wenn ein Element untergeordnete Elemente hat, werden diese mit dem Hauptelement verschoben, soweit es möglich ist.

Bearbeiten eines Elements

Rufen Sie den Artikel-Editor mit einem Doppelklick auf, oder wählen Sie "Bearbeiten..." aus dem Kontextmenü. Sie können den Editor auch mit der Enter-Taste aufrufen.

Gruppieren

Wie oben beschrieben, kann eine Matrixüberschrift zwei Arten von Elementen enthalten: Gruppen und einfache Elemente. Der Elementtyp wird im Überschriftenedtor eingestellt:



> Ein einfaches Element kann zu einer Gruppe gemacht werden und umgekehrt. Für ein einfaches Element sind einige Einstellungen im Editorfenster nicht verfügbar.

Mit der Gruppe können Sie eine Liste von Werten anzeigen, die nach einer Bedingung gruppiert sind. Im obigen Beispiel ist die Bedingung `[MatrixDemo.Year]` angegeben. Das bedeutet, dass das Element eine Liste von Jahren anzeigt. Die Liste enthält sich nicht wiederholende Werte; identische Werte werden gruppiert.

Standardmäßig zeigt die Gruppe die in der Gruppierungsbedingung angegebenen Werte an. Wenn zum Beispiel die Bedingung `[MatrixDemo.Month]` angegeben wird, werden die Monatszahlen angezeigt. Mit der Eigenschaft "Anzeigetext" kann ein anderer Wert angezeigt werden, z. B.:

```
[MonthName([MatrixDemo.Month])]
```

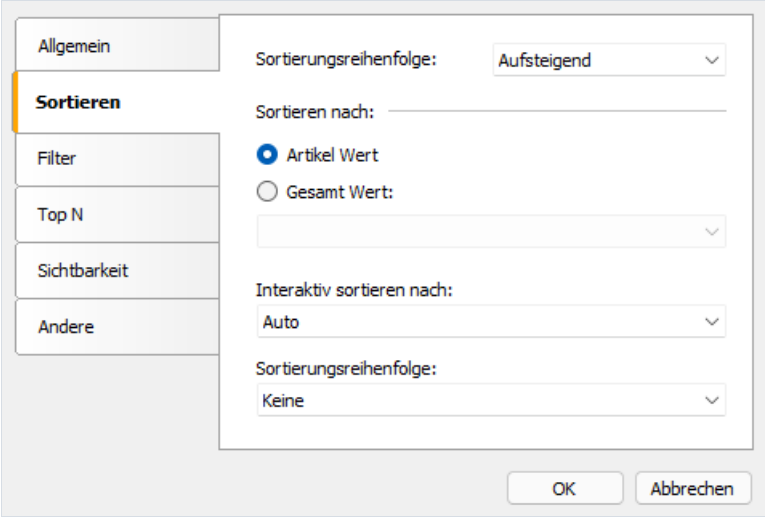
zeigt den Namen des Monats anstelle seiner Zahl an.

In dieser Eigenschaft kann auf spezielle Matrixfunktionen zugegriffen werden (siehe Abschnitt Eigenschaften, auf die über Kopfzellen zugegriffen werden kann), z.B.:

```
[Matrix1.RowNo]. [MatrixDemo.Name]
```

Sortieren

Das Sortieren der Ausgabewerte wird in der gleichnamigen Registerkarte des Überschriften-Editors konfiguriert. Die Einstellungen sind aktiv, wenn der "Gruppe" Elententyp ausgewählt ist:



Beschreibung der Einstellungen:

"Sortierunsreihenfolge" legt die Sortierreihenfolge fest - aufsteigend, absteigend oder keine Sortierung.

"Sortieren nach" - wählen Sie eine der beiden Sortieroptionen: nach Ausgabewert oder nach Gesamtwert. Die folgenden Abbildungen zeigen, wie das Element "Jahr" sortiert wird:

Nach Elementenwert:

	2011	2012	2013	2014	2015
Andrew Fuller	$3,900.00	$2,100.00			$1,800.00
Janet Leverling	$6,100.00	$3,200.00			
Nancy Davolio	$3,300.00	$2,700.00	$3,100.00		$1,700.00
Steven Buchanan			$3,999.00	$8,100.00	
Total	$13,300.00	$8,000.00	$7,099.00	$8,100.00	$3,500.00

Nach Gesamtwert:

	2015	2013	2012	2014	2011
Andrew Fuller	$1,800.00		$2,100.00		$3,900.00
Janet Leverling			$3,200.00		$6,100.00
Nancy Davolio	$1,700.00	$3,100.00	$2,700.00		$3,300.00
Steven Buchanan		$3,999.00		$8,100.00	
Total	$3,500.00	$7,099.00	$8,000.00	$8,100.00	$13,300.00

"Interaktives Sortieren nach" legt fest, wie die Überschriftenwerte sortiert werden sollen, wenn interaktives Sortieren aktiv ist (siehe Abschnitt Interaktives Sortieren). Mögliche Varianten sind "Nein", "Automatisch" und Gesamtwertname.

"Die Sortierreihenfolge wird durch die Schaltfläche umgeschaltet". Hier wird der Name der `MatrixSortButton` -Schaltfläche angegeben, der sich in der Ecke oder außerhalb der Matrix befindet. Wenn die Schaltfläche angeklickt wird, wird im Fenster Berichtvorschau die Sortierreihenfolge geändert und die Matrix aktualisiert.

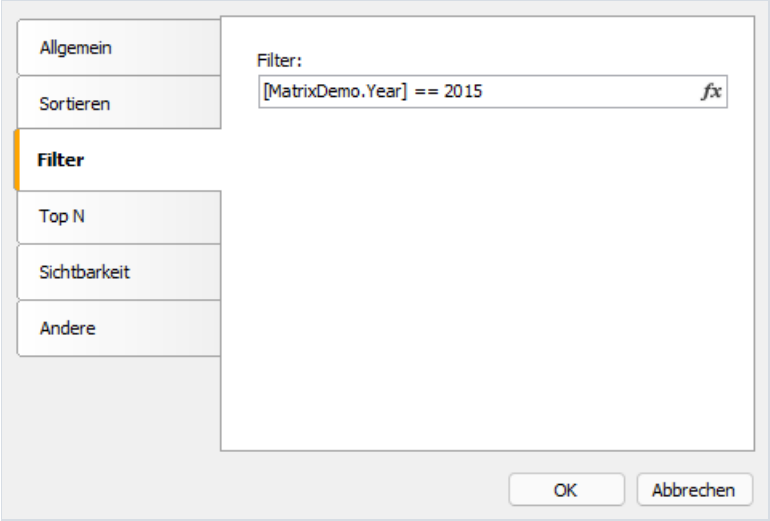
Datenfiltern

Es gibt zwei Optionen, die in der Matrix angezeigten Daten zu filtern.

Option 1: Filtern mit Hilfe der `Filter` -Matrixeigenschaft. Wählen Sie dazu das Objekt "Matrix" aus und füllen Sie im Eigenschaftenfenster die `Filter` Eigenschaft aus:

```
[MatrixDemo.Year] == 2015
```

Option 2: Filtern im Überschrifteneditor mit einem ähnlichen Ausdruck:



In beiden Fällen gibt der Filterausdruck den Typ `bool` zurück.

Der Unterschied zwischen den beiden beschriebenen Optionen liegt in der Menge der Daten, die in die Matrix einfließen. Im ersten Fall werden alle Werte ausgeworfen, außer denen, bei denen das Jahr 2015 ist. Das Ergebnis kann wie folgt aussehen:

	2015
Andrew Fuller	$1,800.00
Nancy Davolio	$1,700.00

Im zweiten Fall werden die Werte des jeweiligen Elements in der Überschrift gefiltert. Im Gegensatz zur vorherigen Option kann das Ergebnis leere Werte enthalten:

	2015
Andrew Fuller	$1,800.00
Janet Leverling	
Nancy Davolio	$1,700.00
Steven Buchanan	

TopN Gruppieren

Wenn die Anzahl der Werte in der Überschriftengruppe groß ist, wird eine übermäßige Anzahl von Berichtsseiten generiert. Mit dem TopN Gruppieren können die ersten N Werte angezeigt und die übrigen Werte minimiert angezeigt werden:

	Top 5 customers						Total
	Lonesome Pine Restaurant	Rattlesnake Canyon Grocery	Save-a-lot Markets	QUICK-Stop	Ernst Handel	⊞ Others (84)	
Beverages (12)	$2 708,00	$26 876,15	$65 498,00	$36 216,43	$12 709,30	$192 684,30	**$336 692,18**
Condiments (12)	$82 480,00	$6 075,20	$11 567,00	$9 214,94	$14 070,06	$105 665,89	**$229 073,09**
Confections (13)	$549,00	$10 947,21	$11 981,07	$18 530,09	$12 815,76	$112 615,09	**$167 438,23**
Dairy Products (10)	$815,00	$42 854,87	$21 107,10	$13 800,85	$24 496,46	$166 433,01	**$269 507,29**
Grains/Cereals (7)	$190,00	$4 831,31	$8 298,10	$5 310,90	$12 737,70	$64 376,58	**$95 744,59**
Meat/Poultry (6)	$140 098,40	$83 657,28	$27 659,18	$9 754,96	$8 325,56	$113 526,98	**$383 022,36**
Produce (5)	$16 379,20	$13 836,05	$16 387,90	$8 081,40	$12 469,67	$71 955,36	**$139 109,58**
Seafood (12)	$625,00	$884,73	$13 604,60	$9 367,74	$7 250,48	$99 591,19	**$131 323,74**
Total	**$243 844,60**	**$189 962,80**	**$176 102,95**	**$110 277,31**	**$104 874,98**	**$926 848,41**	**$1 751 911,04**

Funktionsweise

Der TopN-Modul verwendet vier Elemente, um Daten anzuzeigen:

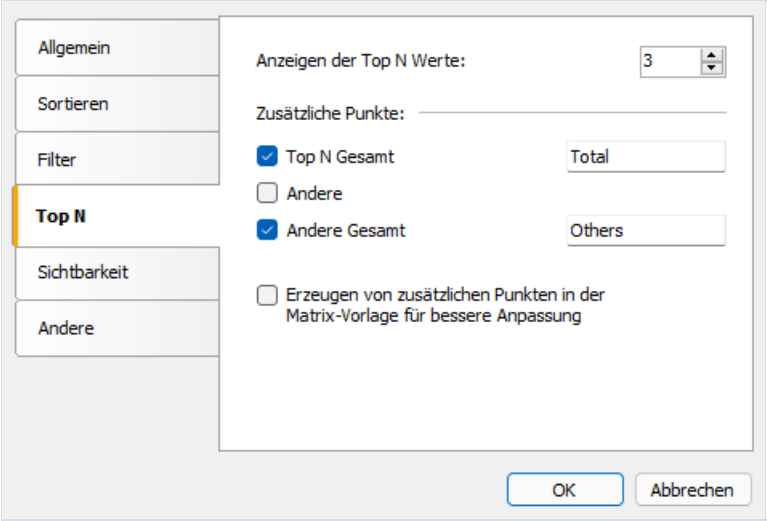
1. Die TopN-Gruppe ist eine Ausgangsgruppe mit einer großen Anzahl von Werten.
2. Der TopN-Gesamtwert ist der Wert, der die Summe der TopN-Werte anzeigt.
3. Die Gruppe "Sonstige" ist die Gruppe, die Werte anzeigt, die in TopN nicht enthalten sind.
4. Gesamtwert der Gruppe "Sonstige".

Hat die Ausgangsgruppe weniger Werte als in der TopN.Count-Eigenschaft angegeben, wird sie üblicherweise ohne TopN-Gruppieren angezeigt. Ansonsten passiert folgendes:

- in der Hauptgruppe bleiben N Werte;
- die übrigen Werte werden in die Gruppe "Sonstige" übertragen;
- die Daten werden in der Hauptgruppe und in der Gruppe "Sonstige" aggregiert;
- die resultierenden Werte werden im TopN-Gesamtwert und im Gesamtwert der Gruppe "Sonstige" angezeigt.

Konfigurieren

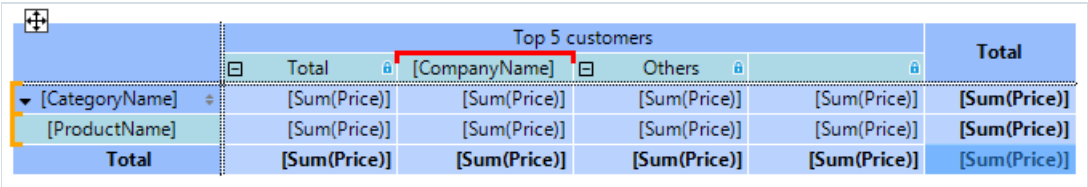
Das TopN-Konfigurieren erfolgt für die Hauptgruppe. Doppelklicken Sie dazu auf das Element oder wählen Sie aus dem Kontextmenü den Punkt "Bearbeiten ...".



Es gibt zwei Optionen mit zusätzlichen Elementen umzugehen:

- die Elemente "Top N Gesamt", "Andere", "Andere Gesamt" werden beim Aufbau der Matrix automatisch erstellt. Ihre visuelle Gestaltung wird vom Hauptelement kopiert. Sie können die Sichtbarkeit der Elemente steuern sowie die Überschriften für die Gesamtwert-Elemente festlegen. Es gibt keine anderen Optionen zum Anpassen des Erscheinungsbilds;
- die oben genannten Elemente werden der Matrixvorlage hinzugefügt. So können Sie das Erscheinungsbild vollständig anpassen sowie die Reihenfolge der Elemente ändern. Sie können Schaltflächen zum Ausblenden hinzufügen, um die Sichtbarkeit einzelner Elemente interaktiv zu steuern.

Unten sehen Sie, wie eine Matrixvorlage aussieht, wenn zusätzliche Elemente hinzugefügt werden:



In diesem Fall ist die Hauptgruppe rot hervorgehoben und die zusätzlichen Elemente sind mit einem Schloss-Symbol gekennzeichnet. Sie können sie im Fenster des Hauptelementeditors löschen, indem Sie "Zusätzliche Elemente zur Matrixvorlage hinzufügen" deaktivieren.

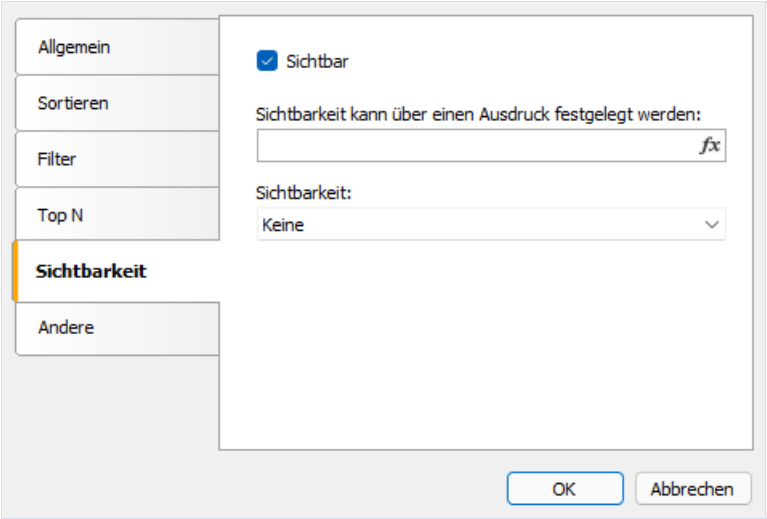
TopN, BottomN, FirstN, LastN

Der TopN-Modul verwendet die ersten N Werte der Ausgangsgruppe. Die Bedeutung der resultierenden Werte hängt davon ab, wie die Ausgangsgruppe sortiert wurde:

- die Gruppe wird nach Überschriftenwert sortiert: es werden die ersten N (aufsteigende Sortierung) oder die letzten N Werte (absteigende Sortierung) angezeigt;
- die Gruppe wird nach dem Gesamtwert sortiert: es werden die größten N (absteigende Sortierung) oder die kleinsten N Werte (aufsteigende Sortierung) angezeigt.

Sichtbarkeit eines Elements

Auf der Registerkarte "Sichtbarkeit" können Sie die Sichtbarkeit des Elements steuern:



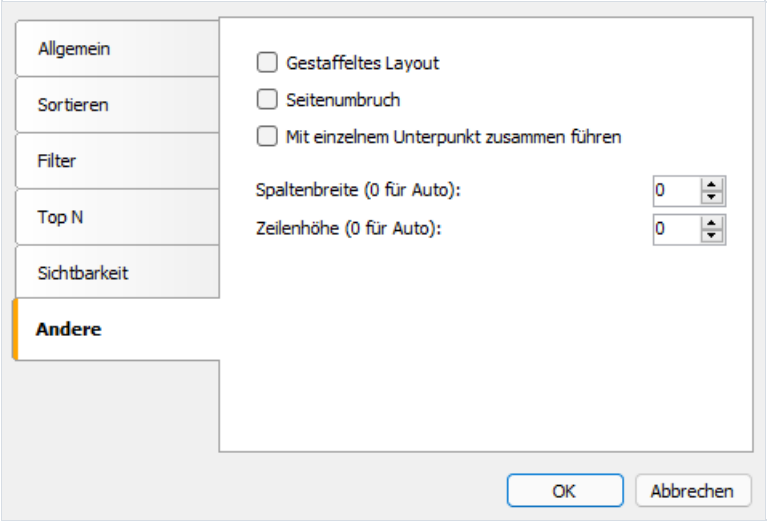
"Sichtbar" - legt die anfängliche Sichtbarkeit fest.

"Die Sichtbarkeit wird durch einen Ausdruck gesetzt" ist ein Ausdruck, der einen "bool"-Typ zurückgibt, der die Sichtbarkeit festlegt. Wenn der Ausdruck nicht leer ist, wird sein Wert anstelle der Option "Sichtbar" verwendet.

"Die Sichtbarkeit wird mit der Schaltfläche umgeschaltet" ist der Name der MatrixCollapseButton-Schaltfläche, die die Sichtbarkeit dieses Elements steuert. Siehe Abschnitt Ein-/Ausblenden von Elementen.

Weitere Eistellungen

Weitere Einstellungen sind in der Registerkarte des Überschriteneditors "Sonstiges" zusammengefasst:



Die Option "Stufenweise Anzeige" schaltet die Anzeige von verschachtelten Elementen zwischen einer blockweisen (Standard) und einer stufenweisen Anordnung um. Betrachten wir ein Beispiel für eine Matrix mit einer verschachtelten Zeilenüberschrift (Year, Month). Die blockweise Anordnung sieht so aus:

		[Name]
[Year]	[Month]	[Sum(Revenue)]
Total		[Sum(Revenue)]

Der fertige Bericht sieht wie folgt aus:

		Andrew Fuller	Janet Leverling
2011	2		
	10	$1,900.00	$3,000.00
	11	$2,000.00	$3,100.00
	12		
2012	1		
	2	$2,100.00	
	3		$3,200.00
2013	1		
	2		
	3		
2014	1		
	2		
2015	1	$1,800.00	
Total		$7,800.00	$9,300.00

Wenn für das Element "Year" die Option "Stufenweise Anzeige" aktiviert ist, ändert sich die Anordnung der Elemente wie folgt:

	[Name]
[Year]	[Sum(Revenue)]
[Month	[Sum(Revenue)]
Total	[Sum(Revenue)]

Der fertige Bericht sieht wie folgt aus:

	Andrew Fuller	Janet Leverling
2011	$3,900.00	$6,100.00
2		
10	$1,900.00	$3,000.00
11	$2,000.00	$3,100.00
12		
2012	$2,100.00	$3,200.00
1		
2	$2,100.00	
3		$3,200.00
2013		
1		
2		
3		
2014		
1		
2		
2015	$1,800.00	
1	$1,800.00	
Total	$7,800.00	$9,300.00

Wenn die Option "Stufenweise Anzeige" aktiviert ist, ändert sich die Padding.Left-Eigenschaft des verschachtelten Elements, die den Texteinzug bestimmt. Sie können diesen Wert im Eigenschaftenfenster anpassen.

Diese Option kann für jedes Element verwendet werden, das verschachtelte Elemente enthält.

Die Option "Seitenumbruch" fügt eine neue Seite ein, bevor das Element gedruckt wird. Keine neue Seite wird vor dem ersten Element eingefügt.

Die Option "Mit einem einzelnen untergeordneten Element verbinden" wird für dynamisch ausgeblendeten Überschriften verwendet (siehe Abschnitt Ein-/Ausblenden von Elementen). Mit dieser Option können Sie das Element "Gesamt" ausblenden. Unten ist der fertige Bericht mit deaktivierter Option (standartmäßig).

		Andrew Fuller
⊞ 2011	Total	$3,900.00
⊞ 2012	Total	$2,100.00
⊟ 2013	1	
	2	
	3	
	Total	
⊞ 2014	Total	
⊞ 2015	Total	$1,800.00
Total		$7,800.00

und aktivierter Option dargestellt. Beachten Sie die Anzeige des Elements "Jahr":

		Andrew Fuller
⊞ 2011		$3,900.00
⊞ 2012		$2,100.00
⊟ 2013	1	
	2	
	3	
	Total	
⊞ 2014		
⊞ 2015		$1,800.00
Total		$7,800.00

Mit den Optionen "Zeilen zusammenführen" und "Spalten zusammenführen" können Sie die Zellen in Zeilen und Spalten zusammenführen, wenn Sie ein Element drucken. In der Standardeinstellung steuert das Element selbst diese Optionen.

Eigenschaften, auf die über Kopfzellen zugegriffen werden kann

Die Matrix besitzt eine Reihe von Eigenschaften, die beim Drucken von Überschriftenzellen verwendet werden können. Der Matrixname wird verwendet, um auf die Eigenschaft zuzugreifen:

```
[Matrix1.RowNo]
```

Eigenschaft	Rückgabewert	Beschreibung
RowNo	int	Laufende Nummer des Überschriftenelements.
ItemCount	int	Anzahl der untergeordneten Elemente im aktuellen Überschriftenelement.

Diese Eigenschaften können im Feld der Überschriftenzelle "Anzeigetext" verwendet werden, z.B.:

```
[Matrix1.RowNo]. [MatrixDemo.Name]
```

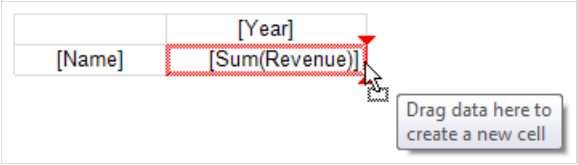
zeigt den folgenden Text an

```
1. Andrew Fuller
```

Konfigurieren des Datenbereichs

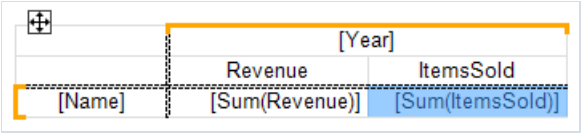
Hinzufügen eines Elements

Sie können ein Element hinzufügen, indem Sie das Datenbankfeld aus dem Datenfenster ziehen. Beim Ziehen wird angezeigt, in welchem Teil des Datenbereichs das neue Element eingefügt wird:



> Wenn ein Element zu einer leeren Matrix hinzugefügt wird, wird ihre DataSource-Eigenschaft automatisch konfiguriert.

Wenn ein Element neben dem bestehenden Element hinzugefügt wird, wird seine Überschrift ebenso hinzugefügt:



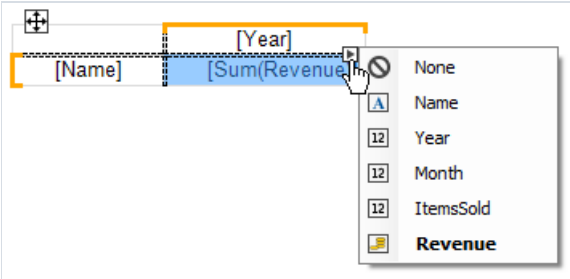
Löschen eines Elements

Sie können den Text eines Elements löschen, indem Sie die Delete Taste drücken oder das "Löschen" aus dem Elementmenü auswählen. Das Datenelement selbst kann nicht gelöscht werden, dazu muss das entsprechende Element in der Matrixüberschrift gelöscht werden. Um also im obigen Beispiel das Element "[Sum (Revenue)]" aus dem Datenbereich zu entfernen, müssen Sie das Element "Revenue" aus der Überschrift entfernen.

Bearbeiten eines Elements

Um den Text eines Elements zu bearbeiten, doppelklicken Sie darauf, um das Texteditor-Fenster aufzurufen. Sie können den Editor auch durch Drücken der Enter-Taste aufrufen.

Sie können das im Element angezeigte Datenbankfeld mit der Schnellauswahltaste (smarttag) schnell auswählen. Diese Taste wird angezeigt, wenn sich der Mauszeiger im Element befindet:



Sie können auch über das Kontextmenü des Elements die folgenden Operationen ausführen:

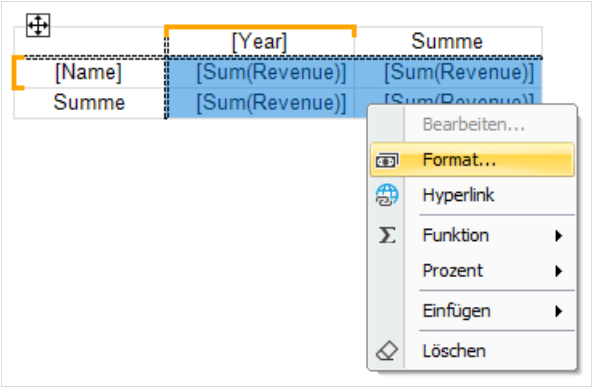
- den Element-Editor aufrufen;
- Datenformatierung konfigurieren;
- den Typ der Aggregatfunktion ändern (siehe Abschnitt Aggregatfunktionen);
- Prozentrechnung hinzufügen;
- eine Fortschrittsanzeige oder eine Sparkline in die Zelle einfügen.

Bearbeiten von mehreren Elementen

Für eine Massenbearbeitung von Zellen können Sie den gesamten Datenbereich oder mehrere Zellen auswählen. Dazu können sie die Shift-Taste oder die Maus verwenden:

- die Startzelle auswählen;
- die linke Maustaste gedrückt halten und die Maus ziehen, um eine Gruppe von Zellen auszuwählen;

Für ausgewählte Zellen können Sie das Kontextmenü aufrufen oder die Schnellwahltaste des Datenbankfeldes verwenden:



Aggregatfunktionen

Aggregatfunktionen werden in Datenzellen verwendet, um Zellwerte zu aggregieren und Gesamtwerte zu berechnen. Eine Aggregatfunktion wird wie folgt abgerufen:

```
[Funktion(Ausdruck)]
```

> Eckige Klammern werden verwendet, um Ausdrücke im Zellentext anzugeben. Sie können mehrere Aggregatfunktionen in einer Zelle gemischt mit einfachem Text verwenden.

Ein Ausdruck ist normalerweise ein Datenquellenfeld. Beispiel für die Verwendung einer Aggregatfunktion:

```
[Sum([MatrixDemo.Revenue])]
```

Nachfolgend finden Sie eine Liste der Aggregatfunktionen:

Funktion	Beschreibung
Sum	gibt die Summe von Werten zurück.
Min	gibt den Minimalwert zurück.
Max	gibt den Maximalwert zurück.
Avg	gibt den Durchschnittswert zurück.
Count	gibt die Anzahl der Werte zurück.
CountDistinct	gibt die Anzahl der verschiedenen (einmaligen) Werte zurück.
StDev	gibt die Standardabweichung einer Stichprobe zurück.
StDevP	Gibt die Standardabweichung einer Population zurück.
Var	Gibt die Varianz einer Stichprobe zurück.
VarP	Gibt die Varianz einer Population zurück.
First	Gibt den ersten Wert zurück.
Last	Gibt den letzten Wert zurück.
ValuesList	Gibt die Liste aller Werte in der Zelle zurück. Diese Aggregation wird in Verbindung mit den Objekten "Diagramm" und "Sparkline" verwendet.
_Name	Benutzerdefinierte Aggregatfunktion, die im Berichtscode festgelegt wird.

Eine benutzerdefinierte Funktion hat einen Namen, der mit einem Unterstrich beginnt. Ihr Code soll in die ReportScript Hauptklasse eingefügt werden. Die Funktion ist wie folgt definiert:

```
object _FuncName(List<dynamic> l)
```

Beispiel einer benutzerdefinierten Funktion _Sum:

```
public class ReportScript
{
    public object _Sum(List<dynamic> l)
    {
        dynamic value = 0;
        foreach (dynamic v in l)
            value += v;
        return value;
    }
}
```

Sonderfunktionen

In einer Matrixdatenzelle können spezielle Funktionen verwendet werden. Sie ermöglichen es Ihnen, den Wert einer anderen Zelle in derselben Zeile oder Spalte zu ermitteln.

GrandColumnTotal

Gibt den Spaltengesamtwert zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
GrandColumnTotal(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / GrandColumnTotal()
```

GrandRowTotal

Gibt den Zeilengesamtwert zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
GrandRowTotal(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / GrandRowTotal()
```

GrandTotal

Gibt den Gesamtwert zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
GrandTotal(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / GrandTotal()
```

ColumnTotal

Gibt den maximalen Spaltengesamtwert für die aktuelle Gruppe zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
ColumnTotal(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / ColumnTotal()
```

RowTotal

Gibt den Zeilengesamtwert für die aktuelle Gruppe zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
RowTotal(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / RowTotal()
```

ColumnMaxValue

Gibt den maximalen Spaltengesamtwert für die aktuelle Gruppe zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
ColumnMaxValue(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / ColumnMaxValue()
```

ColumnMinValue

Gibt den minimalen Spaltengesamtwert für die aktuelle Gruppe zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
ColumnMinValue(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / ColumnMinValue()
```

RowMaxValue

Gibt den maximalen Zeilengesamtwert für die aktuelle Gruppe zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
RowMaxValue(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / RowMaxValue()
```

RowMinValue

Gibt den minimalen Zeilengesamtwert für die aktuelle Gruppe zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
RowMinValue(Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / RowMinValue()
```

FirstColumn

Gibt den Wert der ersten Zelle in der Spalte zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Parameter	Beschreibung
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der ersten Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den ersten Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der ersten Zelle in der Gruppe zurück.

Beispiele:

```
FirstColumn(Sum([MatrixDemo.Revenue]))
FirstColumn(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / FirstColumn()
```

FirstRow

Gibt den Wert der ersten Zelle in der Zeile zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der ersten Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den ersten Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der ersten Zelle in der Gruppe zurück.

Beispiele:

```
FirstRow(Sum([MatrixDemo.Revenue]))
FirstRow(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / FirstRow()
```

LastColumn

Gibt den Wert der letzten Zelle in der Spalte zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der letzten Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den letzten Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der letzten Zelle in der Gruppe zurück.

Beispiele:

```
LastColumn(Sum([MatrixDemo.Revenue]))
LastColumn(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / LastColumn()
```

LastRow

Gibt den Wert der letzten Zelle in der Zeile zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der letzten Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den letzten Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der letzten Zelle in der Gruppe zurück.

Beispiele:

```
LastRow(Sum([MatrixDemo.Revenue]))
LastRow(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / LastRow()
```

PreviousColumn

Gibt den Wert der vorherigen Zelle in der Spalte zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der vorherigen Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den vorherigen Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der vorherigen Zelle in der Gruppe zurück.

Beispiele:

```
PreviousColumn(Sum([MatrixDemo.Revenue]))
PreviousColumn(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / PreviousColumn()
```

PreviousRow

Gibt den Wert der vorherigen Zelle in der Zeile zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der vorherigen Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den vorherigen Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der vorherigen Zelle in der Gruppe zurück.

Beispiele:

```
PreviousRow(Sum([MatrixDemo.Revenue]))
PreviousRow(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / PreviousRow()
```

NextColumn

Gibt den Wert der nächsten Zelle in der Spalte zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der nächsten Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den nächsten Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der nächsten Zelle in der Gruppe zurück.

Beispiele:

```
NextColumn(Sum([MatrixDemo.Revenue]))
NextColumn(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / NextColumn()
```

NextRow

Gibt den Wert der nächsten Zelle in der Zeile zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwertes: gibt den Wert der nächsten Zelle in der Gruppe zurück, wenn der Parameter `useThisGroup` `true` ist, andernfalls gibt den nächsten Gesamtwert zurück;
- Gruppenzelle: gibt den Wert der nächsten Zelle in der Gruppe zurück.

Beispiele:

```
NextRow(Sum([MatrixDemo.Revenue]))
NextRow(Sum([MatrixDemo.Revenue]), true)
Sum([MatrixDemo.Revenue]) / NextRow()
```

SpecificColumn

Gibt den Wert der Zelle mit dem angegebenen Index in der Spalte zurück.

Parameter	Beschreibung
`index`	Indexwert
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwerts: gibt den Wert der Zelle in der Gruppe zurück, zu der der Gesamtwert gehört;
- gibt andernfalls den Wert der Zelle in der Spalte der aktuellen Gruppe zurück.

Beispiele:

```
SpecificColumn("Andrew Fuller", Sum([MatrixDemo.Revenue]))
SpecificColumn(2011, Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / SpecificColumn(2011)
```

SpecificRow

Gibt den Wert der Zelle mit dem angegebenen Index in der Zeile zurück.

Parameter	Beschreibung
`index`	Indexwert
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwerts: gibt den Wert der Zelle in der Gruppe zurück, zu der der Gesamtwert gehört;
- gibt andernfalls den Wert der Zelle in der Zeile der aktuellen Gruppe zurück.

Beispiele:

```
SpecificRow("Andrew Fuller", Sum([MatrixDemo.Revenue]))
SpecificRow(2011, Sum([MatrixDemo.Revenue]))
Sum([MatrixDemo.Revenue]) / SpecificRow(2011)
```

PercentOfColumnTotal

Gibt den Wert der aktuellen Zelle geteilt durch den Spaltengesamtwert zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

> Aufruf der Funktion
>
> PercentOfColumnTotal(Sum([MatrixDemo.Revenue]))
>
> entspricht dem folgenden Code:
>
> Sum([MatrixDemo.Revenue]) / ColumnTotal()

Beispiele:

```
PercentOfColumnTotal(Sum([MatrixDemo.Revenue]))
[Sum([MatrixDemo.Revenue])] [PercentOfColumnTotal()]
```

PercentOfRowTotal

Gibt den Wert der aktuellen Zelle geteilt durch den Zeilengesamtwert zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
PercentOfRowTotal(Sum([MatrixDemo.Revenue]))
[Sum([MatrixDemo.Revenue])]  [PercentOfRowTotal()]
```

PercentOfGrandTotal

Gibt den Wert der aktuellen Zelle geteilt durch den Wert des Gesamtergebnisses zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.

Beispiele:

```
PercentOfGrandTotal(Sum([MatrixDemo.Revenue]))
[Sum([MatrixDemo.Revenue])]  [PercentOfGrandTotal()]
```

PercentOfPreviousColumn

Gibt den Wert der aktuellen Zelle geteilt durch den Wert der vorherigen Zelle in der Spalte zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwerts: der Wert der vorherigen Zelle in der Gruppe wird verwendet, wenn der Parameter `useThisGroup` `true` ist, andernfalls wird der vorherige Gesamtwert verwendet;
- Gruppenzelle: der Wert der vorherigen Zelle in der Gruppe wird verwendet.

Beispiele:

```
PercentOfPreviousColumn(Sum([MatrixDemo.Revenue]))
PercentOfPreviousColumn(Sum([MatrixDemo.Revenue]), true)
[Sum([MatrixDemo.Revenue])]  [PercentOfPreviousColumn()]
```

PercentOfPreviousRow

Gibt den Wert der aktuellen Zelle geteilt durch den Wert der vorherigen Zelle in der Zeile zurück.

Parameter	Beschreibung
`aggregate`	(optional) Aggregatfunktion. Der Parameter kann weggelassen werden, wenn auf die Aggregation bereits in einer Zelle zugegriffen wurde.
`useInteractiveSort=false`	(optional) Berücksichtigung der Ergebnisse des interaktiven Sortierens, bei der die Reihenfolge der Elemente geändert werden kann.
`useThisGroup=true`	(optional) Verwendung der Gruppe auf der gleichen Ebene, um nach einem Wert zu suchen.

Der Rückgabewert hängt davon ab, wo die Funktion verwendet wird:

- Zelle des Gesamtwerts: der Wert der vorherigen Zelle in der Gruppe wird verwendet, wenn der Parameter `useThisGroup` `true` ist, andernfalls wird der vorherige Gesamtwert verwendet;
- Gruppenzelle: der Wert der vorherigen Zelle in der Gruppe wird verwendet.

Beispiele:

```
PercentOfPreviousRow(Sum([MatrixDemo.Revenue]))
PercentOfPreviousRow(Sum([MatrixDemo.Revenue]), true)
[Sum([MatrixDemo.Revenue])]  [PercentOfPreviousRow()]
```

Eigenschaften, auf die über Datenzellen zugegriffen werden kann

Die Matrix verfügt über eine Reihe von Eigenschaften, die Sie beim Drucken von Datenzellen verwenden können. Der Matrixname wird verwendet, um auf die Eigenschaft zuzugreifen:

```
[Matrix1.RowIndex]
```

Eigenschaft	Rückgabewert	Beschreibung
ColumnIndex	int	Index der aktuellen Spalte.
RowIndex	int	Index der aktuellen Zeile.
ColumnValues	object[]	Array von Werten aus der Spaltenüberschrift.
RowValues	object[]	Array von Werten aus der Zeilenüberschrift.

Diese Eigenschaften können nützlich sein, um die Zellen je nach der Bedingung farblich hervorzuheben.

Sie können auch von einer Zelle aus auf die Datenquellenfelder zugreifen. Dies ist normalerweise erforderlich, damit die bedingte Hervorhebung funktioniert (siehe Abschnitt Bedingte Hervorhebung). Für eine Datenzelle können Sie beispielsweise die folgende Bedingung für die Hervorhebung von Werten in Bezug auf 2012 festlegen:

```
[MatrixDemo.Year] == 2012
```

Der Wert des Datenbankfeldes, der zum Drucken der Zelle verwendet wurde, wird aus der Matrixüberschrift übernommen. Da es 2 (Zeilen- und Spalten-) Überschriften gibt, muss angegeben werden, welche Überschriftenwerte Vorrang haben sollen. Dafür ist die `DataRowPriority` Matrix-Eigenschaft zuständig. Standardmäßig ist die Eigenschaft auf `Rows` gesetzt.

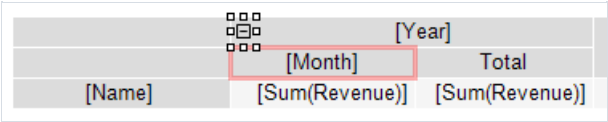
Interaktive Funktionen

Dieser Abschnitt befasst sich mit den interaktiven Funktionen des Objekts "Erweiterte Matrix":

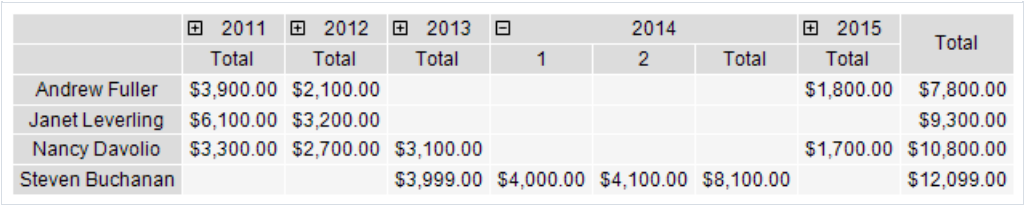
- Ein-/Ausblenden von Überschriftenelementen;
- Interaktives Sortieren.

Aus-/Einblenden von Elementen

Sie können die Sichtbarkeit einzelner Überschriftelemente mit der speziellen Schaltfläche wie `MatrixCollapseButton` interaktiv steuern. Die Schaltfläche wird in das Überschriftelement eingefügt und steuert die Sichtbarkeit der anderen Elemente. Die folgende Abbildung zeigt die Schaltfläche und das von ihr gesteuerte Element (durch einen roten Rahmen hervorgehoben, wenn die Schaltfläche ausgewählt ist)):



Wenn Sie auf die Schaltfläche im Vorschaufenster klicken, werden die zugehörigen Elemente ein- oder ausgeblendet. Dabei wird der Bericht neu erstellt:



	2011	2012	2013	2014			2015	Total
	Total	Total	Total	1	2	Total	Total	
Andrew Fuller	$3,900.00	$2,100.00					$1,800.00	$7,800.00
Janet Leverling	$6,100.00	$3,200.00						$9,300.00
Nancy Davolio	$3,300.00	$2,700.00	$3,100.00				$1,700.00	$10,800.00
Steven Buchanan			$3,999.00	$4,000.00	$4,100.00	$8,100.00		$12,099.00

Hinzufügen einer Schaltfläche

Über das Kontextmenü können Sie einem Überschriftelement eine Schaltfläche hinzufügen. Wählen Sie das Element aus, klicken Sie mit der rechten Maustaste darauf und wählen Sie "Ausblenden-Schaltfläche". Die Schaltfläche wird auf der linken Seite des Elements hinzugefügt.

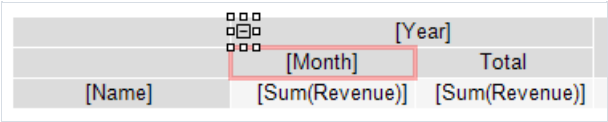
> Wenn eine Schaltfläche hinzugefügt wird, ändert sich die Padding.Left-Eigenschaft des Elements, sodass sich der Text nicht mit der Schaltfläche überschneidet.

Konfigurieren einer Schaltfläche

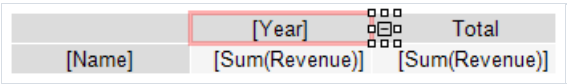
Wenn eine Schaltfläche hinzugefügt wird, konfiguriert FastReport automatisch die Verknüpfung zwischen der Schaltfläche und dem Element, das sie steuert. In einigen Fällen kann es erforderlich sein, die Verknüpfung manuell zu konfigurieren. Öffnen Sie dazu den Editor des Elements, das von der Schaltfläche abhängig sein soll, und geben Sie den Namen der Schaltfläche auf der Registerkarte "Sichtbarkeit/Über die Schaltfläche gesteuerte Sichtbarkeit" an.

> Die Schaltfläche kann mehrere Elemente gleichzeitig steuern.

Die Schaltfläche kann über dem gesteuerten Element:



oder auf derselben Ebene wie es positioniert werden:



Der Ausgangszustand des gesteuerten Elements, seine Sichtbarkeit, wird im Element-Editor auf der Registerkarte

"Sichtbarkeit/Sichtbar" eingestellt.

Entfernen einer Schaltfläche

Es gibt zwei Möglichkeiten, eine Schaltfläche zu entfernen:

- wählen Sie die Schaltfläche aus und drücken Sie die Delete Taste;
- deaktivieren Sie im Kontextmenü des Elements die Option "Ausblenden-Schaltfläche".

Verschieben einer Schaltfläche

Standardmäßig besitzt eine Schaltfläche die Eigenschaft Dock = Left. Dies bedeutet, dass sie auf der linken Seite des Elements verankert wird. Um die Schaltfläche an eine neue Stelle zu verschieben, setzen Sie im Eigenschaftenfenster die Eigenschaft Dock = None.

> Sie können auch die Anchor-Eigenschaft der Schaltfläche verwenden, um sie an einer bestimmten Stelle im Element zu verankern.

Einstellen des Erscheinungsbildes einer Schaltfläche

Über die Symbolleiste "Rahmen" können Sie das Schaltflächensymbol anpassen: Rahmenfarbe und -stil, Hintergrundfarbe. Im Eigenschaftenfenster können Sie außerdem die folgenden Schaltflächeneigenschaften einstellen:

Eigenschaft	Standardwert	Beschreibung
Cursor	Hand	Ansicht des Mauszeigers.
Exclusive	False	Wenn True, kann nur ein Element ausgeblendet werden.
Exportable	False	Wenn True, wird die Schaltfläche beim Exportieren des Berichts angezeigt.
Printable	False	Wenn True, wird die Schaltfläche beim Drucken des Berichts angezeigt.
ShowCollapseExpandMenu	False	Legt fest, ob das Menü mit "Alle aus-/einblenden" angezeigt werden soll, wenn mit der rechten Maustaste auf diese Schaltfläche geklickt wird.
Symbol	PlusMinus	In der Schaltfläche angezeigtes Symbol.
SymbolSize	5	Größe des Schaltflächensymbols.

Interaktives Sortieren

Mit der Schaltfläche `MatrixSortButton` können Zeilen und Spalten einer Matrix interaktiv sortiert werden. Die Schaltfläche wird in das untergeordnete Überschriftenelement eingefügt:

	[Year]	Total
[Name]	[Sum(Revenue)]	[Sum(Revenue)]

Wenn Sie im Vorschaufenster auf die Schaltfläche klicken, wird die gegenüberliegende Überschrift sortiert. Das folgende Beispiel sortiert die Zeilen nach dem Wert in der ausgewählten Spalte:

	2011	2012	2013	2014	2015	Total
Andrew Fuller	$3,900.00	$2,100.00			$1,800.00	$7,800.00
Janet Leverling	$6,100.00	$3,200.00				$9,300.00
Nancy Davolio	$3,300.00	$2,700.00	$3,100.00		$1,700.00	$10,800.00
Steven Buchanan			$3,999.00	$8,100.00		$12,099.00

Mit jedem Tastendruck wird der Sortiermodus umgeschaltet: aufsteigend/absteigend/keine Sortierung.

Hinzufügen einer Schaltfläche

Über das Kontextmenü können Sie einem Überschriftenelement eine Schaltfläche hinzufügen. Wählen Sie das Element aus, klicken Sie mit der rechten Maustaste darauf und wählen Sie "Sortierschaltfläche". Die Schaltfläche wird auf der rechten Seite des Elements hinzugefügt.

> Wenn eine Schaltfläche hinzugefügt wird, ändert sich die Padding.Right-Eigenschaft des Elements, sodass sich der Text nicht mit der Schaltfläche überschneidet.

Einstellung der Schaltfläche

Der Sortiermodus der Überschrift wird in ihrem Editor auf der Registerkarte Sortieren/Interaktives Sortieren nach Gesamtwert eingestellt. Mögliche Werte sind:

- "Nein" –diese Überschrift wird nicht sortiert.
- "Automatisch" ist ein Standard-Modus. Die Sortierung erfolgt nach dem Wert des ersten Gesamtwertes (Aggregation).
- (Aggregations-) Gesamtwertname: Wenn die Überschrift mehrere Ausgabewerte enthält, können Sie einen davon zum Sortieren auswählen. Im folgenden Beispiel müssen Sie den Aggregationsnamen `Sum([MatrixDemo.ItemsSold])` auswählen, um die Zeilenüberschrift nach dem ItemsSold-Wert zu sortieren:

		Total
Nancy Davolio	Revenue	$10,800.00
	ItemsSold	12
Steven Buchanan	Revenue	$12,099.00
	ItemsSold	11
Janet Leverling	Revenue	$9,300.00
	ItemsSold	9
Andrew Fuller	Revenue	$7,800.00
	ItemsSold	8

Löschen einer Schaltfläche

Es gibt zwei Optionen, eine Schaltfläche zu löschen:

- Wählen Sie die Schaltfläche aus und drücken Sie die Delete-Taste;
- Deaktivieren Sie im Kontextmenü des Elements die Option "Sortierschaltfläche".

Verschieben einer Schaltfläche

Standardmäßig hat eine Schaltfläche die Eigenschaft Dock = Right. Dies bedeutet, dass sie auf der rechten Seite des Elements verankert wird. Um die Schaltfläche an eine neue Stelle zu verschieben, setzen Sie im Eigenschaftenfenster die Eigenschaft Dock = None.

> Sie können auch die Anchor-Eigenschaft der Schaltfläche verwenden, um sie an einer bestimmten Stelle im Element zu verankern.

Einstellen des Erscheinungsbildes einer Schaltfläche

Über die Symbolleiste "Rahmen" können Sie das Schaltflächensymbol anpassen: Rahmenfarbe und -stil, Hintergrundfarbe. Im Eigenschaftenfenster können Sie außerdem die folgenden Schaltflächeneigenschaften einstellen:

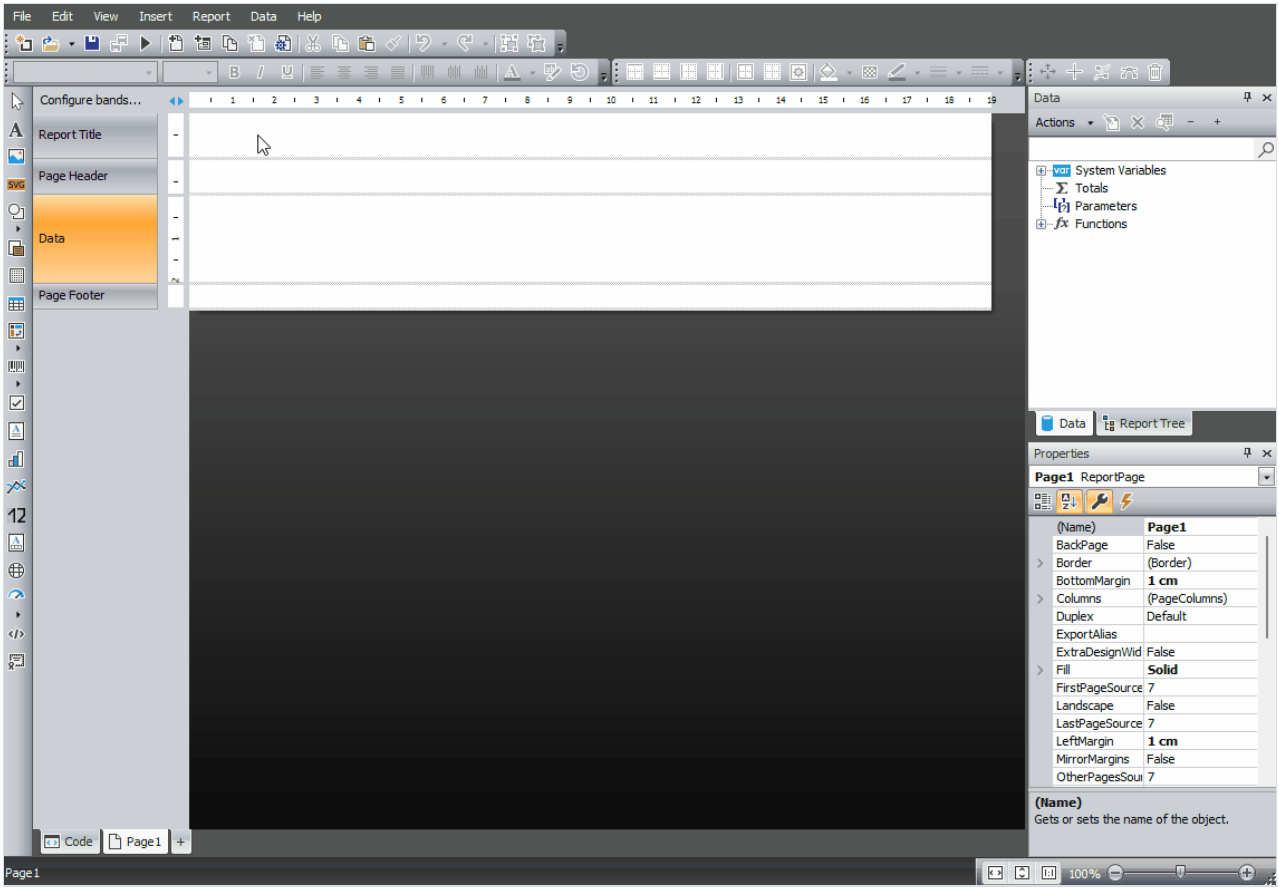
Eigenschaft	Standardwert	Beschreibung
AllowInactiveSort	True	Legt fest, ob die Schaltfläche inaktiv sein kann (Modus "Keine Sortierung").
Cursor	Hand	Ansicht des Mauszeigers.
Exportable	False	Wenn True, wird die Schaltfläche beim Exportieren des Berichts angezeigt.
InactiveSortColor	Gray	Farbe der inaktiven Schaltfläche.
Printable	False	Wenn True, wird die Schaltfläche beim Drucken des Berichts angezeigt.
Symbol	Arrow	Schaltflächensymbol.
SymbolSize	7	Größe des Schaltflächensymbols.

Anwendungsbeispiele

Im Folgenden finden Sie Anwendungsbeispiele des Objektes "Erweiterte Matrix" als animierte GIF-Dateien.

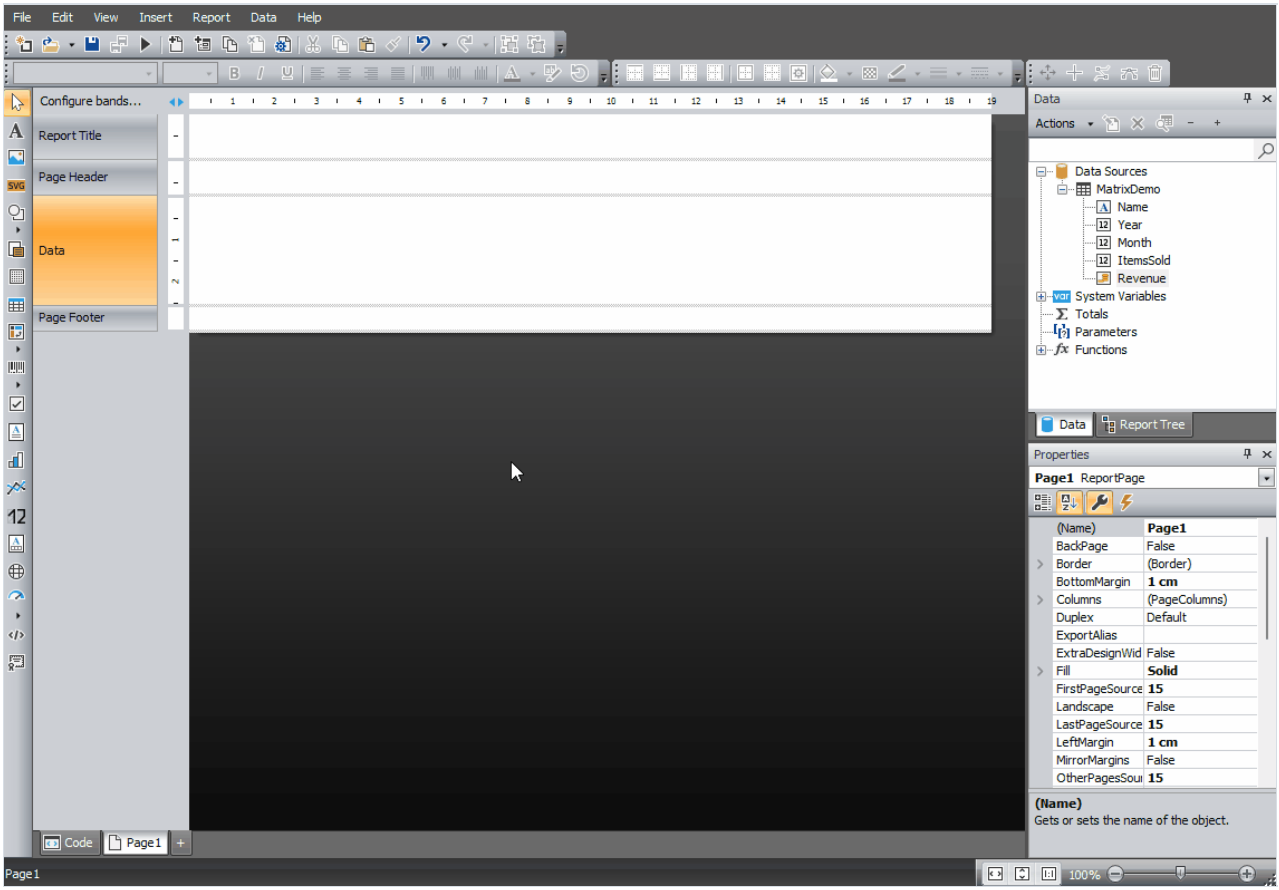
Beispiel 1. Einfache Matrix

Erstellung einer einfachen Matrix, Auswahl eines Stils, Hinzufügen von Summen:



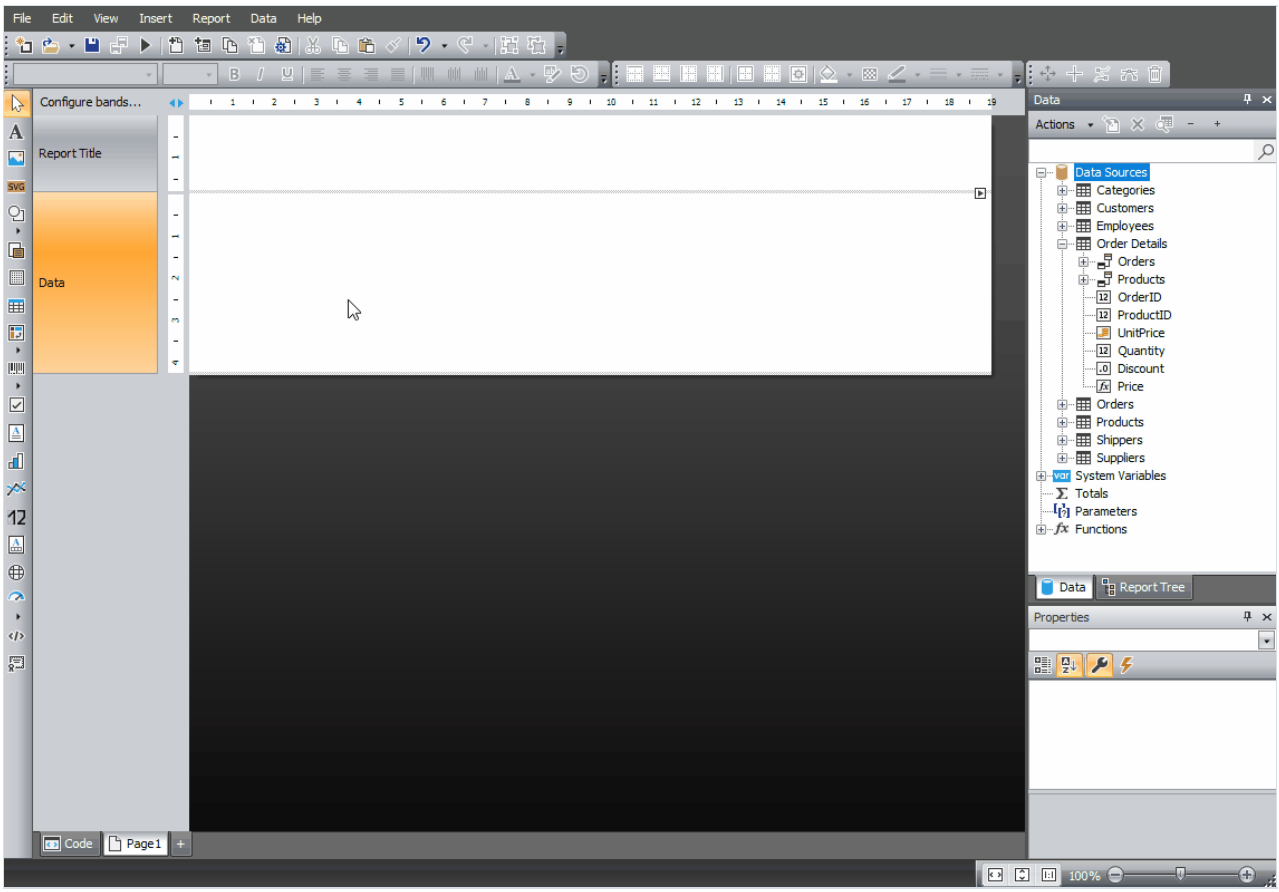
Beispiel 2. Matrix mit zusammengesetzten Kopfzeilen

Erstellen einer Matrix mit zusammengesetzten Kopfzeilen, Hinzufügen von Summen, Verwendung der DisplayText-Eigenschaft, stufenweise Anzeige der Kopfzeile, Ein-/Ausblenden-Schaltflächen:



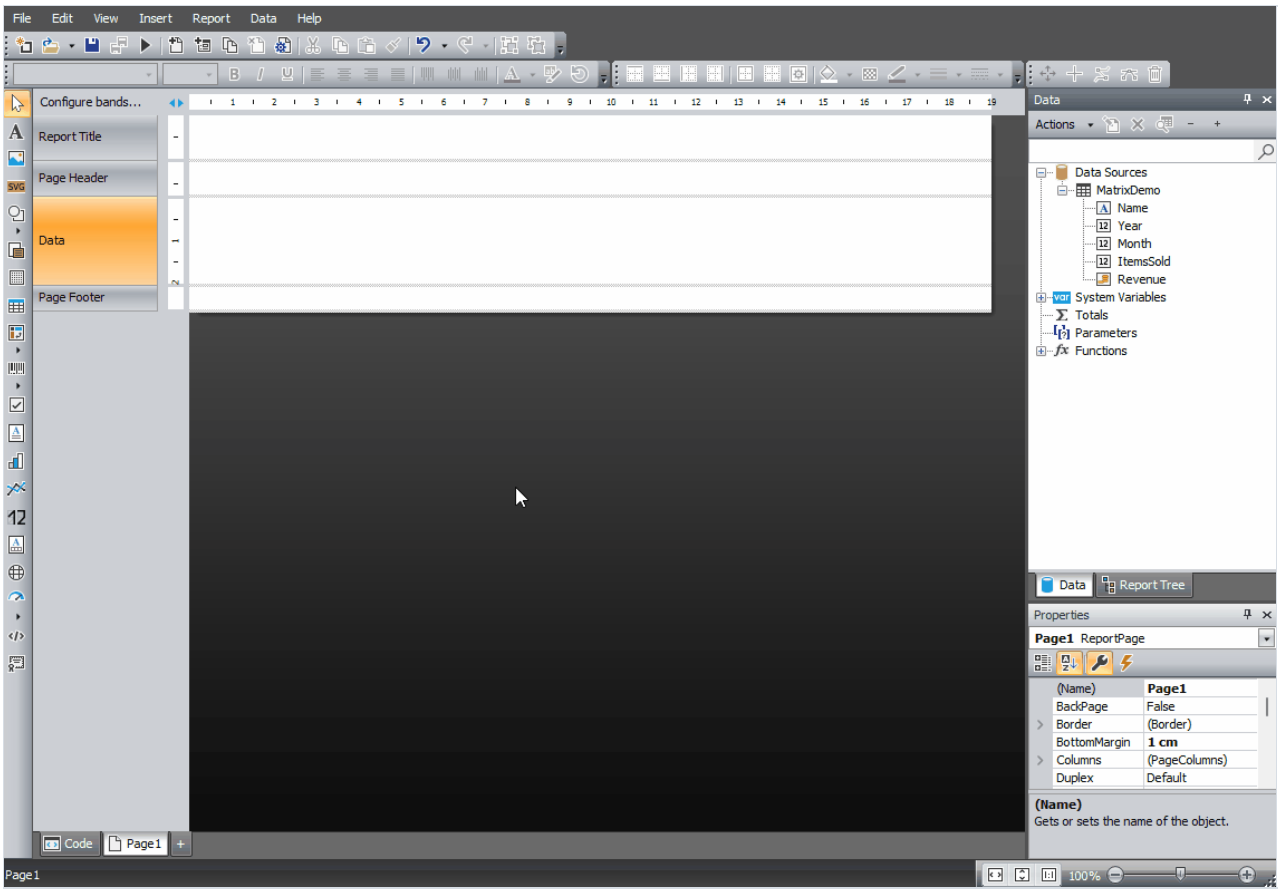
Beispiel 3. TopN-Gruppieren

Erstellen und Konfigurieren einer TopN-Gruppe (es gibt zwei Varianten: eine einfache und eine anpassbare), Einblenden von Elementen, Verwendung der ItemCount-Matrixeigenschaft:



Beispiel 4. Interaktive Sortierung

Hinzufügen von interaktiven Sortierschaltflächen, Hinzufügen eines zweiten Datenbankfeldes zum Datenbereich (ItemsSold), Einrichten einer interaktiven Sortierung für dieses Feld.



Interaktive Berichte

Ein von FastReport erstellter Bericht kann auf interaktive Weise gestaltet werden. Dies bedeutet, dass dieser aus Reaktionen der Nutzer/innen in einem Vorschaufenster reagiert. Die folgenden Interaktionen können genutzt werden:

· Durch Klicken des Bericht Objektes wird eine Operation ausgeführt. Sie können beispielsweise detaillierte Berichte laufen lassen und diese in separaten Fenstern anzeigen;

· Das Vorschaufenster kann die Gliederung des Berichts anzeigen. Dieses kann zur Navigation des Berichts genutzt werden.

Hyperlink

Fast alle Berichtsobjekte weisen die Eigenschaft "Hyperlink" auf. Die Nutzung dieser Eigenschaft erlaubt es festzulegen, welche Reaktion das Objekt bei Klicken der Maus in dem Vorschaufenster zeigt.

Durch Klicken eines solchen Objektes wird eine der folgenden Aktionen eintreten:

· Navigation zur URL-Adresse;

· Versand einer E-Mail;

· Ausführung eines beliebigen Systemkommandos;

· Navigation zur Berichtseite mit der angegebenen Zahl;

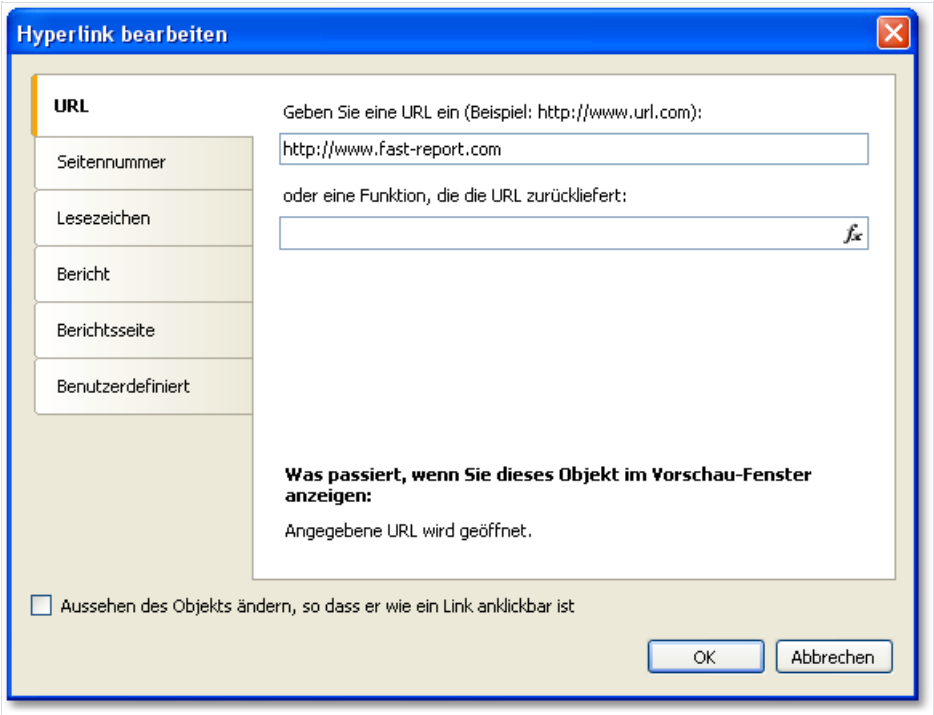
· Navigation zum Lesezeichen, das in einem anderen Bericht Objekt definiert ist;

· Durchführung eines detaillierten Berichts in einem separaten Fenster;

· benutzerdefinierte Aktion, die in einem Skript festgelegt ist.

Konfiguration eines Hyperlink

Zur Konfiguration eines Hyperlink wird das Objekt ausgewählt, dass interaktiv werden soll und mit der rechten Maustaste angeklickt. Im Kontextmenü wird das Item "Hyperlink" ausgewählt. Es öffnet sich ein Fenster mit dem Hyperlink Editor:



hier kann die Art des Hyperlinks, durch Auswahl einer der Registerkarten auf der linken Seite des Fensters, bestimmt werden. Nach dieser Auswahl kann man das Kontrollkästchen für "Aussehen des Objekts ändern so dass er wie ein Link anklickbar ist" am unteren Rand des Fensters aktivieren. Das Aussehen der Matrix ändert sich wie folgt:

· Die Textfarbe wird blau und unterstrichen;

· Der Zeiger ändert sich von einem Pfeil in eine Handform.

In einigen Fällen müssen Hyperlinks in Vorschaufenstern angezeigt werden ohne das diese gedruckt werden müssen. Durch Deaktivierung der Objekteigenschaft "Printable" kann dies recht einfach geändert werden. Dies kann im Fenster "Eigenschaften" geschehen.

Verknüpfung mit einer URL

Mit dieser Art Verknüpfung lässt sich:

· zu einer angegebenen Internet-Adresse navigieren;

· einige Systemkommandos, wie "mailto:" zum Versand von E-Mails, durchführen.

> Beim Klicken auf eine derartige Verknüpfung wird die System.Diagnostics.Process.Start-Methode, mit dem Wert der Verknüpfung als Parameter, durchgeführt.

Der Wert einer Verknüpfung kann mittels zweier Methoden angezeigt werden:

· Der Wert wird direkt angezeigt, beispielweise, "http://www.fast-report.com";

· Ein Ausdruck wird angezeigt, der den Wert der Verknüpfung zurück liefert. Dieser Wert wird während der Durchführung des Berichts berechnet.

Verknüpfung mit einer Seitenzahl

Diese Art der Verknüpfung hilft bei die Navigation der Seiten des erstellten Berichtes zu organisieren. Am häufigsten wird eine Navigation auf die erste Seite genutzt. Hierzu wird die Seitenzahl (in diesem Beispiel 1) als Wert der Verknüpfung angegeben.

Die Seitenzahl kann mittels zweier Methoden angezeigt werden:

· Der Wert wird direkt angezeigt, beispielsweise, 1;

· Ein Ausdruck wird angezeigt, der die Seitenzahl zurück liefert. Dieser Wert wird bei der Durchführung des Berichts berechnet.

Verknüpfung mit einem Lesezeichen

Die Nutzung dieses Links ermöglicht die Navigation zu einem Lesezeichen, das in einem anderen Bericht definiert ist.

Für Benutzer mit HTML-Kenntnissen, reicht die Information, dass Lesezeichen als Anker dienen. Ein Lesezeichen hat einen Namen und eine festgelegte Position in einem erstellten Bericht (Seitenzahl und Position auf der Seite). Man navigiert zur angegebenen Position mittels der Bewegung zu dem Namen eines Lesezeichens.

Um diese Art der Verknüpfung nutzen zu können, muss zuerst das Lesezeichen festgelegt werden. Hierzu wird das Objekt, zu dem man sich bei Klicken des Links bewegen will, ausgewählt. Fast alle Berichtsobjekte weisen die Eigenschaft "Lesezeichen" auf. Diese Eigenschaft kann mit Hilfe des Fensters "Eigenschaften" geändert werden.

Das "Lesezeichen" enthält einen Ausdruck, der folgendermaßen genutzt werden kann:

· Anzeige des Namen des Lesezeichen als Zeichenfolge:

```
"MyBookmark"
```

· Ein Ausdruck wird angezeigt, der den Namen des Lesezeichens zurück liefert. Beispielsweise kann die Datenspalte als Ausdruck genutzt werden. Der Wert des Ausdrucks wird bei der Durchführung des Berichts berechnet.

Nach der Definition des Lesezeichens kann der Name im Konfigurationsfenster des Lesezeichens angezeigt werden. Dies kann auf zwei Arten erfolgen:

· Direkte Anzeige des Namens des Lesezeichens;

· Anzeige eines Ausdrucks, der den Namen des Lesezeichens zurück liefert. Hierbei kann es sich z.B. um eine Datenspalte handeln. Der Wert des Ausdrucks wird bei der Durchführung des Berichts berechnet.

Verknüpfung mit einem detaillierten Bericht

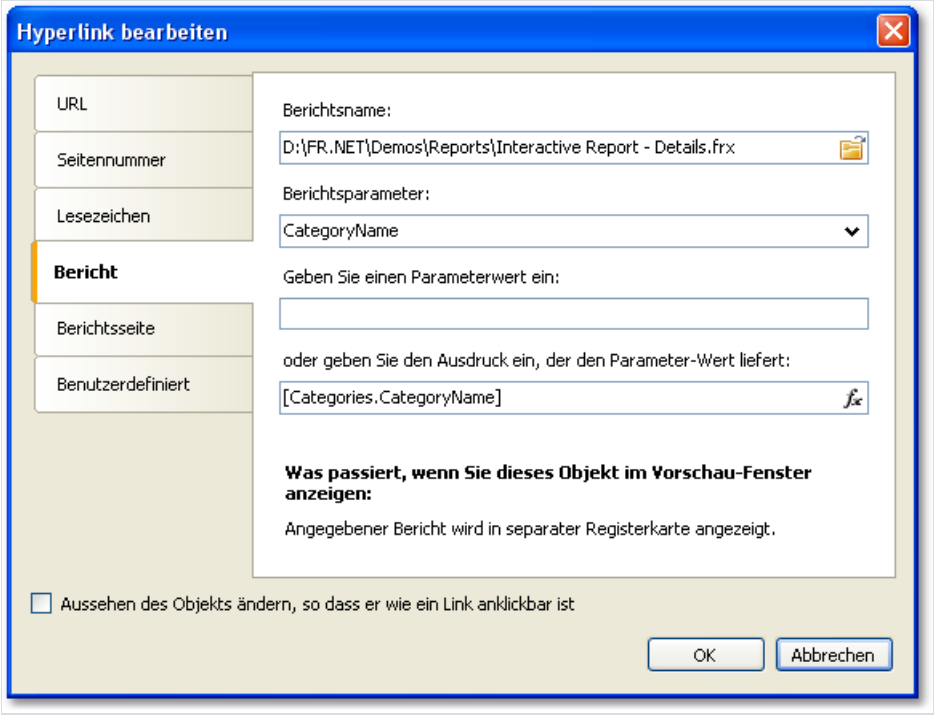
Die Nutzung dieser Verknüpfungsart erlaubt die Durchführung eines anderen Berichts und dessen Anzeige in einem separaten Vorschaufenster.

Für diese Art von Verknüpfung müssen die folgenden Parameter festgelegt werden:

· Detaillierter Berichtsname;

· Name des Bericht Parameters, der den Wert des Hyperlinks annimmt;

· Wert des Hyperlinks.



Wird die Verknüpfung angeklickt, geschieht das Folgende:

· Der anzeigte Bericht wird geladen;

· Die Berichtsparameter werden auf den Wert des Hyperlinks festgesetzt;

· Der Bericht wird in einem separaten Vorschaufenster erstellt und ausgeführt.

Der Wert des Berichtsparameters kann durch Nutzung zweier Methoden angezeigt werden:

· Direkte Anzeige des Wertes;

· Ein Ausdruck wird angezeigt, der den Wert zurück liefert. Dieser Ausdruck wird berechnet wenn der Bericht durchgeführt wird.

Verknüpfung mit einer detaillierten Seite

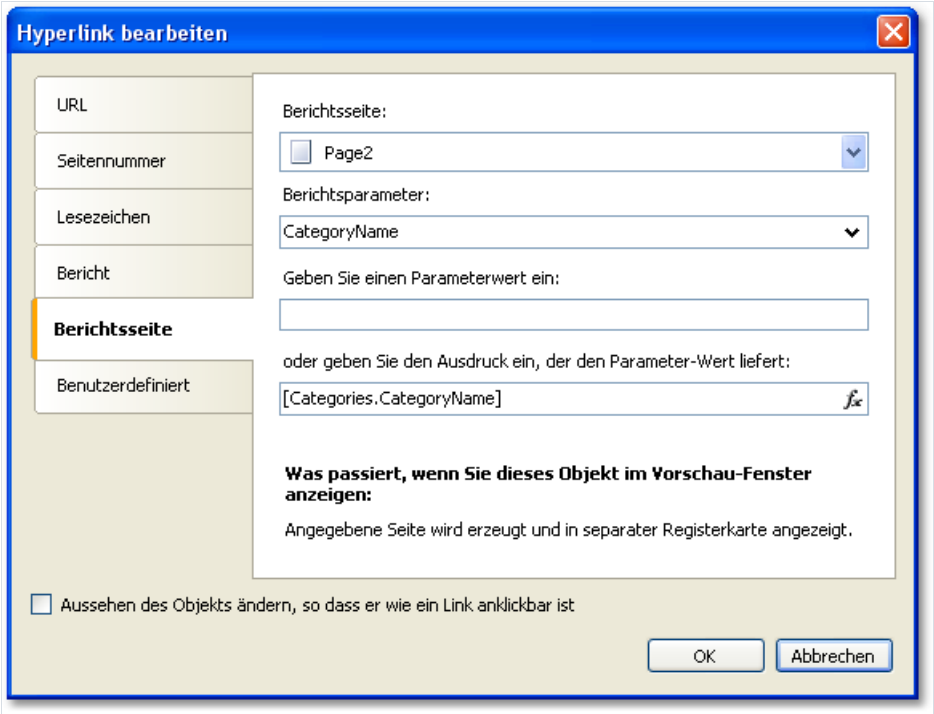
Diese Verknüpfungsart funktioniert auf die gleiche Art, mit der Ausnahme, dass eine andere Seite im aktuellen Bericht als Detailbericht genutzt wird. Hierzu muss der Bericht mindestens zwei Seiten umfassen: eine Seite mit dem Hauptbericht, eine andere mit einem detaillierten Bericht.

Für diese Art von Verknüpfung müssen die folgenden Parameter festgelegt werden:

· Seitenname in diesem Bericht;

· Name des Berichtsparameters, der den Wert des Hyperlinks annimmt;

· Wert des Hyperlinks.



Wird die Verknüpfung angeklickt, geschieht das Folgende:

· Die Berichtsparameter werden auf den Wert des Hyperlinks festgesetzt;

· Die angezeigte Berichtseite wird in einem separaten Vorschaufenster erstellt und ausgeführt.

Der Wert des Berichtsparameters kann durch Nutzung einer von zwei Methoden angezeigt werden:

· Direkte Anzeige des Wertes;

· Ein Ausdruck wird angezeigt, der den Wert zurück liefert. Dieser Ausdruck wird beim Durchlauf des Berichts berechnet.

> Wird eine Berichtseite gewählt, setzt sich die Eigenschaft "Sichtbar" auf false zurück. Diese Seite wird während der Erstellung des Hauptberichts übersprungen.

Benutzerdefinierte Verknüpfung

Durch Nutzung dieser Art von Verknüpfung lässt sich eine benutzerdefinierte Reaktion auf den Mausklick festlegen. Hierzu wird der Ereignishandler "Click" des Objektes genutzt. Um dies zu tun:

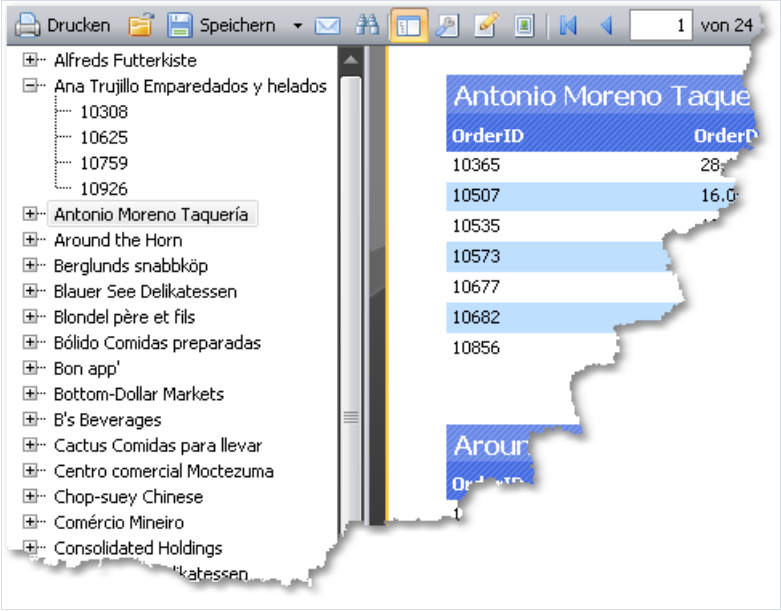
- Auswahl des Objektes und Öffnen des Fensters "Eigenschaften";
- Klicken der Schaltfläche zur Anzeige der Ereignisse des Objektes;
- Doppelklick auf das Ereignis "Click". FastReport wechselt zum "Code"-Fenster und erstellt einen leeren Ereignishandler.

Hier sollten alle notwendigen Schritte durchgeführt werden. Wahrscheinlich wird eine Verknüpfung mit dem Objekt, den der Handler aufruft, und einen Hyperlink-Wert benötigt. Nutzung des Parameter Senders des Handlers:

```
private void Text2_Click(object sender, EventArgs e)
{
  // sender - this is the object which was clicked.
  // In order to receive the value of the hyperlink, you need
  // to cast the sender to ReportComponentBase type.
  object hyperlinkValue = (sender as ReportComponentBase).Hyperlink.Value;

  MessageBox.Show("Hyperlink value = " + hyperlinkValue.ToString());
}
```

Berichtsgliederung

Die Gliederung des Berichts (auch Dokumentenstruktur genannt) ist eine Steuerung in Baumansicht, die im Vorschaufenster angezeigt wird:



Diese Steuerung zeigt die während der Berichterstellung erzeugte Baumstruktur an. Durch Klicken auf ein Baumelement wird das entsprechende Berichtselement angesteuert.

Hat der Bericht eine Gliederung wird diese automatisch angezeigt. Die Gliederung kann durch Klicken der Schaltfläche auf der Symbolleiste angezeigt oder verborgen werden. Dieser Bericht erstellt keine automatische Gliederung: sie muss selbst initiiert werden.

Die Berichtseite und all seine Leisten weisen die Eigenschaft "Outline Expression" auf. Zur Erstellung der Gliederung wird ein Ausdruck angezeigt, der den Text des Elements in dieser Eigenschaft zurück liefert. Dieser Ausdruck wird beim Druck einer Leiste berechnet und sein Wert zur Gliederung hinzugefügt. Ein Bericht des Typen Master-Detail oder Gruppe, weist eine Gliederungsstruktur auf, die der einer Berichtsstruktur ähnelt.

Die Eigenschaft "OutlineExpression" kann im Fenster "Eigenschaften" festgelegt werden.

Hier einige Empfehlungen zur Konfiguration der Gliederung in unterschiedlichen Berichtstypen:

· Sollen die Blätter eines erstellten Berichtes in der Gliederung angezeigt werden, muss die Eigenschaft "OutlineExpression" der Berichtseite festgelegt werden. Der Ausdruck wird die Seitenzahl zurückgeben:

```
[PageN]
```

· In der Berichtsart "Einfache Liste" mit einer Datenleiste, wird die Eigenschaft "OutlineExpression" dieser Leiste festgelegt. Als Ausdruck kann jede beliebige Datenspalte, die auf der Leiste gedruckt ist, genutzt werden;

· In der Berichtsart Master-Detail mit zwei "Daten"-Leisten, wird die Eigenschaft "OutlineExpression" der entsprechenden Leisten festgelegt. Die Berichtsart "Category/Product" beinhaltet die "OutlineExpression" für die erste Leiste den Namen der Kategorie und für die zweite Leiste den Produktnamen;

· Im Gruppenbericht, wird die "OutlineExpression" Eigenschaft des Gruppenkopfes und eine "Daten"-Leiste

konfiguriert . Die Gruppenbedingung wird als ein Ausdruck des Gruppenkopfes genutzt. Für die "Daten"-Leiste kann jede beliebige Datenspalte, die auf der Leiste gedruckt ist, genutzt werden.

Beispiele

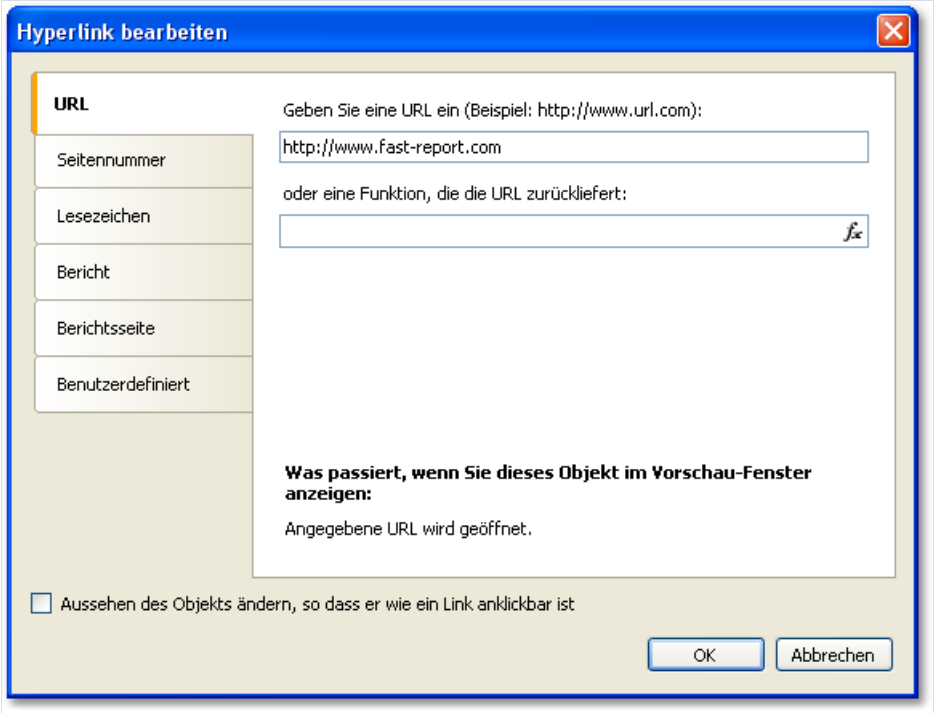
Beispiel 1: Verknüpfung mit einer Webseite

In diesem Beispiel wird ein einfacher Bericht mit einem "Text"-Objekt erstellt. Durch Klicken auf das Objekt im Vorschaufenster wird der Benutzer an die FastReport Webseite weitergeleitet.

Erstellung eines neues Berichts und hinzufügen eines "Text"-Objektes. Der folgende Text wird eingegeben:

```
Go FastReport home page
```

Klick mit der rechten Maustaste aus das Objekt und Auswahl des Elements "Hyperlink" aus dem Kontextmenü. Die Objekte werden folgendermaßen konfiguriert:



Hiernach wird das Kontrollkästchen "Aussehen des Objekts ändern, so dass er wie ein Link anklickbar ist" aktiviert, um einige Verknüpfungsattribute anzuwenden (blaue Textfarbe, unterstreichen und ein Zeiger in der Form einer Hand).

Durchführung des Berichts und klicken auf das Objekt. Das Fenster des Webbrowsers öffnet sich und es wird an die FastReport Startseite weitergeleitet.

Beispiel 2: Erstellen eines Detailberichts

In diesem Beispiel wird ein Bericht erstellt, der die Kategorienliste anzeigt. Durch Klicken auf den Kategoriennamen wird ein Bericht angezeigt, der eine Liste der Produkte in den gegebenen Kategorien umfasst.

Hierzu müssen die folgenden Schritte beachtet werden:

· Zuerst wird ein Detailbericht erstellt;

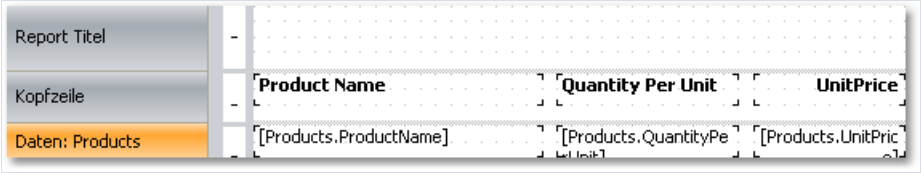
· Definition der Bericht Parameter die eine Kategorie identifizieren;

· Festlegen der Datenfilterung für diesen Parameter;

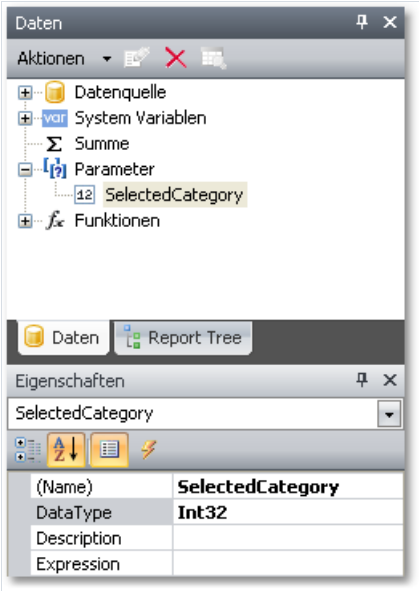
· Erstellen des Hauptberichtes;

· Konfiguration des Hyperlinks im Hauptbericht, so dass der Detailbericht, mit dem Parameter für die gewählte Kategorie, durchgeführt werden kann.

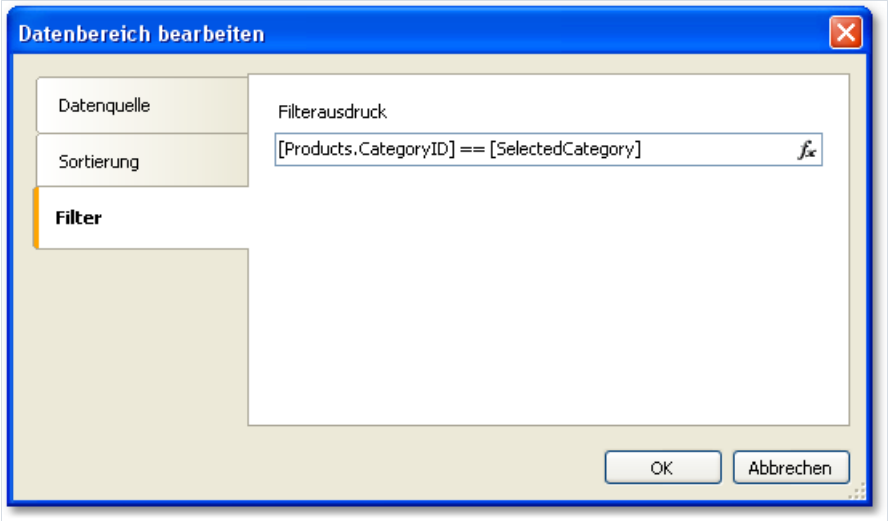
Zuerst wird ein Detailbericht erstellt, der die Liste der Produkte druckt. Hierzu wird ein neuer Bericht erstellt und die "Products"-Tabelle aus der Datenquelle ausgewählt. Die Objekte müssen folgendermaßen positioniert werden:



Erstellen Sie den Parameter, der genutzt wird um eine ausgewählte Kategorie von dem Hauptbericht an den detaillierten Bericht weiterzugeben. Zur Identifizierung der Kategorie wird die Spalte "CategoryID", die sowohl in der Tabelle "Categories", wie auch in der Tabelle "Products" enthalten ist, genutzt. Die Parameter sollten folgendermaßen konfiguriert werden:

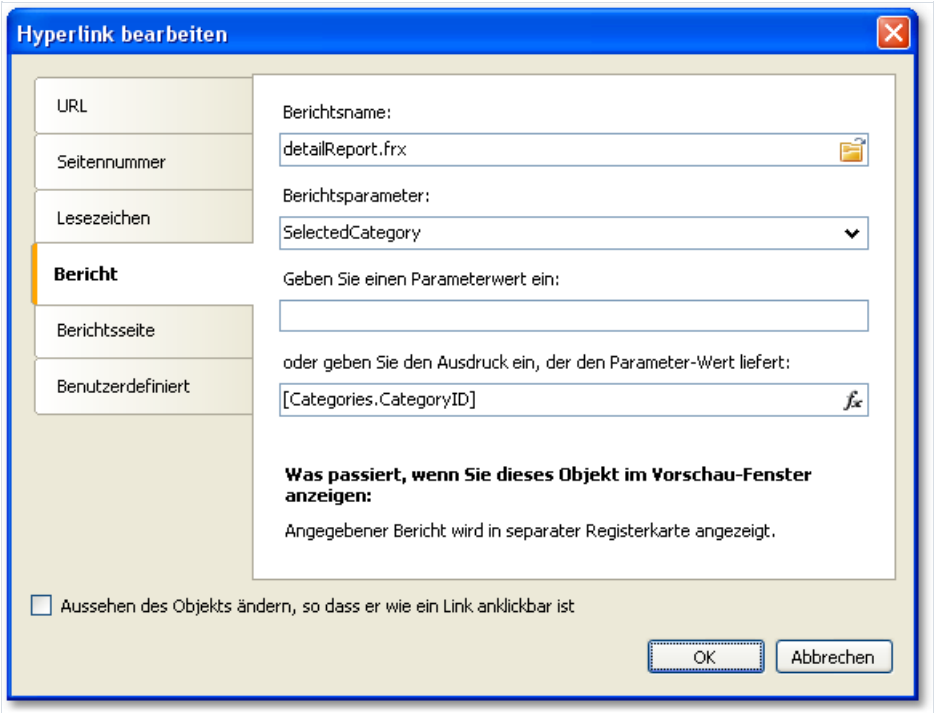


Hiernach wird die Datenfilterung, zur Filterung aller Produkte, die in die spezifische Kategorie gehören, festgelegt. Um dies zu tun, wird die "Daten"-Leiste doppelgeklickt. Dann wird zur Registerkarte "Filter" gewechselt und die folgenden Bedingungen werden eingestellt:



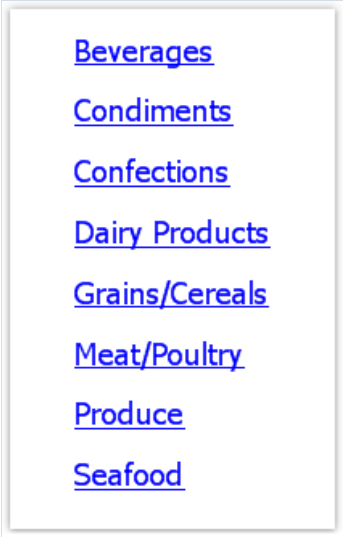
Hiernach wird der Hauptbericht erstellt. Ein neuer Bericht wird erstellt und die "Categories"-Tabelle aus der Datenquelle ausgewählt. Die Objekte müssen folgendermaßen positioniert werden:

Mit der rechten Maustaste auf das "Text"-Objekt klicken und das Menüelement "Hyperlink" klicken. Die Objekte sollten folgendermaßen konfiguriert werden:

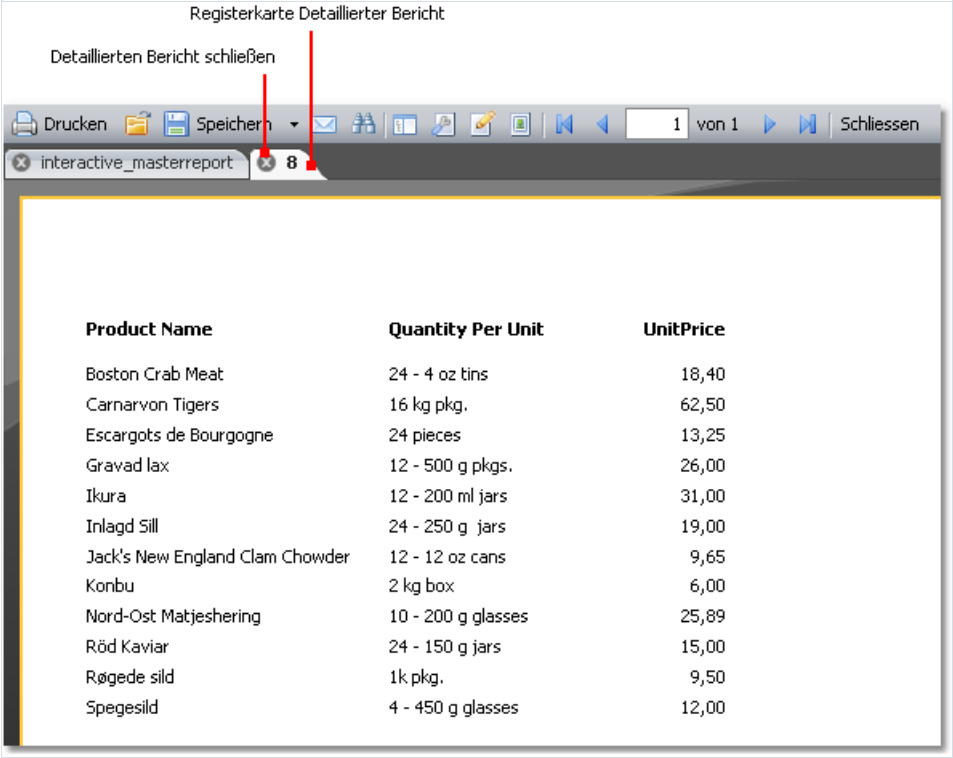


Als Berichtsname wird der Name der detaillierten Berichtsdatei angegeben. Die Berichtsparameter können, durch Klicken der Schaltfläche auf der rechten Seite der Liste, aus einer Dropdownliste ausgewählt werden. Geben Sie als Parameterwert den Ausdruck "[Categories.CategoryID]" an.

Beim Durchlauf des Berichts wird die Kategorienliste sichtbar:



Durch Klicken auf eine der Kategorien wird ein detaillierter Bericht erstellt. Dieser wird auf einer separaten Registerkarte des Vorschaufensters angezeigt:



Wie auf dem Bild zu erkennen ist, wird der Titel der Registerkarte auf den Wert des Hyperlinks eingestellt. In dem hier vorgestellten Beispiel handelt es sich um einen numerischen Wert, der in der Datenspalte "CategoryID" enthalten ist. Diese Darstellung ist weder sonderlich informativ noch gut aussehend. Der Bericht sollte so geändert werden, dass anstelle der Nummer der Kategorienname verwendet wird. Um dies zu tun, befolgen Sie bitte die folgenden Anweisungen:

Im Detailbericht:

· Änderung der Eigenschaft "DatenType" des Parameters zu "String";

· Hinzufügen der Datenquelle "Categories" in den Bericht. Diese wird bei der Datenfilterung zur Referenzierung der Spalte "Categoryname" genutzt;

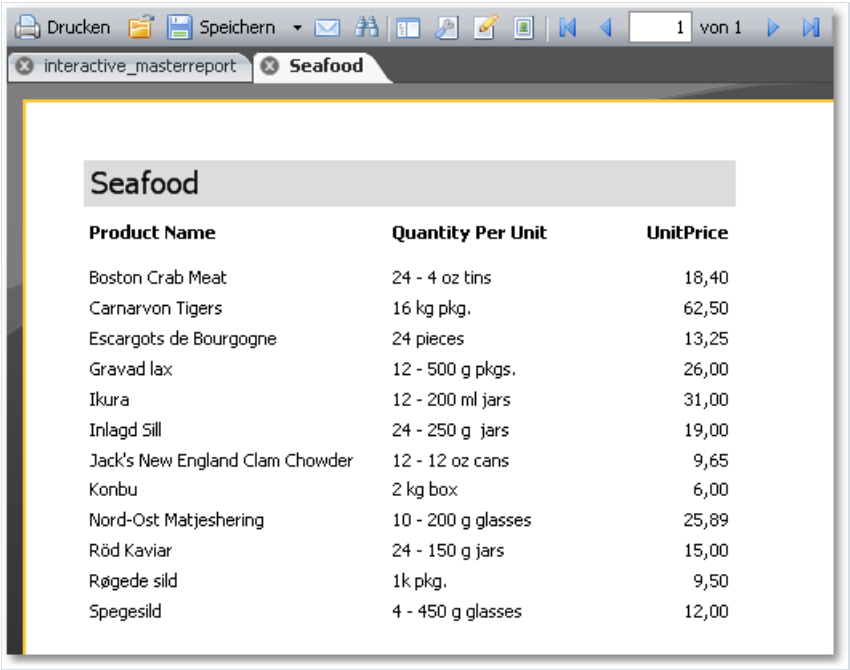
· Änderung des Filterausdrucks der "Daten"-Leiste:

```
[Products.Categories.CategoryName] == [SelectedCategory]
```

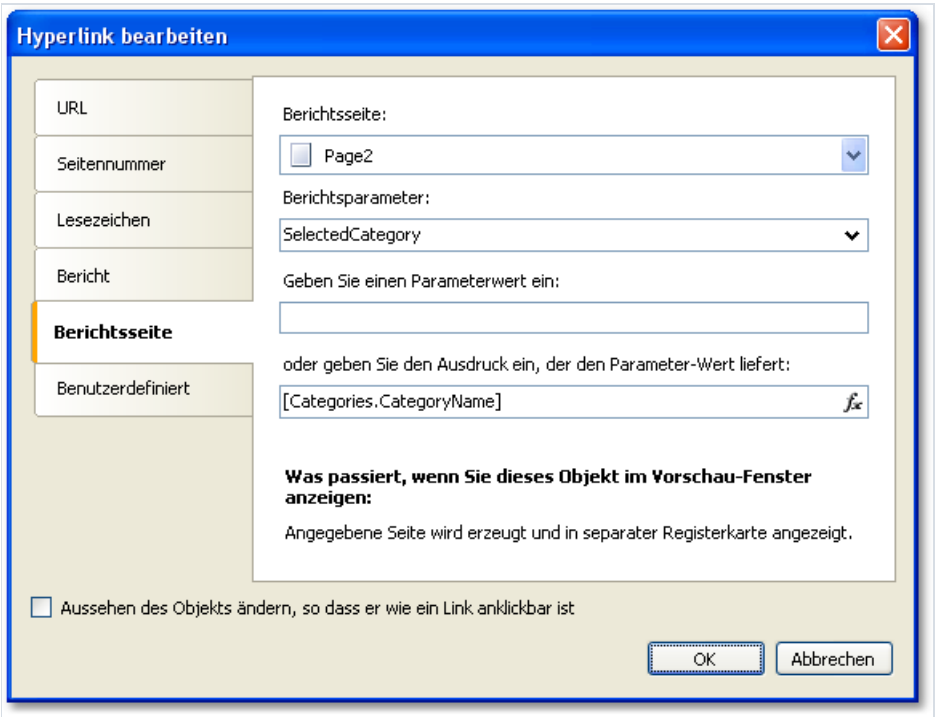
Im Hauptbericht:

· Änderung der Einstellungen des Hyperlinks. Danach wird der Wert "[Categories.CategoryName]" in die Bericht Parameter eingegeben.

Wird der Bericht durchgeführt, kann man erkennen, dass der Titel der Registerkarte auf den Kategoriennamen umgeändert wurde. Nun kann der Detailbericht verbessert werden. Hinzufügen des "Text"-Objektes. Dies wird den Namen der ausgewählten Kategorie im Berichtstitel drucken:



In diesen Beispielen, haben wir zwei Berichte erstellt und mehrfach zwischen diesen gewechselt. Diese Vorgehensweise ist verbesserungsfähig. Zur Vereinfachung dieser Aufgabe können zwei Berichte in einen platziert werden: der Hauptbericht auf der ersten und der Detailbericht auf der zweiten Seite. Hierzu muss der Hyperlink auf folgende Art festgelegt werden:



In diesem Fall muss die Seite 2 als Detailbericht gewählt werden.

Beispiel 3: Das interaktive "Matrix"-Objekt

Dieses Beispiel verdeutlicht, wie durch klicken auf die Zelle des "Matrix"-Objektes, ein Detailbericht erstellt wird. Als Beispiel nutzen wir eine Matrix, die die Verkäufe der Mitarbeiter/innen nach Jahren gruppiert anzeigt.

Als Datenquelle für diese Matrix wird die "MatrixDemo"-Tabelle genutzt. Sie repräsentiert die Verkäufe der Mitarbeiter/innen, gruppiert nach Jahren und Monaten:

Name	Year	Month	ItemsSold	Revenue
Nancy Davolio	1999	2	1	1000
Nancy Davolio	1999	11	1	1100
Nancy Davolio	1999	12	1	1200
Nancy Davolio	2000	1	1	1300
Nancy Davolio	2000	2	2	1400
Nancy Davolio	2001	2	2	1500
Nancy Davolio	2001	3	2	1600
Nancy Davolio	2002	1	2	1700
Andrew Fuller	2002	1	2	1800
Andrew Fuller	1999	10	2	1900
Andrew Fuller	1999	11	2	2000
Andrew Fuller	2000	2	2	2100
Janet Leverling	1999	10	3	3000
Janet Leverling	1999	11	3	3100
Janet Leverling	2000	3	3	3200
Steven Buchanan	2001	1	3	4000
Steven Buchanan	2001	2	4	4100
Steven Buchanan	2000	1	4	3999

Die Matrix sollte folgendermaßen konfiguriert werden:

· Hinzufügen der Datenspalte "MatrixDemo.Year" zu dem Spaltenkopf;

· Hinzufügen der Datenspalte "MatrixDemo.Name" zu dem Zeilenkopf;

· Hinzufügen der Datenspalte "MatrixDemo.Revenue" in die Zelle;

Hiernach sieht eine vorbereitete Matrix wie folgt aus:

	Year				
Employee	1999	2000	2001	2002	Summe
Andrew Fuller	3 900,00	2 100,00		1 800,00	7 800,00
Janet Leverling	6 100,00	3 200,00			9 300,00
Nancy Davolio	3 300,00	2 700,00	3 100,00	1 700,00	10 800,00
Steven Buchanan		3 999,00	8 100,00		12 099,00
Summe	13 300,00	11 999,00	11 200,00	3 500,00	39 999,00

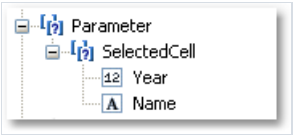
Wie angezeigt, stellt der Zellenwert die Summe der Verkäufe der Mitarbeiter/innen für das Gesamtjahr dar. Hier wird ein Detailbericht erstellt, der angezeigt wird, wenn auf die Zelle geklickt wird. In diesem Beispiel kann der Detailbericht die Verkäufe eines/einer ausgewählten Mitarbeiter/in für jeden Monat eines ausgewählten Jahres beinhalten.

Wie wird eine Zelle mit Daten verbunden auf dessen Basis diese gedruckt wurde? Jede Zelle der Matrix hat ihre eigene Adresse. Dies ist eine Kombination der Werte aus den Spalten-, und Zeilenköpfen. In diesem Beispiel ist die Adresse eine Kombination des Jahres und des Namen des Mitarbeiters. Genau diese Daten können an den Detailbericht weitergegeben werden. Wie funktioniert dies? Dies ist prinzipiell recht einfach: der Hyperlink wird festgelegt. Er zeigt lediglich den Bericht Namen und den Namen des Parameter. Die Parameter Werte müssen nicht anzeigt werden: für die Matrixzelle wird der Wert von FastReport erstellt und an den Parameter weitergegeben.

Unter der Annahme, dass wir auf die obere linke Zelle, mit der Nummer 3900, geklickt haben: Dies ist die Summe der Verkäufe des Mitarbeiters "Andrew Fuller" für das Jahr 1999. Welche Form wird zur Weitergabe dieses Wertes in den Parameter genutzt? FastReport verbindet Spalten-, und Zeilenwerte mittels eines Trennzeichens:

```
1999;Andrew Fuller
```

Bedeutet dies, dass der Wert des Jahres und der Name des Mitarbeiters aus dieser Zeichenfolge extrahiert werden, das Jahr in ein int konvertiert und diese Daten zur Datenfilterung genutzt werden müssen? Nein, das Verfahren ist um einiges einfacher. Hier muss lediglich ein Parameter erstellt werden, der geschachtelte Parameteraufweist. Weitere Informationen zu diesem Objekt werden im Kapitel "Daten" erläutert. In diesem Beispiel können die übergeordneten Parameter folgendermaßen aussehen:



Bei der Erstellung der Parameter sollte das Folgende beachtet werden:

· Der übergeordnete Parameter muss nicht festgelegt werden. Es bedarf lediglich einer Namensgebung;

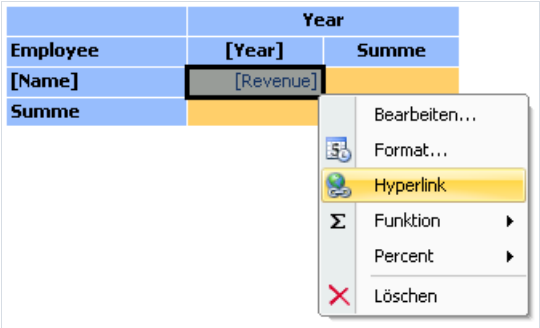
· Ein übergeordneter Parameter muss so viele geschachtelte Parameter aufweisen, wie Werte von der Matrix weitergegeben werden. In diesem Beispiel hat er zwei Werte;

· Die Reihenfolge der geschachtelten Parameter muss mit der Reihenfolge der Werte, die von der Matrix weitergegeben werden, übereinstimmen. In diesem Beispiel wird das Jahr im ersten Parameter und der Name des Mitarbeiters im zweiten Parameter weitergegeben;

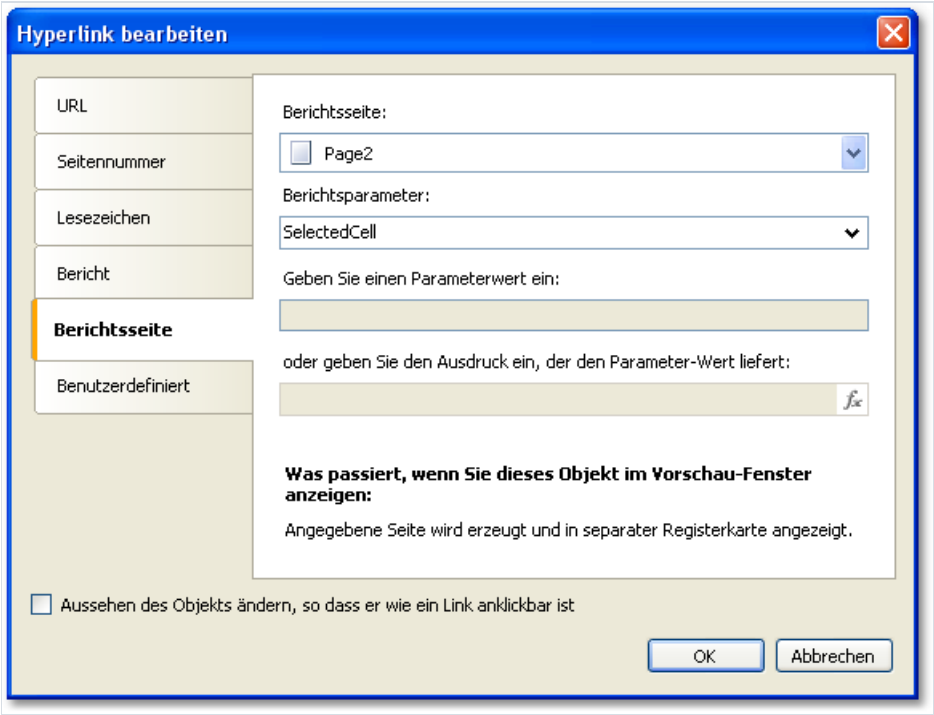
· geschachtelte Parameter können beliebig benannt werden. Es ist jedoch ratsam Ihnen Namen zu geben, die mit dem Namen des Matrix-Elements korrespondieren;

· Die korrekte Festlegung des Datentypen für jeden geschachtelten Parameter ist von großer Wichtigkeit. Die Datentypen müssen mit dem Wert korrespondieren, der an den Parameter weitergegeben wird. In diesem Beispiel muss der erste Parameter (Jahr) eine ganze Zahl und der zweite Parameter (Name des Mitarbeiters) eine Zeichenfolge sein.

Nachdem alle benötigten Details geklärt wurden wird der Bericht erstellt. Die Zelle der Matrix wird ausgewählt und der Hyperlink Editor aufgerufen:

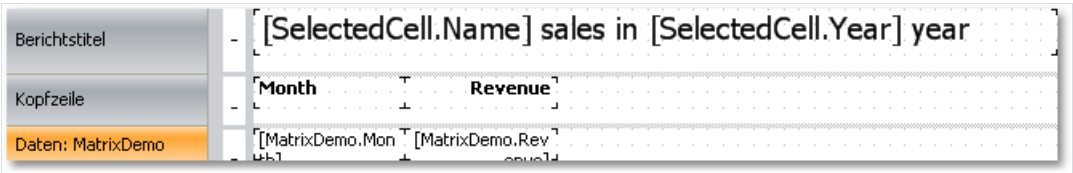


In der Konfiguration des Hyperlink wird der übergeordnete Parameter als ein Berichtsparameter (in unserem Beispiel - "SelectedCell") angezeigt:



FastReport gibt die Werte an die übergeordneten Parameter SelectedCell.Year und SelectedCell.Name weiter. Diese Werte werden in Datentypen konvertiert, die in der Parameterkonfiguration angezeigt sind: daher müssen die Parameter des Datentypen korrekt konfiguriert werden.

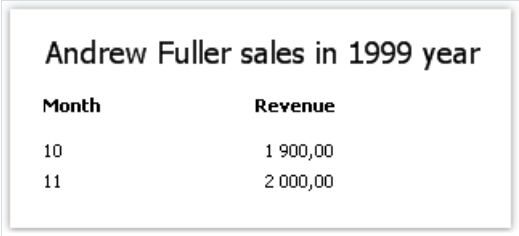
Der Detailbericht wird auf einer separatem Seite des Hauptberichts platziert und nutzt dieselbe Datenquelle:



Zur Anzeige der Verkäufe eines ausgewählten Mitarbeiters für ein Jahr wird die Filterung eingestellt. Hierzu muss der "Daten"-Leisten Editor aufgerufen und die folgende Filterbedingung angezeigt werden:

```
[MatrixDemo.Year] == [SelectedCell.Year] && [MatrixDemo.Name] == [SelectedCell.Name]
```

Der Bericht ist erstellt. Er wird durchgeführt und es wird auf die obere linke Zelle geklickt. Ein Detailbericht, mit den folgenden Daten, wird geöffnet:



Andrew Fuller sales in 1999 year

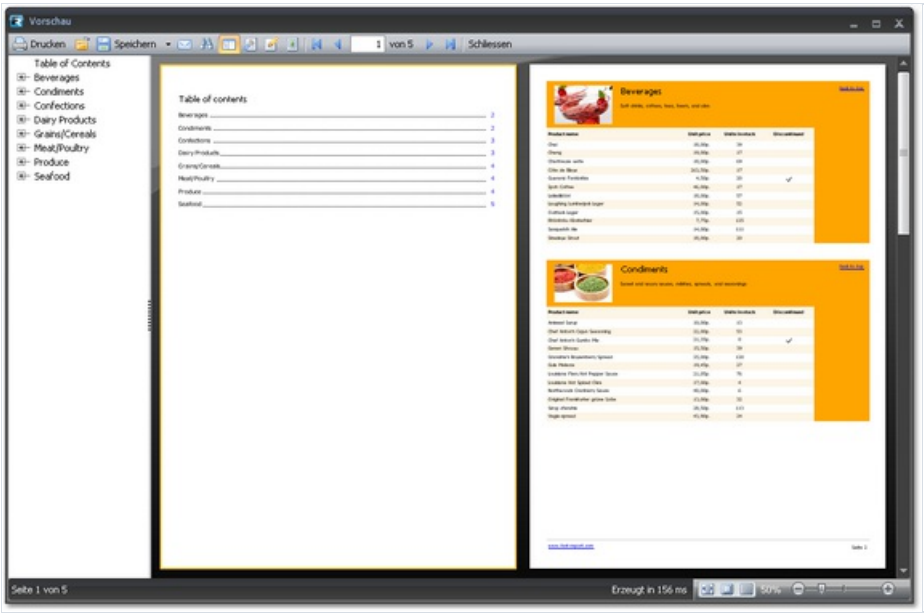
Month	Revenue
10	1 900,00
11	2 000,00

Wie hier dargestellt, korrespondiert die Summe der Werte (1900+2000) mit der Zelle der angeklickten Matrix.

Beispiel 4: Bericht mit Inhaltsangabe, Navigation und Gliederung

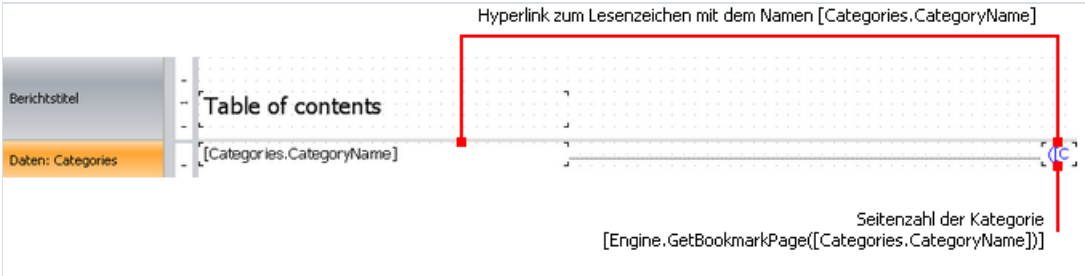
Dieses Beispiel veranschaulicht das Erstellen eines Berichts mit den folgenden Eigenschaften:

- Auf der ersten Seite wird die interaktive Inhaltsangabe, d.h. die einzelnen Elemente können zur Navigation der entsprechenden Seite angeklickt werden, gedruckt;
- Im Vorschaufenster wird die Gliederung angezeigt. Diese ist auch interaktiv.

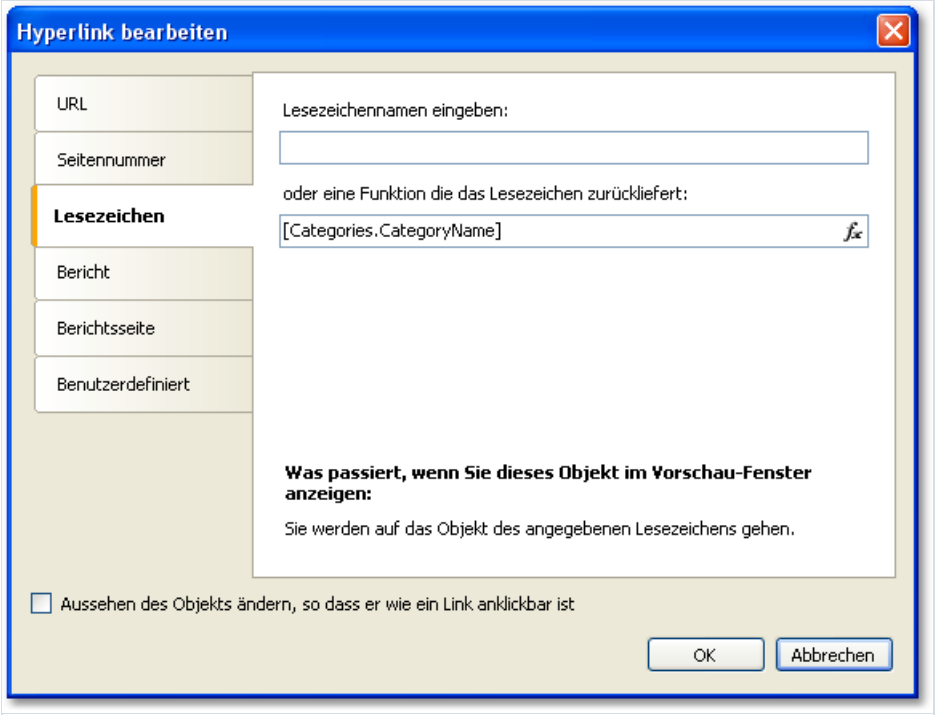


Der Bericht nutzt die Tabellen "Categories" und "Products". In der Inhaltsangabe wird die Kategorienliste gedruckt. Der Rest des Berichts wird die kategorisierte Produktliste drucken. Die Berichtsvorlage besteht aus zwei Seiten: die erste Seite wird zum Druck der Inhaltsangabe genutzt und die zweite Seite zum Druck des Hauptberichts.

Zuerst soll die Inhaltsangabe erläutert werden. Es wird ein neuer Bericht erstellt und diesem werden die Datenquellen "Categories" und "Products" hinzugefügt. Die "Daten"-Leiste wird mit der Tabelle "Categories" verbunden und die Objekte auf die folgende Art platziert.



Zur Erstellung interaktiver Inhaltsangaben-Objekte, müssen die "Hyperlink" Eigenschaften konfiguriert werden:



Anzeige des Kategorienamens als Lesezeichen. Das Lesezeichen des Objektes wird später festgelegt.

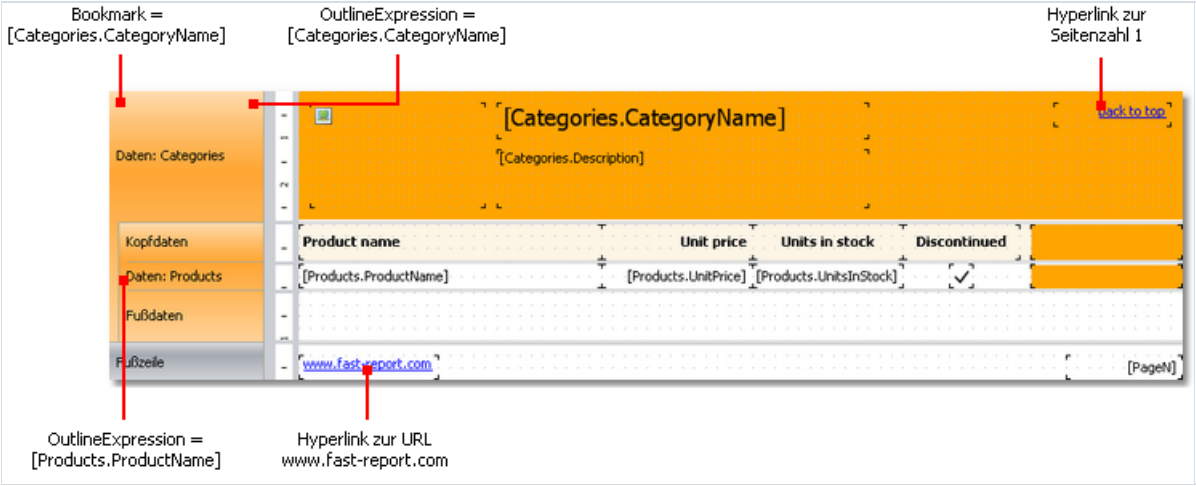
Um die Seitenzahl in der Inhaltsangabe zu drucken, muss Folgendes durchgeführt werden:

· Aktivierung der Einstellung "Double Pass" des Berichts. Dazu wird das Menü "Bericht|Stile" genutzt. Dies muss durchgeführt werden, da die Inhaltsangabe vor anderen Seiten gedruckt wird. Zu diesem Zeitpunkt weiß FastReport nicht, wo die Kategorien gedruckt werden;

· Nutzung der Funktion "Engine.GetBookmarkPage", der die Seitenzahl für das spezifische Lesezeichen zurück gibt. In diesem Beispiel wird der Ausdruck "[Categories.CategoryName]" als Lesezeichenname verwendet, so dass der Funktionsaufruf wie folgt aussieht:

```
[Engine.GetBookmarkPage([Categories.CategoryName])]
```

Auf der zweiten Berichtseite wird ein Master-Detail Bericht, wie in der unteren Abbildung dargestellt, erstellt:



Festlegen des Lesezeichens zu dem navigiert wird, wenn auf ein Element in der Inhaltsangabe geklickt wird. Hierzu wird die erste "Daten"-Leiste ausgewählt und der folgende Ausdruck in ihrer Eigenschaft "Lesezeichen" angezeigt:

```
[Categories.CategoryName]
```

Zur Festlegung der Gliederung des Berichts, sollte das Folgende durchgeführt werden:

· Auswahl der ersten Berichtseite. Dies kann durch Wechsel auf die Seite erfolgen;

· Im Fenster "Eigenschaften" muss der folgende Wert für die Eigenschaft "OutlineExpression" festgelegt werden:

```
"Contents"
```

- Wechsel zur zweiten Berichtsseite;

· Auswahl der ersten "Daten"-Leiste und Festlegung seiner Eigenschaft "OutlineExpression":

```
[Categories.CategoryName]
```

- Auswahl der ersten "Daten"-Leiste und Festlegung der Eigenschaft "OutlineExpression":

```
[Products.ProductName]
```

Bericht Vererbung

Berichte beinhalten häufig die gleichen Daten; beispielsweise wird häufig dieselbe Kopf-, und Fußzeile mit einem Unternehmenslogo und einigen Daten, wie E-Mail, Adresse etc. angezeigt. Es besteht oftmals nur die Notwendigkeit eine der zahlreichen Informationen, z.B. E-Mail, zu ändern. Das müsste dann in jedem Bericht einzeln durchgeführt werden. Um diesen Mehraufwand zu vermeiden kann die Berichtsvererbung genutzt werden. Was ist das?

Es bestehen beispielsweise eine Reihe an identischen Elementen in einem Bericht (Logo, Firmenname, E-Mail, etc.). Üblicherweise sind diese Elemente auf der Titel und/oder dem Seitenkopf platziert. Es kann ein Basisbericht erstellt werden, der lediglich allgemein genutzte Elemente umfasst. Alle anderen Berichte nutzen diesen Basisbericht und beinhalten entsprechend die allgemeinen Elemente sowie zusätzlich für den spezifischen Bericht definierte Elemente.

Soll ein Elemente (Logo, E-Mail) geändert werden, wird der Basisbericht geöffnet und die erforderlichen Änderungen vorgenommen. Alle anderen Berichte, die aus diesem Basisbericht vererben, werden automatisch geändert. Wird ein vererbter Bericht geöffnet, wird zuerst der Basisbericht geöffnet und danach der vererbte Bericht.

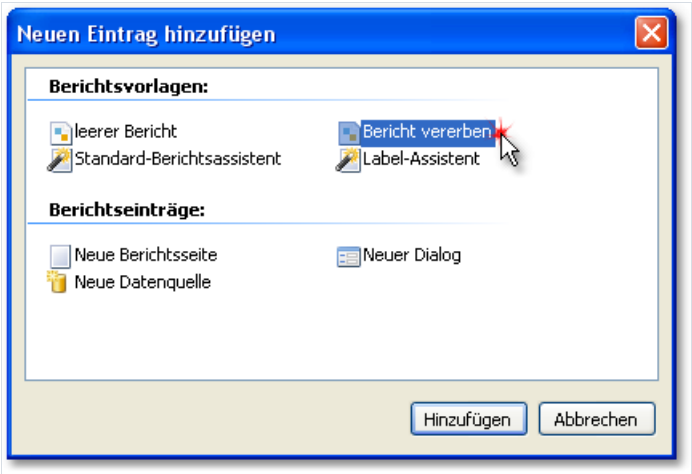
Erstellen eines Berichts

Zur Nutzung der Vererbung wird folgendermaßen vorgegangen:

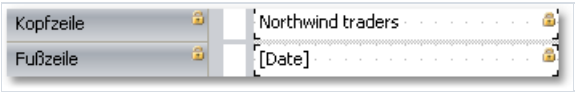
· Erstellen eines Basisberichtes und speichern als Datei;

· Erstellen eines neuen Berichts, der aus dem Basisbericht vererbt wird.

Zur Erstellung eines vererbten Berichts, wird das Menüelement "Datei|Neu..." ausgewählt und dann das Element "Bericht vererben" in dem Fenster:



Eine Basisberichtsdatei muss ausgewählt werden. Diese Datei wird zu diesem Zeitpunkt erstellt. Hiernach wird der Basisbericht in den Designer geladen. Dieser kann, je nach Wunsch, geändert werden. Objekte des Basisberichts sind mit einem "Lock"-Symbol gekennzeichnet:



Derart gekennzeichneten Elemente können weder gelöscht, noch umbenannt oder auf eine andere Leiste verschoben werden.

Es können neue Objekte oder Leisten hinzugefügt und das Aussehen, die Größe und die Platzierung der Objekte geändert werden. Nach diesen Schritten wird der Bericht gespeichert.

Änderung des Basisberichts

Hier wird dargestellt, was bei der Änderung des Basisberichts geschieht. Man kann:

· Ein Objekt des Basisberichts löschen. Dieses Objekt wird auch aus dem vererbten Bericht gelöscht.

· Ein Objekt in den Basisbericht hinzufügen. Dieses Objekt erscheint automatisch im vererbten Bericht.

· Die Größe, Platzierung, den Text oder das Aussehen der Objekte ändern. Für den Fall das dieses Objekte nicht im vererbten Bericht geändert wurde, spiegeln sich alle Änderungen im vererbten Bericht wider.

Bezüglich des letzten Punktes bedarf es zusätzlicher Erläuterungen. Betrachtung zweier Beispiele der Nutzung der Vererbung. Im ersten Beispiel werden folgende Schritte durchgeführt:

· Erstellen eines Basisberichts, der das Text1-Objekt beinhaltet;

· Erstellen eines vererbten Berichts und dessen Speicherung ohne die Änderungen zu speichern;

· Öffnen des Basisberichts und Verschiebung des Text1-Objekts;

· Öffnen des vererbten Berichts und es lässt sich erkennen, dass das Text1 Objekt auch verschoben wurde.

Im zweiten Beispiel werden folgende Schritte durchgeführt:

· Erstellen eines Basisberichts, der das Text1-Objekt beinhaltet;

· Erstellen eines vererbten Berichts;

· Verschieben des Text1 Objekts im vererbten Bericht zu einer neuen Position und Speichern des Berichts;

· Öffnen des Basisberichts und Verschiebung des Text1-Objekts zu einer anderen Position;

· Öffnen des vererbten Berichts und es lässt sich erkennen, dass das Text1 Objekt nicht verschoben wurde.

Dies ist der Fall, da das Objekt im vererbten Bericht geändert wurde. Diese Änderung wurde in der Datei des vererbten Berichts gespeichert. Wird nun das Originalobjekt im Basisbericht geändert, wird diese Änderung im vererbten Bericht ignoriert. Im hiesigen Beispiel wird die neue Platzierung des Objektes ignoriert. Alle anderen Änderungen (wie Textfarbe etc.) werden im vererbten Bericht angezeigt.

Dies wird deutlicher, wenn alle Inhalte der vererbten Berichtsdatei betrachtet werden. Hier beispielsweise ist wie das Originalobjekt im vererbten Bericht gespeichert wurde, wenn das Objekt nicht geändert wurde:

```
<inherited Name="Text1"/>
```

Wird die Position des Objektes im vererbten Bericht geändert, wird es folgendermaßen gespeichert:

```
<inherited Name="Text1" Left="255.15" Top="28.35"/>
```

Beim Öffnen des vererbten Berichts wird FastReport alle im Basisbericht definierten Objekteigenschaften, plus der im vererbten Bericht gespeicherten Eigenschaften laden.

Beschränkungen

Die Berichtsvererbung wurde zu folgendem Zweck entwickelt: Speichern gemeinsamer Berichtselemente, wie Kopf- ,und Fußzeile, in getrennten Dateien und erneute Nutzung in vererbten Berichten. Die Vererbung sollte nicht zur Ausführung komplexerer Aufgaben genutzt werden. Im speziellen sollten die folgenden Aktionen vermieden werden:

· Keine Vererbung des Berichts aus einem vererbten Bericht (nicht zweimal vererben);

· Komplexe Objekte, wie Tabellen oder Matrizen, sollten nicht im Basisbericht genutzt werden;

· Skript sollte nicht im Basisbericht genutzt werden;

· Parameter sollten nicht im Basisbericht genutzt werden;

Berichte mit Diagrammen

Zum Anzeigen von Diagrammen nutzt FastReport die Microsoft Chart Bibliothek. Diese Bibliothek wird im .Net Framework 4.0 enthalten sein. Derzeit ist sie hier als separater Download erhältlich:

http://www.microsoft.com/downloads/details.aspx?FamilyID=130f7986-bf49-4fe5-9ca8-910ae6ea442c

Diese Bibliothek erfordert .Net Framework 3.5 SP1. Hier können Sie mehr über die Eigenschaften dieser Freeware erfahren:

http://code.msdn.microsoft.com/mschart

Einige Eigenschaften von Microsoft Chart:

· mehr als 30 Arten von Reihen (Balken, Spalten, Bereiche, Linien, Blasen, Kuchen, zirkulär, Finanzen, Pyramide, Bereiche);

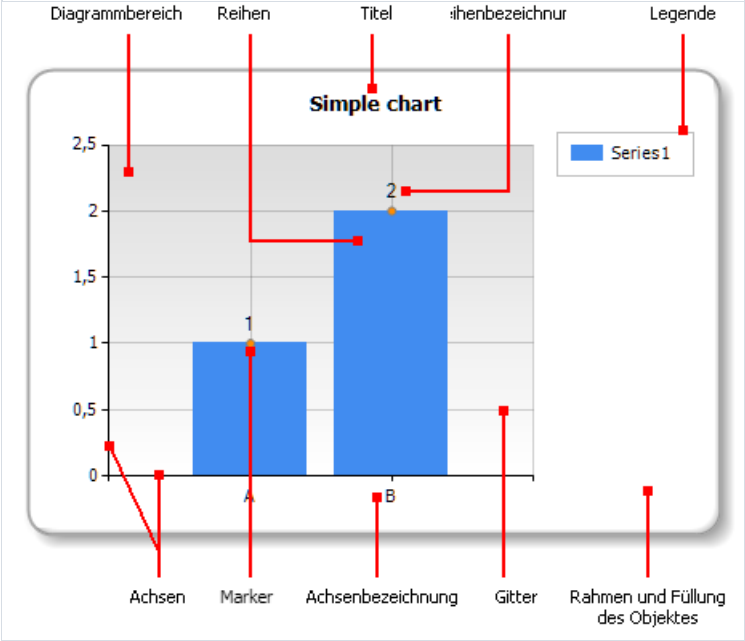
· 3-D Unterstützung;

· unterstützt mehrere Reihen, unterschiedlicher Arten, in einem Diagramm;

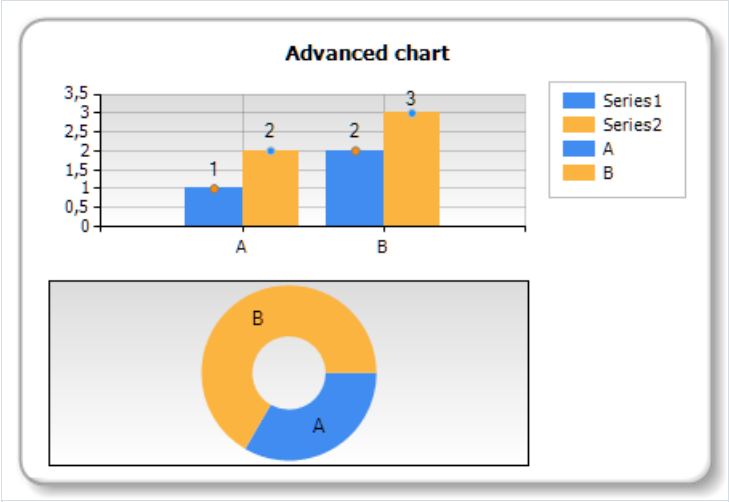
· vollständige Steuerung des Aussehens und des Verhaltens eines jeden Diagrammelementes.

Diagrammelemente

Microsoft Chart umfasst die folgenden Elemente:



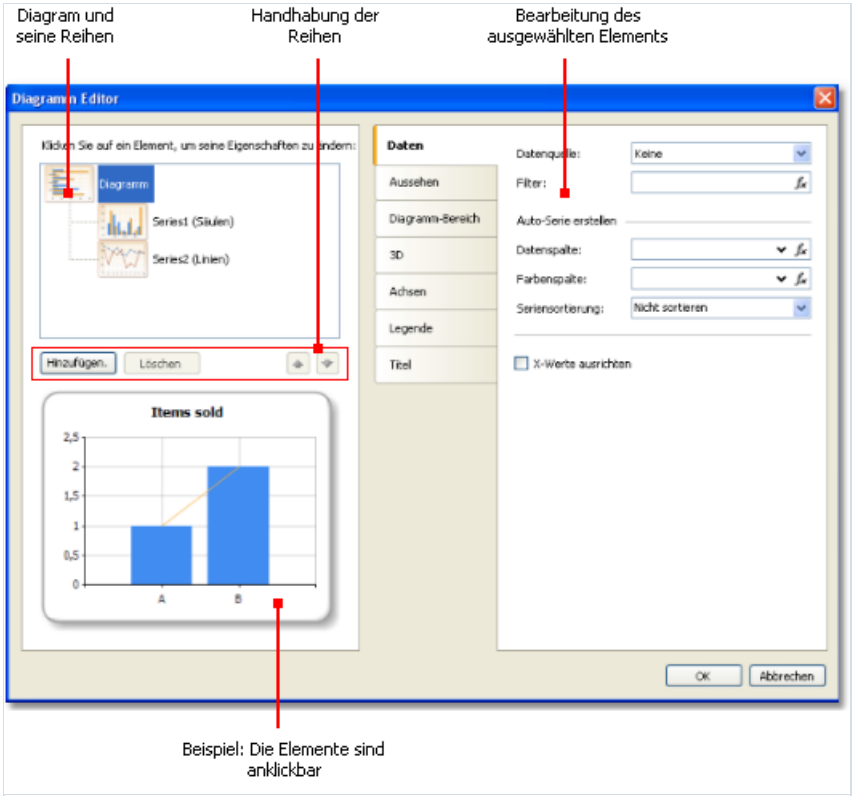
Ein Diagramm kann einen oder mehrere Bereiche aufweisen. Ein Diagramm kann eine oder mehrere Reihen aufweisen. Weiter unten ist ein Diagramm mit zwei Diagrammbereichen abgebildet (der erste Bereich beinhaltet zwei Reihen, der zweite Bereich beinhaltet eine Reihe):



Einige Reihen (z.B. Kuchenreihen) erfordern einen exklusiven Diagrammbereich.

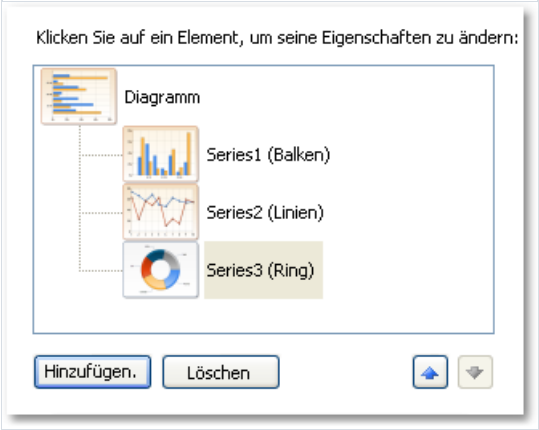
Diagramm Editor

Das "Diagramm"-Objekt umfasst eine Vielzahl an Einstellungen, die mittels des Diagramm Editors geändert werden können. Doppelklicken des "Diagramm"-Objektes ruft den Editor auf:

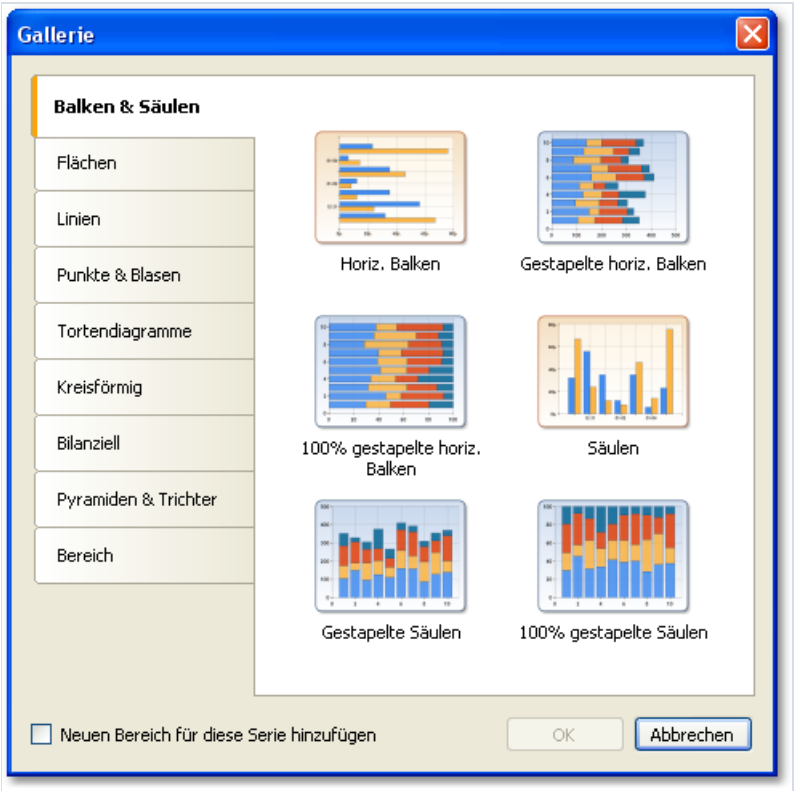


Bearbeitung von Reihen

Das "Diagramm"-Objekt kann eine oder mehrere Reihen aufweisen. Die Liste der Reihen wird im Editor angezeigt:



Zum Hinzufügen einer neuen Reihe wird die Schaltfläche "Hinzufügen..." geklickt. Das Dialogfeld "Gallerie" wird angezeigt:



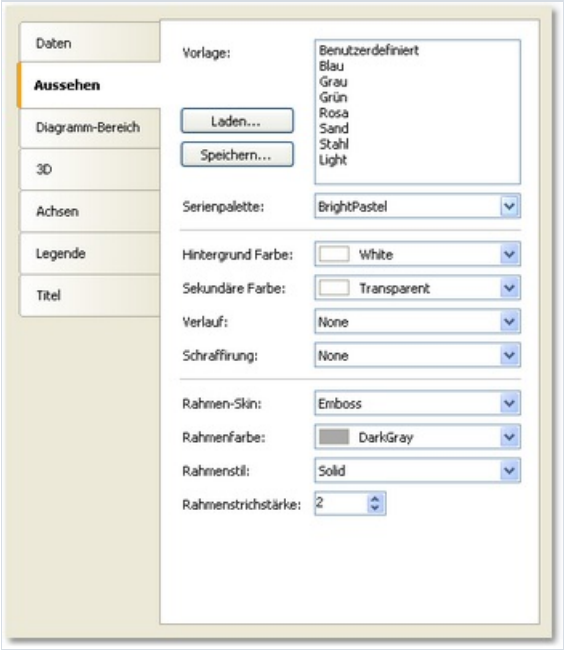
Zuerst wird die benötigte Kategorie, dann die benötigte Reihenart ausgewählt. Soll die Reihe in einen eigenen Diagrammbereich platziert werden, muss das Kontrollkästchen "Neuen Bereich für diese Serie hinzufügen" markiert werden. Für einige Diagrammarten (wie Kuchen, zirkulär, Finanzen, Pyramide) wird dieser neue Diagrammbereich automatisch, ungeachtet der Markierung des Kontrollkästchens, hinzugefügt.

Zum Löschen der Reihe wird die Schaltfläche "Löschen" geklickt. Zur Änderung der Reihenfolge der Reihen werden die Schaltflächen "Auf" und "Ab" genutzt.

Einstellen des Aussehens

Mittels des Diagramm Editors kann das Aussehens eines jeden Diagrammelementes eingestellt werden. Alle Eigenschaften (mehr all 100) sind in mehrere Kategorien aufgeteilt. Einige sind für die "Diagramm"-Objekte spezifisch, während andere einen Teil einer Reihe bilden.

Wird das "Diagramm"-Objekt aus der Reihenliste ausgewählt, werden die folgenden Eigenschaftenseiten angezeigt:



· "Aussehen" - Rahmenlinie und Füllung des Diagramms;

· "Diagramm-Bereich" - Rahmen, Füllung, Schatten;

· "3D" - 3D Einstellungen;

· "Achsen" - Einstellung des Aussehens der Achsen, deren Titel, Bezeichnung, Gitter, Marker, benutzerdefinierte Benennung und Leisten;

· "Legende" - Stil der Legende, Andocken, Rahmen, Füllung, Schatten und Schriftart;

· "Titel" - Stil der Legende, Andocken, Rahmen, Füllung, Schatten und Schriftart;

Wird das Objekt Reihe aus der Reihenliste ausgewählt, werden die folgenden Eigenschaftenseiten angezeigt:

· "Aussehen" - einige Einstellungen sind für die ausgewählte Reihenart spezifisch;

· "Füllung & Rahmen" - Füllung und Rahmen der Reihenwerte;

· "Labels" - Reihenbezeichnung. Die Bezeichnungsart, Farbe und Füllung können ausgewählt werden.

· "Marker" - Reihenmarker; Die Art des Markers, seine Farbe und sein Rahmen können ausgewählt werden.

Verbindung der Daten mit dem Diagramm

Die Diagramme können auf mehrere Arten mit Daten gefüllt werden:

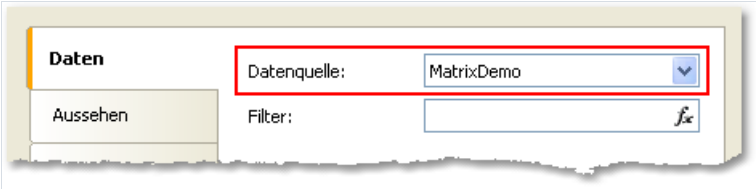
1. Nutzung der Datenquelle. Hierzu muss die Datenquelle für das "Diagramm"-Objekt angezeigt und jede Reihe mit einer Datenspalte verbunden werden.
2. Nutzung fixer Werte für alle Reihen.
3. Füllung des Objektes mit Daten mittels eines Skripts.

Zur Verbindung des Diagramms mit einer Datenquelle müssen die folgenden Schritte befolgt werden:

· Auswahl des "Diagramm"-Objektes in der Reihenliste;

· Wechsel zur Registerkarte "Daten";

· Auswahl der Datenquelle.

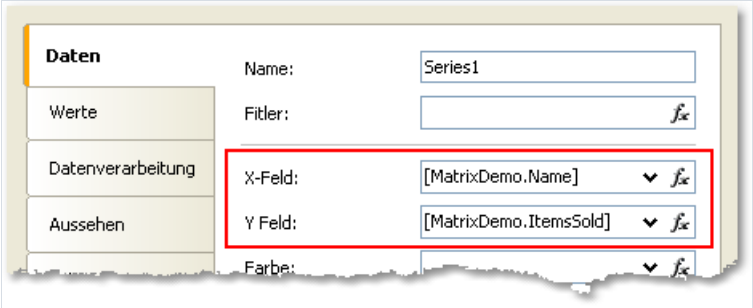


· Festlegung des Ausdrucks des Datenfilters, falls notwendig. Dieser Filter wird auf alle Diagrammreihen angewandt;

· Auswahl der Reihen aus der Reihenliste;

· Wechsel zur Registerkarte "Daten";

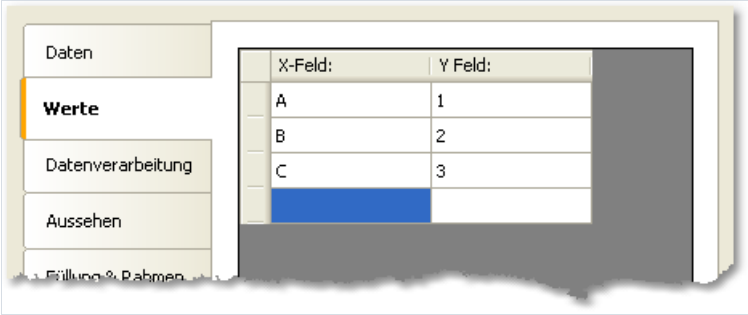
· Auswahl der Datenspalten für alle Reihenwerte. Abhängig von der Reihenart, haben diese zwei oder mehr Werte. Die meisten Reihenarten weisen zwei Werte auf - einen X-, und einen Y-Feld:



· Festlegung des Ausdrucks des Datenfilters, falls notwendig. Dieser Filter wird nur auf aktuelle Reihen angewandt;

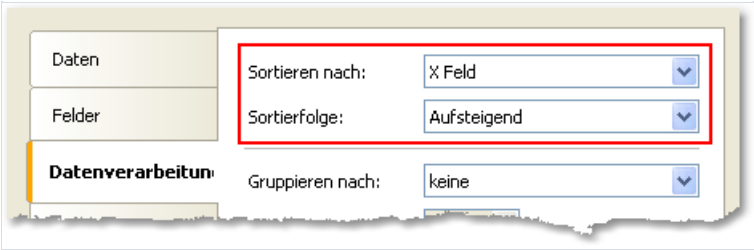
· in der Steuerung der "Farbe" lässt sich eine Datenspalte angeben, die einen Farbwert zurück liefert.

Hier kann auch eine Liste an Werten für diese Reihen geliefert werden. Für einen solchen Fall wird die Datenverbindung nicht benötigt. Um dies zu tun wird eine Reihe aus der Reihenliste ausgewählt und zur Registerkarte "Werte" gewechselt. Dann wird die Tabelle mit Werten gefüllt:


Daten
Werte
Datenverarbeitung
Aussehen
X-Feld:
Y Feld:
A
1
B
2
C
3

Sortierung der Daten

Standardmäßig zeigt das "Diagramm"-Objekt die Daten in natürlicher Reihenfolge an. Die Sortierrichtung kann geändert werden; hierzu wird die Reihe aus der Reihenliste ausgewählt und zur Registerkarte "Datenverarbeitung" gewechselt



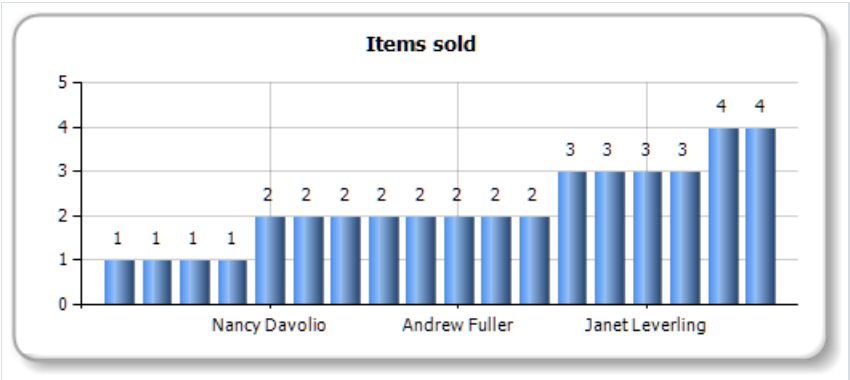
Es kann eine der Sortierrichtungen ausgewählt werden - nicht sortieren , nach X-Feld sortieren und nach Y-Feld sortieren.

Gruppierung der Daten

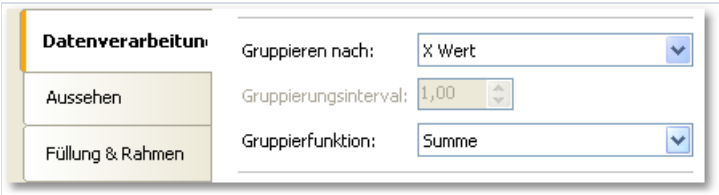
Sind alle Reihen mit Daten angefüllt, die einen identischen X-Werte aufweisen, kann dies beizeiten problematisch sein. Die MatrixDemo, die zur Demonstration der Diagramme genutzt wird, hat beispielsweise die folgenden Daten:

Name	Jahr	Monat	VerkaufteArtikel	Umsatz
Andrew Fuller	2002	1	2	1800
Andrew Fuller	1999	10	2	1900
Andrew Fuller	1999	11	2	2000
Andrew Fuller	2000	2	2	2100
Janet Leverling	1999	10	3	3000
Janet Leverling	1999	11	3	3100
Janet Leverling	2000	3	3	3200
...				

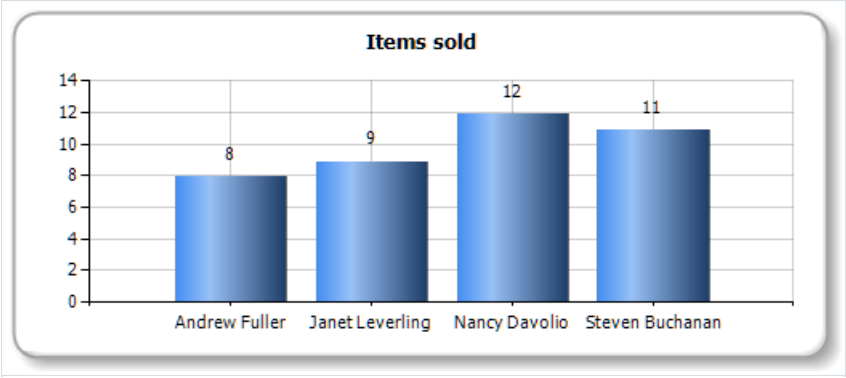
Bei der Erstellung eines Diagramms auf der Basis dieser Daten (z.B. Verkauf der Mitarbeiter: X-Wert ist die Namensspalte und Y-Wert ist die Spalte VerkaufteArtikel) erhalten wir das folgende falsche Ergebnis:



Hier müssen die Mitarbeiter in einen Wert gruppiert werden. Um dies zu tun wird eine Reihe aus der Reihenliste ausgewählt und zur Registerkarte "Datenverarbeitung" gewechselt. Auswahl der Gruppenart "X Wert" und Auswahl der Gruppenfunktion "Summe":



Als Ergebnis werden alle identischen Mitarbeiter in einen Wert gruppiert und deren Verkäufe zusammengefasst. Hier wird das Folgende dargestellt:

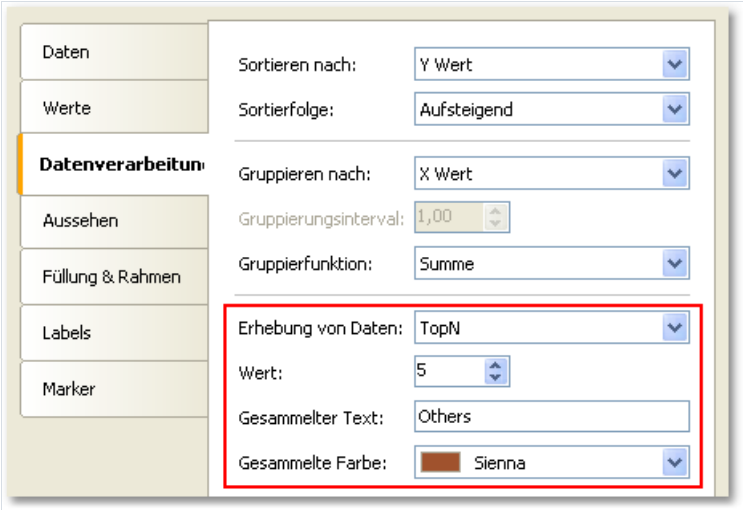

Items sold
14
12
10
8
6
4
2
0
8
9
12
11
Andrew Fuller
Janet Leverling
Nancy Davolio
Steven Buchanan

Datensammlung

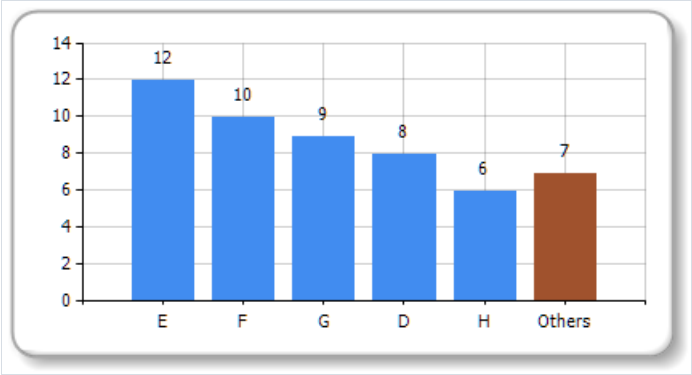
Das Instrument zur Datenbearbeitung ermöglicht die Sammlung mehrerer Reihenwerte in einen Wert. Es stehen die folgenden Algorithmen zur Verfügung:

Algorithmus	Beschreibung
TopN	Nur die höchsten N Werte werden angezeigt. Alle anderen Werte werden gesammelt und als "andere" Werte angezeigt (die Bezeichnung für diesen Wert kann frei gewählt werden).
BottomN	Die unteren N Werte werden angezeigt. Ist der Text für diesen Sammelwert nicht festgelegt, wird dieser Wert nicht angezeigt.
Less than value	Reihenwerte, die geringer als der festgelegte Wert sind, werden gesammelt und als "andere" Werte angezeigt.
Less than percent	Reihenwerte, die geringer als das festgelegte Prozent sind, werden gesammelt und als "andere" Werte angezeigt.
Greater than value	Reihenwerte, die größer als der festgelegte Wert sind, werden gesammelt und als "andere" Werte angezeigt.
Greater than percent	Reihenwerte, die größer als das festgelegte Prozent sind, werden gesammelt und als "andere" Werte angezeigt.

Um beispielsweise die ersten 5 Werte anzuzeigen wird die Reihe auf folgende Art festgelegt:

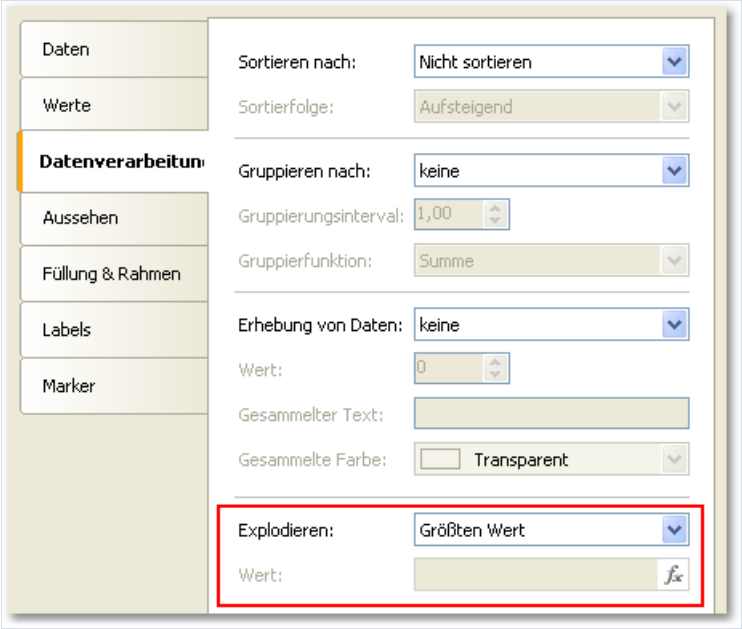


Das folgende Ergebnis wird angezeigt:


14
12
10
8
6
4
2
0
12
10
9
8
6
7
E
F
G
D
H
Others

Exploding the values

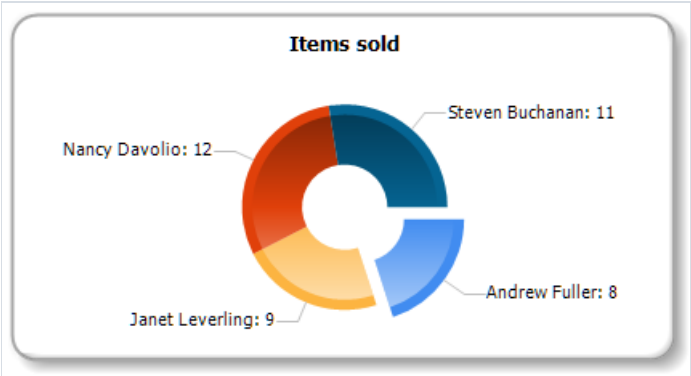
Für kuchenartige Reihen können die Werte explodiert werden. Um dies zu tun wird eine Reihe aus der Reihenliste ausgewählt und zur Registerkarte "Datenverarbeitung" gewechselt:



Eine der folgenden Explode-Arten stehen zur Auswahl: höchster Wert, niedrigster Wert und bestimmter Wert. Für den letzte Modus muss ein Wert, der explodiert werden soll, festgelegt werden. Hierbei kann es sich um einen Ausdruck handeln (siehe auch das Kapitel "Ausdrücke" für Details). Soll beispielsweise der Wert Andrew Fuller aufgelöst werden, wird der folgende Ausdruck genutzt:

```
"Andrew Fuller"
```

Das folgende Ergebnis wird angezeigt:



Einstellen von Auto-Reihen

Die Diagramme können so eingestellt werden, dass neue Datenreihen, abhängig von den Daten in den Datenquellen, automatisch erstellt werden. Zur Einstellung von Auto-Reihen sollte das Folgende durchgeführt werden:

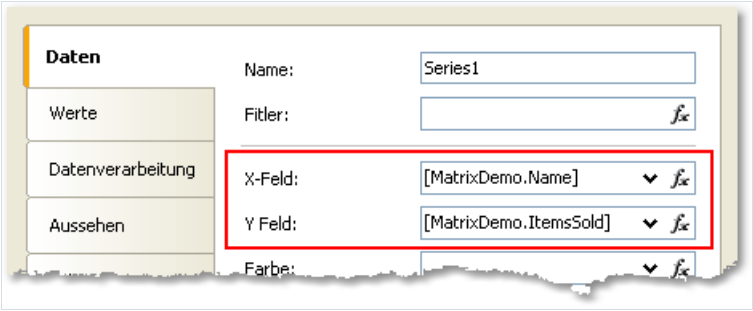
· Erstellen einer Reihe und Festlegen seiner Eigenschaften. Diese Reihe wird als Vorlage für alle weiteren, neuen Reihen genutzt;

· Auswahl des "Diagramm"-Objektes und Festlegung der Datenspalte Auto-Reihe. Der Wert dieser Spalte wird ein Name der neuen Reihe. Sollte keine Reihe mit einem solchen Namen existieren, wird eine neue Reihe hinzugefügt.

Lassen Sie uns demonstrieren, wie man eine neue Auto-Reihe erstellt. Um ein Diagramm der jährlichen Mitarbeiterverkäufe zu erhalten, wird hierzu die MatrixDemo Datentabelle genutzt. Eine Datenreihe repräsentiert ein Jahr. Folgende Schritte sind notwendig:

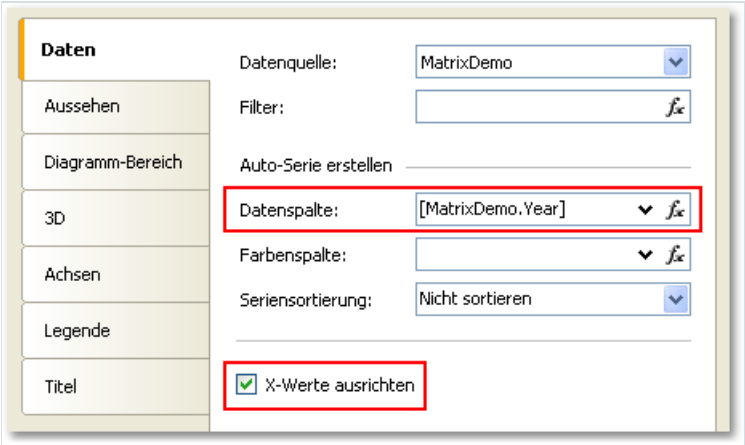
· Verbindung des Diagramms mit der MatrixDemo Datenquelle;

· Erstellen einer Reihe und Einstellen seiner Daten.



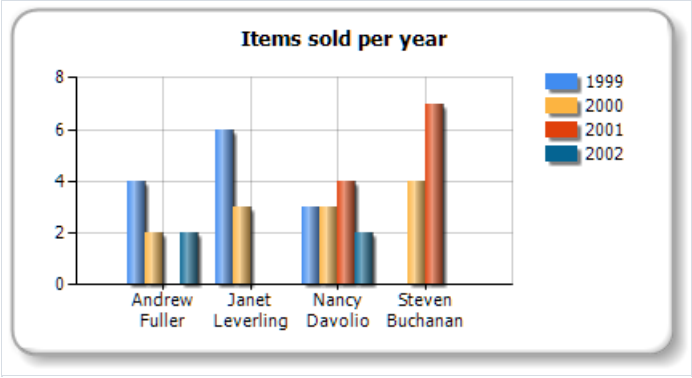
· Markierung des Kontrollkästchens "Nach X Wert gruppieren" in der Registerkarte "Datenverarbeitung". Dies ist notwendig, da die Datenquelle mehrere Mitarbeiter mit dem gleichen Namen aufweist.

· Auswahl des Diagramms aus der Liste der Datenreihen und festlegen der Spalte Auto-Reihe in der Registerkarte "Daten":



· Die Reihen können eine unterschiedliche Anzahl von Werten aufweisen (da einige Mitarbeiter in diesem Jahr keine Verkäufe getätigt haben). Zur Ausrichtung der Reihenwerte wird das Kontrollkästchen "X-Werte ausrichten" genutzt.

Das folgende Ergebnis wird angezeigt:


Items sold per year
8
6
4
2
0
Andrew Fuller
Janet Leverling
Nancy Davolio
Steven Buchanan
1999
2000
2001
2002

Interaktive Diagramme

Die Diagramme, wie andere Berichtsobjekte auch, können interaktiv gestaltet werden. Die Diagramme können so angelegt werden, dass beim Klicken eines der Werte, ein weiterer (detaillierter) Bericht ausgeführt und angezeigt wird. Um dies zu tun, muss die Eigenschaft "Hyperlink" wie im Kapitel "Interaktive Berichte" beschrieben, eingestellt werden. Beim Klicken des Elements wird das Diagramm den Wert von selbst an einen Hyperlink weitergeben.

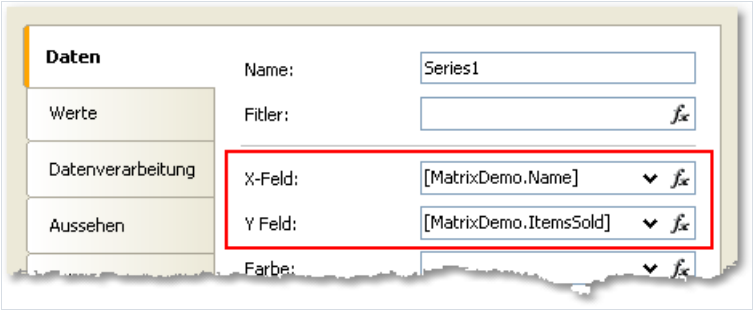
Schauen wir uns den Bericht "Diagramme/Interaktives Diagramm" im FastReport Demonstrationsprogramm an.

Es wird ein Bericht mit zwei Seiten erstellt. Die erste Seite beinhaltet ein Diagramm, die zweite Seite beinhaltet einen Detailbericht, der beim Klick eines Diagrammwertes angezeigt wird.

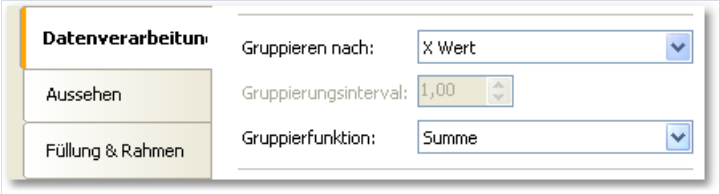
Platzieren Sie das "Diagramm"-Objekt auf der ersten Berichtsseite und legen Sie die Eigenschaften im Diagramm Editor fest:

· Auswahl des "Diagramm" Elements aus der Liste der Datenreihen und Auswahl der Datenquelle MatrixDemo;

· Auswahl der Datenreihe aus der Liste der Datenreihen und festlegen der X-, und Y-Felder: X = [MatrixDemo.Name], Y = [MatrixDemo.ItemsSold]:



- Wechsel zur Registerkarte "Datenverarbeitung" und Auswahl der Gruppenart "X Wert":



Auf der zweiten Berichtsseite wird die Listenart Bericht erstellt:

· Erstellen eines neuen Berichtsparameter mit dem Namen "SelectedEmployee" im "Daten"-Fenster;

· Erstellen des folgenden Berichtslayouts;

- Hierzu muss der Datenleisten Editor aufgerufen und die folgende Filterbedingung angezeigt werden:

```
[MatrixDemo.Name] == [SelectedEmployee]
```

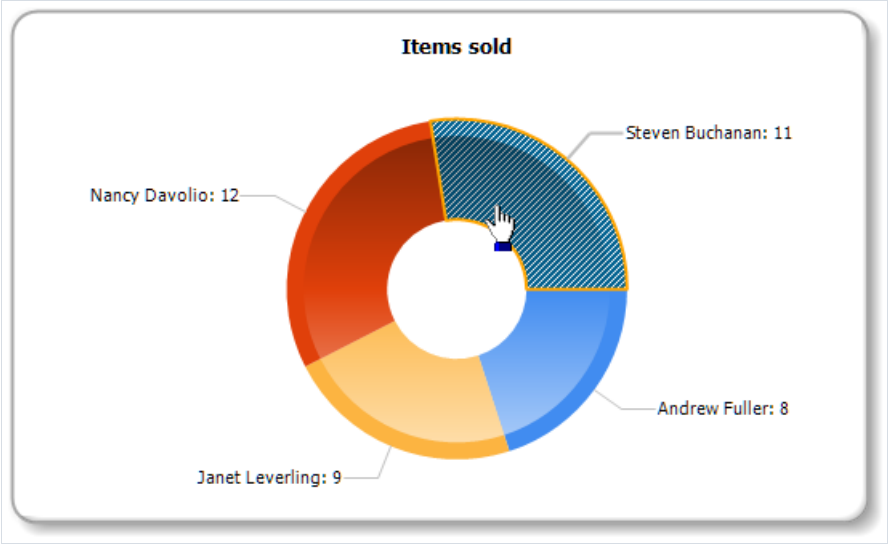
Hiernach wird der Hyperlink des "Diagramm"-Objektes festgelegt:

· Auswahl "Hyperlink" im Kontextmenü des "Diagramm"-Objektes;

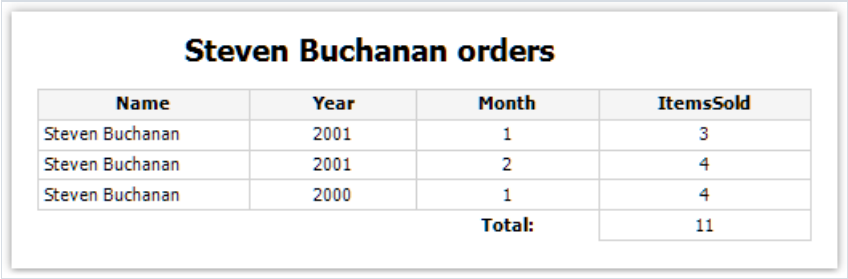
· Auswahl der Hyperlinkart: "Berichtseite";

· Auswahl der zweiten Berichtsseite und Anzeige des Prameternamens: SelectedEmployee.

Der Bericht ist erstellt. Führen Sie den Bericht durch und bewegen Sie die Maus auf einen beliebigen Diagrammwert. Dieser Wert wird visuell ausgewählt und der Zeiger der Maus ändert seine Form:



Wird der Wert geklickt sieht man den folgenden, detaillierten Bericht:



Steven Buchanan orders

Name	Year	Month	ItemsSold
Steven Buchanan	2001	1	3
Steven Buchanan	2001	2	4
Steven Buchanan	2000	1	4
		Total:	11

Berichte mit Karten

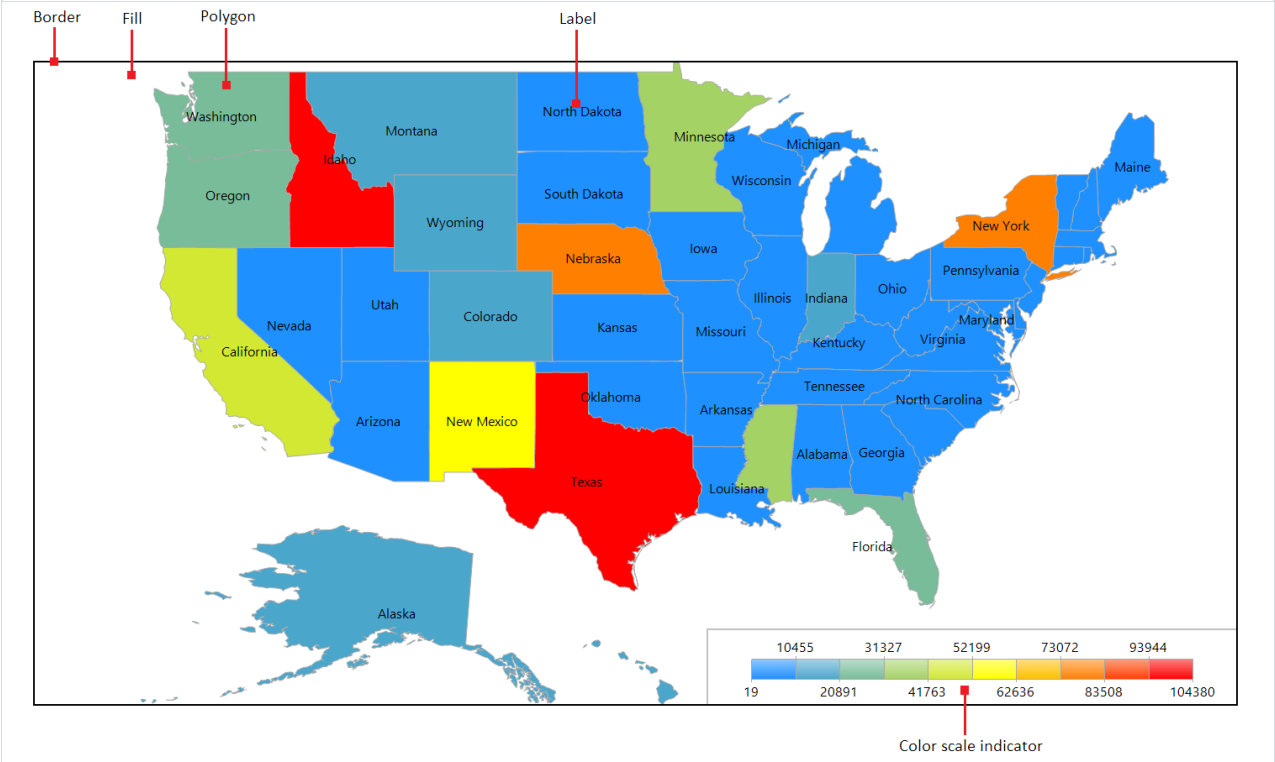
Die Komponente "Kartenobjekt" (MapObject) dient dazu, zweidimensionale grafische Karten im Format ESRI Shapefile darzustellen. Einzelheiten über dieses Format finden Sie hier:

http://de.wikipedia.org/wiki/Shapefile

Für die Arbeit benötigt man die Dateien .shp (Geometrische Daten) und .dbf (Beschreibungen).

Die Elemente der Karte

Das Objekt "Kartenobjekt" besteht aus folgenden Elementen:



Ein Kartenobjekt kann eine oder mehrere Ebenen darstellen. Jede Ebene enthält eine eigene Karte.

Bedienung der Anzeige

Im Gestaltungsmodus sowie im Fenster des fertigen Berichts können Sie die Anzeige der Karte mit der Maus bedienen:

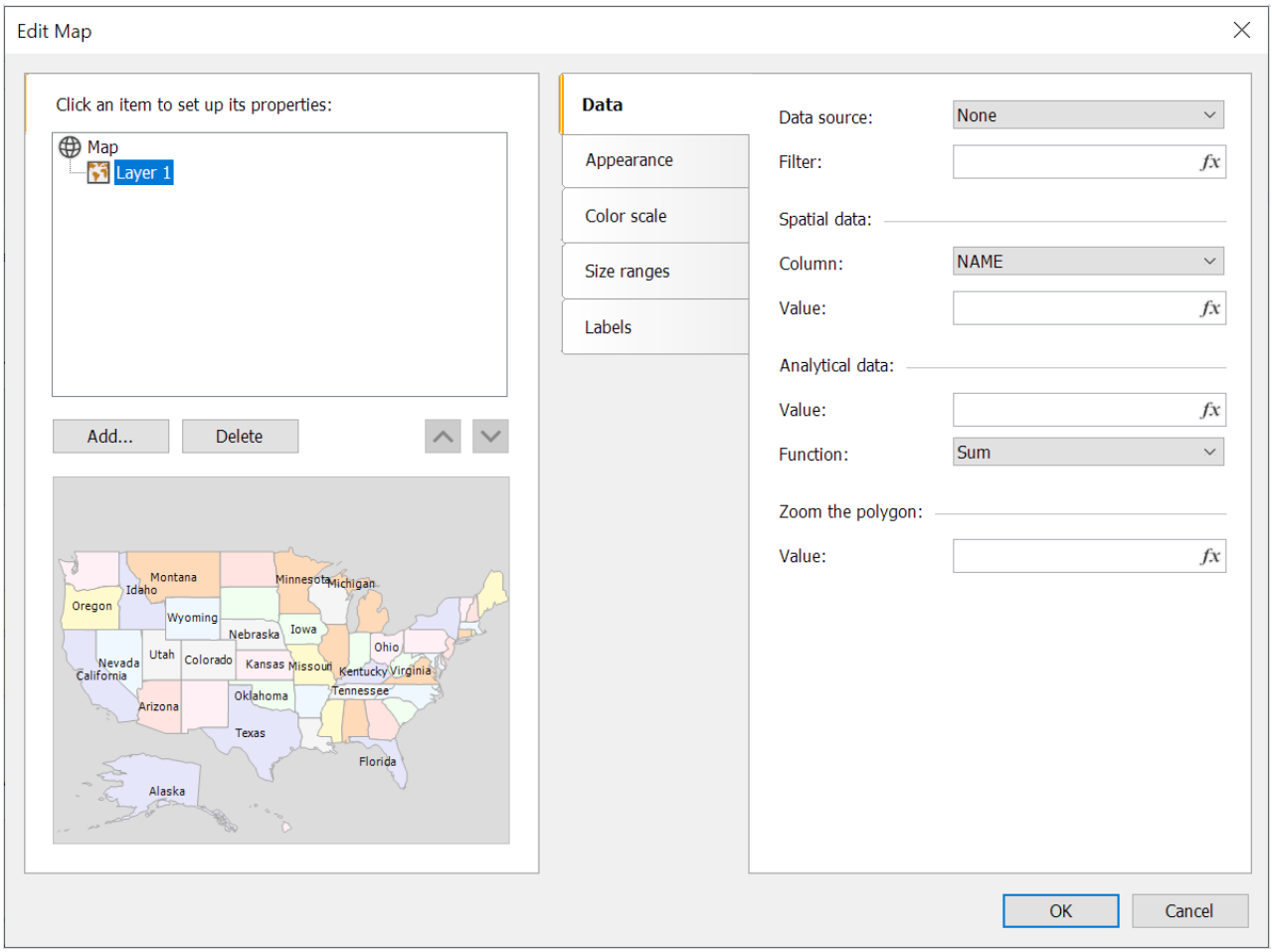
- Das Mausrad steuert den Maßstab;
- mit dem Druck auf die linke Maustaste kann man die Karte verschieben;
- Durch einen Klick auf ein Polygon kann man seine Eigenschaften im "Eigenschaften"-Fenster einstellen.

Der kleinste und der größte Maßstab werden in den Eigenschaften MinZoom und MaxZoom definiert. Diese Werte lassen sich im "Eigenschaften"-Fenster einstellen.

> In asp.net kann die Karte nicht bedient werden.

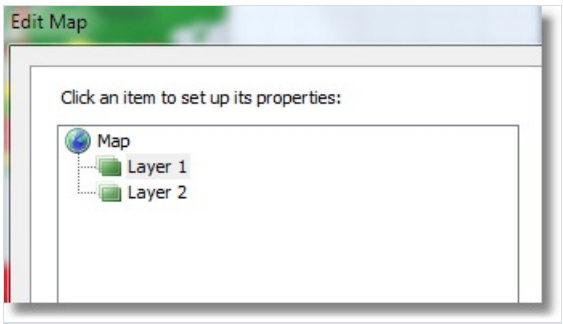
Karten-Editor

Das Kartenobjekt enthält eine Vielzahl an Einstellungen, die man im Objekt-Editor bearbeiten kann. Dafür klicken Sie doppelt auf das Objekt bzw. wählen Sie den Punkt "Bearbeiten..." in seinem Kontextmenü:

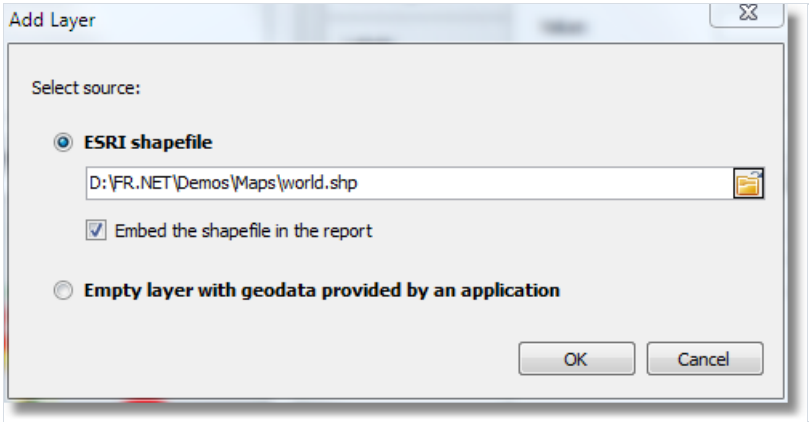


Ebenen hinzufügen

Das Kartenobjekt kann eine oder mehrere Ebenen enthalten. Die Liste der Ebenen wird in der linken oberen Ecke des Editors dargestellt:



Zum Hinzufügen einer neuen Ebene klicken Sie auf den Button "Hinzufügen...". Das folgende Fenster wird angezeigt:



Bei diesem Schritt muss man den Typ der Ebene wählen:

- Karte aus einer Shapefile-Datei (.shp/.dbf). Das ist die am meisten verwendete Kartenart. Beispielsweise können Sie die Weltkarte ausdrucken und die Länder farblich markieren, in denen die Verkäufe einen bestimmten Wert überschritten haben;
- Kartographische Daten aus einer Anwendung. Ihre Anwendung muss geographische Koordinaten liefern (ein Wertepaar der Länge und Breite), die als Punkt auf der Karte dargestellt werden. Der Punkt kann eine Überschrift sowie verschiedene Größen und Farben haben, je nach bestimmten Datenangaben. In der Regel verwendet man diese Ebenenart als zweite Ebene (die erste, die die Basis darstellt, wird einer Shapefile-Datei entnommen). Beispielsweise kann die Basisebene eine Landkarte darstellen, und die zweite Ebene enthält die Punkte mit den Städtenamen, in denen Verkäufe getätigt wurden. Die Größe und die Farbe der Punkte kann man so einstellen, dass sie mit der Anzahl der Verkäufe in der jeweiligen Stadt korellieren.

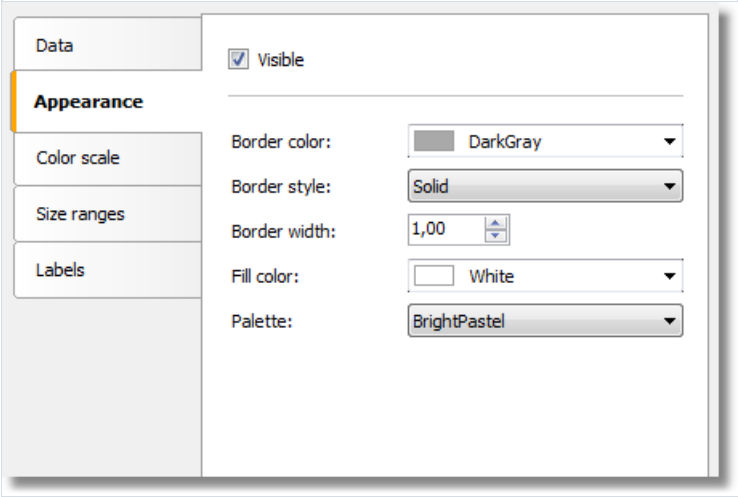
Wenn Sie eine Ebene auf der Grundlage einer Shapefile-Datei ausgewählt haben, können Sie zudem definieren, wie kartographische Daten gespeichert werden:

- Die Daten werden in die Berichtsdatei eingefüg. Dabei kann die Größe der Berichtsdatei sehr groß werden.
- Die Berichtsdatei referenziert Shapefile-Dateien nur, sie werden nicht eingebettet. Dieser Modus macht Sinn, wenn Sie mehrere Berichte haben, die alle die gleichen Karten verwenden.

> Karten mit großem Umfang (über 30MB) oder mit vielen Polygonen (über 20000) können die Arbeit mit dem Bericht stark verlangsamen.

Ansichtseinstellungen

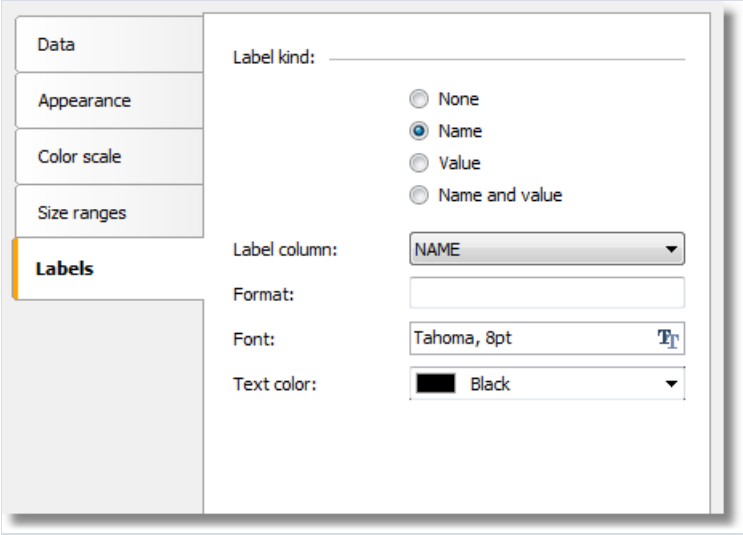
Die Erscheinung der Ebene kann man einstellen, indem man sie auswählt und den Reiter "Aussehen" anwählt:



Hier kann man die Farbe und den Stil der Polygonränder einstellen sowie die Farbpalette auswählen. Wenn Sie die Färbung der Polygone in Abhängigkeit von analytischen Daten eingerichtet haben (dazu später mehr), wird die Palette ignoriert.

Einstellung der dargestellten Werte

Man kann Überschriften auf der Karte darstellen, z.B., Ländernamen auf einer Weltkarte. Den Typ und das Aussehen der Überschriften können Sie im Reiter "Überschriften" angeben:



Wenn die gewählte Ebene eine Karte aus einer Shapefile-Datei ist, muss man das Feld angeben, aus dem die Überschrift entnommen wird. In der Regel ist es das Feld "NAME". Für die Weltkarte, die im Demoprogramm FastReport enthalten ist, kann man folgende Felder verwenden:

- NAME (z.B. Germany)
- ABBREV (z.B. Ger.)
- ISO_A2 (z.B. DE)
- ISO_A3 (z.B. DEU)

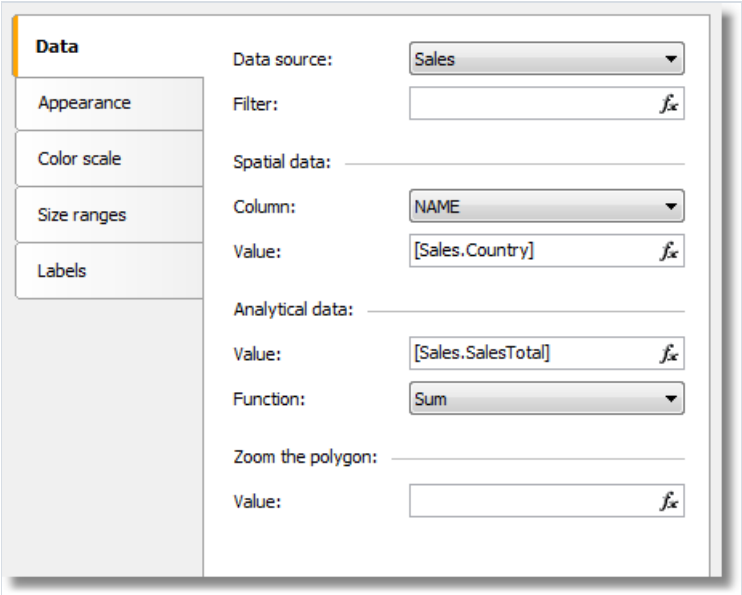
Für andere Karten kann die Liste der Felder anders sein.

Wenn kartographische Daten aus einer Anwendung als Ebenenart angegeben sind, kann man in diesem Fenster die untere Schwelle angeben, bei der die Überschriften angezeigt werden. Dieser Wert beträgt standardmäßig "-1", was bedeutet, es werden immer die Überschriften angezeigt.

Datenverbindung

Die meisten Berichte verwenden das Kartenobjekt nicht an und für sich, sondern zum Darstellen von analytischen Informationen. Das kann zum Beispiel der Umfang der Verkäufe in verschiedenen Ländern sein. Dafür muss eine Ebene an eine Datenquelle angebunden werden. Das kann man im Karteneditor machen, indem man die Ebene auswählt und den Reiter "Daten" anwählt. Die Anbindung an die Daten unterscheidet sich je nach Ebenenart (ob es eine Shapefile-Datei ist oder Geodaten aus einer Anwendung sind):

- Wenn die Ebenenart als Karte aus einer Shapefile-Datei angegeben ist, sieht der "Daten"-Reiter so aus:



In diesem Fall muss die Anwendung folgende Daten bereitstellen:

- Name (z.B., Ländername);
- Zahlenwert (z.B., die Verkäufe im gegebenen Land).

Nehmen wir an, Sie haben eine Tabelle "Sales" mit folgenden Feldern und Daten:

Country SalesTotal

------- ----------

USA 500000

Germany 1200000

Russia 300000

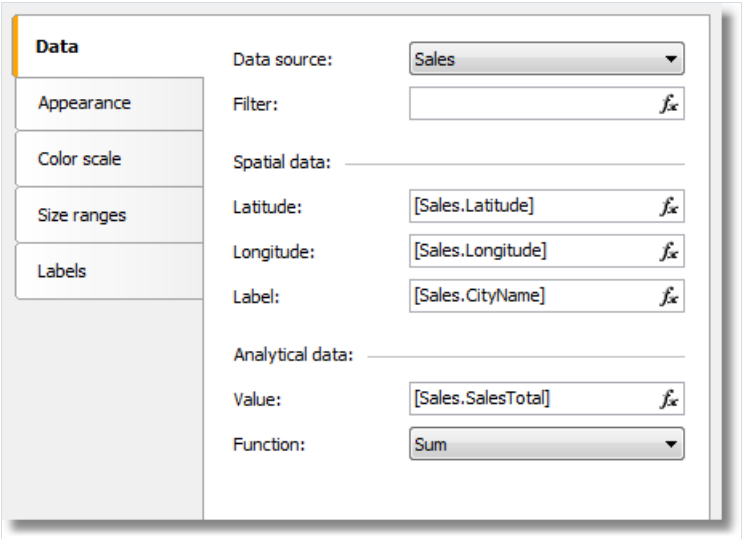
Dann kann man die Datenverbindung wie folgt einstellen:

- Datenquelle: Sales
- Geographische Daten, Feld: wählen Sie das Feld aus, das in der Shapefile-Datei den Namen des Landes enthält. In der Regel ist es das Feld "NAME".
- Geographische Daten, Wert: [Sales.Country]
- Analytische Daten, Wert: [Sales.SalesTotal]

- Analytische Daten, Funktion: "Summe". Die Funktion wird verwendet, wenn für ein Land mehrere Einträge mit verschiedenen Werten enthalten sind.

Das Feld "Polygon vergrößern" erlaubt, ein bestimmtes Polygon auf die ganze Fläche des Kartenobjekts zu bringen. Beispielsweise, wenn Sie eine Weltkarte haben und nur Russland darstellen wollen, geben Sie in diesem Feld den Namen "Russia" an (in Anführungsstrichen).

- Wenn die Ebenenart als Daten, die aus einer Anwendung kommen, eingestellt ist, sieht der Reiter "Daten" wie folgt aus:



In diesem Fall muss die Anwendung folgende Daten bereitstellen:

- Geographische Daten: Länge und Breite;
- Name (z.B., Name einer Stadt);
- Zahlenwert (z.B. die Verkäufe in einer gegebenen Stadt).

Nehmen wir an, Sie haben eine Tabelle "Sales" mit folgenden Feldern und Daten:

Latitude Longitude CityName SalesTotal

-------- --------- -------- ----------

48.13641 11.57753 München 50000

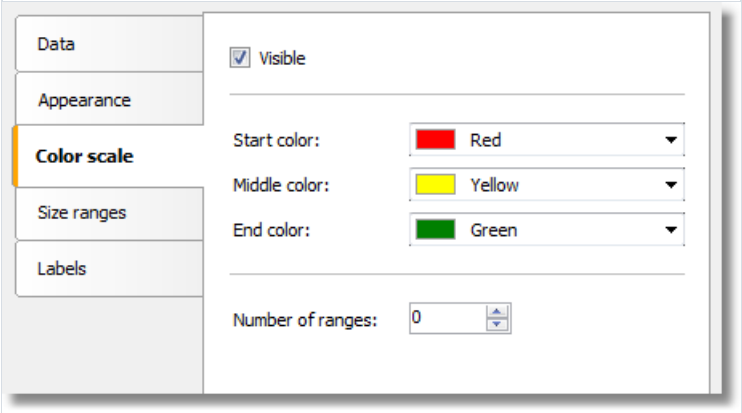
50.94165 6.95505 Köln 36000

In diesem Fall stellen Sie die Daten wie folgt ein:

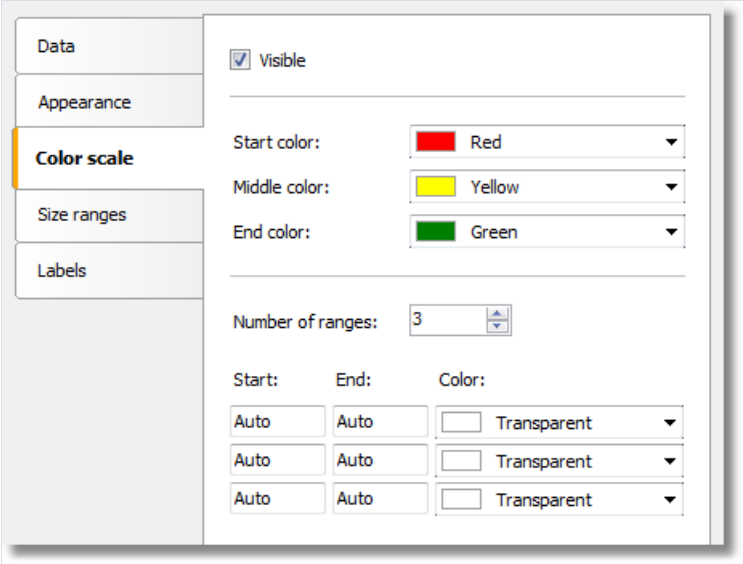
- Datenquelle: Sales
- Geographische Daten, Breite: [Sales.Latitude]
- Geographische Daten, Länge: [Sales.Longitude]
- Geographische Daten, Überschrift: [Sales.CityName]
- Analytische Daten, Wert: [Sales.SalesTotal]
- Analytische Daten, Funktion: "Summe". Die Funktion wird verwendet, wenn für eine Stadt mehrere Einträge mit verschiedenen Werten enthalten sind.

Farbliche Hervorhebung von Daten

Nachdem eine Ebene mit einer Datenquelle verbunden ist, stellt sich die Frage, auf welche Weise man analytische Informationen darstellt (z.B. Verkaufszahlen in verschiedenen Ländern). Die einfachste Variante ist, neben den Überschriften zusätzlich auch die Verkaufszahlen auszugeben (vgl. den Abschnitt "Einstellungen der angezeigten Werte") Doch die anschaulichste Variante wäre, die Länder farblich zu differenzieren, je nach Verkaufszahlen. Dafür muss man eine Farbskala einrichten. Das kann man auf dem Reiter "Farbskala" tun:



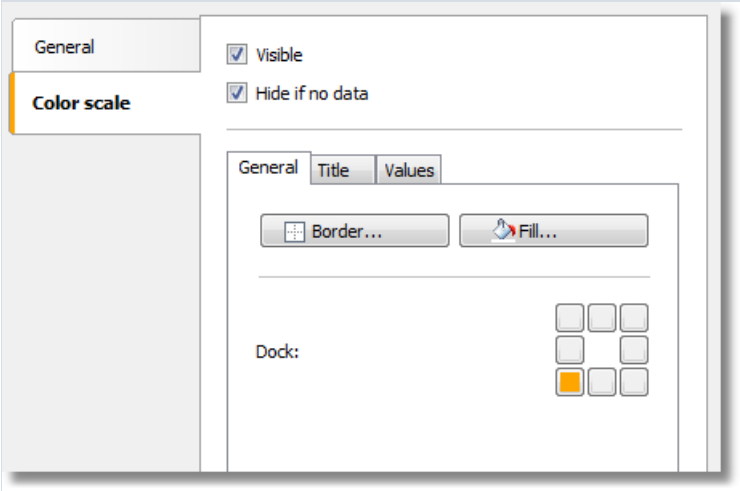
In der Farbskala wird eine Menge von Werten definiert: Mindestwert, Höchstwert, Farbe. Man kann mehrere solche Sets (Bereiche) haben. Zum Einstellen der Farbskala muss man angeben, wie viele solche Bereiche sie besitzt. Danach kann man die Mindest- und die Höchstwerte für jeden Bereich sowie die Farbe einstellen:



Standardmäßig sind alle Werte als "Auto" gekennzeichnet. In diesem Fall berechnet FastReport die Mindest- und die Höchstwerte für jeden Bereich automatisch; die Farben werden aus den Voreinstellungen "Anfangsfarbe", "Mittelfarbe" und "Endfarbe" entnommen. Diesen Modus kann man für die meisten Fälle verwenden.

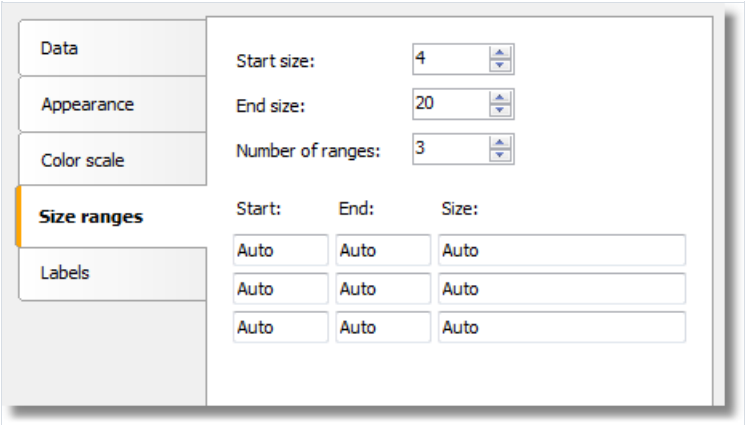
Wenn die Farbskala eingerichtet ist, erscheint im unteren Teil der Karte eine Legende: ein Streifen aus mehreren Rechtecken von verschiedenen Farben:

Das Aussehen und die Lage der Legende kann man einstellen, indem man in der Ebenenliste den Eintrag "Karte" auswählt und den Reiter "Farbskala" wählt:



Hervorhebung der Daten durch Größe

Wenn die Ebenenart als Daten, die durch eine Anwendung bereitgestellt werden, spezifiziert wurde, dann werden diese Daten als ein Punkt mit einer Überschrift abgebildet. Die Größe des Punkts kann man ungefähr genauso an die Daten anbinden, wie es mit der Farbe passiert. Das kann man auf dem Reiter "Größenskala" tun:



Die Größenskala besteht aus einer Menge von Werten: Minimaler Wert; Maximaler Wert; Größe in Pixel. Man kann mehrere solche Bereiche haben. Um die Skalen einzurichten, stellt man die Anzahl der Bereiche ein, die sie enthält. Danach kann man die Mindest- und Höchstwerte für jeden Bereich sowie die Größe der Punkte einrichten.

Standardmäßig sind alle Werte als "Auto" gekennzeichnet. In diesem Fall berechnet FastReport die Mindest- und die Höchstwerte für jeden Bereich automatisch; die Größen werden aus den Voreinstellungen "Anfangsgröße" und "Endgröße" entnommen.

Daten

Alle Berichte drucken Daten. Die folgenden Daten können in FastReport bearbeitet werden:

· Datenquellen;

· Systemvariablen;

· Summenwerte;

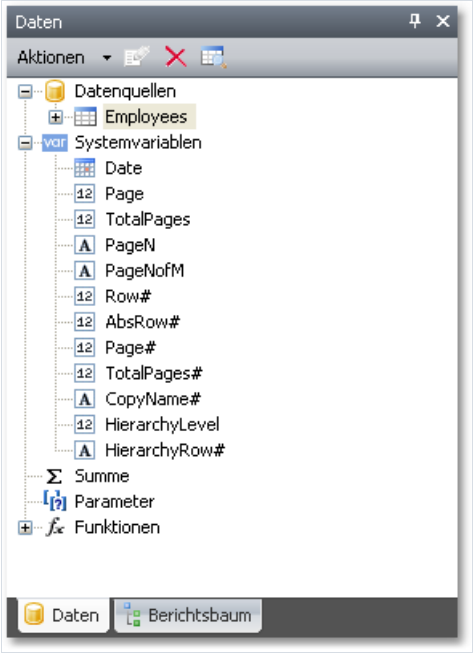
· Bericht Parameter;

· Ausdrücke, die die oben erwähnten Daten beinhalten.

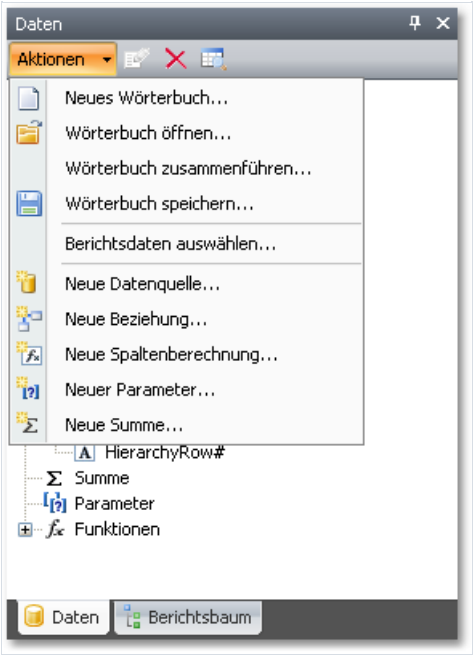
In diesem Kapitel schauen wir uns an, wie man mit diesen Daten arbeitet.

Das "Daten"-Fenster

Alle Daten sind über das "Daten"-Fenster zugänglich. Dieses Fenster kann durch Auswahl des Menüs "Daten|Anzeige des Datenfensters" angezeigt werden.



Das "Daten"-Fenster ermöglicht die Arbeit mit allen Datenelementen und erlaubt diese auf die Berichtsseite zu ziehen. Alle Operationen können mittels der Symbolleiste und über das Menü "Aktionen" durchgeführt werden.



Ein Teil dieser Operationen befinden sich auch im Kontextmenü des "Daten"-Fensters. Wird beispielsweise eine Datenquelle ausgewählt kann das Kontextmenü genutzt werden, um eine berechnete Spalte zu erstellen, eine Datenquelle zu löschen oder deren Daten anzuzeigen.

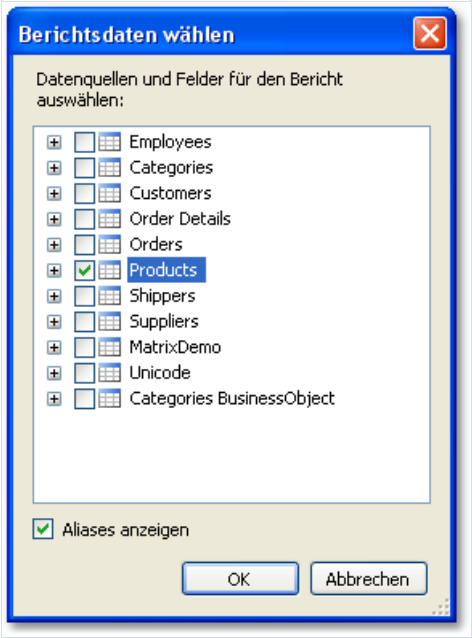
Datenquellen

Üblicherweise repräsentiert die Datenquelle eine DB-Tabelle oder eine SQL-Abfrage. Ein Bericht kann mehrere Datenquellen haben. Für die meisten Berichte wird lediglich eine Datenquelle benötigt. Ein Master-Detail Bericht benötigt zwei Datenquellen die miteinander über eine Relation verbunden sind (hierzu später mehr).

Eine Datenquelle hat eine oder mehrere Datenspalten. Jede Spalte hat eine bestimmte Datenart. Zur Anzeige der Spaltenart muss diese ausgewählt und das Fenster "Eigenschaften" geöffnet werden. Die Spaltenart ist in der Eigenschaft "Datenart" angezeigt. Auch das Symbol nahe des Spaltennames hilft bei der Bestimmung der Spaltenart.

Um eine Datenquelle für einen Bericht zu definieren, können zwei Methoden herangezogen werden:

Die erste Methode: Die Datenquelle wird in der Anwendung festgelegt und in einem Bericht registriert. Dies bleibt dem Programmierer, der die Anwendung erstellt hat, überlassen (siehe hierzu auch Details im Handbuch für Programmierer). Ein Benutzer sollte die benötigte Datenquelle lediglich auswählen, um diese in einem Bericht zu nutzen. Dies kann im Menü "Daten|Berichtsdaten auswählen..." stattfinden.



Alle in einem Bericht registrierten Daten sind in diesem Fenster aufgelistet. In einem Bericht benötigte Daten können angekreuzt werden. Dies kann jederzeit während der Arbeit mit dem Bericht geschehen.

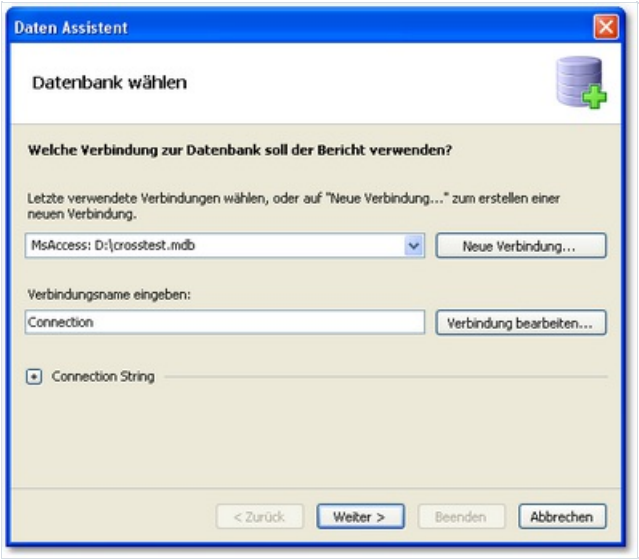
Die zweite Methode: Sie erstellen die neue Datenquelle selbst. Dabei kann es sich sowohl um eine DB-Tabelle, wie auch um eine SQL-Abfrage handeln. In einem solchen Fall wird die Definition der Datenquelle in der Berichtsdatei gespeichert.

FastReport ermöglicht die Verbindung mit einer Vielzahl von populären DBMS (Datenbank Management Systemen), wie MS, SQL, Oracle, Interbase, Access. Es lassen sich auch Datendateien speichern, die in xml/xsd-Format gespeichert sind.

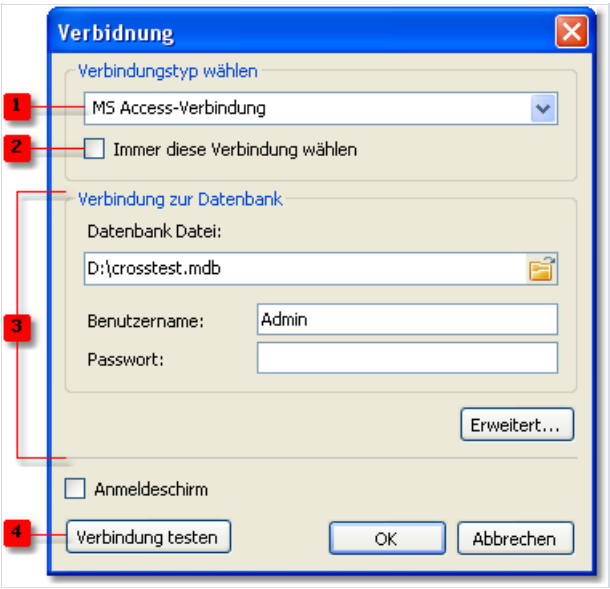
> Der Inhalt der DB-Tabelle ist nicht in der Berichtsdatei gespeichert. Stattdessen werden die Verbindungszeichenfolgen und das Datenquellenschema gespeichert. Eine Verbindungszeichenfolge kann personenbezogene Daten, wie Anmeldeinformationen und Passworte, enthalten. Daher wird sie in dechiffrierter Form in einer Berichtsdatei gehalten. Durch die Nutzung eines eigenen Schlüssels zur Datendechiffrierung kann die Sicherheit bei Bedarf erhöht werden. In einem solchen Fall kann die Berichtsdatei lediglich in dem eigenen Programm ordnungsgemäß geöffnet werden.

Erstellen einer Datenquelle

Zur Erstellung einer neuen Datenquelle, wird das Menüelement "Daten|Datenquelle hinzufügen..." ausgewählt oder die Schaltfläche "Aktionen" im "Daten"-Fenster gedrückt und das Element "Neue Datenquelle..." gewählt. Das Fenster "Daten Assistent" wird angezeigt:



Zuerst erfolgt eine Aufforderung eine Verbindung herzustellen. Hierzu wird die Schaltfläche "Neue Verbindung..." gedrückt. Es erscheint ein Fenster mit Verbindungseinstellungen:



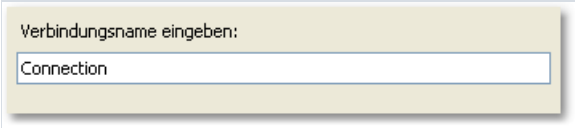
Die folgenden Elemente werden angeboten:

1. Verbindungsart;
2. Bei Aktivierung wird die ausgewählte Verbindungsart standardmäßig genutzt.
3. Verbindungseinstellungen;
4. Schaltfläche Testverbindung

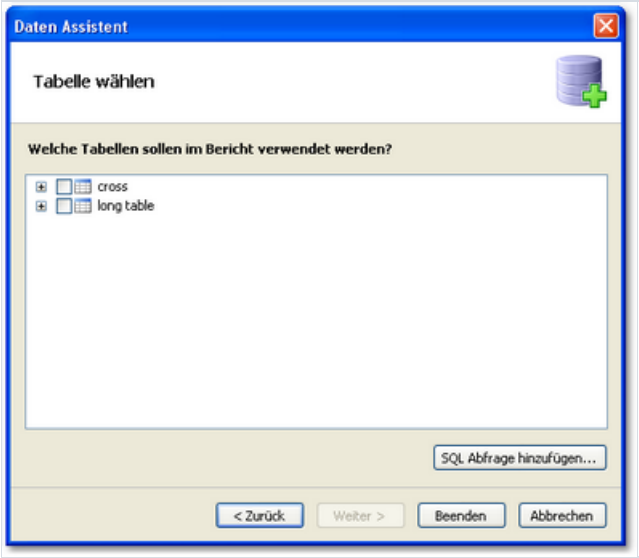
Die Verbindung mit der MS-Access-Datenbank wird auf in der Abbildung angezeigt. Wird eine andere Verbindungsart gewählt, dann wird der Bereich der Verbindungseinstellungen (3) geändert. Die Verbindung zu den Datenbanken, MS und SQL, hat beispielsweise die folgenden Einstellungen:

Auswahl der benötigten Verbindungsart und Festlegen der Parameter. Nach dem Druck der Schaltfläche OK wird das Fenster geschlossen und das Fenster des Datenassistenten angezeigt.

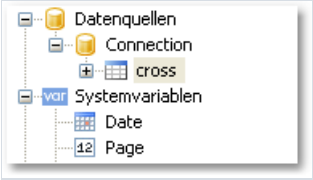
Als nächstes muss ein Verbindungsname eingegeben werden. Dieser Name erscheint im "Daten"-Fenster.



Drücken der Schaltfläche "Weter" zum Fortfahren. Hier werden Tabellen, die über die Datenbank zugänglich sind, angeboten:



Markieren der benötigten Tabellen und Schließen des Assistenten durch Klicken der Schaltfläche "Beenden". Im Datenfenster wird nun eine von Ihnen erstellte Verbindung angezeigt, die die ausgewählten Datenquellen beinhaltet:

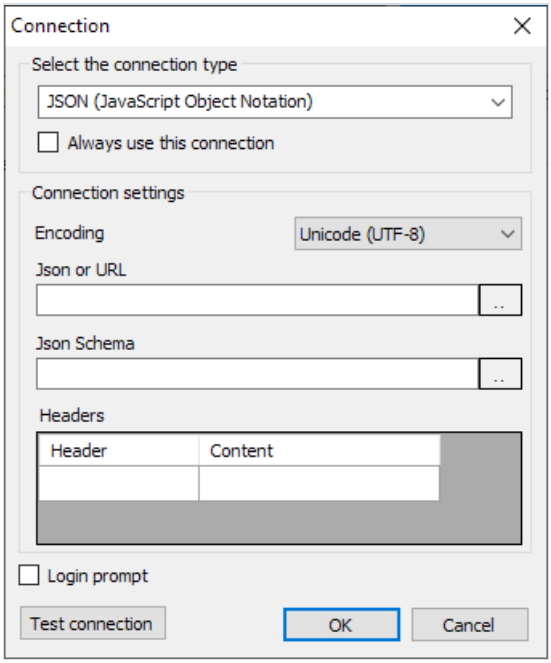

Datenquellen
Connection
cross
Systemvariablen
Date
Page

JSON-Verbindung (JavaScript Object Notation)

Diese Verbindung ermöglicht es Ihnen, eine statische JSON-Datei als Datenquelle oder eine URL-Adresse zu übergeben, von der diese Datei vor der Erstellung eines Berichts empfangen wird.

JSON-Verbindungsassistent

Wenn Sie eine neue Verbindung im FastReport-Designer erstellen, sieht sie wie folgt aus:



Parameter	Beschreibung
Encoding	legt die Codierung fest, in der die JSON-Anforderung verarbeitet wird, wenn ein Link zum Empfangen von JSON angegeben ist.
JSON or URL	legt eine statische JSON-Datei oder einen Link zum Abrufen von JSON-Daten über die API fest.
JSON Schema	legt das JSON-Schema fest.
Headers	Legt die erforderlichen HTTP-Header für die JSON-Verbindung über die API fest (optional).

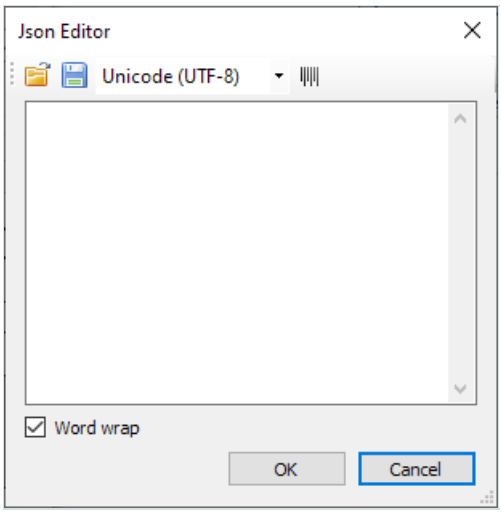
Rechts neben den Feldern befinden sich Schaltflächen zum Bearbeiten des Feldes in einem separaten Fenster.

Wenn das JSON-Schema leer ist, wird beim Klicken auf OK oder Test connection ein JSON-Schema automatisch erstellt.

Wenn Sie die JSON oder URL ändern, wenn das Schema bereits angegeben ist, wird FastReport Sie auffordern, das Schema zu aktualisieren.

JSON-Bearbeitungsfenster

Mit diesem Formular können Sie JSON bearbeiten, sie sieht wie folgt aus:



Beschreibung der Symbolleiste des Editors von links nach rechts:

1. `Datei öffnen` — ermöglicht es Ihnen, die JSON zu öffnen und den Dateiinhalt in den Editor einzufügen.
2. `Datei speichern` — ermöglicht es Ihnen, den Inhalt des Editors in einer Datei zu speichern.
3. `Codierung` — legt die Codierung fest, in der eine JSON-Datei geöffnet werden soll.
4. `Formatierung` — enthält die JSON-Formatierung sowie validiert die JSON anhand von Spezifikationen.

Als nächstes folgt das Editor-Feld, in dem Sie den JSON-Text ändern können.

Mit dem Kontrollkästchen "Word Wrap" ("Zeilenumbruch ") können Sie den Zeilenumbruch im Editor aktivieren oder deaktivieren.

Wie funktioniert die Verbindung?

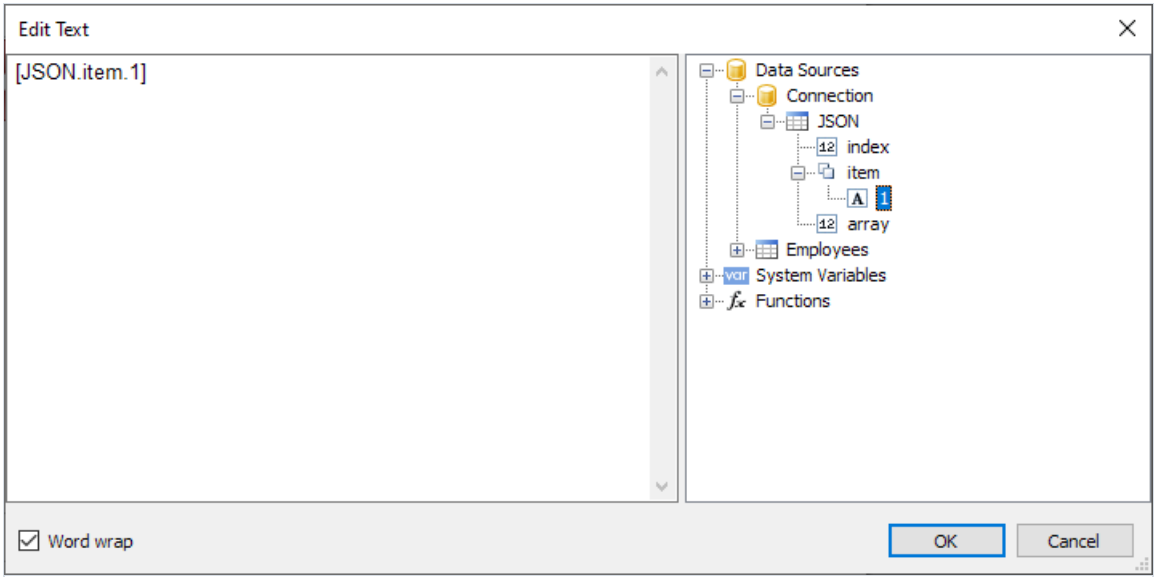
Die JSON ist keine Tabelle, daher behandelt FastReport eine JSON-Verbindung nicht als Datentabelle.

Stattdessen behandelt FastReport jedes JSON-Array als eine hierarchische Datenquelle mit drei Feldern:

Feld	Beschreibung
index	Elementnummer.
item	Element.
array	Link auf ein Array von Elementen.

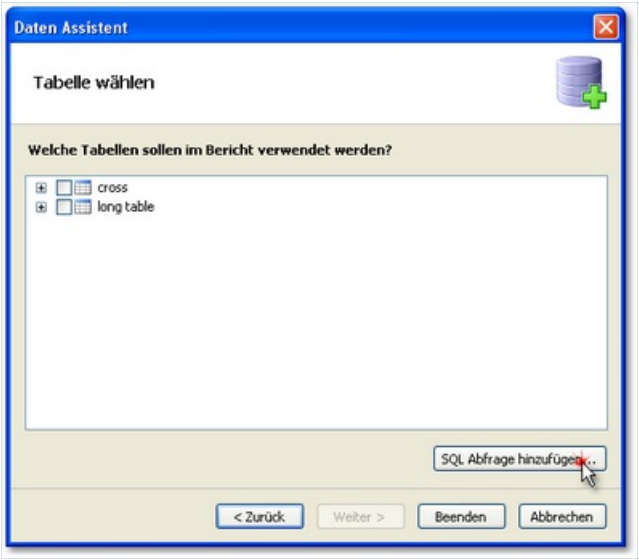
Jedes JSON-Array kann mit dem DataBand verbunden werden.

Sie können dann Felder aus JSON in Berichtsobjekten verwenden.


Edit Text
[JSON.item.1]
Data Sources
Connection
JSON
index
item
array
Employees
System Variables
Functions
Word wrap
OK
Cancel

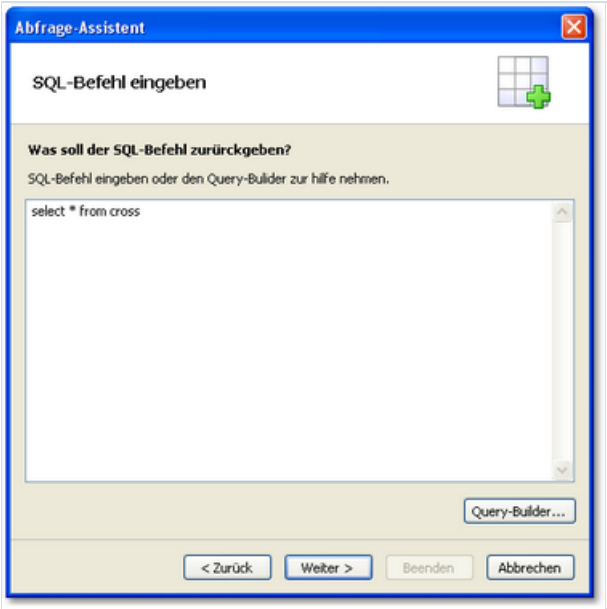
Erstellen einer SQL-Anfrage

Der Datenassistent ermöglicht die schnelle Auswahl einer Tabelle, die in einer Datenbank enthalten ist. Das Erstellen einer SQL-Abfrage erfordert zusätzliche Schritte. Hierzu wird, in einem zweiten Schritt des Assistenten, die Schaltfläche "SQL Abfrage hinzufügen..." gedrückt.



Das Fenster "Abfrage Assistent" wird angezeigt: Der Abfrage Assistent besteht aus 4 Seiten. Durch Nutzung der Schaltflächen "Weiter" und "Zurück" kann zwischen den Seiten gewechselt werden.

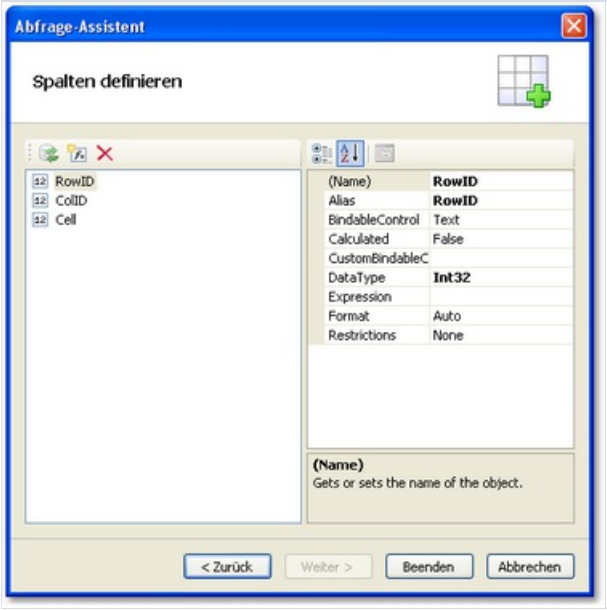
Im ersten Schritt wird der Name der Abfrage festgelegt. Dieser Name erscheint im "Daten"-Fenster. Geben Sie einen einzigartigen Namen ein und klicken Sie die Schaltfläche "Weiter".



In einem zweiten Schritt muss eine Abfrage in SQL-Sprache eingegeben werden. Es sollte die Sprache genutzt werden, die von der eigenen MSDB unterstützt wird. Der Abfrage-Builder kann zur visuellen Abfrageerstellung genutzt werden. Hierzu wird die Schaltfläche "Query-Builder" gedrückt. Der Abfrage-Builder wird später im Detail erklärt.

Nach Eingabe des Abfragetextes, wird die Schaltfläche "Weiter" gedrückt. In einem dritten Schritt werden die Abfrageparameter definiert. Dieser Schritt ist dann notwendig, wenn Ihre Abfrage Parameter aufweist. Parameter werden später in diesem Kapitel näher betrachtet.

Während des letzten Schritt des Assistenten werden die Spalten, die von der Abfrage zurückgegeben werden, festgelegt:



Sollte während der Definition des Abfragetextes oder der Parameter ein Fehler auftreten, wird diese Fehlermeldung im letzten Schritts des Assistenten angezeigt.

Generell gilt, dass es ausreicht sicherzustellen, dass die Abfrage alle benötigten Spalten zurückgegeben hat. In diesem Schritt kann Folgendes getan werden:

· Löschen unnötiger Spalten mittels der Schaltfläche "Löschen";

· Zurücksetzen der Spalten durch Drücken der Schaltfläche "Zurücksetzen";

· Hinzufügen einer berechneten Spalte durch Drücken der Schaltfläche "Berechnete Spalte hinzufügen". Es ist für eine neue Spalte notwendig, die Eigenschaften "Name", "DatenType" und "Ausdruck" festzulegen.

Durch Drücken der Schaltfläche "Beenden" wird der Assistent geschlossen und man wird zum "Daten"-Fenster zurückgeführt.

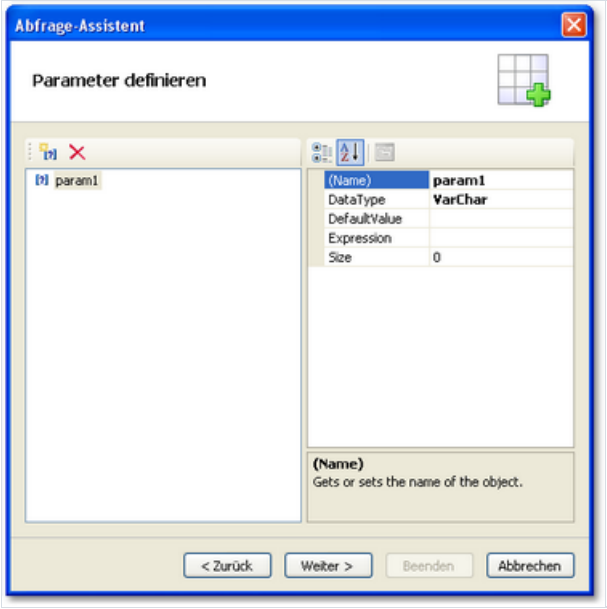
Abfrageparameter

Ein Abfragetext kann Parameter enthalten. Dies soll an dem folgenden Beispiel verdeutlicht werden:

```
select * from DVDs
where Title = @param1
```

Dies ist die Abfrage an die MS SQL-Demo-Datenbank. In dieser Abfrage ist der Parameter mit dem Namen "param1" definiert. Hier sollte folgendes beachtet werden: Die Methode zur Beschreibung von Parametern in einer Abfrage unterscheidet sich für verschiedene DBMS. Für MS und SQL wird ein Parameter durch ein "@" Symbol gekennzeichnet. MS-Access-Parameter haben keine Namen und sind mit einem "?"-Symbol gekennzeichnet.

Sollte die SQL-Abfrage Parameter beinhalten, müssen diese bezeichnet werden. Dies geschieht während des dritten Schritts, des weiter oben betrachteten "Abfrage Assistenten". Zur Erstellung eines Parameters wird die Schaltfläche "Parameter hinzufügen" gedrückt. Es wird ein neuer Parameter erstellt:



Die folgenden Parametereigenschaft sollten im Fenster "Eigenschaften" festgelegt werden:

Eigenschaft	Beschreibung
Name	Parametername. Hier muss der gleiche Name, der auch im Abfragetext genutzt wurde, angezeigt werden. Einige DBMS (z.B. MS Access) unterstützen die benannten Parameter nicht. In diesem Beispiel sollte diese Eigenschaft nicht geändert werden.
DataType	Datenart des Parameters
DefaultValue	Wert der bei einem nicht bestimmten, oder nicht berechenbaren (z.B. beim Betrieb mit einer Abfrage im Designmodus des Berichts) "Ausdruck" Eigenschaft genutzt wird.
Expression	Ausdruck der den Parameterwert zurückgibt. Dieser Wert wird bei der Durchführung des Berichts bearbeitet. In dieser Eigenschaft kann jeder Ausdruck angezeigt werden (siehe Details im Kapitel "Ausdrücke").

Eigenschaft	Beschreibung
Size	Datengröße des Parameters. Diese Eigenschaft sollte angezeigt sein, wenn der Parameter Datentyp eine "Zeichenfolge" ist.

Werden die Parametereigenschaften nicht ordnungsgemäß eingestellt, wird auf der letzten Seite des Assistenten eine Fehlermeldung angezeigt.

Weitergabe eines Wertes an den Parameter

Zur Abfrage eines Wertes vom Benutzer werden häufig Parameter genutzt. In deisem Beispiel werden zwei Arten der Weitergabe des Wertes an den Abfrage Parameter, betrachtet.

Die erste Methode gibt den Wert programmatisch weiter. Es existiert keine einfache Art den Wert direkt an den Abfrage-Parameter weiterzugeben. Daher muss hier der Bericht Parameter genutzt werden, der wiederum leicht mittels Code festgelegt werden kann. Hierzu müssen die folgenden Schritte beachtet werden:

· Erstellen der Bericht-Parameter (diese werden später in diesem Kapitel näher betrachtet). Festlegen der gleichen Datenart (DataType) für die Bericht-Parameter, wie für die Abfrage-Parameter.

· Eine Referenzierung zur einem Bericht-Parameter wird in der Eigenschaft "Ausdruck" der Abfrage Parameter hergestellt, z.B.

```
[MyReportParameter]
```

- Weitergabe eines Wertes an den Bericht-Parameter:

```
report1.SetParameterValue("MyReportParameter", 10);
```

Die zweite Art nutzt die Dialogfeldformulare (diese werden später näher erläutert). Soll beispielsweise ein Wert der Art Zeichenfolge abgefragt werden, wird das folgende Vorgehen angewandt:

· Hinzufügen eines Dialogfeldes in den Bericht;

· Einfügen des Steuerelementes "TextBoxControl" zur Eingabe der Werte der Zeichenfolge genutzt;

· Festlegen der Parameter, wie folgt:

```
Name=param1
DataType=VarChar
DefaultValue= (empty string)
Expression=TextBox1.Text
Size=255
```

Immer dann wenn TextBox1 ein Steuerelement ist, das einen, vom Benutzer eingegeben Wert, beinhaltet.

Bearbeiten einer Verbindung

Die mittels des "Daten Assistenten" erstellte Datenverbindung kann bearbeitet werden. Hierzu wird die Datenverbindung im Fenster "Daten" ausgewählt und die Schaltfläche "Bearbeiten" auf der Symbolleiste gedrückt. Es öffnet sich das bereits bekannte Fenster des Datenassistenten. In diesem Fenster können die Verbindungseinstellungen, durch Drücken der Schaltfläche "Verbindung bearbeiten...", geändert werden. Die Art der Verbindung kann nicht geändert werden.

Auf der zweiten Seite des Assistenten können die für den Bericht erwünschten Tabellen ausgewählt werden. Nach diesem Schritt wird die Schaltfläche "Beenden" gedrückt.

Bearbeiten einer Datenquelle

Die mittels des "Daten Assistenten" erstellte Datenverbindung kann bearbeitet werden. Hierzu wird die Datenquelle im Fenster "Daten" ausgewählt und die Schaltfläche "Bearbeiten" auf der Symbolleiste gedrückt. Es öffnet sich das bereits bekannte Fenster des Datenassistenten. Dieses Fenster ermöglicht die Änderung des SQL Abfragetextes und die Festlegung der Abfrageparameter und Datenspalten.

Zur Löschung der Datenquelle wird diese ausgewählt und die Schaltfläche "Löschen" aus der Symbolleiste ausgewählt. Dies führt nicht zu einer physikalischen Löschung der Quelle, sondern ändert deren Zugänglichkeit. Die Aktivierung einer solchen Datenquelle findet mittels des Menüs "Daten|Berichtsdaten auswählen..." statt. Gelöschte Datenquellen werden niemals in der Berichtsdatei gespeichert und entsprechend nicht wiederhergestellt wenn der Bericht das nächste mal gelesen wird.

Hinzufügen einer Datenquelle zu einer bestehenden Verbindung

Es gibt zwei Möglichkeiten, eine Datenquelle (eine Tabelle oder Anfrage) zu einer bestehenden Verbindung hinzufügen:

- durch den Aufruf des Verbindungseditors, wie im Abschnitt "Bearbeiten einer Verbindung" beschrieben. Es wird der Verbindungsassistent angezeigt, auf dessen zweiter Seite Sie eine neue Datenquelle auswählen bzw. erstellen können;
- durch Klicken mit der rechten Maustaste auf "Verbindung" und Auswählen des Menüpunktes "Neue Datenquelle..." Wie im ersten Fall wird anschließend der Verbindungsassistent angezeigt.

Aliases

Alle Datenelemente (Datenquellen und Spalten) tragen Ihren eigenen Namen. Standardmäßig ist dies der Name der in der Datenbank gespeichert ist. In manchen Fällen sind diese Namen nicht selbsterklärend und es ist nicht offensichtlich was sich hinter den Namen verbirgt, z.B. ProdID.

Datenelemente haben einen zweiten Namen, auch "Alias" genannt. Ein Alias ermöglicht die erneute Benennung eines Elementes. Beispielweise haben wir eine Datenquelle CATEGORY_TABLE mit der Spaltenbenennung "PROD_ID". Diesem kann der folgende Alias gegeben werden

```
CATEGORY_TABLE --> Categories
PROD_ID --> Product ID
```

Die Datenspalten können folgendermaßen referenziert werden:

```
[Categories.Product ID]
```

> Bei der Referenzierung zu einem Datenelement muss dann, wenn er definiert wurde, der Alias genutzt werden. Ein Element sollte niemals mit seinem Originalnamen referenziert werden.

Zur Umbenennung eines Datenelements wird dieses im Fenster "Daten" ausgewählt und die Taste F2 gedrückt. Hier kann auch das Item "Umbenennen" aus dem Kontextmenü ausgewählt werden. Hiernach erfolgt die Eingabe der erforderlichen Namens und drücken von OK.

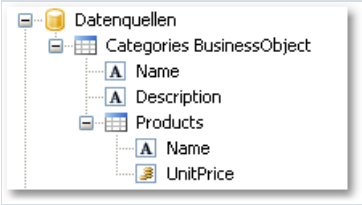
Das Element kann auch über das Fenster "Eigenschaften" umbenannt werden. Auswahl eines Elementes im Fenster "Daten", Wechsel zum Fenster "Eigenschaften" und Änderung des Wertes der Eigenschaft "Alias".

Zur Löschung eines Alias (Zurücksetzen auf den Originalnamen), wird erst ein Element und dann das Item "Alias löschen" in dessen Kontextmenü ausgewählt.

Hierarchische Datenquellen

Die betrachteten Datenquellen sind relationaler Natur, d.h. sie kommen aus relationalen DBMS (auch RDBMS genannt). FastReport unterstützt auch andere Arten von Daten: hierarchische Datenquellen. Derlei Daten kommen häufig aus so-genannten Business-Objekten, die häufig in Anwendungen genutzt werden , um eine relationale Datenquelle als .Net Klasse zu repräsentieren.

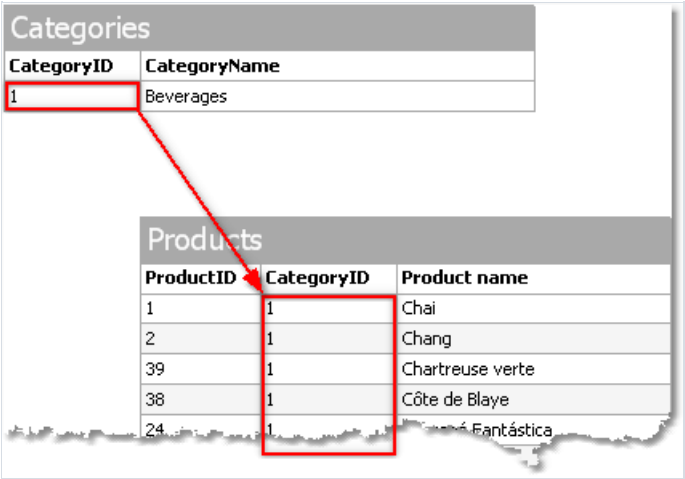
Einem Bericht kann eine hierarchische Datenquelle nur über eine programmatische Registrierung hinzugefügt werden. Dies wird im "Handbuch für Programmierer" näher erläutert. Hier werden einige Unterschiede zwischen ordinalen und hierarchischen Datenquellen betrachtet. In der folgenden Abbildung werden zwei Datenquellen angezeigt: "Categories BusinessObject" and "Products". Es ist zu erkennen, dass die Datenquelle "Products" im übergeordneten Element "Categories BusinessObject" enthalten ist.



Dies bedeutet, dass beide Datenquellen miteinander in Beziehung stehen und als "Master-Detail" Berichtsart genutzt werden können. Beide Datenquellen können seperat in der Berichtsart "einfache Liste" genutzt werden.

Relationen

Zwischen diesen beiden Datenquellen können Relationen festgelegt werden. Die Relation wird zur Definition der "Master-Detail" Beziehung verwendet. So kann beispielsweise ein Eintrag in der Tabelle "Categories" mehrere Einträge in der Tabelle "Products" haben:



Zur Erstellung einer Relation muss das Folgende angegeben werden:

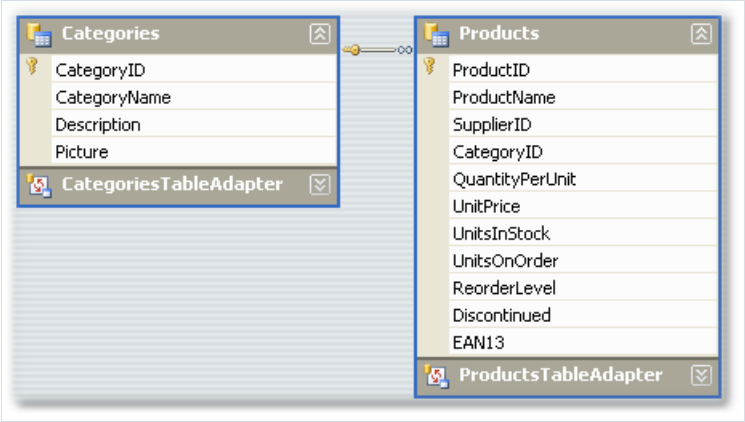
· Tabelle des übergeordneten Elementes;

· Kind-Tabelle;

· Reihe an Schlüsselspalten in der übergeordneten Tabelle;

· Reihe an Schlüsselspalten in der Kind-Tabelle;

Als Beispiel dienen die Tabellen "Categories" und "Products" der Demo-Datenbank. Diese zeigen die folgende Struktur:

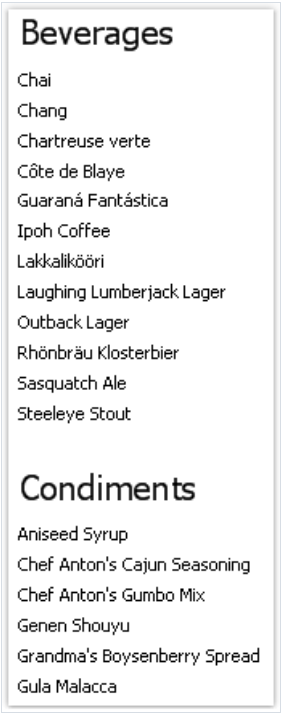


Beide Tabellen haben das CategoryID-Feld, für das die Relation festgelegt werden kann. Eine Kategorie kann also mehrere Produkte beinhalten.

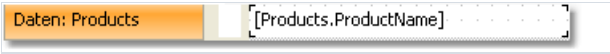
Wie können zusammenhängende Datenquellen in FastReport genutzt werden? Hier eignen sich zwei Methoden.

Mit der ersten Methode können Berichte der Art "Master-Detail" erstellt werden. Hierzu werden zwei "Daten"-Leisten genutzt. Die Masterleiste ist mit der Master-Datenquelle verbunden und die Detailleiste mit der Detail-Datenquelle. Das Beispiel schaut folgendermaßen aus:

Ein solcher Bericht wird, nach seinem Durchlauf, eine Liste an Produkten in jeder Kategorie drucken.



Die zweite Methode ermöglicht die Referenzierung des Masters aus der Detai-Datenquelle. Dies wird im nächsten Beispiel erläutert: Hier soll eine Liste aller Produkte gedruckt werden. Hierzu wird eine "Daten"-Leiste benötigt, die mit der "Products"-Tabelle verbunden ist.



Ein solcher Bericht druckt alle Produkte aus allen Kategorien. Angenommen es soll neben allen Produkten, auch der Kategoriename, zu dem dieses gehört, gedruckt werden; Ohne Relation würde sich dies sehr kompliziert gestalten. Wir kennen lediglich die ID der Produktkategorie (repräsentiert in der Spalte "CategoryID" in der Tabelle "Products"). Der Name der Kategorie, der gedruckt werden soll, ist in der Spalte "CategoryName" der Tabelle "Categories" gespeichert. Mittels der Relation können wir zu der Kategorie auf folgende Weise referenzieren:

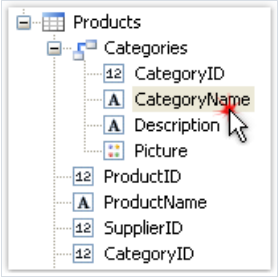
```
[Products.Categories.CategoryName]
```

Für die aktuelle Zeile der Tabelle "Products" findet FastReport die entsprechende Zeile des übergeordneten Elementes in der Tabelle "Categories" und gibt einen Wert an die Spalte "CategoryName" zurück.

Allgemein ermöglicht die Referenzierung auf ein Tabellenfeld eines übergeordneten Elements eine unbeschränkte Anzahl an Tabellenahnen:

```
[Child_table.Its_parent.Parent_of_a_parent.And _so_on.Column_name]
```

Zum Hinzufügen einer solchen Datenspalte in einen Bericht, wird die Tabelle "Products" im "Daten"-Fenster geöffnet. Bei den Spalten lässt sich eine Verknüpfung mit der Tabelle "Categories" erkennen:



Ziehen wir die Spalte (obigen Abbildung) in den Bericht, erhält man ein "Text"-Objekt mit dem Text:

```
[Products.Categories.CategoryName]
```

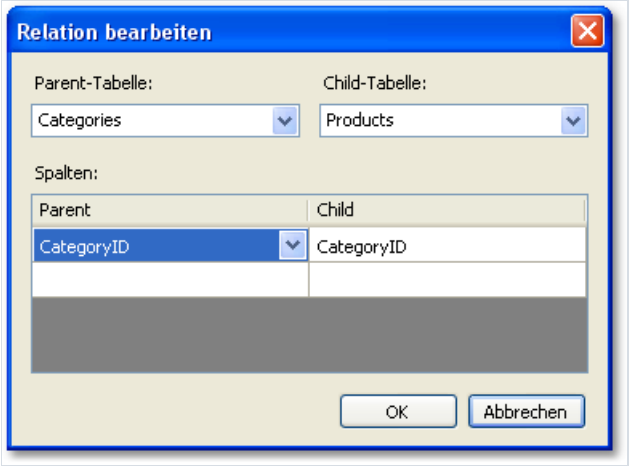
Der Bericht wird folgendermaßen aussehen:

Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

Alice Mutton	Meat/Poultry
Aniseed Syrup	Condiments
Boston Crab Meat	Seafood
Camembert Pierrot	Dairy Products
Carnarvon Tigers	Seafood
Chai	Beverages
Chang	Beverages

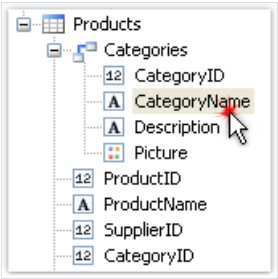
Erstellen einer Beziehung

Zum Erstellen einer Beziehung wird die Schaltfläche "Aktionen" im "Daten"-Fenster gedrückt und das Item "Neue Beziehung..." ausgewählt. Der Beziehung Editor wird angezeigt:



Zuerst muss die Eltern-, und Kind-Tabelle ausgewählt werden. Hiernach wird, im unteren Teil des Fensters, die verknüpfte Datenspalte gewählt. Die Tabellen können mittels einer oder mehrerer Datenspalten verbunden werden. Nachdem die Spalten festgelegt wurden, wird der Beziehung Editor, durch Drücken der Schaltfläche OK, geschlossen.

Die erstellte Beziehung kann im "Daten"-Fenster eingesehen werden. Hierzu wird die Kind-Datenquelle ausgewählt und eine Liste seiner Spalten geöffnet. Die Beziehung zur Elternquelle wird für diese Spalten ersichtlich:



Die Datenspalte der Quelle des übergeordneten Elements kann mittels der drag&drop-Methode in den Bericht eingefügt werden. Werden also die in der Abbildung angegebenen Spalten ausgewählt und auf die Berichtseite gezogen, erhält man ein "Text"-Objekt mit dem folgenden Inhalten:

```
[Products.Categories.CategoryName]
```

Beenden einer Beziehung

Zur Bearbeitung einer Beziehung wird die Liste der Spalten der Datenquelle des Kindes geöffnet, die benötigte Beziehung gefunden und die Schaltfläche "Bearbeiten" auf der Symbolleiste gedrückt. Dies ruft den weiter oben betrachteten Beziehung Editor auf.

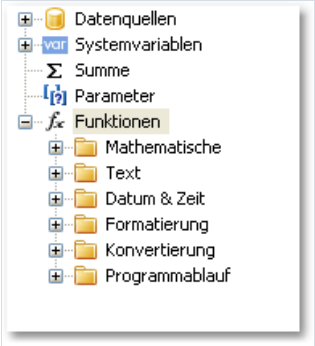
Systemvariablen

FastReport bietet Liste von Systemvariablen, die im Bericht genutzt werden können:

Variable	Beschreibung	
Date	Datum und Zeit des Berichtstarts.	
Page	Aktuelle Seitenzahl.	
TotalPages	Gesamtanzahl der Seiten im Bericht. Zur Nutzung dieser Variable muss "Double Pass" im Bericht aktiviert werden. Dies kann über das Menü "Bericht	Eigenschaften..." geschehen.
PageN	Seitenzahl in der Form: "Seite N".	
PageNofM	Seitenzahl in der Form: "Seite N von M".	
Row#	Zeilennummer der Daten innerhalb der Gruppe. Dieser Wert wird zum Start einer neuen Gruppe zurückgesetzt.	
AbsRow#	Absolute Anzahl an Datenzeilen. Dieser Wert wird zu Beginn einer neuen Gruppe niemals zurückgesetzt.	
Page#	Aktuelle Seitenzahl. Werden mehrere erstelle Berichte in einem Paket vereinigt, gibt diese Variable die aktuelle Seitenzahl in einem Paket zurück.Diese Variable ist ein Makro. Wenn die Komponente im Vorschaufenster angezeigt wird, wird ihr Wert wird ersetzt. Dies bedeutet, dass sie nicht in einem Ausdruck genutzt werden kann.	
TotalPages#	Gesamtanzahl der Seiten im Bericht. Werden mehrere erstelle Berichte in einem Paket vereinigt, gibt diese Variable die Anzahl der Seiten in einem Paket zurück. Um den korrekten Wert zu erhalten, kann hier kein Double Pass genutzt werden.Diese Variable ist ein Makro. Ihr Wert wird ersetzt wenn die Komponente im Vorschaufenster angezeigt wird. Dies bedeutet, dass sie nicht in einem Ausdruck genutzt werden kann.	
HierarchyLevel	Die aktuelle Hierarchieebene in einem hierarchischen Bericht (siehe "Druck Hierarchie"). Die oberste Ebene entspricht 1.	
HierarchyRow#	Vollständige Zeilennummer wie "1.2.1" eines hierarchischen Berichts.	

Funktionen

FastReport.Net umfasst mehr als 60 eingebaute Funktionen. Alle Funktionen sind in mehrere Kategorien unterteilt und über das "Daten"-Fenster zugänglich:



Funktionen können für jeden Ausdruck oder im Skript (siehe auch das Kapitel "Skript") genutzt, oder sein Wert kann im "Text"-Objekt gedruckt werden. Beispielweise wird der folgende Text im "Text"-Objekt:

```
[Sqrt(4)]
```

als "2" (Quadratwurzel von 4) gedruckt.

Der folgende Ausdruck gibt 4 zurück:

```
Sqrt(4) + 2
```

Hier einige Vorgehensweisen zum Einfügen von Funktionen in einen Bericht:

· Eine Funktion kann, mittels drag&drop, aus dem "Daten"-Fenster auf eine Berichtseite gezogen werden. Das "Text"-Objekt wird erstellt und beinhaltet einen Funktionsaufruf. Um Parameter zum Funktionsaufruf hinzuzufügen, muss der Text bearbeitet werden;

· Eine Funktion kann, mittels drag&drop, zum Skriptcode gezogen werden;

· Im Expression Editor kann eine Kopie des "Daten"-Fensters eingesehen werden. Dies funktioniert auf die gleiche Art. Hier können Items mittels drag&drop herausgezogen und in den Text des Ausdrucks gebracht werden.

Im folgenden werden alle Funktionen im Detail beschrieben.

Mathematische Funktionen

Abs

Funktion	Parameter	Rückgabewert
`Abs`	`sbyte value`	`sbyte`
`Abs`	`short value`	`short`
`Abs`	`int value`	`int`
`Abs`	`long value`	`long`
`Abs`	`float value`	`float`
`Abs`	`double value`	`double`
`Abs`	`decimal value`	`decimal`

Gibt den absoluten Wert zurück.

Beispiel:

```
Abs(-2.2) = 2.2
```

Acos

Funktion	Parameter	Rückgabewert
`Acos`	`double d`	`double`

Gibt den Winkel (in Radianten) zurück, dessen Kosinus d ist. d muss zwischen -1 und 1 liegen.

> Multiplizieren des Rückgabewertes mit 180/Math.PI zur Konvertierung von Radianten zu Grad.

Beispiel:

```
Acos(0) * 180 / Math.PI = 90
```

Asin

Funktion	Parameter	Rückgabewert
`Asin`	`double d`	`double`

Gibt den Winkel (in Radianten), dessen Sinus d ist, zurück. d muss zwischen -1 und 1 sein.

> Zur Konvertierung von Radianten zu Grad wird der Rückgabewert mit 180/Math.PI mutipliziert.

Beispiel:

```
Asin(0) = 0
```

Atan

Funktion	Parameter	ückgabewert
`Atan`	`double d`	`double`

Gibt den Winkel (in Radianten), dessen Tangente d ist, zurück.

Zur Konvertierung von Radianten zu Grad wird der Rückgabewerte mit 180/Math.PI multipliziert.

Beispiel:

```
Atan(1) * 180 / Math.PI = 45
```

Ceiling

Funktion	Parameter	Rückgabewert
`Ceiling`	`double d`	`double`
`Ceiling`	`decimal d`	`decimal`

Gibt den kleinsten Integer, größer als oder gleich d, zurück.

Beispiel:

```
Ceiling(1.7) = 2
```

Cos

Funktion	Parameter	Rückgabewert
`Cos`	`double d`	`double`

Gibt den Kosinus des bestimmten Winkels (d) zurück. Der Winkel muss in Radianten sein.

Zur Konvertierung von Grad zu Radianten wird mit Math.PI/180 multipliziert.

Beispiel:

```
Cos(90 * Math.PI / 180) = 0
```

Exp

Funktion	Parameter	Rückgabewert
`Exp`	`double d`	`double`

Gibt e (2.71828), hoch des bestimmten d, zurück.

Beispiel:

```
Exp(1) = 2.71828
```

Floor

Funktion	Parameter	Rückgabewert
`Floor`	`double d`	`double`
`Floor`	`decimal d`	`decimal`

Gibt den größten Integer, kleiner als oder gleich d, zurück.

Beispiel:

```
Floor(1.7) = 1
```

Log

Funktion	Parameter	Rückgabewert
`Log`	`double d`	`double`

Gibt den Logarithmus einer bestimmten Zahl d zurück.

Beispiel:

```
Log(2.71828) = 1
```

Maximum

Funktion	Parameter	Rückgabewert
`Maximum`	`int val1, int val2`	`int`
`Maximum`	`long val1, long val2`	`long`
`Maximum`	`float val1, float val2`	`float`

Funktion	Parameter	Rückgabewert
`Maximum`	`double val1, double val2`	`double`
`Maximum`	`decimal val1, decimal val2`	`decimal`

Gibt den größeren der Werte val1 und val2 zurück.

Beispiel:

```
Maximum(1,2) = 2
```

Minimum

Funktion	Parameter	Rückgabewert
`Minimum`	`int val1, int val2`	`int`
`Minimum`	`long val1, long val2`	`long`
`Minimum`	`float val1, float val2`	`float`
`Minimum`	`double val1, double val2`	`double`
`Minimum`	`decimal val1, decimal val2`	`decimal`

Gibt den kleineren der Werte val1 und val2 zurück.

Beispiel:

```
Minimum(1,2) = 1
```

Round

Funktion	Parameter	Rückgabewert
`Round`	`double d`	`double`
`Round`	`decimal d`	`decimal`

Rundet d auf den nächsten Integer.

Beispiel:

```
Round(1.47) = 1
```

Funktion	Parameter	Rückgabewert
`Round`	`double d, int digits`	`double`
`Round`	`decimal d, int digits`	`decimal`

Rundet d auf eine Präzisierung die in dem "Ziffer" Parametern festgelegt wurde.

Beispiel:

```
Round(1.478, 2) = 1.48
```

Sin

Funktion	Parameter	Rückgabewert
`Sin`	`double d`	`double`

Gibt den Sinus des spezifischen Winkels (d) zurück. Der Winkel muss in Radianten sein.

Zur Konvertierung von Grad zu Radianten wird mit Math.PI/180 multipliziert.

Beispiel:

```
Sin(90 * Math.PI / 180) = 1
```

Sqrt

Funktion	Parameter	Rückgabewert
`Sqrt`	`double d`	`double`

Gibt die Quadratwurzel von d zurück.

Beispiel:

```
Sqrt(4) = 2
```

Tan

Funktion	Parameter	Rückgabewert
`Tan`	`double d`	`double`

Gibt die Tangente des spezifischen Winkels (d) zurück. Der Winkel muss in Radianten sein.

Multiplizieren mit Math.PI/180 zur Konvertierung von Grad zu Radianten.

Beispiel:

```
Tan(45 * Math.PI / 180) = 1
```

Truncate

Funktion	Parameter	Rückgabewert
`Truncate`	`double d`	`double`
`Truncate`	`decimal d`	`decimal`

Berechnet den integralen Teil von d.

Beispiel:

```
Truncate(1.7) = 1
```

Text

Hinweis:

· Diese Funktionen ändern nicht den weitergereichten Wert der Zeichenfolge. Stattdessen geben sie eine neue geänderte Zeichenfolge zurück;

· Das erste Zeichen einer Zeichenfolge hat einen 0 Index. Dies sollte für Funktionen beachtet werden, die einen Zeichenindex haben, wie Insert.

Asc

Funktion	Parameter	Rückgabewert
`Asc`	`char c`	`int`

Gibt einen ganzzahligen Wert zurück, der den Zeichencode entsprechend eines Zeichens repräsentiert.

Beispiel:

```
Asc('A') = 65
```

Chr

Function	Parameters	Return value
`Chr`	`int i`	`char`

Gibt das Zeichen, dass mit einem bestimmten Zeichencode assoziiert ist, zurück.

Beispiel:

```
Chr(65) = 'A'
```

Insert

Funktion	Parameter	Rückgabewert
Insert	`string s, int startIndex, string value`	`string`

Fügt eine Teilzeichenfolge in die "s"-Zeichenfolge zu einer bestimmten Indexposition "startIndex" ein und gibt eine neue Zeichenfolge zurück.

Beispiel:

```
Insert("ABC", 1, "12") = "A12BC"
```

Length

Funktion	Parameter	Rückgabewert
Length	`string s`	`int`

Gibt die Länge von "s" zurück.

Beispiel:

```
Length("ABC") = 3
```

LowerCase

Funktion	Parameter	Rückgabewert
`LowerCase`	`string s`	`string`

Konvertiert alle Zeichen von "s" zu Kleinbuchstaben und gibt ein Ergebnis zurück.

Beispiel:

```
LowerCase("ABC") = "abc"
```

PadLeft

Funktion	Parameter	Rückgabewert
`PadLeft`	`string s, int totalWidth`	`string`

Richtet die Zeichen der Zeichenfolge "s", mit Leerräumen auf der linken Seite, nach rechts aus über eine Gesamtbreite, die im Parameter "totalWidth" bestimmt wurde.

Beispiel:

```
PadLeft("ABC", 5) = "  ABC"
```

Funktion	Parameter	Rückgabewert
`PadLeft`	`string s, int totalWidth, char paddingChar`	`string`

Richtet die Zeichen der Zeichenfolge "s", mit Abschnitten auf der linken Seite, nach rechts aus, über eine Gesamtbreite, die im Parameter "totalWidth" bestimmt wurde.

Beispiel:

```
PadLeft("ABC", 5, '0') = "00ABC"
```

PadRight

Funktion	Parameter	Rückgabewert
`PadRight`	`string s, int totalWidth`	`string`

Richtet die Zeichen der Zeichenfolge "s", mit Leerräumen auf der linken Seite, nach links aus über eine Gesamtbreite, die im Parameter "totalWidth" bestimmt wurde, aus.

Beispiel:

```
PadRight("ABC", 5) = "ABC  "
```

Funktion	Parameter	Rückgabewert
`PadRight`	`string s, int totalWidth, char paddingChar`	`string`

Richtet die Zeichen der Zeichenfolge "s", mit Abschnitten auf der linken Seite, nach links aus über eine Gesamtbreite, die im Parameter "totalWidth" bestimmt wurde.

Beispiel:

```
PadRight("ABC", 5, '0') = "ABC00"
```

Remove

Funktion	Parameter	Rückgabewert
Remove	`string s, int startIndex`	`string`

Löscht alle Zeichen der Zeichenfolge "s". Beginnt mit der Position "startIndex" und setzt dies bis zur letzten Position fort.

Beispiel:

```
Remove("ABCD", 3) = "ABC"
```

Funktion	Parameter	Rückgabewert
Remove	`string s, int startIndex, int count`	`string`

Löscht eine Anzahl von Zeichen, die im "Count" Parameter der "s" Zeichenfolge bestimmt wurde. Beginnt mit der "startIndex" Position.

Beispiel:

```
Remove("A00BC", 1, 2) = "ABC"
```

Replace

Funktion	Parameter	Rückgabewert
`Replace`	`string s, string oldValue, string newValue`	`string`

Gibt eine Zeichenfolge zurück, in der eine bestimmte Teilzeichenfolge mit einer anderen Teilzeichenfolge "newValue" ersetzt wurde.

Beispiel:

```
Replace("A00", "00", "BC") = "ABC"
```

Substring

Funktion	Parameter	Rückgabewert
`Substring`	`string s, int startIndex`	`string`

Gibt eine Teilzeichenfolge au der Zeichenfolge "s" zurück. Die Teilzeichenfolge beginnt mit einer in dem Parameter "startIndex" bestimmten Zeichenposition.

Beispiel:

```
Substring("ABCDEF", 4) = "EF"
```

Funktion	Parameter	Rückgabewert
`Substring`	`string s, int startIndex, int length`	`string`

Gibt eine Teilzeichenfolge au der Zeichenfolge "s" zurück. Die Teilzeichenfolge beginnt mit einer Zeichenposition, die im Parameter "startIndex" bestimmt wurde und hat die im "length" Parameter festgelegte Länge.

Beispiel:

```
Substring("ABCDEF", 1, 3) = "BCD"
```

TitleCase

Funktion	Parameter	Rückgabewert
`TitleCase`	`string s`	`string`

Konvertiert die bestimmte Zeichenfolge zu der Titelschrift.

Beispiel:

```
TitleCase("john smith") = "John Smith"
```

Glätten

Funktion	Parameter	Rückgabewert
`Trim`	`string s`	`string`

Entfernt alle Vorkommnisse der Leerzeichenfolge vom Anfang und Ende der Zeichenfolge "s".

Beispiel:

```
Trim("  ABC  ") = "ABC"
```

UpperCase

Funktion	Parameter	Rückgabewert
`UpperCase`	`string s`	`string`

Konvertiert alle Zeichen von "s" zu Großbuchstaben und gibt ein Ergebnis zurück.

Beispiel:

```
UpperCase("abc") = "ABC"
```

Datum & Zeit

AddDays

Funktion	Parameter	Rückgabewert
`AddDays`	`DateTime date, double value`	`DateTime`

Fügt dem Datum "date" die bestimmte Anzahl an Tagen ("value") und gibt ein neues Datum zurück.

Beispiel:

```
AddDays(#7/29/2009#, 1) = #7/30/2009#
```

AddHours

Funktion	Parameter	Rückgabewert
`AddHours`	`DateTime date, double value`	`DateTime`

Fügt dem Datum "Date" die bestimmte Anzahl an Stunden ("Wert") zu und gibt ein neues Datum zurück.

Beispiel:

```
AddHours(#7/29/2009 1:30#, 1) = #7/29/2009 2:30#
```

AddMinutes

Funktion	Parameter	Rückgabewert
`AddMinutes`	`DateTime date, double value`	`DateTime`

Fügt dem Datum "Date" die bestimmte Anzahl an Minuten ("Wert") zu und gibt ein neues Datum zurück.

Beispiel:

```
AddMinutes(#7/29/2009 1:30#, 1) = #7/29/2009 1:31#
```

AddMonth

Funktion	Parameter	Rückgabewert
`AddMonths`	`DateTime date, int value`	`DateTime`

Fügt dem Datum "date" die spezifizierte Anzahl an Monaten ("value") zu und gibt ein neues Datum zurück.

Beispiel:

```
AddMonths(#7/29/2009#, 1) = #8/29/2009#
```

AddSeconds

Funktion	Parameter	Rückgabewert
`AddSeconds`	`DateTime date, double value`	`DateTime`

Fügt dem Datum "date" die besimmte Anzahl an Sekunden ("value") zu und gibt ein neues Datum zurück.

Beispiel:

```
AddSeconds(#7/29/2009 1:30:01#, 1) = #7/29/2009 1:30:02#
```

AddYears

Funktion	Parameter	Rückgabewert
`AddYears`	`DateTime date, int value`	`DateTime`

Fügt dem Datum "date" die bestimmte Anzahl an Jahren ("value") zu und gibt ein neues Datum zurück.

Beispiel:

```
AddYears(#7/29/2009#, 1) = #7/29/2010#
```

DateDiff

Funktion	Parameter	Rückgabewert
`DateDiff`	`DateTime date1, DateTime date2`	`TimeSpan`

Gibt das Intervall (Anzahl an Tagen, Stunden, Minuten, Sekunden) zwischen zwei Datumsangaben zurück.

Beispiel:

```
DateDiff(#1/2/2009#, #1/1/2009#) = 1.00:00:00
```

DateSerial

Funktion	Parameter	Rückgabewert
`DateSerial`	`int year, int month, int day`	`DateTime`

Erstellt einen neuen DateTime-Wert für das festgelegte Jahr, den Monat und/oder Tag.

Beispiel:

```
DateSerial(2009, 7, 29) = #7/29/2009#
```

Day

Funktion	Parameter	Rückgabewert
`Day`	`DateTime date`	`int`

Holt den Tag des Monats (1-31), dasdurch ein bestimmtes Datum repräsentiert wird.

Beispiel:

```
Day(#7/29/2009#) = 29
```

DayofWeek

Funktion	Parameter	Rückgabewert
`DayOfWeek`	`DateTime date`	`string`

Holt den lokalisierten Namen des Wochentages, der durch ein bestimmtes Datum repräsentiert wird.

Beispiel:

```
DayOfWeek(#7/29/2009#) = "wednesday"
```

DayOfYear

Funktion	Parameter	Rückgabewert
`DayOfYear`	`DateTime date`	`int`

Holt den Tag des Jahres (1-365), der durch ein bestimmtes Datum repräsentiert wird.

Beispiel:

```
DayOfYear(#7/29/2009#) = 210
```

DaysInMonth

Funktion	Parameter	Rückgabewert
`DaysInMonth`	`int year, int month`	`int`

Gibt die Anzahl der Tage in einem bestimmten Monat oder Jahr zurück.

Beispiel:

```
DaysInMonth(2009, 7) = 31
```

Stunden

Funktion	Parameter	Rückgabewert
`Hour`	`DateTime date`	`int`

Holt die Stundenkomponente (0-23), die durch ein bestimmtes Datum repräsentiert wird.

Beispiel:

```
Hour(#7/29/2009 1:30#) = 1
```

Minute

Funktion	Parameter	Rückgabewert
`Minute`	`DateTime date`	`int`

Holt die Minutenkomponente (0-59), die durch bestimmtes Datum repräsentiert wird.

Beispiel::

```
Minute(#7/29/2009 1:30#) = 30
```

Month

Funktion	Parameter	Rückgabewert
`Month`	`DateTime date`	`int`

Holt die Monatskomponente (1-12), die durch ein bestimmtes Datum repräsentiert wird.

Beispiel:

```
Month(#7/29/2009#) = 7
```

MonthName

Funktion	Parameter	Rückgabewert
`MonthName`	`int month`	`string`

Holt den lokalisierten Namen des bestimmten Monats (1-12).

Beispiel:

```
MonthName(1) = "January"
```

Sekunde

Funktion	Parameter	Rückgabewert
`Second`	`DateTime date`	`int`

Holt die Sekundenkomponente (0-59), die durch ein bestimmtes Datum repräsentiert wird.

Beispiel:

```
Second(#7/29/2009 1:30:05#) = 5
```

Year

Funktion	Parameter	Rückgabewert
`Year`	`DateTime date`	`int`

Holt die Jahreskomponente, die durch ein bestimmtes Datum repräsentiert wird.

Beispiel:

```
Year(#7/29/2009#) = 2009
```

Formatierung

Format

Funktion	Parameter	Rückgabewert
`Format`	`string format, params object[] args`	`string`

Ersetzt das formatierte Element in einer bestimmen "Format"-Zeichenfolge mit dem Wert eines korrespondierenden Objektinstanz in einer bestimmten "args" Array.

Zum Beispiel beinhaltet der folgende Funktionsaufruf:

```
Format("Name = {0}, hours = {1:hh}", myName, DateTime.Now)
```

die folgenden formatierten Elemente: "{0}" and "{1:hh}". Diese werden durch Werte der myNme und DateTime.Now ersetzt. Die Ergebnisse können folgendermaßen aussehen:

```
Name = Alex, hours = 12
```

Jedes formatierte Element nimmt die folgende Form an:

```
{index[,alignment][:formatString]}
```

· Index - ein nullbasierter Integer der anzeigt welches Element in einer Liste von Objekten formatiert werden soll;

· Ausrichtung - ein optionaler Integer, der die minimale Weite des Bereichs anzeigt, der den formatierten Wert beinhaltet. Ist die Länge des formatierten Wertes geringer als die Ausrichtung, wird der Bereich mit Leerzeichen aufgefüllt. Ist die Ausrichtung negativ wird der formatierte Wert nach links im Bereich ausgerichtet; ist die Ausrichtung positiv wird der formatierte Wert nach rechts ausgerichtet.

· formatString - eine optionale Zeichenfolge für Spezifizierer der Formatierung.

Die folgende Tabelle beschreibt die standardmäßigen, numerischen Zeichenfolgen zur Formatierung.

Spezifizierer der Formatierung	Name	Beschreibung
C or c	Währung	Die Zahl wird in eine Zeichenfolge umgewandelt, die den Währungsbetrag repräsentiert. `Format("{0:C}", 10) = "$10.00"`
D or d	Dezimalwert	Diese Format unterstützt lediglich Ganzzahltypen. Die Zahl wird in eine Zeichenfolge von Dezimalzahlen (0-9) umgewandelt. `Format("{0:D}", 10) = "10"`

Spezifizierer der Formatierung	Name	Beschreibung
E oder e	Wissenschaftlich	Die Zahl wird in eine Zeichenfolge der folgenden Art umgewandelt: "-d.ddd...E+ddd" oder "-d.ddd...e+ddd". Jede Ziffer 'd' steht für eine Ziffer (0-9). `Format("{0:E}", 10) = "1,000000E+001"`
F oder f	Fester Punkt	Die Zahl wird in eine Zeichenfolge des Formats "-ddd.ddd..." umgewandelt, wobei "d" eine Ziffer anzeigt (0-9). `Format("{0:F}", 10) = "10.00"`
G oder g	Allgemein	Die Zahl wird in die komprimierteste Notation umgewandelt. `Format("{0:G}", 10) = "10"`
N oder n	Zahl	Die Zahl wird in eine Zeichenfolge des Formats "-d,ddd,ddd.ddd" umgewandelt, wobei "d" eine Ziffer anzeigt (0-9). `Format("{0:N}", 1234.56) = "1,234.56"`
P oder p	Prozentsatz	Die Zahl wird in eine Zeichenfolge umgewandelt, den Prozentsatz repräsentiert. Zur Darstellung des Prozentsatzes, wird die konvertierte Zahl wird mit 100 multipliziert. `Format("{0:P}", 0.15) = "15.00%"`
X oder x	Hexadezimal	Die Zahl wird in eine Zeichenfolge von hexadezimalen Ziffern konvertiert. Der Fall der Spezifizierer des Formats zeigt an, ob Klein-, oder Großbuchstaben für die hexadezimalen Ziffern größer als 9 angezeigt werden. Beispielsweise wird "X" zur Erstellung von "ABCDEF" und "x" zur Erstellung von "abcdef" genutzt. `Format("{0:X}", 26) = "1A"`

Bei der Formatierung der Gleitkommata kann eine Anzahl an Dezimalstellen nach der Zeichenfolge angezeigt werden:

```
Format("{0:C1}", 12.23) = "$12.2"
```

Bietet der standardmäßige numerische Spezifizierer des Formats nicht die gewünschte Formatierungsart, kann eine benutzerdefinierte Zeichenfolge zur Formatierung eingegeben werden:

Format Zeichen	Beschreibung
0	Platzhalter Null Hat der formatierte Wert in der Zeichenfolge eine Ziffer in der Position an der die "0" erscheint, wird diese Ziffer in die Ergebniszeichenfolge kopiert. Die Position der am weitesten Links positionierten "0" vor der Dezimalstelle und der am weitesten rechts positionieren "0" nach der Dezimalstelle bestimmt den Ziffernbereich der stets in der Ergebniszeichenfolge vorhanden ist.
#	Platzhalter Ziffer Hat der formatierte Wert in der Zeichenfolge eine Ziffer in der Position an der das "#" erscheint, wird diese Ziffer in die Ergebniszeichenfolge kopiert. Anderweitig wird nichts in diese Position des Ergebniszeichenfolge kopiert.
.	Dezimalstelle Das erste "."-Symbol im Formatierungszeichenfolge bestimmt die Positionierung der Dezimalstelle in dem formatierten Wert.
,	Tausendertrennzeichen. Beinhaltet die Zeichenfolge einen ","-Symbol dann hat wird die Ausgabe ein Tausendertrennzeichen zwischen Gruppen von drei Ziffern zur linken der Dezimalstelle anzeigen.

Format Zeichen	Beschreibung
%	Platzhalter Prozent Die Anwesenheit eines "%"-Symbols in einer Zeichenfolge resultiert in einer Zahl multipliziert mit 100 vor der Formatierung.
;	Abschnittszeichen Das ";"-Symbol wird zur Trennung von Abschnitten für positive, negative und Null-Zahlen in einer Zeichenfolge genutzt.

Anwendungsbeispiele:

```
Format("{0:$#,##0.00}", 1024.25) = "$1,024.25"
Format("{0:00%}", 0.25) = "25%"
Format("{0:$#,##0.00;($#,##0.00);Zero}", 1024.25) = "$1,024.25"
Format("{0:$#,##0.00;($#,##0.00);Zero}", -1024.25) = "($1,024.25)"
Format("{0:$#,##0.00;($#,##0.00);Zero}", 0) = "Zero"
```

Die folgende Tabelle gibt einen Überblick über die standardmäßigen Spezifizierer des Formats der DateTime-Werte:

Spezifizierer der Formatierung	Name	Beispiel
d	Kurzes Datenmuster(amerikanisches Datenformat)	"8/9/2009"
D	Langes Datenmuster	"Sunday, August 09, 2009"
f	Komplettes Datum-,/Zeit-Muster (Kurzes Zeitformat)	"Sunday, August 09, 2009 2:44 PM"
F	Komplettes Datum-,/Zeit-Muster (langes Zeitformat)	"Sunday, August 09, 2009 2:44:01 PM"
g	Allgemeines Datum-,/Zeit-Muster (kurzes Zeitformat)	"8/9/2009 2:44 PM"
G	Allgemeines Datum-,/Zeit-Muster (langes Zeitformat)	"8/9/2009 2:44:01 PM"
t	Kurzes Zeitmuster	"2:44 PM"
T	Langes Zeitmuster	"2:44:01 PM"

In der folgenden Tabelle werden die benutzerdefinierten Datum/Zeit Spezifizierer für das Format beschrieben und deren produziertes Ergebnis.

Spezifizierer der Formatierung	Beschreibung
d	Zeigt den aktuellen Tag des Monats, gemessen als Zahl zwischen 1 und 31 (einschließlich), an. Ist der Tag nur einstellig (1-9) wird er entsprechend als einzelne Ziffer angezeigt.

Spezifizierer der Formatierung	Beschreibung
dd	Zeigt den aktuellen Tag des Monats, gemessen als Zahl zwischen 1 und 31 (einschließlich), an. Ist der aktuelle Tag einstellig (1-9) wird er mit einer vorangehenden 0 (01-09) formatiert.
ddd	Zeigt den abgekürzten Namen des Tages an.
dddd	Zeigt den vollständigen Namen des Tages an.
f or F	Zeigt die wichtigste Ziffer des zweiten Bruchs an.
h	Zeigt die Stunde im Bereich 1-12 an. Ist die Stunde einstellig (1-9) wird sie entsprechend als einzelne Ziffer angezeigt.
hh	Zeigt die Stunde im Bereich 1-12 an. Ist die Stunde einstellig (1-9) wird sie mit einer vorangehenden 0 (01-09) formatiert.
H	Zeigt die Stunde im Bereich 0-23 an. Ist die Stunde einstellig (1-9) wird sie entsprechend als einzelne Ziffer angezeigt.
HH	Zeigt die Stunde im Bereich 0-23 an. Ist die Stunde einstellig (1-9) wird sie mit einer vorangehenden 0 (01-09) formatiert.
m	DZeigt die Minute im Bereich 0-59 an. Ist die Minute einstellig (0-9) wird sie entsprechend als einzelne Ziffer angezeigt.
mm	Zeigt die Minute im Bereich 0-59 an. Ist die Minute einstellig (0-9) wird sie mit einer vorangehenden 0 (01-09) formatiert.
M	Zeigt den Monat, gemessen als Zahl zwischen 1 und 12 (einschließlich), an. Ist der Monat einstellig (1-9) wird er mit einer vorangehenden 0 (01-09) formatiert.
MM	Zeigt den Monat, gemessen als Zahl zwischen 1 und 12 (einschließlich), an. Ist der Monat einstellig (1-9) wird er mit einer vorangehenden 0 (01-09) formatiert.
MMM	Zeigt den abgekürzten Namen des Monats an.
MMMM	Zeigt den vollständigen Namen des Monats an.
s	Zeigt die Sekunden im Bereich 0-59 an. Ist die Sekunde einstellig (0-9) wird sie entsprechend als einzelne Ziffer angezeigt.
ss	Zeigt die Sekunden im Bereich 0-59 an. Ist die Sekunde einstellig (0-9) wird sie mit einer vorangehenden 0 (01-09) formatiert.
t	Zeigt das ersten Zeichen des A.M./P.M. Kennzeichners an.
tt	Zeigt den A.M./P.M. Kennzeichner an.
y	Zeigt das Jahr maximal als zweistellige Zahl an. Die beiden ersten Ziffern des Jahres werden ausgelassen. Ist das Jahr einstellig (1-9) wird es entsprechend als einzelne Ziffer angezeigt.

Spezifizierer der Formatierung	Beschreibung
yy	Zeigt das Jahr maximal als zweistellige Zahl an. Die beiden ersten Ziffern des Jahres werden ausgelassen. Ist das Jahr einstellig (1-9) wird es mit einer vorangehenden 0 (01-09) formatiert.
yyyy	Zeigt das Jahr, einschließlich des Jahrhunderts an. Ist das Jahr kürzer als vier Ziffern werden, falls notwendig, vorangehende Nullen hinzugefügt, um das angezeigte Jahr auf vier Stellen zu erweitern.
z	Zeigt die Zeitzone für die aktuelle Zeitzone des Systems in ausschließlich ganzen Stunden an. Die zeitliche Verschiebung wird stets mit dem führenden Symbol (Null wird als "+0" angezeigt) zur Greenwich Mean Time, als vor (+) oder nach (-) angezeigt. Der Wertebereich liegt zwischen -12 und +13. Ist die Verschiebung einstellig (0-9) wird sie entsprechend als einzelne Ziffer mit dem entsprechenden Führungssymbol angezeigt.
zz	Zeigt die Verschiebung der Zeitzone für die aktuelle Zeitzone des Systems in ausschließlich ganzen Stunden an. Die zeitliche Verschiebung wird stets mit einem Führungs-, oder Spätsymbol (null wird als "+00" angezeigt), die den Unterschied zur Greenwich Mean Time, als vor (+) oder nach (-) anzeigen, dargestellt. Der Wertebereich liegt zwischen -12 und +13. Ist die Verschiebung einstellig (0-9) wird sie entsprechend mit einer vorangehenden 0 (01-09) und dem entsprechenden Führungszeichen angezeigt.
zzz	Zeigt die Verschiebung der Zeitzone für die aktuelle Zeitzone des Systems in Stunden und Minuten an. Die zeitliche Verschiebung wird stets mit einem Führungs-, oder Spätsymbol (null wird als "00:00" angezeigt), die den Unterschied zur Greenwich Mean Time, als vor (+) oder nach (-) anzeigen, dargestellt. Der Wertebereich liegt zwischen -12:00 und +13:00. Ist die Verschiebung einstellig (0-9) wird sie entsprechend mit einer vorangehenden 0 (01-09) und dem entsprechenden Führungszeichen angezeigt.
:	Das Trennzeichen für die Zeit
/	Das Trennzeichen für das Datum.

Anwendungsbeispiele:

```
Format("{0:d MMM yyyy}", DateTime.Now) = "9 Aug 2009"
Format("{0:MM/dd/yyyy}", DateTime.Now) = "08/09/2009"
Format("{0:MMMM, d}", DateTime.Now) = "August, 9"
Format("{0:HH:mm}", DateTime.Now) = "16:07"
Format("{0:MM/dd/yyyy hh:mm tt}", DateTime.Now) = "08/09/2009 04:07 PM"
```

FormatCurrency

Funktion	Parameter	Rückgabewert
`FormatCurrency`	`object value`	`string`

Formatiert den bestimmten Wert in eine Währung mittels der Ländereinstellungen.

Beispiel:

```
FormatCurrency(1.25) = "$1.25"
```

Funktion	Parameter	Rückgabewert
`FormatCurrency`	`object value, int decimalDigits`	`string`

Formatiert den bestimmten Wert in eine Währung. Der Parameter "decimalDigits" zeigt an, wie viele Stellen zur rechten der Dezimalstelle angezeigt werden.

Beispiel:

```
FormatCurrency(1.25, 1) = "$1.3"
```

FormatDateTime

Funktion	Parameter	Rückgabewert
`FormatDateTime`	`DateTime value`	`string`

Formatiert den bestimmten Wert in eine Datums-,/Zeitangabe mittels der Ländereinstellungen. Diese Funktion beinhaltet keine neutralen Werte in der resultierenden Zeichenfolge.

Beispiel:

```
FormatDateTime(#1/1/2009#) = "01/01/2009"
FormatDateTime(#1/1/2009 1:30#) = "01/01/2009 1:30:00 AM"
FormatDateTime(#1:30#) = "1:30:00 AM"
```

Funktion	Parameter	Rückgabewert
`FormatDateTime`	`DateTime value, string format`	`string`

Formatiert den bestimmten Wert in eine Datums-,/Zeitangabe mittels der in dem "Format" Parameter benannten Spezifizierer für das Format. Die gültigen Werte für diesen Parameter sind:

"Long Date"

"Short Date"

"Long Time"

"Short Time"

Beispiel:

```
FormatDateTime(#1/1/2009 1:30#, "Long Date") = "Thursday, January 01, 2009"
FormatDateTime(#1/1/2009#, "Short Date") = "01/01/2009"
FormatDateTime(#1:30#, "Short Time") = "01:30 AM"
FormatDateTime(#1:30#, "Long Time") = "1:30:00 AM"
```

FormatNumber

Funktion	Parameter	Rückgabewert
`FormatNumber`	`object value`	`string`

Formatiert, mittels der Ländereinstellungen von Windows, den bestimmten Wert in eine Zahl.

Beispiel:

```
FormatNumber(1234.56) = "1,234.56"
```

Funktion	Parameter	Rückgabewert
`FormatNumber`	`object value, int decimalDigits`	`string`

Formatiert den bestimmten Wert in eine Zahl. Der Parameter "decimalDigits" zeigt an, wie viele Stellen zur rechten der Dezimalstelle angezeigt werden.

Beispiel:

```
FormatNumber(1234.56, 1) = "1,234.6"
```

FormatPercent

Funktion	Parameter	Rückgabewert
`FormatPercent`	`object value`	`string`

Formatiert, mittels der Ländereinstellungen von Windows, den bestimmten Wert in Prozent.

Beispiel:

```
FormatPercent(0.15) = "15.00%"
```

Funktion	Parameter	Rückgabewert
`FormatPercent`	`object value, int decimalDigits`	`string`

Formatiert den bestimmten Wert in Prozent. Der Parameter "decimalDigits" zeigt an, wie viele Stellen zur rechten der Dezimalstelle angezeigt werden.

Beispiel:

```
FormatPercent(0.15, 0) = "15%"
```

Umrechnung

ToBoolean

Funktion	Parameter	Rückgabewert
`ToBoolean`	`object value`	`bool`

Wandelt den bestimmten Wert in Boolean um.

Beispiel:

```
ToBoolean(1) = true
ToBoolean(0) = false
```

In Byte

Funktion	Parameter	Rückgabewert
`ToByte`	`object value`	`byte`

Wandelt den bestimmten Wert in Byte um.

Beispiel:

```
ToByte("55") = 55
```

In Char

Funktion	Parameter	Rückgabewert
`ToChar`	`object value`	`char`

Wandelt den bestimmten Wert in Char um.

Beispiel:

```
ToChar(65) = 'A'
```

ToDateTime

Funktion	Parameter	RRückgabewert
`ToDateTime`	`object value`	`DateTime`

Konvertiert den bestimmten Wert in Datum/Zeit.

Beispiel:

```
ToDateTime("1/1/2009") = #1/1/2009#
```

ToDecimal

Funktion	Parameter	Rückgabewert
`ToDecimal`	`object value`	`decimal`

Wandelt den bestimmten Wert in Dezimal um.

Beispiel:

```
ToDecimal(1) = 1m
ToDecimal("1") = 1m
```

ToDouble

Funktion	Parameter	Rückgabewert
`ToDouble`	`object value`	`double`

Wandelt den bestimmten Wert in Doppel um.

Beispiel:

```
ToDouble(1) = 1
ToDouble("1") = 1
```

ToInt32

Funktion	Parameter	Rückgabewert
`ToInt32`	`object value`	`int`

Wandelt den bestimmten Wert in int um.

Beispiel:

```
ToInt32(1f) = 1
ToInt32("1") = 1
```

ToRoman

Funktion	Parameter	Rückgabewert
`ToRoman`	`object value`	`string`

Wandelt den bestimmten Zahlenwert in die römische Entsprechung um. Der Wert muss im Bereich 1-3998 liegen.

Beispiel:

```
ToRoman(9) = "IX"
```

ToSingle

Funktion	Parameter	Rückgabewert
`ToSingle`	`object value`	`float`

Wandelt den bestimmten Wert in einen Gleitwert um.

Beispiel:

```
ToSingle(1m) = 1f
ToSingle("1") = 1f
```

ToString

Funktion	Parameter	Rückgabewert
`ToString`	`object value`	`string`

Wandelt den bestimmten in eine Zeichenfolge um.

Beispiel:

```
ToString(false) = "False"
ToString(DateTime.Now) = "08/09/2009 4:45:00 PM"
```

ToWords

Funktion	Parameter	Rückgabewert
`ToWords`	`object value`	`string`

Wandelt die bestimmte Währung in Worte um.

Beispiel:

```
ToWords(1024.25) = "One thousand and twenty-four dollars and 25 cents"
```

Funktion	Parameter	Rückgabewert
`ToWords`	`object value, string currencyName`	`string`

Wandelt die bestimmte Währung in Worte um. Der Parameter "currencyName" zeigt die Währung an. Gültige Werte dieses Parameters sind:

"USD"

"EUR"

"GBP"

Beispiel:

```
ToWords(1024.25, "EUR") = "One thousand and twenty-four euros and 25 cents"
```

Funktion	Parameter	Rückgabewert
`ToWords`	`object value, string one, string many`	`string`

Wandelt den bestimmten Integer in Worte um. Der Parameter "one" beinhaltet den Namen im Singular; der Parameter "many" beinhaltet den Namen im Plural.

Beispiel:

```
ToWords(124, "page", "pages") = "One hundred and twenty-four pages"
ToWords(1, "page", "pages") = "One page"
```

ToWordsEnGb

Funktion	Parameter	Rückgabewert
`ToWordsEnGb`	`object value`	`string`

Wandelt den bestimmten Währungswert in Worte in britisches Englisch um. Es bestehen die folgenden Unterschiede zwischen dieser Funktion und ToWords:

· die Währung GBP wird standardmäßig genutzt;

· unterschiedliche Wortwahl für die Umwandlung in Milliarden und Trilliarden.

Beispiel:

```
ToWordsEnGb(121) = "One hundred and twenty-one pounds and 00 pence"
```

Funktion	Parameter	Rückgabewert
`ToWordsEnGb`	`object value, string currencyName`	`string`

Wandelt den bestimmten Währungswert in Worte (Britisches Englisch) um. Der Parameter "currencyName" zeigt die Währung an. Gültige Werte dieses Parameters sind:

"USD"

"EUR"

"GBP"

Beispiel:

```
ToWordsEnGb(1024.25, "EUR") = "One thousand and twenty-four euros and 25 cents"
```

Funktion	Parameter	Rückgabewert
`ToWordsEnGb`	`object value, string one, string many`	`string`

Wandelt den bestimmten Integerwert in Worte (Britisches Englisch) um. Der Parameter "one" beinhaltet den Namen im Singular; der Parameter "many" beinhaltet den Namen im Plural.

Beispiel:

```
ToWordsEnGb(124, "page", "pages") = "One hundred and twenty-four pages"
ToWordsEnGb(1, "page", "pages") = "One page"
```

ToWordsRu

Funktion	Parameter	Rückgabewert
`ToWordsRu`	`object value`	`string`

Wandelt die bestimmten Währungswerte in russische Worte um.

Beispiel:

```
ToWordsRu(1024.25) = "Одна тысяча двадцать четыре рубля 25 копеек"
```

Funktion	Parameter	Rückgabewert
`ToWordsRu`	`object value, string currencyName`	`string`

Wandelt die bestimmten Währungswerte in russische Worte um. Der Parameter "currencyName" zeigt die Währung an. Gültige Werte dieses Parameters sind:

"RUR"

"UAH"

"USD"

"EUR"

Beispiel:

```
ToWordsRu(1024.25, "EUR") = "Одна тысяча двадцать четыре евро 25 евроцентов"
```

Funktion	Parameter	Rückgabewert
`ToWordsRu`	`object value, bool male, string one, string two, string many`	`string`

Wandelt die bestimmten Integerwerte in russische Worte um. Der Parameter "male" zeigt das Geschlecht des Namen an. Die Parameter '"one", "two" und "five" beinhalten eine Form des Namen, die mit den Zahlen "1", "2" und "5" genutzt werden.

Beispiel:

```
// the "страница" word is of female gender, male = false
ToWordsRu(122, false, "страница", "страницы", "страниц") =
"Сто двадцать две страницы"
// the "лист" word is of male gender, male = true
ToWordsRu(122, true, "лист", "листа", "листов") =
"Сто двадцать два листа"
```

Programmablauf

Choose

Funktion	Parameter	Rückgabewert
`Choose`	`double index, params object[] choice`	`object`

Gibt ein Element des Array "Choice" mit dem im "Index" bestimmten Indexparameter zurück. Das erste Array-Element hat 1 Index.

Beispiel:

```
Choose(2, "one", "two", "three") = "two"
```

IIf

Funktion	Parameter	Rückgabewert
`IIf`	`bool expression, object truePart, object falsePart`	`object`

Ist der "Ausdruck" = true wird der Wert "truePart" zurückgegeben. Wenn nicht, gibt er den "falsePart" Wert zurück.

Beispiel:

```
IIf(2 > 5, "true", "false") = "false"
```

Switch

Funktion	Parameter	Rückgabewert
`Switch`	`params object[] expressions`	`object`

Das Argument, dass mit den Ausdrücken zurück gegeben wird, besteht aus gepaarten Ausdrücken und Werten. Die Switch-Funktion wertet die ungeraden Ausdrücke, vom niedrigsten bis zum höchsten Index, aus und gibt die geraden Werte, die mit dem ersten Ausdruck, der mit true ausgewertet ist, assoziiert sind, zurück.

Beispiel:

```
// returns one of the following values - "a greater than 0",
// "a less than 0", "a equals to 0", depending on "a" value
Switch(
  a > 0, "a greater than 0",
  a < 0, "a less than 0",
  a == 0, "a equals to 0")
```

Summen

In vielen Berichten muss ggf. eine Summeninformation bereit gestellt werden: Summe der Gruppe, Anzahl der Reihen in der Liste und viele mehr. Zur Durchführung dieser Aufgabe nutzt FastReport Summen. Für die Summe muss der folgende Parameter angezeigt werden:

· Die Funktionsart der Summe;

· Der Ausdruck, der berechnet werden soll. Für die "Count"-Funktion muss der Ausdruck nicht angezeigt werden;

· Die Bedingung; Die Funktion wird berechnet, wenn die Bedingung erfüllt ist. Eine Festlegung der Bedingung ist nicht obligatorisch.

· Die Datenleiste für die die Funktion bearbeitet wird;

· Die Leiste in der der Gesamtwert gedruckt wird.

Eine Liste aller Summenfunktionen kann hier eingesehen werden:

Funktion	Beschreibung
Sum	Berechnet die Summe des Ausdrucks.
Min	Berechnet den Mindestwert des Ausdrucks.
Max	Berechnet den Maximalwert des Ausdrucks.
Average	Berechnet den Durchschnittswert des Ausdrucks.
Anzahl der verschiedenen	Gibt die Anzahl der verschiedenen Zeilen im Bereich.
Count	Gibt die Anzahl der Zeilen zurück.

Erstellen einer Summe

Als Beispiel wird die Nutzung der Summenfunktion betrachtet. Zuerst wird ein "Master-Detail"-Bericht erstellt, der die zwei Tabellen "Categories" und "Products" nutzt.

Hiernach sieht der vorbereitete Bericht wie folgt aus:

Beverages

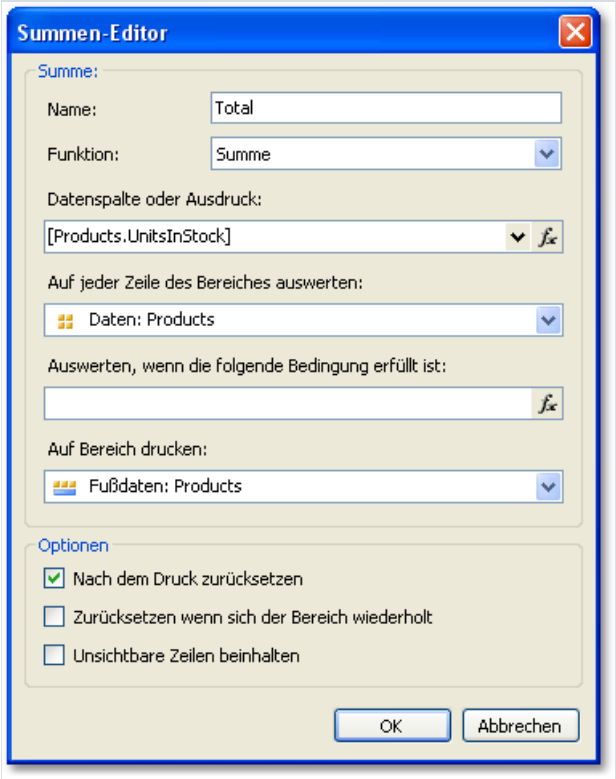
Product Name	Units In Stock
Chai	39
Chang	17
Chartreuse verte	69
Côte de Blaye	17
Guaraná Fantástica	20
Ipoh Coffee	17
Lakkalikööri	57
Laughing Lumberjack Lager	52
Outback Lager	15
Rhönbräu Klosterbier	125
Sasquatch Ale	111
Steeleye Stout	20

Condiments

Product Name	Units In Stock
Aniseed Syrup	13
Chef Anton's Cajun Seasoning	53
Chef Anton's Gumbo Mix	0
Genen Shouyu	39
Grandma's Boysenberry Spread	120
Gula Malacca	27
Louisiana Fiery Hot Pepper Sauce	76
Louisiana Hot Spiced Okra	4
Northwoods Cranberry Sauce	6
Original Frankfurter grüne Soße	32
Sirop d'érable	113
Vegie-spread	24

Lassen Sie uns in diesen Bericht eine Summe einfügen, die die Gesamtmenge der Einheiten auf Lager für jede Kategorie druckt: Summe der Datenspalte "UnitsInStock". Die Summe wird in der Leiste "Datenfuß" gedruckt.

Zum Druck der Gesamtsumme muss diese zuerst erstellt werden. Hierzu wird die Schaltfläche "Aktionen" im "Daten"-Fenster gedrückt und das Element "Neue Summe" ausgewählt. Eine weitere Methode: Klicken mit der rechten Maustaste auf der entsprechende Element im Datenbaum und Auswahl des Menüelements "Neue Summe". Das Editorfenster zur Summe wird angezeigt:

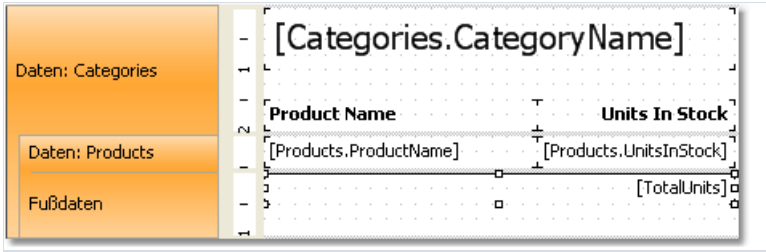


Zuerst wird der Name der Summe eingetragen. Die Summe wird über den Namen referenziert; Daher sollte die Summe so benannt werden, dass der Name einfach verständlich und selbsterklärend ist. In diesem Beispiel wird die Summe "Total" genannt.

Danach wird die Funktion "Summe" für die Summe ausgewählt.

Nun muss der Datenbereich, für den die Summe berechnet wird, anzeigt werden. Für das Feld "Auf jeder Zeile des Bereiches auswerten:" wählen wir die "Daten"-Leiste in der eine Produktliste gedruckt wird. Für das Feld "Auf Bereich drucken:" wird eine Leiste, in die die Summe gedruckt wird, ausgewählt. In diesem Beispiel ist dies die Leiste "Fußdaten".

Der Editor wird durch Drücken der Schaltfläche "OK" geschlossen. Die neue Summe erscheint im Fenster "Daten". Nun kann sie in den Bericht gezogen werden.



Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

Beverages

Product Name	Units In Stock
Chai	39
Chang	17
Chartreuse verte	69
Côte de Blaye	17
Guaraná Fantástica	20
Ipoh Coffee	17
Lakkalikööri	57
Laughing Lumberjack Lager	52
Outback Lager	15
Rhönbräu Klosterbier	125
Sasquatch Ale	111
Steeleye Stout	20
	559

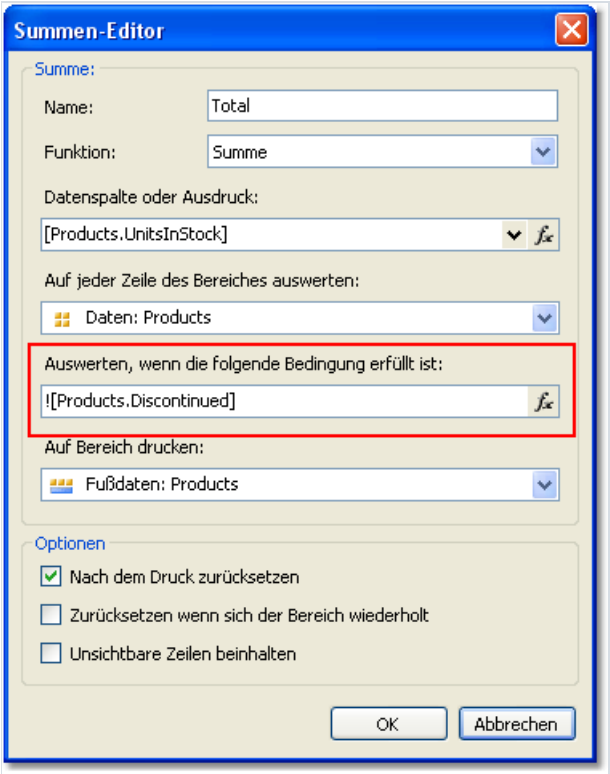
Condiments

Product Name	Units In Stock
Aniseed Syrup	13
Chef Anton's Cajun Seasoning	53
Chef Anton's Gumbo Mix	0
Genen Shouyu	39
Grandma's Boysenberry Spread	120
Gula Malacca	27
Louisiana Fiery Hot Pepper Sauce	76
Louisiana Hot Spiced Okra	4
Northwoods Cranberry Sauce	6
Original Frankfurter grüne Soße	32
Sirop d'érable	113
Vegie-spread	24
	507

Bedingte Summen

Im vorherigen Beispiel wurde die Summe für alle Datenreihen berechnet. Der Datenbereich kann eingegrenzt werden, indem die Bedingung im Summen-Editor angezeigt wird. Die Summe wird nur für solche Reihen berechnet, deren Bedingung als true zurück gegeben wird.

Beispielweise können die folgenden Bedingungen festgelegt werden:



Dies bedeutet, dass die Summe für diejenigen Produkte berechnet werden soll, dessen Kennzeichen nicht auf "Discontinued" gesetzt ist.

Ergebnis

In dem Beispiel wurden die Summen nach dem Druck der Leiste "Fußdaten" zurückgesetzt. Der Grund hierfür: der Summen-Editor macht ein Zurücksetzen der Summe nach dem Druck notwendig macht. Als Ergebnis druckt jede Spalte ihre eigenen Summenwerte.

Wird die Summe nach dem Druck nicht zurückgesetzt, wird das Kontrollkästchen "Nach dem Druck zurücksetzen" deaktiviert. Dies wird auch Ergebnis (Running Total) genannt.

Sollen die zwei Arten von Summen, reguläre Summen und Ergebnisse, gleichzeitig gedruckt werden, wird eine weitere Summe mit ähnlichen Einstellungen und der deaktivierten Kennzeichnung "Nach dem Druck zurücksetzen" erstellt.

Seitensummen

Um die Summe, zum Druck auf dem Seitenfuß, zu erstellen, muss der Seitenfuß im Feld "Print on the band" angezeigt werden.

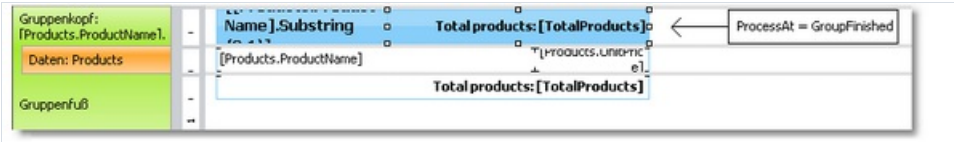
Druck der Summe in der Kopfzeile

Üblicherweise werden Summenwerte auf die Fußleisten (wie Datenfuß, Gruppenfuß, etc.) gedruckt. Dies entspricht einer natürlichen Druckreihenfolge, da beim Druck der Summe sein Wert ordnungsgemäß berechnet wird und gebrauchsfertig ist. Es kann jedoch erforderlich werden die Summen auf die Kopfzeile (beispielsweise auf den Gruppenkopf) zu drucken. Hierbei wird ein Nullwert angezeigt. Zum Zeitpunkt des Drucks der Summe ist diese noch nicht berechnet.

FastReport bietet die Eigenschaft "delayed print" zur Lösung dieses Problems. Das "Text"-Objekt hat die Eigenschaft mit dem Namen "ProcessAt", die eine der folgenden Werte annehmen kann:

Wert	Beschreibung
Default	Der standardmäßige Druckmodus. Dies ist der Standardwert.
ReportFinished	Der Wert des Objektes wird am Ende des Berichts berechnet.
ReportPageFinished	Der Wert des Objektes wird berechnet, nachdem alle Leisten auf der Seite beendet wurden.
PageFinished	Der Wert des Objektes wird am Ende der Seite berechnet.
ColumnFinished	Der Wert des Objektes wird am Ende der Spalte berechnet.
DataFinished	Der Wert des Objektes wird am Ende der Datenleiste (wenn die Fußzeile gedruckt wird) berechnet.
GroupFinished	Der Wert des Objektes wird am Ende der Gruppe (wenn seine Fußzeile gedruckt wird) berechnet.

Lassen Sie uns betrachten, wie dies funktioniert. Das "Text"-Objekt, das die Summe druckt, wird auf den Gruppenkopf positioniert. Die Eigenschaft "ProcessAt" des "Text"-Objektes wird auf "GroupFinished" festgelegt:



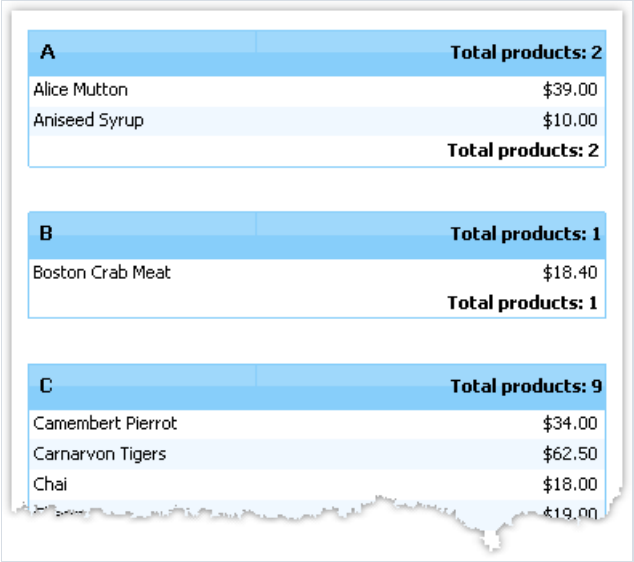
Beim Durchlauf eines solchen Berichtes wird FastReport folgende Aktionen durchführen:

· Der Gruppenkopf wird gedruckt. Die Gesamtsumme wird als 0 (falsch) gedruckt. FastReport merkt sich dieses Objekt, um es später zu bearbeiten;

· Alle Datenreihen werden gedruckt;

· Der Gruppenkopf wird gedruckt. Zu diesem Zeitpunkt nimmt FastReport das im Gruppenkopf gedruckte Objekt und bearbeitet es erneut, um die korrekte Summe zu drucken.

Hiernach sieht der vorbereitete Bericht wie folgt aus:



A	Total products: 2
Alice Mutton	$39.00
Aniseed Syrup	$10.00
	Total products: 2

B	Total products: 1
Boston Crab Meat	$18.40
	Total products: 1

C	Total products: 9
Camembert Pierrot	$34.00
Carnarvon Tigers	$62.50
Chai	$18.00
[illegible]	$19.00

Die Auswahl anderer Werte der Eigenschaft "ProcessAt" ermöglichen es, die Berichtssumme in den Berichtstitel (ProcessAt = ReportFinished festlegen) oder die Summe der Seite in den Seitenkopf (ProcessAt = PageFinished) zu drucken.

> Die verzögerte Druckeigenschaft funktioniert nicht, wenn der Dateicache des Berichts (Menü "Bericht|Optionen...", Kontrollkästchen "Filecache ausnutzen") angestellt ist.

Berichtsparameter

Die Parameter eines Berichts können definiert werden. Ein Parameter ist eine Variable, dessen Wert sowohl in dem Bericht selbst und auch außerhalb des Berichts (z.B. ein Programm, dass einen Bericht aufruft kann Parameterwerte in diesen übertragen) Für Details siehe auch "Handbuch für Programmierer") definiert werden kann. Ein Parameter kann in Ausdrücken genutzt und in Berichtsobjekten, wie das "Text"-Objekt, angezeigt werden.

Die gebräuchlichsten Methoden zur Nutzung von Parametern:

· Datenfilterung durch die Bedingung die in einem Parameter festgelegt ist;

· Druck eines Parameterwertes in einem Bericht.

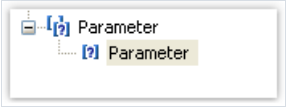
Ein Parameter weist die folgenden Eigenschaften auf:

Eigenschaft	Beschreibung
Name	Der Name eines Parameters kann jedes beliebige Symbol, außer dem Punkt-Symbol ".", enthalten.
DataType	Datenart des Parameters.
Expression	Ausdruck der den Parameterwert zurückgibt. Detaillierte Informationen über Ausdrücke sind im Kapitel "Ausdrücke" erläutert. Dieser Wert wird beim Aufruf eines Parameters bearbeitet.
Value	Parameterwert. Diese Eigenschaft steht im Designer nicht zur Verfügung und kann programmatisch gefüllt werden.

Es müssen die Eigenschaften "Name" und "DataType" festgelegt werden. Die Eigenschaft "Ausdruck" kann leer gelassen werden. In diesem Beispiel sollten die Werte des Parameters programmatisch weitergegeben werden.

Erstellen eines Parameters

Zum Erstellen eines Parameters wird das Element "Parameter" im "Daten"-Fenster mit der rechten Maustaste geklickt und das Element "Neuer Parameter" aus dem Kontextmenü ausgewählt:



Drücken von F2 und benennen des Parameternamens. Wechsel zum Fenster "Eigenschaften" und Festlegen der Parameter Eigenschaft "DataType".

Parameter können verschachtelt werden. Zum Erstellen eines verschachtelten Parameters wird ein Eltern-Parameter ausgewählt, mit der rechten Maustaste geklickt und das Element "Neuer Parameter" aus dem Kontextmenü ausgewählt:

Hier kann sich sowohl auf den Eltern-Parameter, wie auch auf den verschachtelten Parameter referenziert werden. Es besteht keine Beschränkung hinsichtlich der Verschachtelungsebenen.

Nutzung von Parametern in einem Bericht

Man kann, durch Nutzung eckiger Klammern, auf einen Parameter aus einem Ausdruck referenzieren:

```
[Parameter name]
```

Mittels dieser Methode wird auf einen verschachtelten Parameter referenziert:

```
[Parent parameter.Child parameter]
```

Da ein Parameter einen festgelegten Typ (dieser ist in der Eigenschaft DataType angezeigt) hat, sind mit Parametern dieselben Aktionen , wie mit Datentypen, erlaubt. Daher können Parameter des Typus Zeichenfolge folgendermaßen in einem Ausdruck genutzt werden:

```
[StringParameter].Substring(0, 2)
```

Lassen Sie uns ein Anwendungsbeispiel für Parameter anschauen. Angenommen wir haben einen Bericht der eine Tabelle der Mitarbeiter ("employees") druckt. Der Bericht soll dahingehend modifiziert werden, dass er Informationen über einen Mitarbeiter mit einer angegebenen Nummer druckt. Hierzu müssen die Daten in der Datenspalte "EmployeeID" gefiltert werden: Erstellen eines Parameters mit dem Namen "EmployeeID". Anzeige des Parametertypen Int32, da genau dieser Typ die "EmployeeID" Datenspalte hat. Zur Filterung des Mitarbeiters mit der angegebenen ID, muss der Leisteneditor "Daten" eingegeben und der folgende Ausdruck in der Registerkarte "Filter" angezeigt werden:

```
[Employees.EmployeeID] == [EmployeeID]
```

Zum Weiterreichen des Parameterwertes aus dem Programm an den Bericht wird der folgende Code genutzt:

```
report1.SetParameterValue("EmployeeID", 2);
```

Ausdrücke

An vielen Stellen in FastReport werden Ausdrücke genutzt. Das "Text"-Objekt kann beispielsweise Ausdrücke in eckigen Klammern beinhalten.

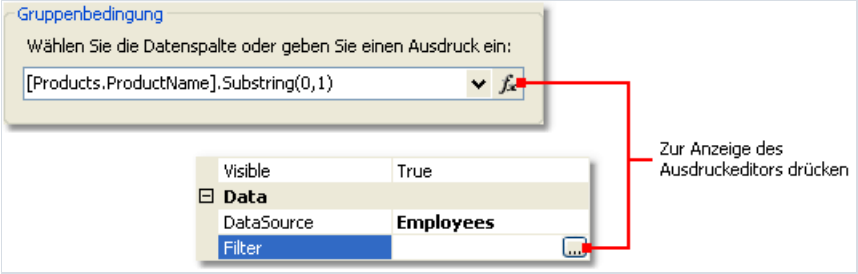
Ein Ausdruck ist ein Code in C# oder VB.Net der einen beliebigen Wert zurückgibt. Zum Beispiel:

```
2 + 2
```

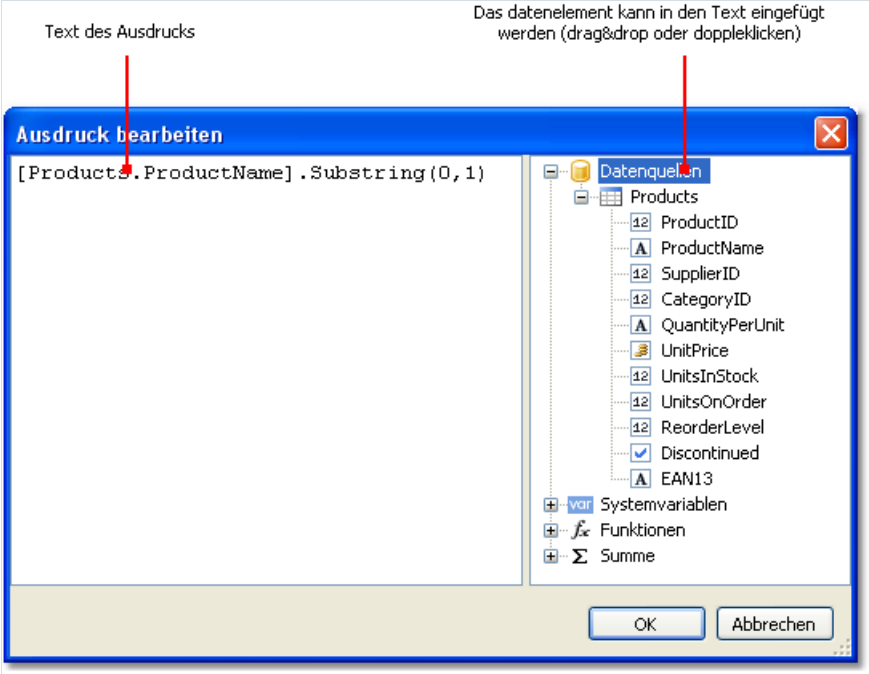
Ein Ausdruck sollte in einer Sprache geschrieben werden, die im Bericht als Skript gewählt wurde. Standardmäßig ist diese C#. Die Sprache kann im Menü "Bericht|Optionen..." durch Auswahl des Elementes "Script" in einem Fenster, geändert werden.

Ausdruck-Editor

Zum schnellen Verfassen eines Ausdrucks wird der Ausdruck-Editor genutzt. Er kann an solchen Stellen der FastReport-Benutzeroberfläche aufgerufen werden, wo ein Ausdruck eingegeben werden kann:



Der Ausdruck-Editor ist ein Fenster in das ein Ausdruck und einige Datenelemente eingegeben werden können:



Referenzierung zu einem Berichtsobjekt

Zur Referenzierung zu einem Berichtsobjekt wird der Objektname verwendet. Das folgende Beispiel gibt die Höhe des Text1 Objektes zurück.

```
Text1.Height
```

Zur Referenzierung von Berichtseigenschaften wird die Berichtsvariable genutzt. Das folgende Beispiel gibt den Dateinamen, aus dem ein Bericht geladen wurde, zurück.

```
Report.FileName
```

Es kann auch zu verschachtelten Objekteigenschaften referenziert werden. Das folgende Beispiel gibt einen Berichtsnamen zurück:

```
Report.ReportInfo.Name
```

Nutzung von .Net Funktionen

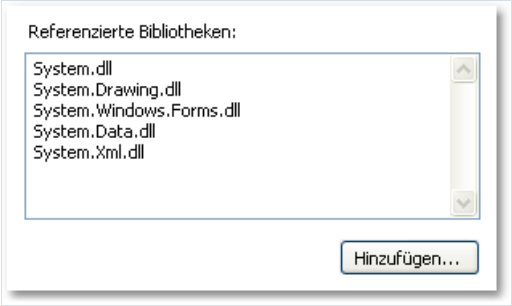
Alle .Net Objekte können in Ausdrücken verwendet werden. Das folgende Beispiel veranschaulicht die Nutzung der Max-Funktion:

```
Math.Max(5, 10)
```

Standardmäßig nutzt ein Bericht die folgenden .Net Assemblys.

```
System.dll
System.Drawing.dll
System.Windows.Forms.dll
System.Data.dll
System.Xml.dll
```

Es besteht Zugang zu allen in den .Net Assemblies aufgeführten Objekten. Sollte Bedarf nach einem weiteren Assembly bestehen, kann dieses in die Bericht Assembly-Liste hinzugefügt werden. Dies kann über das Menü "Bericht|Optionen..." durch Auswahl des "Script"-Elementes in dem Fenster vorgenommen werden:



Soll beispielsweise eine Funktion im Bericht, der in der Anwendung deklariert wurde, genutzt werden, wird die Assembly der Anwendnung (.exe oder .dll) in die Bericht Assembly-Liste hinzugefügt. Hiernach kann die Funktion mittels des Namespace der Anwendung aufgerufen werden. Wenn beispielsweise die folgende Funktion in der Anwendung definiert ist:

```
namespace Demo
{
  public static class MyFunctions
  {
    public static string Func1()
    {
      return "Hello!";
    }
  }
}
```

Diese kann im Bericht folgendermaßen genutzt werden:

```
Demo.MyFunctions.Func1()
```

Wird die Zeile "using Demo" im oberen Teil des Berichtskripts hinzugefügt, erlaubt dies eine Verkürzung der Syntax:

```
MyFunctions.Func1()
```

Zur Referenzierung der Funktion oder Variable die in einem Skript definiert wurde, wird lediglich der Name genutzt:

```
myPrivateVariableThatIHaveDeclaredInScript
MyScriptFunction()
```

> In einem Ausdruck können nur solche Funktionen genutzt werden die einen Wert zurückgeben.

Referenzierung zu Datenelementen

Neben den Standard-Sprachelementen, können die folgenden Berichtselemente in Ausdrücken genutzt werden:

· Datenquellenspalten;

· Systemvariablen;

· Summenwerte;

· Berichtparameter.

Alle diese Elemente sind im Fenster "Daten" zu finden. Weitere Details hierzu werden im Kapitel "Daten" erläutert. Jedes dieser Element kann in einem Ausdruck genutzt werden, indem es in die eckigen Klammern integriert wird. Zum Beispiel:

```
[Page] + 1
```

Dieser Ausdruck gibt die nächste gedruckte Seitenzahl zurück. Dieser Ausdruck nutzt eine Systemvariable "Page" die die aktuelle Berichtseitenzahl zurückgibt. Diese ist in den eckigen Klammern eingeschlossen.

Referenzierung zu Datenquellen

Zur Referenzierung zu Datenquellen wird das folgende Format genutzt:

```
[DataSource.Column]
```

Der name der Quelle wird von dem Spaltennamen durch ein Komma getrennt, z.B.:

```
[Employees.FirstName]
```

Wenn wir zur Datenuelle durch Nutzung einer Relation referenzieren, kann im Fall der Quellname ein Verbund sein. Weitere Details hierzu werden im Kapitel "Daten" erläutert. Hier ein Beispiel wie zu einer verwandten Datenquellenspalte referenziert werden kann:

```
[Products.Categories.CategoryName]
```

Das folgende Beispiel illustriert die Nutzung von Spalten in einem Ausdruck:

```
[Employees.FirstName] + " " + [Employees.LastName]
```

Bitte beachten: Jede Spalte hat einen bestimmten Datentyp, der in der Eigenschaft "DatenType" (einsehbar vorher mittels des Fensters "Eigenschaften" bei Auswahl Datenspalte im "Daten"-Fenster) festgelegt ist. Die Nutzung der Spalte in einem Ausdruck, hängt von dessen Art ab. Im obigen Beispiel sind beide Spalten (Vorname und Zuname) vom Typ Zeichenfolge und können daher auf diese Art genutzt werden. Im folgenden Beispiel wird die Spalte "Employee.Age" des numerischen Typen genutzt und zu einem Fehler führen:

```
[Employees.FirstName] + " " + [Employees.Age]
```

Dieser Fehler tritt auf, da Zeichenfolgen niemals mit Nummern gemischt werden können. Hierzu muss die Nummer erst in eine Zeichenfolge umgewandelt werden.

```
[Employees.FirstName] + " " + [Employees.Age].ToString()
```

In einem solchen Fall wird die Spalte "Employee.Age" behandelt, als wäre sie eine Integer Variable. Und das ist sie auch. Wir wissen, dass alle Ausdrücke verbunden sind. Alle nicht standardisierten Aktionen (wie eine Referenzierung zu Datenspalten) werden aus Sicht des Compilers in andere Typen, die von einem Compiler verstanden werden können, umgewandelt. Der letzte Ausdruck wird daher in folgende Form umgewandelt:

```
(string)(Report.GetColumnValue("Employees.FirstName")) + " " +
(int)(Report.GetColumnValue("Employees.Age")).ToString()
```

Wie man hier sehen kann, ändert FastReport die Referenzierung zu den Datenspalten folgendermaßen:

`[Employees.FirstName]` --> `(string)(Report.GetColumnValue("Employees.FirstName"))`

`[Employees.Age]` --> `(int)(Report.GetColumnValue("Employees.Age"))`

Die Datenspalten in einem Ausdruck können so benutzt werden als seinen sie eine Variable eines definitiven Typus. Der folgende Ausdruck gibt beispielsweise das Symbol eines Mitarbeiternamens zurück:

```
[Employees.FirstName].Substring(0, 1)
```

Referenzierung zu Systemvariablen

Die folgenden Systemvariablen können in Ausdrücken (zugänglich über das Fenster "Daten") genutzt werden:

Variable	.Net Datenart	Beschreibung	
Date	DateTime	Datum und Zeit des Berichtsstarts	
Page	int	Aktuelle Seitenzahl.	
TotalPages	int	Gesamtanzahl der Seiten im Bericht. Zur Nutzung dieser Variable muss "Double Pass" im Bericht aktiviert werden. Dies kann über das Menü "Bericht	Eigenschaften..." erfolgen:
PageN	string	Seitenzahl in der Form: "Seite N".	
PageNofM	string	Seitenzahl in der Form: "Seite N von M".	
Row#	int	Zeilennummer der Daten innerhalb der Gruppe. Dieser Wert wird zu Beginn einer neuen Gruppe zurückgesetzt.	
AbsRow#	int	Absolute Anzahl an Datenzeilen. Dieser Wert wird zu Beginn einer neuen Gruppe niemals zurückgesetzt.	

Jede Variable weist eine bestimmte Datenart auf. Von dieser hängt ab, wie sie in dem Ausdruck genutzt werden kann. Hier ein Beispiel eines Ausdrucks in dem ein Datum genutzt wird:

```
[Date].Year
```

Dieser Ausdruck gibt das aktuelle Jahr zurück. Da die Variable "Date" den DateTime-Typus aufweist, kann zu deren Eigenschaft "Year" referenziert werden. Der aktuelle Monat kann auf ähnliche Weise ([Date].Month) erhalten werden.

FastReport wandelt die Referenzierung zu der Systemvariablen (z.B. die Variable "Date") auf die folgende Art um:

```
((DateTime)Report.GetVariableValue("Date"))
```

Referenzierung zu Summenwerten

Um zu einem Summenwert zu referenzieren wird dessen Name genutzt:

```
[TotalSales]
```

FastReport wandelt die Referenzierung zu Summen auf die folgende Art um:

```
Report.GetTotalValue("TotalSales")
```

Wie man sehen kann, wird die Datenart hier nicht genutzt. Dies liegt daran, dass der Summenwert von der Art FastReport.Variant ist. Dieser kann direkt in jedem Ausdruck genutzt werden, da er automatisch in jeden Typus konvertiert wird. Zum Beispiel:

```
[TotalSales] * 0.2f
```

Referenzierung zu Berichtparametern

Um zu einem Berichtparameter zu referenzieren wird dessen Name genutzt:

```
[Parameter1]
```

Parameter können verschachtelt werden. In einem solchen Fall sollten sowohl die Namen der Eltern-, wie auch die Kindparameter folgendermaßen genutzt werden:

```
[ParentParameter.ChildParameter]
```

Parameter haben eine eindeutigen Datenart. Diese wird in der Eigenschaft "DatenType" des Parameters festgelegt. Wie diese in einem Ausdruck genutzt werden kann hängt von der Datenart des Parameters ab.

FastReport wandelt die Referenzierung zu Berichtparametern auf die folgende Art um:

```
((string)Report.GetParameterValue("Parameter1"))
```

Script

Skript ist eine Programmiersprache höherer Ordnung, die Teil eines Berichtes ist. Das Skript kann in einer der folgenden .Net-Sparchen geschrieben werden:

· C#

· VisualBasic.Net

Ein Skript kann in einer Vielzahl von Positionen angewandt werden. Mit Hilfe eines Skriptes kann Folgendes durchgeführt werden:

· Durchführung der Datenhandhabung, die nicht mittels der regulären Mittel des FastReport-Engine durchgeführt werden kann;

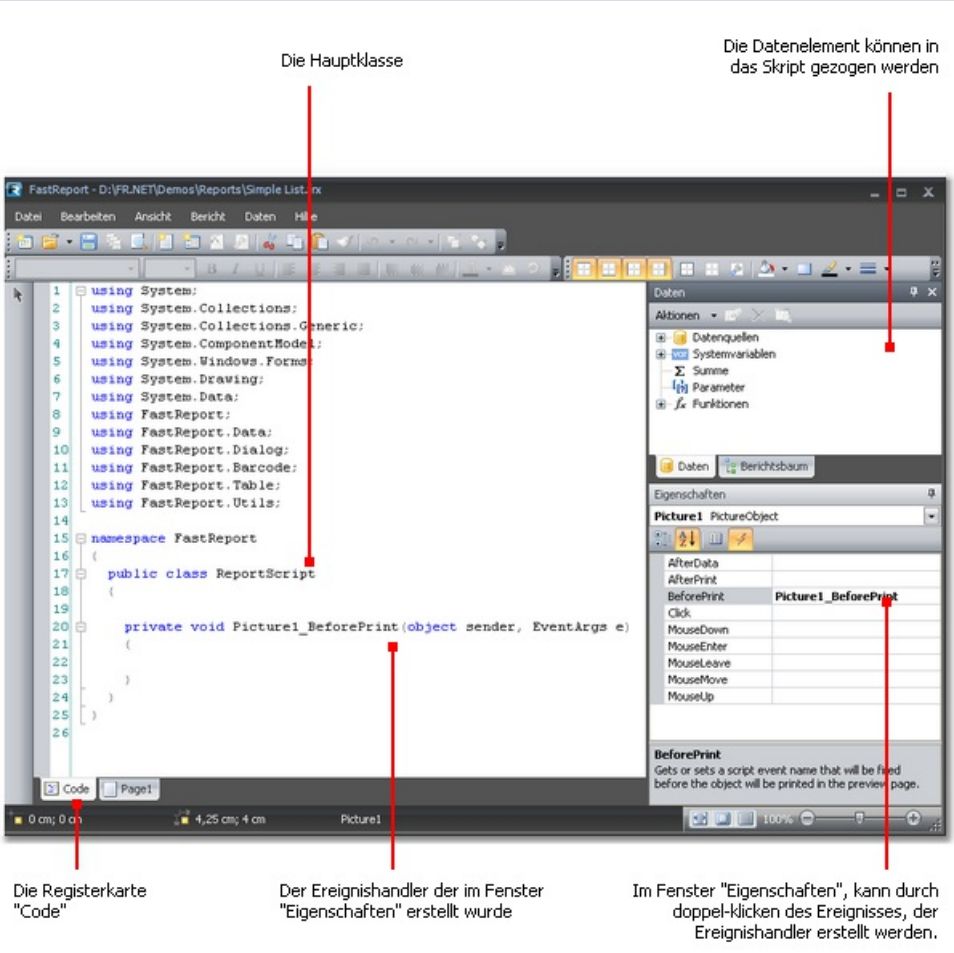
· Steuerung des Drucks der Berichtsseiten und der Leisten auf den Seiten;

· Steuerung der Interaktion zwischen den Elementen in Dialogformularen;

· Steuerung der Formatierung dynamischer "Tabellen"-Objekte;

· und vieles mehr.

Zur Ansicht des Skripts des Berichts wird zur Registerkarte "Code" im Designer gewechselt:



Die Skript-Sprache kann über das Menü "Bericht|Optionen" festgelegt werden. Dies wird nach der Erstellung eines neuen Berichtes durchgeführt, da die Änderung der Sprache zum Verlust des vorhandenen Skriptes führt.

Berichtsoptionen
Allgemein
Beschreibung
Script
Sicherheit
Vererbung
Email
Script-Sprache
C#
VisualBasic.NET
Hinweis: Wenn die Sprache geändert wird, wird das Script gelöscht!
Referenzierte Bibliotheken:
System.dll
System.Drawing.dll
System.Windows.Forms.dll
System.Data.dll
System.Xml.dll
Hinzufügen...
OK
Abbrechen

Allgemeine Informationen

Im Gegensatz zu anderen Berichtsgeneratoren enthält FastReport lediglich das was Sie geschrieben haben. Mit Hilfe des Skripts können/kann:

· Variablen, Methoden und Eigenschaften zur Skript-Klasse hinzugefügt werden;

· ein Eventhandler für die Berichtsobjekte erstellt werden;

· dem Skript, bei Bedarf, neue Klassen hinzugefügt werden. Eine Klasse kann entweder vor oder nach der Hauptklasse des ReportScript hinzugefügt werden.

Man kann nicht:

· die Sichtbarkeitsbereiche der Hauptklasse ReportScript löschen, umbenennen oder ändern;

· ein Namespace, das in der Hauptklasse positioniert ist, umbenennen.

Wird der Bericht durchgeführt tritt das Folgende auf:

· FastReport fügt dem Skript eine Liste von Variablen hinzu, dessen Namen den Namen der Berichtsobjekte entsprechen. Dies wird vor der Zusammenstellung des Skript durchgeführt und erlaubt die Berichtsobjekte durch deren Namen zu referenzieren;

· dem Skript wird ein Ausdruckshandler, der alle im Bericht gefundene Ausdrücke verwaltet, hinzugefügt;

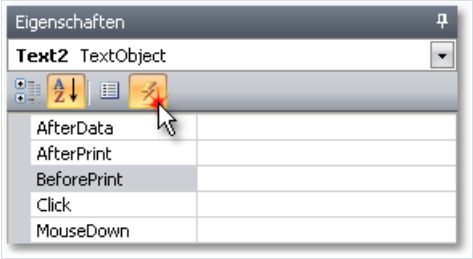
· ein Skript ist zusammengestellt, wenn er nicht leer ist;

· eine Skriptklasse ist initialisiert;

· der Bericht wird durchgeführt.

Eventhandler

Ein Skript wird hauptsächlich zur Erstellung eines Eventhandlers des Objektes genutzt. Zur Erstellung eines Eventhandlers wird das benötigte Objekt ausgewählt. Um die Liste der Ereignisse anzustellen, wird im Fenster "Eigenschaften" die Schaltfläche gedrückt:



Auswahl des gewünschten Ereignisses und doppelklicken. FastReport fügt einen leeren Ereignishandler in den Berichtscode ein:

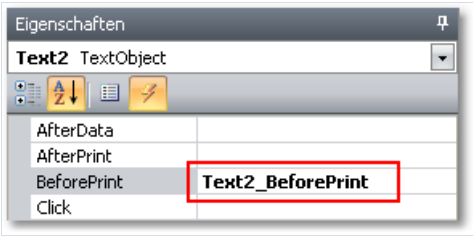
```
private void Text2_BeforePrint(object sender, EventArgs e)
{

}
```

Das "Bericht"-Objekt hat auch Ereignisse. Dieses Objekt kann auf die folgende Art ausgewählt werden:

· Auswahl von "Bericht" im Fenster "Berichtsbaum";

· Auswahl von "Bericht" in der Dropdownliste des Fensters "Eigenschaften".

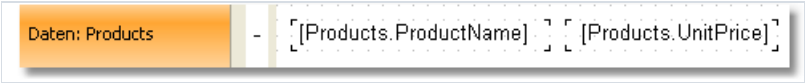
Zum Löschen des Eventhandlers, wird ein Ereignis im Fernster "Eigenschaften" ausgewählt, der Text ausgewählt und die Taste Löschen gedrückt:



Berichtereignisse

Um den Bericht so flexibel wie möglich zu steuern hat jedes Berichtsobjekt mehrere Ereignisse. In einem Handler, der mit der "Daten"-Leiste verbunden ist, können beispielsweise Berichte gefiltert werden, d.h. verbergen oder anzeigen von Leisten abhängig von bestimmten Bedingungen.

Lassen Sie uns die Ereignisse betrachten, die während des Berichtserstellungsprozesses generiert werden. Nehmen wir beispielsweise einen einfachen Bericht, der eine Seite umfasst, mit einer "Daten"-Leiste und zwei "Text"-Objekten auf der Leiste:



Zu Beginn des Berichts generiert das "Bericht"-Objekt das Ereignis StartReport. Das StartPage Ereignis wird vor der Bildung der Berichtseite generiert. Dieses Ereignis wird einmal für jede Vorlageseite (nicht mit der erstellten Berichtsseite verwechseln!) generiert. In unserem Fall, ungeachtet der Anzahl der Seiten des erstellten Berichts, wird das Ereignis einmal generiert, da die Vorlageseite aus einer Seite besteht.

Des weiteren beginnt der Druck der "Daten"-Leiste. Dies geschieht folgendermaßen:

1. das Ereignis BeforePrint Leiste wird generiert;
2. das Ereignis BeforePrint aller Ereignisse, die auf dem Objekt liegen, wird generiert;
3. alle Objekte werden mit Daten gefüllt;
4. das Ereignis AfterData aller Ereignisse, die auf dem Objekt liegen, wird generiert;
5. das Ereignis BeforeLayout Leiste wird generiert;
6. Objekte werden auf der Leiste platziert, die Höhe der Leiste berechnet und das Band gestreckt (wenn dies möglich ist);
7. das Ereignis AfterLayout Leiste wird generiert;
8. wenn die Leiste nicht auf den freien Platz auf der Seite passt, wird eine neue Seite erstellt;
9. die Liste und all ihre Objekte werden auf der erstellten Berichtseite angezeigt;
10. das Ereignis AfterPrint Leiste wird generiert;
11. das Ereignis AfterPrint aller Leistenobjekte wird generiert.

Der Druck der Leistenzeile wird so lange fortgesetzt wie Daten in der Quelle existieren. Für dieses Beispiel endet hiernach die Erstellung des Berichts. Das Ereignis FinishPage einer Seite wird generiert und letztendlich das Ereignis FinishReport des "Bericht"-Objektes.

Durch Nutzung von Ereignissen unterschiedlicher Objekte kann jeder Schritt der Berichterstellung gesteuert werden. Der Schlüssel zur korrekten Nutzung von Ereignissen: ein vollständiges Verständnis des Druckprozesses der Leisten, wie in den 11 Schritten (siehe oben) beschrieben. So lassen sich eine Vielzahl von Operationen lediglich durch Nutzung der Leiste BeforePrint durchführen. Auch jede an dem Objekt durchgeführte Änderung, wird angezeigt. In diesem Ereignis ist es nicht möglich zu analysieren auf welcher Seite die Leiste gedruckt wird, weil die Höhe des Bandes in Schritt 6 berechnet wird. Dies kann mittels des Ereignisses Afterlayout Leiste in Schritt 7 erledigt werden oder AfterPrint in Schritt 10. Im letzteren Fall ist die Leiste jedoch bereits gedruckt und die Operation mit Objekten wird keine Ergebnisse liefern. Es muss klar herausgestellt werden zu welchem Zeitpunkt

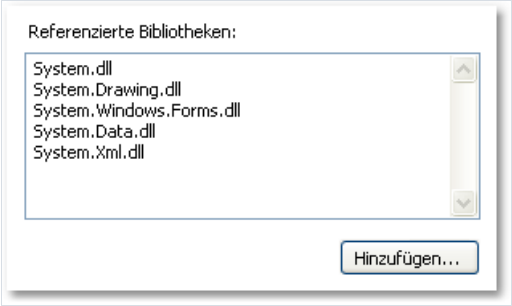
jedes Ereignis generiert und benutzt wird und es müssen solche genutzt werden, die mir der gegebenen Aufgabe korrespondieren.

Nutzung von .Net Objekten

In einem Skript kann jedes .Net-Objekt, das in den folgenden Assemblies definiert ist, genutzt werden:

```
System.dll
System.Drawing.dll
System.Windows.Forms.dll
System.Data.dll
System.Xml.dll
```

Darüber hinaus kann jedes Objekt, das in der Assembly von FastReport definiert ist, genutzt werden. Soll ein anderes Assembly zugänglich sein, kann es zur Liste der Assemblies hinzugefügt werden. Dies kann im Menü "Bericht|Optionen..." durch Auswahl der Registerkarte "Script" vorgenommen werden:



Soll beispielsweise eine Funktion in Ihrem Bericht, der in der Anwendung deklariert wurde, genutzt werden, wird die Assembly der Anwendnung (.exe oder .dll) der Bericht Assembly-Liste hinzugefügt. Hiernach kann die Funktion mittels des Namespace der Anwendung aufgerufen werden. Wenn beispielsweise die folgende Funktion in der Anwendung definiert ist:

```
namespace Demo
{
  public static class MyFunctions
  {
    public static string Func1()
    {
      return "Hello!";
    }
  }
}
```

Findet ein Aufruf im Skript folgendermaßen statt:

```
string hello = Demo.MyFunctions.Func1();
```

Wird die Zeile "using Demo" im oberen Teil des Berichtskripts hinzugefügt, erlaubt dies eine Verkürzung der Syntax:

```
string hello = MyFunctions.Func1();
```

Referenzierung zu einem Berichtsobjekt

Zur Referenzierung zu Berichtsobjekten (zum Beispiel "Text"-Objekt) wird der Name des Objektes genutzt. Das folgende Beispiel gibt die Höhe des Text1 Objektes zurück.

```
float height = Text1.Height;
```

Beachten Sie: die native Messeinheit des Berichts sind Bildschirmpixel. Dies sollte bei der Nutzung solcher Objekteigenschaften wie Left, Top, Width und Height beachtet werden. Zur Umwandlung von Pixel in Zentimeter und umgekehrt, werden die in der Klasse "Units" definierten Konstanten genutzt:

```
float heightInPixels = Text1.Height;
float heightInCM = heightInPixels / Units.Centimeters;
Text1.Height = Units.Centimeters * 5; // 5см
```

Report-, und Engine-Objekte

Neben den in dem Bericht beinhalteten Objekten sind zwei Variablen im Skript definiert: Report und Engine.

Die Report-Variable bezieht sich auf den aktuellen Bericht. Die folgende Liste zeigt eine Liste der Objektmethoden:

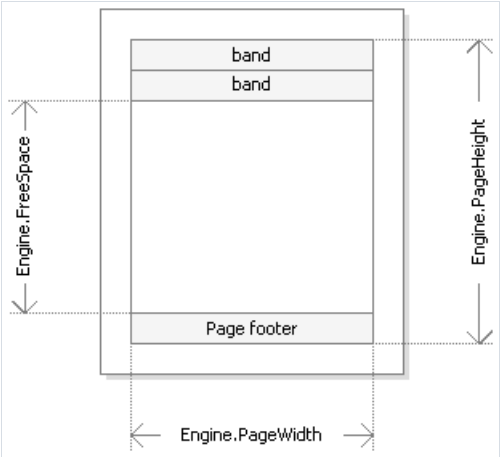
Methode	Beschreibung
`object Calc(string expression)`	Ein Ausdruck wird berechnet und der Wert zurückgegeben. Der erste Aufruf dieser Methode ist zeitintensiv, da ein Audruck erstellt werden muss.
`object GetColumnValue(string complexName)`	Gibt den Wert der Datenspalte zurück. Der Name wird in der Form "DataSource.Column" präsentiert. Hat die Spalte einen Null Wert, wird es standardmäßig in einen Wert (0, leere Zeichenfolge, false) umgewandelt.
`object GetColumnValueNullable(string complexName)`	Gibt den Wert der Datenspalte zurück. Im Gegensatz zur vorherigen Methode wird dieser nicht in einen Standardwert umgewandelt und kann Nullsein.
`Parameter GetParameter(string complexName)`	Gibt den Berichsparameter mit dem angegebenen Namen zurück. Der Name kann, wenn zu dem verschachtelten Parameter referenziert wird, verbunden werden: "MainParam.NestedParam".
`object GetParameterValue(string complexName)`	Gibt den Wert des Berichtsparameters mit dem angegebenen Namen zurück.
`void SetParameterValue(string complexName, object value)`	Legt den Wert des Berichtsparameters mit dem angegebenen Namen fest.
`object GetVariableValue(string complexName)`	Gibt den Wert der Systemvariable, z.B. "Date", zurück.
`object GetTotalValue(string name)`	Gibt den Wert der Summe, definiert im Fenster "Daten", nach seinem Namen zurück.
`DataSourceBase GetDataSource(string alias)`	Gibt die Datenquelle, definiert im Bericht, nach Namen zurück.

Das Engine-Objekt ist ein Engine, der die Berichterstellung steuert. Der Prozess der Platzierung von Leisten auf der Seite kann durch die Nutzung der Methoden und Eigenschaften des Engine gesteuert werden. Die folgenden Eigenschaften des Engine-Objektes können genutzt werden:

Eigenschaft	Beschreibung
`float CurX`	Aktuelle Koordinaten auf der X-Achse. Dieser Eigenschaft kann ein Wert zugeordnet werden, um das gedruckte Objekt zu verschieben.
`float CurY`	Aktuelle Druckposition auf der Y-Achse. Dieser Eigenschaft kann ein Wert zugeordnet werden, um das gedruckte Objekt zu verschieben.
`int CurColumn`	Anzahl der aktuellen Spalten in einem mehrspaltigen Bericht. Die erste Spalte hat die Nummer 0.

Eigenschaft	Beschreibung
`int CurPage`	Nummer auf der gedruckten Seite. Dieser Wert kann aus der Systemvariable "Page" erhalten werden.
`float PageWidth`	Die Seitenbreite minus der Größe der rechten und linken Seitenränder.
`float PageHeight`	Die Seitenhöhe minus der Größe der oberen und unteren Seitenränder.
`float PageFooterHeight`	Die Höhe der Fußzeile (und aller Kind-Leisten).
`float ColumnFooterHeight`	Die Höhe des Spaltenfußes (und aller Kind-Leisten).
`float FreeSpace`	Die Größe des freien Platzes auf der Seite.
`bool FirstPass`	Gibt true zurück, wenn der erste (oder einzige) Berichtsdurchlauf durchgeführt wird. Die Anzahl der Durchläufe kann aus der Eigenschaft Report.DoublePass abgerufen werden.
`bool FinalPass`	Gibt true zurück, wenn der letzte (oder einzige) Berichtsdurchlauf durchgeführt wird.

OnIn der folgenden Abbildung lässt sich die Bedeutung einiger oben aufgeführter Eigenschaften einsehen.



Die Eigenschaften Engine.PageWidth und Engine.PageHeight bestimmen die Größe des Druckbereichs. Diese ist in fast allen Fällen geringer als die aktuelle Seitengröße. Die Größe des Druckbereichs mittels der Seitenränder bestimmt, die in den Seiteneigenschaften LeftMargin, TopMargin, RightMargin und BottomMargin angezeigt sind.

Die Eigenschaft Engine.FreeSpace bestimmt die Höhe des freien Platzes auf der Seite. Befindet sich die Leiste "Seitenfuß" auf der Seite, wird dessen Höhe bei der Berechnung des freien Platzes in Betracht gezogen. Bitte beachten: Nach dem Druck einer Leiste verringert sich der freie Platz.

Wie wird eine Seite des erstellen Berichts erstellt? Der Engine von FastReport zeigt Bänder auf der Leiste an, bis der Platz für den Leisten-Output gefüllt ist. Besteht kein weiterer Platz mehr, wird die Leiste "Berichtsfußzeile" gedruckt und eine neue Seite erstellt. Die Anzeige einer Leiste beginnt in der aktuellen Position, die durch die X-, und Y-Koordinaten bestimmt wird. Diese Position wird von den Eigenschaften Engine.CurX und Engine.CurY zurückgegeben. Nach dem Druck eines Bandes vergrößert CurY automatisch die Höhe der gedruckten Leiste. Nach der Erstellung einer neuen Seite wird die Position der CurY auf 0 zurückgesetzt. Die Position der CurX ändert sich beim Druck eines mehrspaltigen Berichts.

Die Eigenschaften Engine.CurX und Engine.CurY sind nicht nur zum Lesen, sondern auch zum Schreiben zugänglich. Dies bedeutet, dass eine Leiste manuell durch Nutzung eines geeigneten Ereignisses, umgeschaltet

werden kann. Beispiele für die Nutzung dieser Eigenschaft können im Abschnitt "Beispiele" eingesehen werden.

> Bei der Arbeit mit Eigenschaften, die die Größe oder Position zurückgeben, ist es wichtig zu beachten, dass diese Eigenschaften in Bildschirmpixeln gemessen werden.

Im Engine-Objekt werden die folgenden Methoden definiert:

Methode	Beschreibung
`void AddOutline(string text)`	Fügt der Berichtsgliederung ein neues Element hinzu (lesen Sie hierzu das Kapitel "Interaktive Berichte") und legt die aktuelle Position auf das hinzugefügte Element.
`void OutlineRoot()`	Setzt die aktuelle Position auf den Stamm der Gliederung.
`void OutlineUp()`	Ändert die aktuelle Position zu einem Gliederungselement höherer Ebene.
`void AddBookmark(string name)`	Fügt ein Lesezeichen hinzu (mehr hierzu im Kapitel "Interaktive Berichte").
`int GetBookmarkPage(string name)`	Gibt die Seitenzahl zurück, auf der das Lesezeichen mit dem angegebenen Namen positioniert ist.
`void StartNewPage()`	Beginnt eine neue Seite. Ist der Bericht mehrspaltig wird eine neue Spalte begonnen.

Die Berichtsgliederung kann auch durch Nutzung der Methode AddOutline, OutlineRootm, OutlineUp manuell erstellt werden. Üblicherweise wird dies automatisch, mittels der Eigenschaft OutlineExpression, die sich auf jeder Leiste und Berichtseite befindet, erstellt.

Die AddOutline Methode fügt das Kindelement zu dem aktuellen Gliederungselement hinzu und macht diese aktuell. Sowohl die aktuelle Berichtsseite, als auch die aktuelle Position, sind mit dem neuen Element assoziiert. Wird die AddOutline Methode mehrere Male aufgerufen, erhält man die folgende Struktur:

```
Item1
    Item2
        Item3
```

Zur Kontrolle des aktuellen Elementes gibt es OutlineUp-, und OutlineRoot-Methoden. Die erste Methode verschiebt den Zeiger auf das Element, dass sich auf der höheren Ebene befindet. Daher resultiert das Skript

```
Engine.AddOutline("Item1");
Engine.AddOutline("Item2");
Engine.AddOutline("Item3");
Engine.OutlineUp();
Engine.AddOutline("Item4");
```

in der folgenden Gliederung:

```
Item1
    Item2
        Item3
        Item4
```

Die OutlineRoot-Methode bewegt das aktuelle Element zum Stamm der Gliederung. Z.B. erstellt das folgende Skript:

```
Engine.AddOutline("Item1");
Engine.AddOutline("Item2");
Engine.AddOutline("Item3");
Engine.OutlineRoot();
Engine.AddOutline("Item4");
```

die folgende Gliederung:

```
Item1
    Item2
        Item3
Item4
```

Zur Arbeit mit Lesezeichen werden die Methoden AddBookmark und GetBookmarkPage des Engine-Objektes genutzt. Üblicherweise werden die Lesezeichen automatisch bei allen im Bericht vorhandenen Objekten im Bericht hinzugefügt.

Durch Nutzung der Methode Lesezeichen hinzufügen lässt sich ein Lesezeichen programmatisch hinzufügen. Diese Methode erstellt ein Lesezeichen auf der aktuellen Seite in der aktuellen Position.

Die Methode GetBookmarkPage gibt die Seitenzahl in der das Lesezeichen positioniert ist zurück. Diese Methode wird häufig zur Anzeige von Seitenzahlen bei der Erstellung eines Inhaltsverzeichnisses genutzt. In diesem Fall muss der Bericht einen Double Pass haben.

Referenzierung zu Datenquellen

Im Gegensatz zu den Ausdrücken in FastReport (im Abschnitt "Ausdrücke" behandelt) werden hier im Skript niemals eckige Klammern zur Referenzierung der Datenquelle genutzt. Anstelle dessen wird die Methode GetColumnValue des Berichtsobjektes, das den Spaltenwert zurückgibt, genutzt:

```
string productName = (string)Report.GetColumnValue("Products.Name");
```

Hier lässt sich erkennen, dass der Name der Quelle und seine Spalte angegeben werden muss. Im Fall kann der Quellname ein Verbund sein, wenn wir zur Datenuelle durch Nutzung einer Relation referenzieren. Detaillierte Informationen über Beziehungen sind im Kapitel "Daten" erläutert. Zu einer Spalte der verwandten Datenquelle kann beispielsweise auf diese Art referenziert werden:

```
string categoryName = (string)Report.GetColumnValue("Products.Categories.CategoryName");
```

Zu Vereinfachung wird das Fenster "Daten" genutzt. Aus diesem lassen sich die Datenelemente in das Skript ziehen. FastReport erstellt automatisch einen Code der auf das Element verweist.

Zur Referenzierung auf die Datenquelle selbst wird die Methode GetDataSource des Berichtsobjektes genutzt:

```
DataSourceBase ds = Report.GetDataSource("Products");
```

Hilfe zu Eigenschaften und Methoden der Klasse DataSourceBase kann über das Fastreport.Net Klassenreferenz Hilfesystem abgerufen werden. In der Regel wir dieses Objekt im Skript auf folgende Art genutzt:

```
// get a reference to the data source
DataSourceBase ds = Report.GetDataSource("Products");
// initialize it
ds.Init();
// enum all rows
while (ds.HasMoreRows)
{
  // get the data column value from the current row
  string productName = (string)Report.GetColumnValue("Products.Name");
  // do something with it...
  // ...
  // go next data row
  ds.Next();
}
```

Referenzierung zu Systemvariablen

Zur Referenzierung zu Systemvariablen wird die Methode GetVariableValue des Berichtsobjektes genutzt:

```
DateTime date = (DateTime)Report.GetVariableValue("Date");
```

Eine Liste der Systemvariablen kann im Fenster "Daten" eingesehen werden. Aus dieser lässt sich die Variable in das Skript ziehen. FastReport erstellt automatisch einen Code der auf das Element verweist.

Referenzierung zu Summenwerten

Zur Referenzierung zum Summenwert wird die Methode GetTotalValue des Berichtsobjektes genutzt:

```
float sales = Report.GetTotalValue("TotalSales");
```

Eine Liste der Summen kann im Fenster "Daten" eingesehen werden. Aus dieser lässt sich eine Summe in das Skript ziehen. FastReport erstellt automatisch einen Code der auf die Summe verweist.

Der Summenwert hat den Typus FastReport.Variant. Dieser kann direkt in jedem Ausdruck genutzt werden, da der Typus FastReport.Variant automatisch in jeden Typus konvertiert wird. Zum Beispiel:

```
float tax = Report.GetTotalValue("TotalSales") * 0.2f;
```

Eine Referenzierung zum Summenwert kann zur dem Zeitpunkt hergestellt werden an dem dieser bearbeitet wird. Die Summe ist üblicherweise zu dem Zeitpunkt "gebrauchsfertig", an dem es im Bericht positioniert ist.

Referenzierung zu Berichtparametern

Zur Referenzierung zu Berichtparametern wird die Methode GetParameterValue des Berichtsobjektes genutzt:

```
int myParam = (int)Report.GetParameterValue("MyParameter");
```

Parameter können verschachtelt werden. In einem solchen Fall wird der Name des Elternparameters angezeigt und nach diesem Zeitraum der Name des Kindparameters.

```
Report.GetParameterValue("ParentParameter.ChildParameter")
```

Parameter weisen eine bestimmte Datenart auf. Diese wird in der Eigenschaft "DatenType" des Parameters angezeigt. Sie muss bei der Referenzierung zu Parametern in Betracht gezogen werden. Eine Liste an Parametern kann im Fenster "Daten" eingesehen werden. Aus dieser lassen sich Parameter in das Skript ziehen. FastReport erstellt automatisch einen Code der auf die Parameter verweist.

Zur Änderung des Wertes des Parameters wird die Methode SetParameterValue des Berichtsobjektes genutzt:

```
Report.SetParameterValue("MyParameter", 10);
```

Beispiele

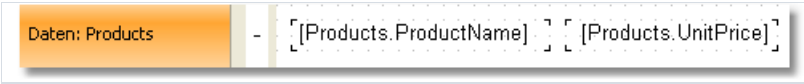
Beispiel 1: Änderung des Aussehens des Objektes

In diesem Beispiel wird erläutert wie die Textfarbe, abhängig von dem Wert der in das Objekt gedruckt wurde, geändert werden kann. Wir nutzen:

· das Ereignis BeforePrint;

· Referenzierung zur Datenspalte des Skriptes.

Erstellen eines einfachen Berichts mit dem folgenden Aussehen:



Auswahl des Objektes, das die Spalte "UnitPrice" druckt und erstellen eines Ereignishandlers BeforePrint:

```
private void Text2_BeforePrint(object sender, EventArgs e)
{
  if (((Decimal)Report.GetColumnValue("Products.UnitPrice")) > 20)
    Text2.TextColor = Color.Red;
}
```

Um die Datenspalte "Products.UnitPrice" in das Skript einzufügen, wird es aus dem Fenster "Daten" gezogen. In Folge dessen wird die folgende Zeichenfolge in das Skript eingefügt:

```
((Decimal)Report.GetColumnValue("Products.UnitPrice"))
```

Bei der Durchführung des Berichtes, lässt sich erkennen, dass alle Produkte mit einem Preis > 20 in rot hervorgehoben werden:

Chai	18,00
Chang	19,00
Aniseed Syrup	10,00
Chef Anton's Cajun Seasoning	22,00
Chef Anton's Gumbo Mix	21,35
Grandma's Boysenberry Spread	25,00

Der gleiche Effekt kann mit Hilfe des bedingten Markierens erreicht werden (mehr hierüber lässt sich im Abschnitt "Bedingtes Hervorheben" des Kapitels "Erstellung des Berichts" nachlesen).

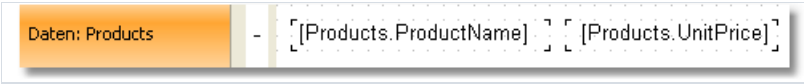
Beispiel 2: Hervorheben gerader Reihen einer Leiste

In diesem Beispiel wird erläutert wie die Füllfarbe der geraden Zeilen der "Daten"-Leiste geändert werden kann. Wir nutzen:

· das Ereignis BeforePrint;

· referenziert zu der Systemvariable "Row#" aus dem Skript.

Erstellen eines einfachen Berichts mit dem folgenden Aussehen:



Erstellen eines Ereignishandlers BeforePrint für die Leiste:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
  if (((Int32)Report.GetVariableValue("Row#")) % 2 == 0)
    Data1.FillColor = Color.Gainsboro;
}
```

Die Systemvariable "Row#" gibt die Anzahl der Zeilen der gedruckten Leiste zurück. Um eine Referenzierung zur Variablen in das Skript einzufügen, wird es aus dem Fenster "Daten" gezogen. Währenddessen wird in das Skript eine Zeichenfolge eingefügt:

```
((Int32)Report.GetVariableValue("Row#"))
```

Bei der Durchführung des Berichts erkennen wir, dass gerade Zeilen in hellgrau hervorgehoben werden:

Chai	18,00
Chang	19,00
Aniseed Syrup	10,00
Chef Anton's Cajun Seasoning	22,00
Chef Anton's Gumbo Mix	21,35

Der gleiche Effekt kann mittels der Eigenschaft "EvenStyle" der "Daten"-Leiste erreicht werden. Mehr hierzu kann im Abschnitt "Hervorheben gerader/ungerader Datenzeilen" des Kapitels "Berichterstellung" gelesen werden.

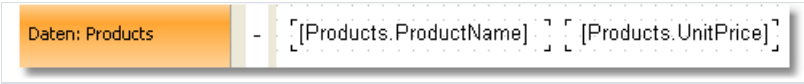
Beispiel 3: Datenfilterung

In diesem Beispiel wird demonstriert, wie die "Daten" Leistenzeile, abhängig von den gegebenen Bedingungen, verborgen werden kann. Wir nutzen:

· das Ereignis BeforePrint;

· Referenzierung zur Datenquelle von einem Skript.

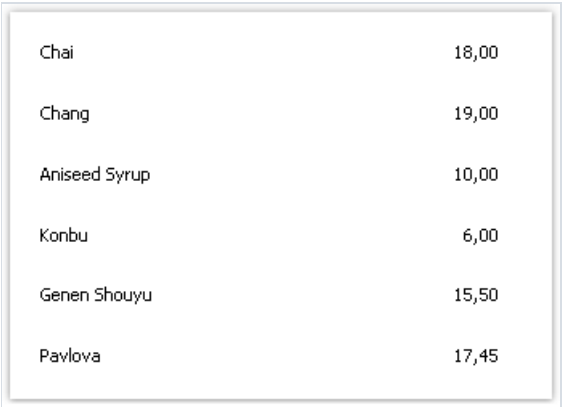
Erstellen eines einfachen Berichts mit dem folgenden Aussehen:



Erstellen eines Ereignishandlers BeforePrint für die Leiste:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
  if (((Decimal)Report.GetColumnValue("Products.UnitPrice")) > 20)
    Data1.Visible = false;
}
```

In diesem Beispiel werden die Leistenzeilen, dessen Stückpreis > 20 sind, verborgen:



Chai	18,00
Chang	19,00
Aniseed Syrup	10,00
Konbu	6,00
Genen Shouyu	15,50
Pavlova	17,45

Der gleiche Effekt kann durch Nutzung des Datenfilter, die im Editor "Daten"-Leiste festgelegt ist, erreicht werden.

Beispiel 4: Berechnung der Summe

In diesem Beispiels wird erläutert wie eine Summe durch Nutzung von Programmiermethoden erstellt werden kann. Hierzu nutzen wird die folgende Vorgehensweise:

· das Ereignis BeforePrint;

· Referenzierung zur Datenspalte von einem Skript;

· lokale Variable, dessen Wert im Bericht gedruckt wird.

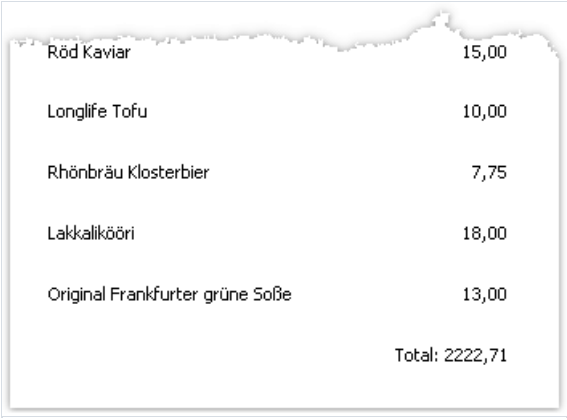
Erstellen eines Berichts der folgenden Form:

Deklaration der Variable "Summe" im Skript und erstellen eines Eventhandlers, der zur Leiste gehört:

```
public class ReportScript
{
  private decimal sum;

  private void Data1_BeforePrint(object sender, EventArgs e)
  {
    sum += (Decimal)Report.GetColumnValue("Products.UnitPrice");
  }
}
```

Durch Ziehen kann die Datenspalte "Products.UnitPrice" aus dem "Daten"-Fenster in das Skript positioniert werden.

Beim Durchlauf eines solchen Berichtes lässt sich das Folgende erkennen:



Der gleiche Effekt kann durch die Nutzung von Summen erreicht werden.

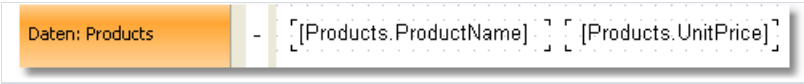
Beispiel 5: Ändern der Druckposition

In diesem Beispiel wird erläutert wie die Position der Leiste manuell, durch Nutzung des Engine-Objektes, geändert werden kann. Wir nutzen:

· das Ereignis BeforePrint;

· das Engine-Objekt.

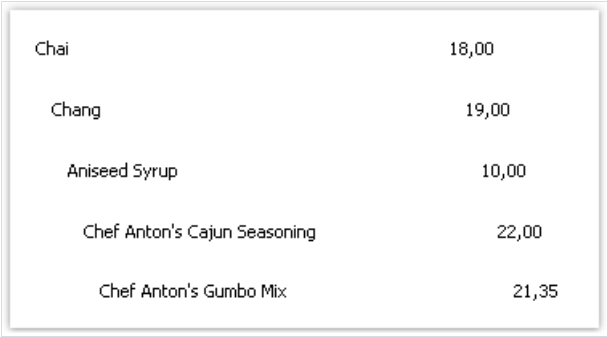
Erstellen eines einfachen Berichts mit dem folgenden Aussehen:



Erstellen eines Ereignishandlers BeforePrint für die Leiste:

```
private void Data1_BeforePrint(object sender, EventArgs e)
{
  Engine.CurX = ((Int32)Report.GetVariableValue("Row#")) * 10;
}
```

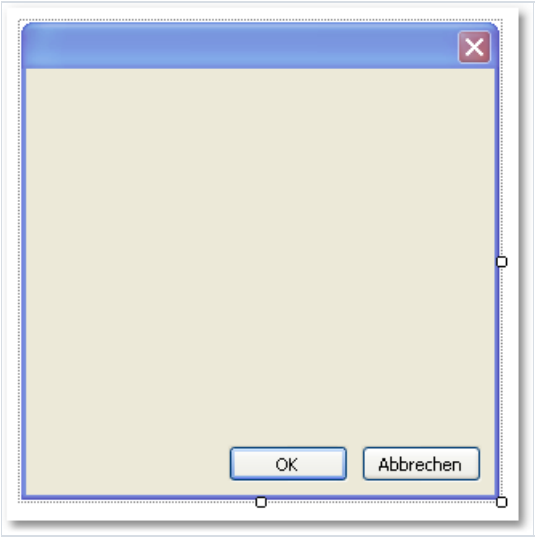
Beim Durchlauf eines solchen Berichtes lässt sich das Folgende erkennen:



Dialogfelder

Neben den regulären Seiten, kann ein Bericht eine oder mehrere Dialogfelder beinhalten. Derartige Dialogfelder werden beiom Start des Berichts angezeigt. In dem Dialogfeld kann jede Art von Information, die zur Erstellung des Berichts benötigt wird, eingegeben werden. Ein Dialog kann auch Daten, die im Bericht angezeigt werden, filtern.

Um einen Dialog in einen Bericht hinzuzufügen wird die Schaltfläche auf der Symbolleiste des Designers gedrückt. Ein neuer Dialog sieht folgendermaßen aus:



Ein Bericht, mit einem oder mehreren Dialogen funktioniert wie folgt:

· zum Start zeigt der Bericht den folgenden Dialog;

· wird der Dialog mit Hilfe der Schaltfläche "OK" geschlossen, wird der folgende Dialog angezeigt - sonst funktioniert der Bericht nicht mehr;

· nachdem alle Dialoge angezeigt wurden, wird der Bericht erstellt.

Steuerelemente

In einem Dialogfeld können die folgenden Steuerelemente genutzt werden:

Symbol	Name	Beschreibung
	ButtonControl	Repräsentiert ein Windows Schaltflächen Steuerelement.
	CheckBoxControl	Repräsentiert eine Windows CheckBox.
	CheckedListBoxControl	Anzeige eine ListBox, in der ein Kontrollkästchen auf der rechten Seite eine jeden Items angezeigt wird.
	ComboBoxControl	Repräsentiert ein Windows Kombinationsfeld-Steuerelement.
	DataGridViewControl	Anzeige von Daten in einem benutzerdefinierten Gitter.
	DataSelectorControl	Anzeige zweier Listen. Dies erlaubt die Neuplatzierung eines Elements von einer Liste in die andere.
	DateTimePickerControl	Repräsentiert ein Windows Steuerelement, dass es dem Nutzer erlaubt ein Datum und eine Zeit auszuwählen und dieses Datum und diese Zeit in einem festgelegten Format anzuzeigen.
	GroupBoxControl	Repräsentiert ein Windows Steuerelement, dass einen Rahmen um eine Gruppe von Steuerelementen, mit optionalen Beschriftungen, anzeigt.
	LabelControl	Repräsentiert ein Standard Windows-Label.
	ListBoxControl	Repräsentiert ein Windows Steuerelement zur Anzeige einer Liste von Elementen.
	ListViewControl	Repräsentiert eine Windows Listenansicht-Steuerelement, das eine Sammlung von Elementen, die mittels einer von vier verschiedenen Views angezeigt werden können, anzeigt.
	MaskedTextBoxControl	Nutzt eine Maske zur Unterscheidung zwischen ordnungsgemäßem und nicht ordnungsgemäßen Benutzereingaben.
	MonthCalendarControl	Repräsentiert ein Windows Steuerelement, dass es dem Nutzer ermöglicht, mittels eines sichtbaren monatlichen Kalenderanzeige, ein Datum auszuwählen.
	NumericUpDownControl	Repräsentiert ein Windows Drehfeld (auch als Dropdownfeld bekannt), dass numerische Werte anzeigt.
	PanelControl	Wird zur Sammlung von Steuerelementen genutzt.
	PictureBoxControl	Repräsentiert ein Windows Bildfeld Steuerelement zur Anzeige eines Bildes.

Symbol	Name	Beschreibung
	RadioButtonControl	Erlaubt dem Benutzer die Wahl einer einzelnen Option aus einer Gruppe an Möglichkeiten wenn es mit einem anderen RadioButton Steuerelement gepaart ist.
	RichTextBoxControl	Repräsentiert ein Windows Feld für Rich-Text Steuerelement.
	TextBoxControl	Repräsentiert ein Windows Textfeld Steuerelement.
	TreeViewControl	Anzeige einer hierarchischen Sammlung von beschrifteten Elementen.

Alle Steuerelemente, ausgenommen der DataSelectorControl, entsprechen den Standard Windows.Forms Steuerelementen. Zur Vermeidung duplizierter Namen haben alle Elementenamen ein Suffix für Steuerelemente. Der ButtonControl von FastReport korrespondiert mit dem Standard Schaltflächen-Steuerelement.

Refrenzierung zu einem Steuerelement von Code

Zu einem Steuerelement kann durch dessen Namen referenziert werden:

```
TextBoxControl1.Text = "my text";
```

Tatsächlich ist das FastReport Steuerelement lediglich ein Wrapper für das Standard .Net Steuerelement. Es umschließt viele, wenn auch nicht alle, Eigenschaften des Standard Steuerelementes. Wird eine Eigenschaft, die nicht durch das FastReport Steuerelement umgesetzt ist, benötigt, kann das Wrapper Standard-Steuerelement folgendermaßen zugänglich gemacht werden:

· Nutzung der Eigenschaft "Steuerelement" des Typus System.Windows.Forms.Control:

```
(TextBox1.Control as TextBox).ShortcutsEnabled = false;
```

· Nutzung der Eigenschaft, die den gleichen Namen wie das Steuerelement selbst trägt, jedoch ohne das Suffix des Steuerelementes. Das TextBoxControl hat z.B. die Eigenschaft "TextBox" des Typus System.Windows.Forms.TextBox und gibt das Wrapper TextBox Steuerelement zurück:

```
TextBox1.TextBox.ShortcutsEnabled = false;
```

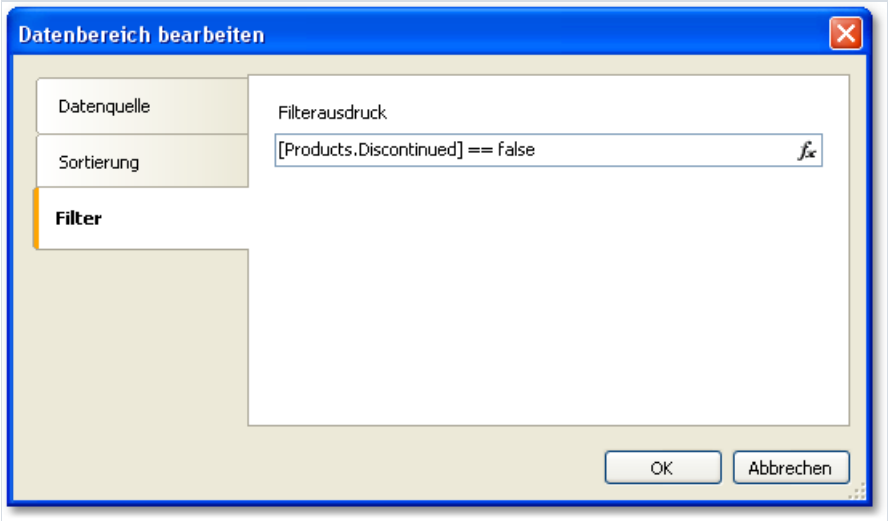
Auf die Hilfe zu Eigenschaften und Methoden kann über das MSDN zugegriffen werden.

Datenfilterung

Zur Filterung von Daten, die in einem Bericht gedruckt werden, können Dialogfelder genutzt werden. Nehmen wir beispielsweise einen Bericht der eine Liste aller Mitarbeiter druckt. Durch Nutzung eines Dialogfeldes können eine oder mehrere Mitarbeiter ausgewählt werden und bei der Erstellung des Berichts werden die Daten derart gefiltert, dass lediglich die ausgewählten Mitarbeiter angezeigt werden.

> Zur Nutzung der Datenfilterung muss der anfängliche Bericht alle Daten enthalten. Das Name "Filterung" zeigt bereits an, dass unnötige Daten bei der Erstellung eines Berichts nicht gedruckt werden.

Die einfachste Methode zur Organisation der Datenfilterung besteht in der Nutzung der Eigenschaft Filter auf der Leiste "Daten". Im Leisten Editor kann der Filter Ausdruck angezeigt werden, z.B.:



Durch Nutzung eines Dialogfeldes kann ein Wert von einem Benutzer abgefragt und im Filterungsausdruck verwendet werden. Mehr hierzu im Beispiel "Einfache Filter" im Abschnitt "Beispiele" .

Diese Methode kann verwendet werden, wenn ein einfacher Wert benötigt wird. Besteht die Aufgabe in der Anzeige einer Liste an Werten und in der Abfrage einer oder mehrerer, gestaltet sich die Umsetzung schwieriger. Die Anzeige einer Liste von Mitarbeitern in dem Steuerlement ListBoxControl und die Auswahl einer oder mehrerer Werte ist nicht so einfach, wie man denken könnte. Zur Umsetzung muss das folgende Skript genutzt werden:

· Holen der Datenquelle nach Namen

· Daten initialisieren;

· Auffüllen der ListBoxControl mit Daten aus der Datenquelle;

· nach Auswahl des Mitarbeiters, Bau eines Filter Ausdrucks, der in der Leiste "Daten" verwendet wird.

FastReport übernimmt dies automatisch. Hierzu wird die automatische Filterung genutzt. Diese wird nachfolgend im Detail betrachtet.

Automatische Filterung - Funktionsweise

Das Steuerelement ist mit der Datenspalte über die Eigenschaft "DataColumn" verbunden. Wenn das Steuerelement eine Liste von Werten (z.B. ListBoxControl) anzeigen kann, wird es mit Werten aus der angegebenen Datenspalte aufgefüllt. Dies geschieht bei Anzeige des Dialogfeldes automatisch. Danach wird im Dialogfeld gearbeitet. Es werden eine oder mehrere Items in dem Steuerelement ausgewählt. Hiernach schließt das Dialogfeld. Die Datenquelle, die in der Eigenschaft "DataColumn" angezeigt wurde, wird dann automatisch gefiltert.

> Es ist von Vorteil, dass diese Methode in jedem Bericht ohne die Notwendigkeit der Codegenerierung genutzt werden kann.

Die automatische Filterung wird von den folgenden Steuerelementen unterstützt:

Icon	Name
	CheckBoxControl
	CheckedListBoxControl
	ComboBoxControl
	DataSelectorControl
	DateTimePickerControl
	ListBoxControl
	MaskedTextBoxControl
	MonthCalendarControl
	NumericUpDownControl
	RadioButtonControl
	TextBoxControl

Filtervorgänge

FastReport filtert standardmäßig die Datenzeilen, die einen Wert, ähnlich dem Wert des Steuerelementes, enthalten. Dieses Verhalten ist in der Eigenschaft "FilterOperation" des Steuerelementes festgelegt. Die folgenden Vorgänge können genutzt werden:

Vorgang	Äquivalent	Effekt
Equal	=	Filtert den Wert wenn dieser gleich dem Wert des Steuerelementes ist.
NotEqual	< >	Filtert den Wert wenn dieser nicht gleich dem Wert des Steuerelementes ist.
LessThan	<	Filtert den Wert wenn dieser kleiner als der Wert des Steuerelementes ist.
LessThanOrEqual	< =	Filtert den Wert wenn dieser kleiner als, oder gleich, dem Wert des Steuerelementes ist.
GreaterThan	>	Filtert den Wert wenn dieser größer als der Wert des Steuerelementes ist.
GreaterThanOrEqual	> =	Filtert den Wert wenn dieser größer als, oder gleich, dem Wert des Steuerelementes ist.

Ist die Eigenschaft "FilterOperation" des Steuerelementes auf "LessThanOrEqual" eingestellt und man gibt den Wert 5 in das Steuerelement ein, dann werden alle Datenzeilen ausgewählt die einen korrespondierenden Wert kleiner als oder gleich 5 aufweisen.

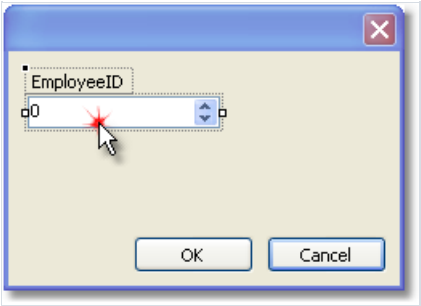
Für die Datenart "Zeichenfolge" können weitere Vorgänge genutzt werden:

Vorgang	Effekt
Contains	Filtert den Wert wenn dieser den Wert des Steuerelementes enthält.
NotContains	Filtert den Wert wenn dieser nicht den Wert des Steuerelementes enthält.
StartsWith	Filtert den Wert wenn dieser mit dem Wert des Steuerelementes beginnt.
NotStartsWith	Filtert den Wert wenn dieser nicht mit dem Wert des Steuerelementes beginnt.
EndsWith	Filtert den Wert wenn dieser mit dem Wert des Steuerelementes endet.
NotEndsWith	Filtert den Wert wenn dieser nicht mit dem Wert des Steuerelementes endet.

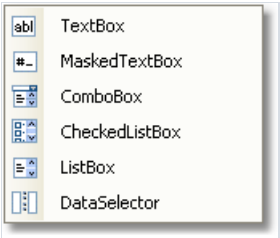
Ist die Eigenschaft "FilterOperation" des Steuerelementes beispielsweise auf "StartsWith" eingestellt und es wird "A" eingegeben, dann werden alle Datenspalten, dessen Wert mit "A" beginnt, ausgewählt.

Hinzufügen von Filtern in einen Bericht

Der FastReport Dialogfeld Designer bietet eine Vielzahl an einfach zu handhabenden Möglichkeiten um Steuerelementen, die die Datenfilterung unterstützen, hinzuzufügen. Mittels drag&drop wird die Datenspalte aus dem Fenster "Daten" in das Dialogfeld gezogen. Währenddessen erstellt FastReport eine Kopfzeile (Steuerelement LabelControl) und ein tatsächliches Steuerelement, das zur Datenfilterung genutzt wird:



Die Art des Steuerelementes hängt von der Art der Datenspalte ab. Ist die Spalte ein Zeichenfolge lässt sich nach dem Einfügen die Art des Steuerelementes auswählen:



Wurden zwei ähnliche Steuerelemente, die mit der gleichen Datenspalte verbunden sind, eingefügt, konfiguriert FastReport mit der Hilfe der Eigenschaft "FilterOperation" automatisch den Datenbereich. Das erste Steuerelement hat FilterOperation = GreaterThanEqual; das Zweite LessThanEqual. Dies wird für den Fall, dass eine Spalte eingefügt wird, die nicht vom Typ Zeichenfolge ist, durchgeführt.

Um die Datenfilterung in einen Bericht hinzuzufügen, muss daher das Folgende vorgenommen werden:

· Hinzufügen eines Neuen Dialogfeldes in den Bericht;

· Ziehen der Datenspalte auf das Dialogfeld, zu dem der Bericht gefiltert werden soll.

Filterung von Datenbereichen

Diese Filterungsmethode ist angesagt, wenn mit Werten gearbeitet wird, die quantitative Charakteristiken aufweisen, z.B. Kosten. Es können Güter gefiltert werden, die geringere Kosten aufweisen, als der gegebene Wert. Um anzuzeigen, wie der in das Element eingetragene Wert interpretiert werden soll, wird die Eigenschaft "FilterOperation", die weiter oben erläutert wurde, genutzt.

Der Start und das Ende eines Datenbereiches, lässt sich mit Hilfe zweier Steuerelemente, die an derselben Datenspalte verbunden sind und verschiedene Einstellungen der Eigenschaft "FilterOperaton" aufweisen, anzeigen. Für das erste Steuerelement muss die FilterOperation = GreaterThanOrEqual und für das Zweite = LessThanOrEqual angezeigt werden.

Filterung verwandter Datenspalten

Zwischen zwei Datenquellen können bekanntermaßen Relationen festgelegt werden. Weitere Details hierzu werden im Kapitel "Data" erläutert. Mit Hilfe der Relationen wird die Filterung der Daten an der Quelle möglich, indem eine Datenspalte aus einer anderen Quelle genutzt wird.

Angenommen es wurde ein ListBoxControl auf das Dialogfeld platziert und die folgende Datenspalte in der Eigenschaft "DataColumn" angezeigt:

```
Products.Categories.CategoryName
```

Wie funktioniert Filterung?

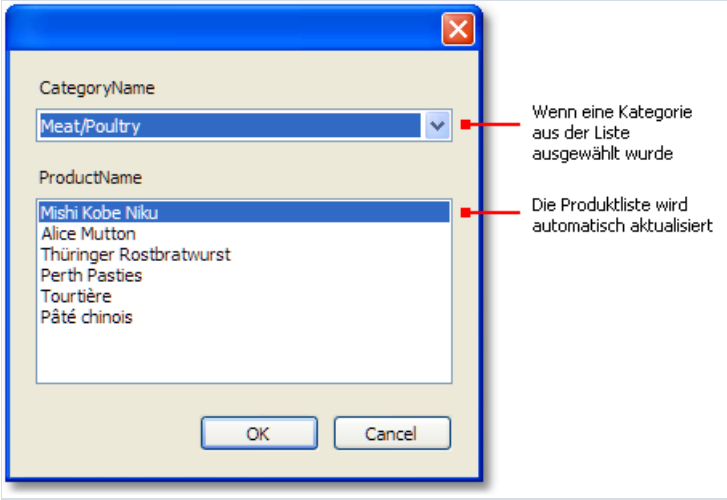
· die Spalte "CategoryName" aus der Datenquelle "Categories" wird bei der Füllung des Steuerelementes mit Werten genutzt;

· der Filter wird auf die Tabelle "Products" angewendet. Diese Datenzeilen werden derart gefiltert, das die folgende Bedingung korrekt ist:

```
the [Products.Categories.CategoryName] contains one of the values, selected by the user
```

Filterung mit Hilfe überlappender Listen

Eine überlappende Liste ist eine Liste mit Wahlmöglichkeiten, die sich basierend auf Werten, die ein Benutzer in einer anderen Liste festgelegt hat, ändert. Ein Formular umfasst beispielsweise zwei Listen: eine mit Kategorien und eine andere mit Produkten. Wird in der ersten Liste eine Kategorie ausgewählt, werden in der zweiten Liste die Produkte in einer ausgewählten Kategorie angezeigt.



Zur Erstellung einer überlappenden Liste werden zwei Datenquellen mit Master-Detail Relationen benötigt (mehr zu Datenquellen und Relationen kann im Kapitel "Daten" nachgelesen werden). Die Masterliste wird an eine Spalte in der Master-Datenquelle angefügt und die Detailliste an eine Spalte in der Detail-Datenquelle. Ebenso wird die Eigenschaft "DetailControl" der Masterliste auf Detailliste eingestellt.

Steuerung der Filterung mittels Code

Auch wenn die automatische Filterung in vielen Fällen ausreicht, besteht zusätzlich die Möglichkeit der manuellen Steuerung. Hierzu werden die folgenden Methoden und Eigenschaften genutzt.

Die Eigenschaft "Autofill" steuert das Füllen der Steuerelemente mit Daten. Es wird von Steuerelementen genutzt, die eine Liste an Werten anzeigen, z.B. die ListControlBox. Vor dem Anzeigen des Dialogfeldes füllt FastReport ein derartiges Steuerelement mit Daten. Standardmäßig ist die Eigenschaft auf true gesetzt. Bei Deaktivierung wird die Kontrolle nicht gefüllt und dies muss händisch durchgeführt werden, indem die Methode "FillData" aufgerufen wird:

```
ListBox1.FillData();
```

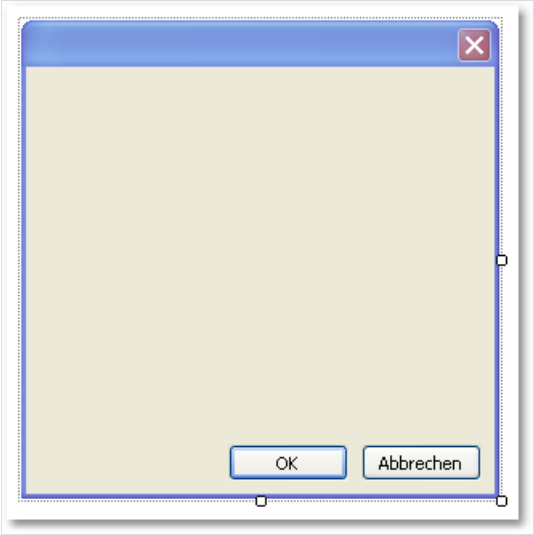
Die Eigenschaft "AutoFilter" steuert die Datenfilterung. Sie wird von allen Steuerelementen genutzt. Nachdem das Dialogfeld mit der Schaltfläche "OK" geschlossen wurde, wendet FastReport automatisch die Datenfilterung an. Standardmäßig ist die Eigenschaft auf true gesetzt. Bei Deaktivierung wird nicht gefüllt und die muss händisch über Aufruf der Methode "FillData", durchgeführt werden:

```
ListBox1.FilterData();
```

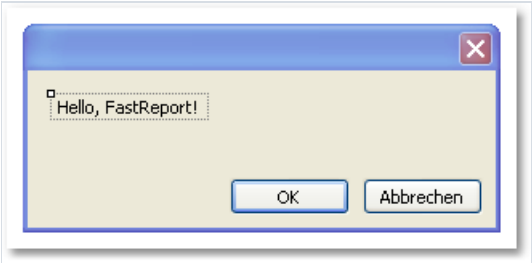
Beispiele

Beispiel 1: "Hello, FastReport!"

In diesem Beispiel wird das Erstellen eines Dialogfeldes, dass eine Begrüßung anzeigt,erläutert. Erstellung eines neues Berichts und hinzufügen eines Dialogfeldes. Hierzu drücken Sie die Schaltfläche auf der Symbolleiste:



Das LabelControl wird in das Dialogfeld platziert und seine "Text"-Eigenschaften im Fenster "Eigenschaften" festgelegt:

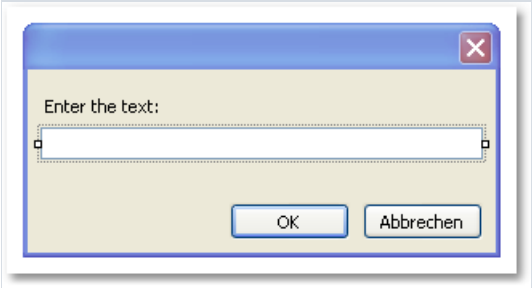


Beim Durchlauf eines solchen Berichtes wird das Dialogfeld angezeigt: Dieses wird mit der Schaltfläche OK geschlossen und der Bericht wird erstellt. Wird der Dialog mit der Schaltfläche "Abbrechen" oder der Schaltfläche "X" geschlossen, wird der Bericht nicht weiterlaufen und der Designer wird erneut angezeigt.

Beispiel 2: Anforderung von Text des Benutzers

In diesem Beispiel wird ein Dialogfeld erstellt, dass den Benutzer nach einem willkürlichen Text fragt und diesen eingegebenen Wert später in dem Bericht druckt.

Erstellung eines neues Berichts und hinzufügen eines Dialogfeldes. Auf das Dialogfeld werden die Steuerelemente LabelControl und TextBoxControl platziert:



In diesem Beispiel ist der eingegebene Wert in der Eigenschaft "Text" des TextControlBox beinhaltet. Zum Druck dieses Wertes im Bericht wird ein neues "Text"-Objekt auf die Leiste "Berichtstitel" gedruckt und das Folgende eingetragen:

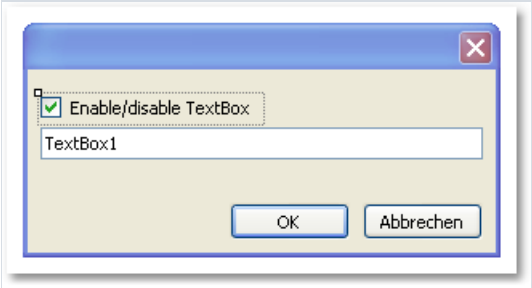
```
You have entered: [TextBox1.Text]
```

mit TextBox1 als Name des Steuerelementes TextBoxControl.

Beispiel 3: Bearbeitung von Dialog-Steuerelementen

Durch Nutzung des Skripts und der Ereignisse der Steuerelemente lassen sich Steuerelemente wie in VisualStudio bedienen. Im Beispiel wird gezeigt, wie die CheckboxControl die Eingabehilfen der TextBoxControl bearbeiten kann.

Erstellung eines neues Berichts und hinzufügen eines Dialogfeldes. Die Steuerelemente CheckBoxControl und TextBoxControl werden, wie weiter unten abgebildet, auf den Dialog platziert.



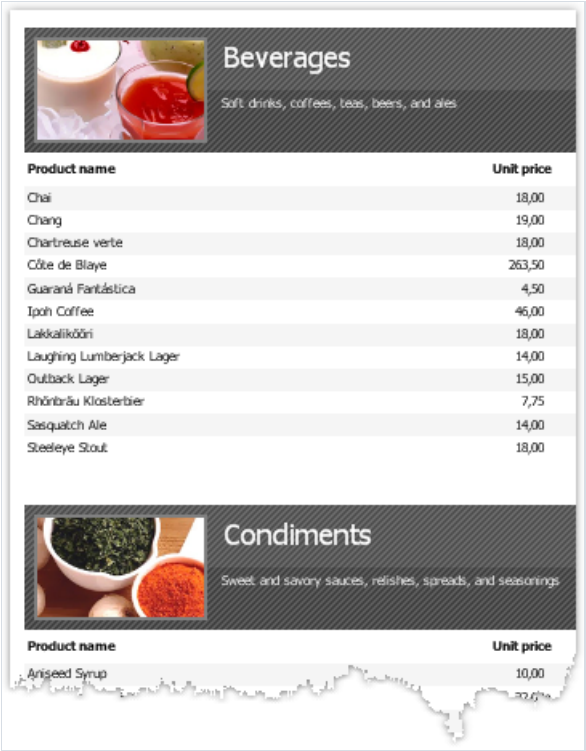
Nun wird die CheckBoxControl ausgewählt, das Fenster "Eigenschaften" geöffnet und die Schaltfläche gedrückt. Doppelklicken des Ereignisses "CheckedChanged" das beim Ändern des Status des Kontrollkästchens generiert wird. FastReport erstellt einen leeren Handler für dieses Ereignis. Der folgende Code wird in den Handler eingegeben:

```
private void CheckBox1_CheckedChanged(object sender, EventArgs e)
{
  TextBox1.Enabled = CheckBox1.Checked;
}
```

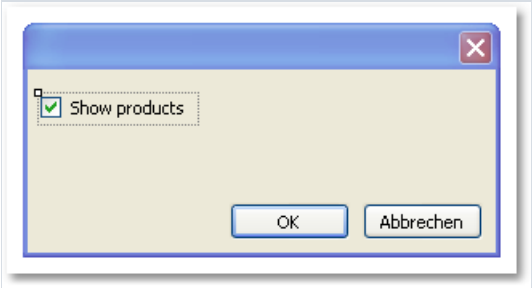
Bei der Durchführung des Berichts, kann das TextBoxControll über das Kontrollkästchen aktiviert oder deaktiviert werden.

Beispiel 4: Bearbeitung von Berichtsobjekten

Hier ein Beispiel eines Berichts, der eine Liste der Kategorien und Produkte in jeder Kategorie druckt:



Es wird demonstriert wie, mittels des Dialoges, der Druck der Produkte angehalten und lediglich Kategorien gedruckt werden. Hierzu wird ein Dialog in den Bericht eingefügt:



Doppelklicken der Schaltfläche "OK". FastReport fügt einen leeren Ereignishandler für das Ereignis "Click" ein: Der folgende Code wird in den Handler eingegeben:

```
private void btnOk_Click(object sender, EventArgs e)
{
  Data2.Visible = CheckBox1.Checked;
}
```

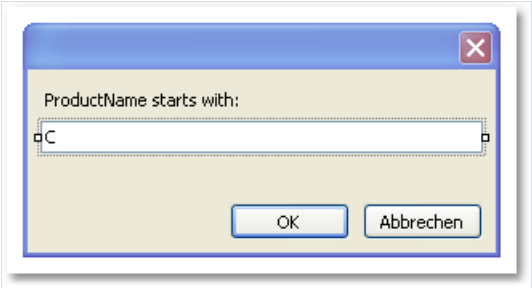
Die Sichtbarkeit der Leiste, die die Produkte druckt, wird gesteuert. In diesem Beispiel handelt es sich um eine Leiste mit dem Namen "Data2". Wird der Bericht durchgeführt und das Kontrollkästchen ist nicht markiert sieht das Ergebnis folgendermaßen aus:

Beverages
Soft drinks, coffees, teas, beers, and ales
Condiments
Sweet and savory sauces, relishes, spreads, and seasonings
Confections
Desserts, candies, and sweet breads

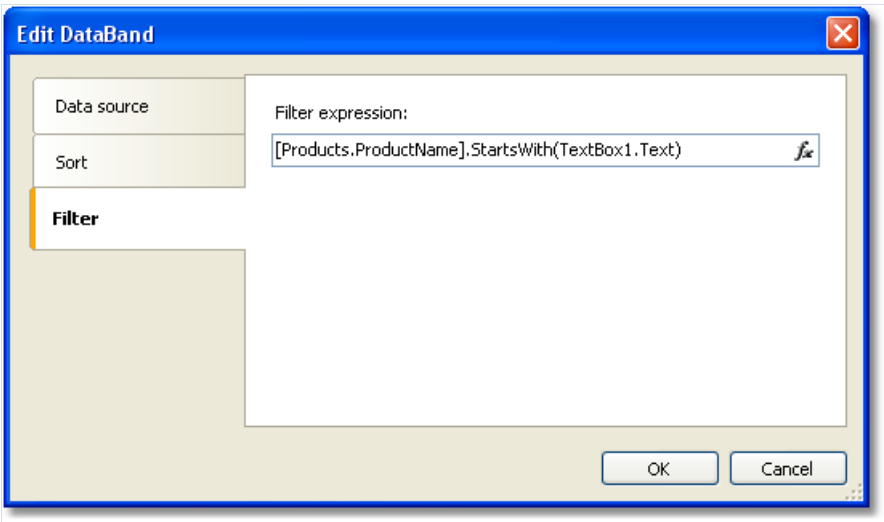
Beispiel 5: Einfacher Filter

Hier ein Beispiel eines Berichts, der eine Produktliste druckt:

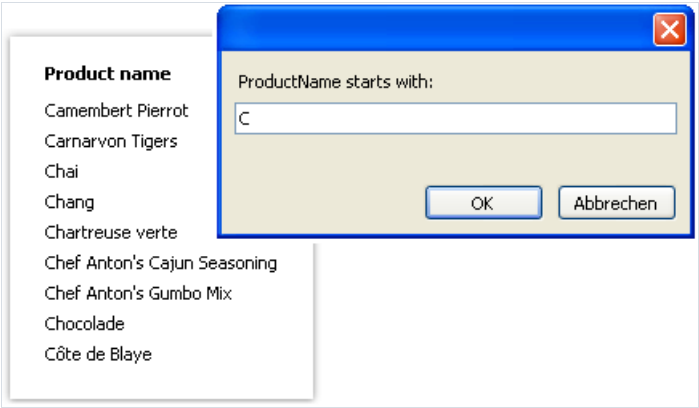
Hier wird erläutert, wie eine Produktliste nach dem ersten Buchstaben des Produktnamens gefiltert wird. Hierzu werden keine automatischen Datenfilterungsmöglichkeiten genutzt. Es wird ein neuer Dialog in den Bericht hinzugefügt und zwei Steuerelemente auf diesem platziert: LabelControl und TextBoxControl.



Hierzu muss der Datenleisten Editor aufgerufen und der folgende Filterausdruck angezeigt werden:



Durchführung des Berichts und Sicherstellung eines reibungslosen Ablaufs.

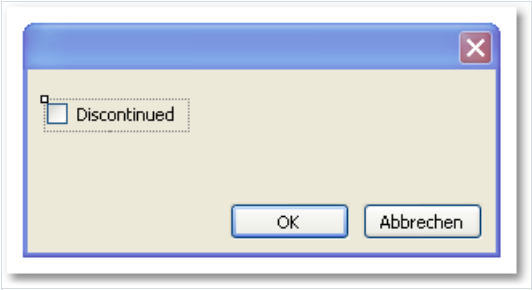


Beispiel 6: Automatische Filterung

Dieses Beispiel verdeutlicht wie ein Filter, der eine Produktliste aus der Tabelle "Products" druckt, in einen Bericht eingefügt wird. Die Daten werden entsprechend der Spalte "Products.Discontinued" gefiltert.

Der Bericht wird folgendermaßen aussehen:

Ein neuer Dialog wird in den Bericht eingefügt und die Schaltfläche auf der Symbolleiste gedrückt. Dann wird die Spalte "Products.Discontinued" aus dem Fenster "Daten" in das Dialogfeld gezogen:



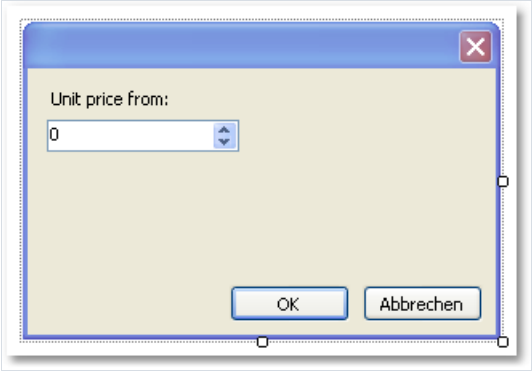
Und schon ist die Filterung fertig: nur zwei Klicks waren hierzu nowendig. FastReport verbindet die Steuerelemente automatisch mit der Datenspalte.

Der Bericht wird durchgeführt und die Flagge Markierung Discontinued aktiviert. Hiernach wird die Schaltfläche "OK" gedrückt und der Bericht wird lediglich Produkte mit der Markierung Discontinued anzeigen.

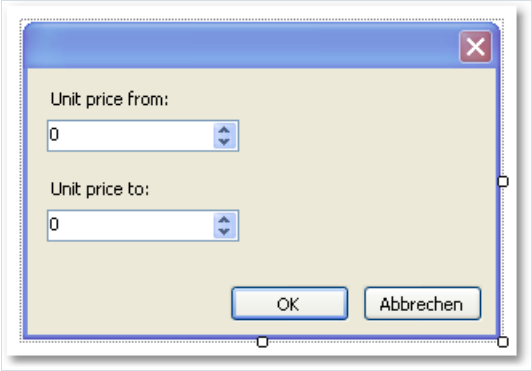
Product name	Discontinued
Alice Mutton	✓
Chef Anton's Gumbo Mix	✓
Guaraná Fantástica	✓
Mishi Kobe Niku	✓
Perth Pasties	✓
Rössle Sauerkraut	✓
Singaporean Hokkien Fried Mee	✓
Thüringer Rostbratwurst	✓

Beispiel 7: Automatische Filterung nach Bereich

Das vorherige Beispiel wird genutzt um zu demonstrieren, wie Produkte, mit den Kosten in den angegebenenn Bereichen, gedruckt werden können. Hierzu wird dem Bericht eine Dialogfeld hinzugefügt und die Datenspalte "Products.UnitPrice" in diesen gezogen. Dann wird der Beschriftungstext korrigiert:

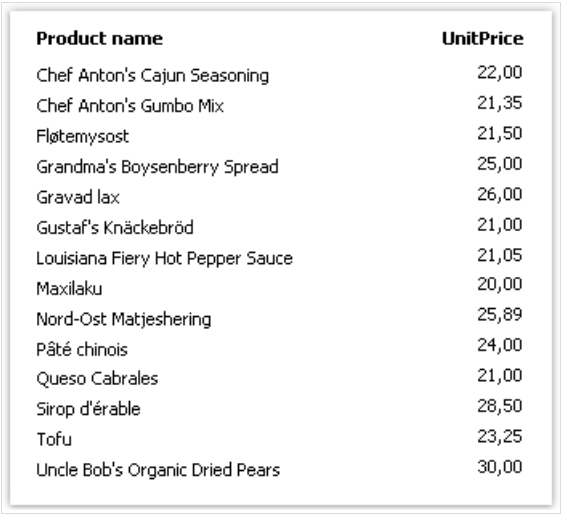


Auf die gleiche Art wird eine weitere Spalte "Products.UnitPrice" hinzugefügt und dessen Kopfzeile korrigiert:



Hiermit ist bereits alles erledigt. FastReport übernimmt den Rest der verbleibenden Arbeit: Verbindung der Steuerelemente mit den Datenspalten und festlegen derer FilterOperation Eigenschaften. Das erste Steuerelement hat FilterOperation = GreaterThanOrEqual; das Zweite LessThanOrEqual.

Der Bericht wird durchgeführt und die Werte, z.B. zwischen 20 und 30, müssen angezeigt werden. Nach drücken der Taste "OK" wird der Bericht erstellt. Er beinhaltet Produkte mit den Werten im angegebenen Bereich.



Product name	UnitPrice
Chef Anton's Cajun Seasoning	22,00
Chef Anton's Gumbo Mix	21,35
Fløtemysost	21,50
Grandma's Boysenberry Spread	25,00
Gravad lax	26,00
Gustaf's Knäckebröd	21,00
Louisiana Fiery Hot Pepper Sauce	21,05
Maxilaku	20,00
Nord-Ost Matjeshering	25,89
Pâté chinois	24,00
Queso Cabrales	21,00
Sirop d'érable	28,50
Tofu	23,25
Uncle Bob's Organic Dried Pears	30,00

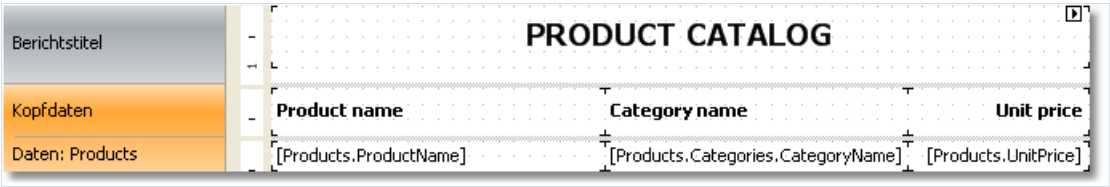
Beispiel 8: Filterung nach verwandter Datenspalte

In diesem Beispiel wird eine Spalte aus einer verwandten Datenquelle zur Filterung genutzt.

Hierzu wird die Berichtsart "einfache Liste", die die Produktliste druckt, verwendet. Der Kategorienname wird neben jedes Produkt gedruckt. Dies geschieht mittels der Relation:

```
[Products.Categories.CategoryName]
```

Der Bericht wird folgendermaßen aussehen:

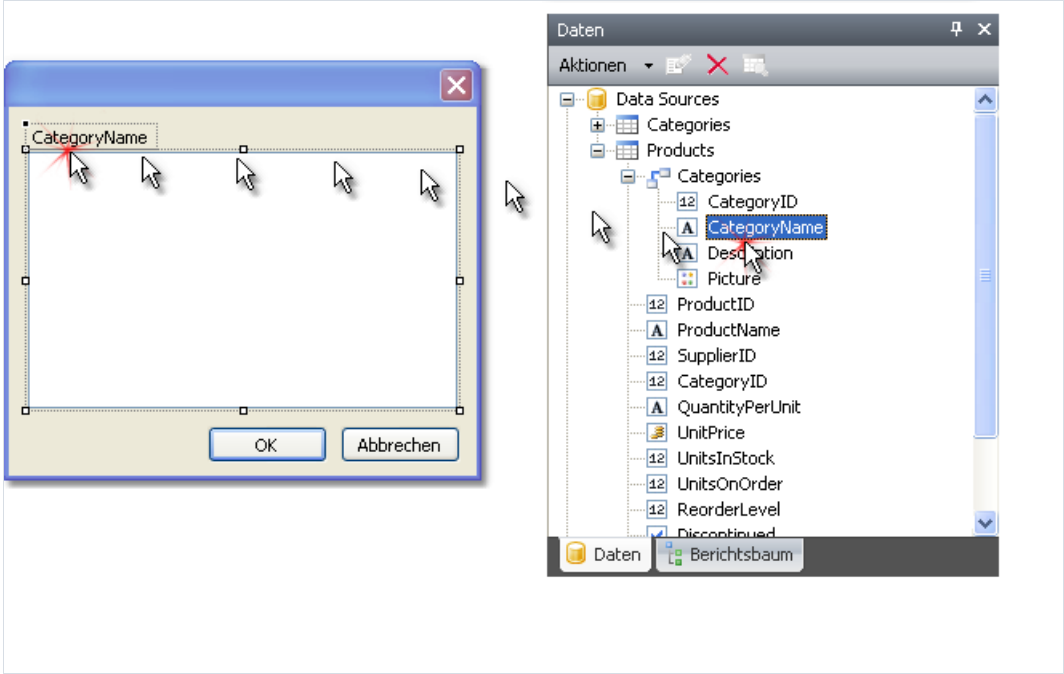


Beim Durchlauf eines solchen Berichtes wird das Folgende zu erkennen sein:

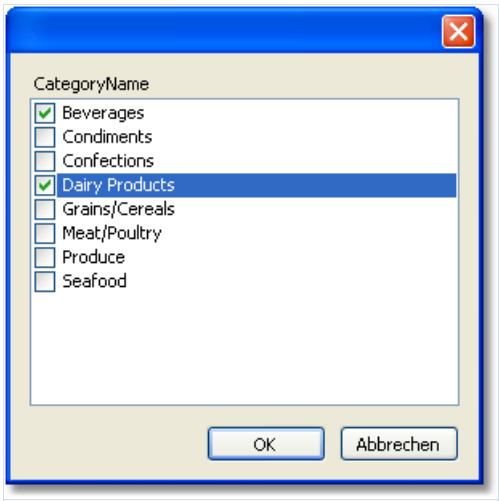
PRODUCT CATALOG

Product name	Category name	Unit price
Alice Mutton	Meat/Poultry	39,00
Aniseed Syrup	Condiments	10,00
Boston Crab Meat	Seafood	18,40
Camembert Pierrot	Dairy Products	34,00
Carnarvon Tigers	Seafood	62,50
Chai	Beverages	18,00
Chang	Beverages	19,00
Chartreuse verte	Beverages	18,00
Chef Anton's Cajun Seasoning	Condiments	22,00
Chef Anton's Gumbo Mix	Condiments	21,35
Chocolade	Confections	12,75

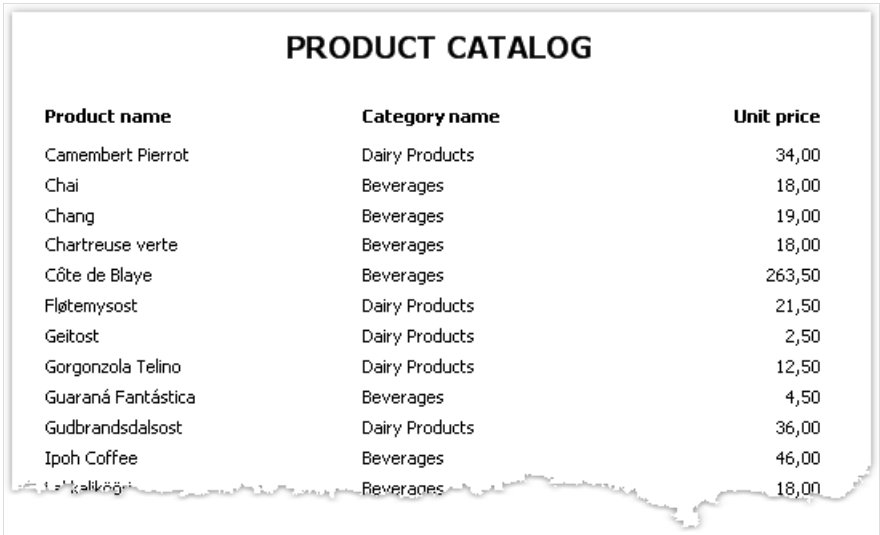
Nun wird die Filterung nach Kategorienname eingefügt. Hierzu wird ein neues Dialogfeld hinzugefügt und die Spalte "Products.Categories.CategoryName" in diese gezogen:



Zur Erstellung des Steuerelementes ist die Auswahl des Typus erforderlich. Es wird CheckedListBoxControl gewählt. Beim Durchlauf eines solchen Berichtes wird das folgende Dialogfeld zu erkennen sein:



Auswahl mehrerer Kategorien und drücken der Taste "OK". Hiernach werden die Daten gefiltert und der folgende Bericht angezeigt:



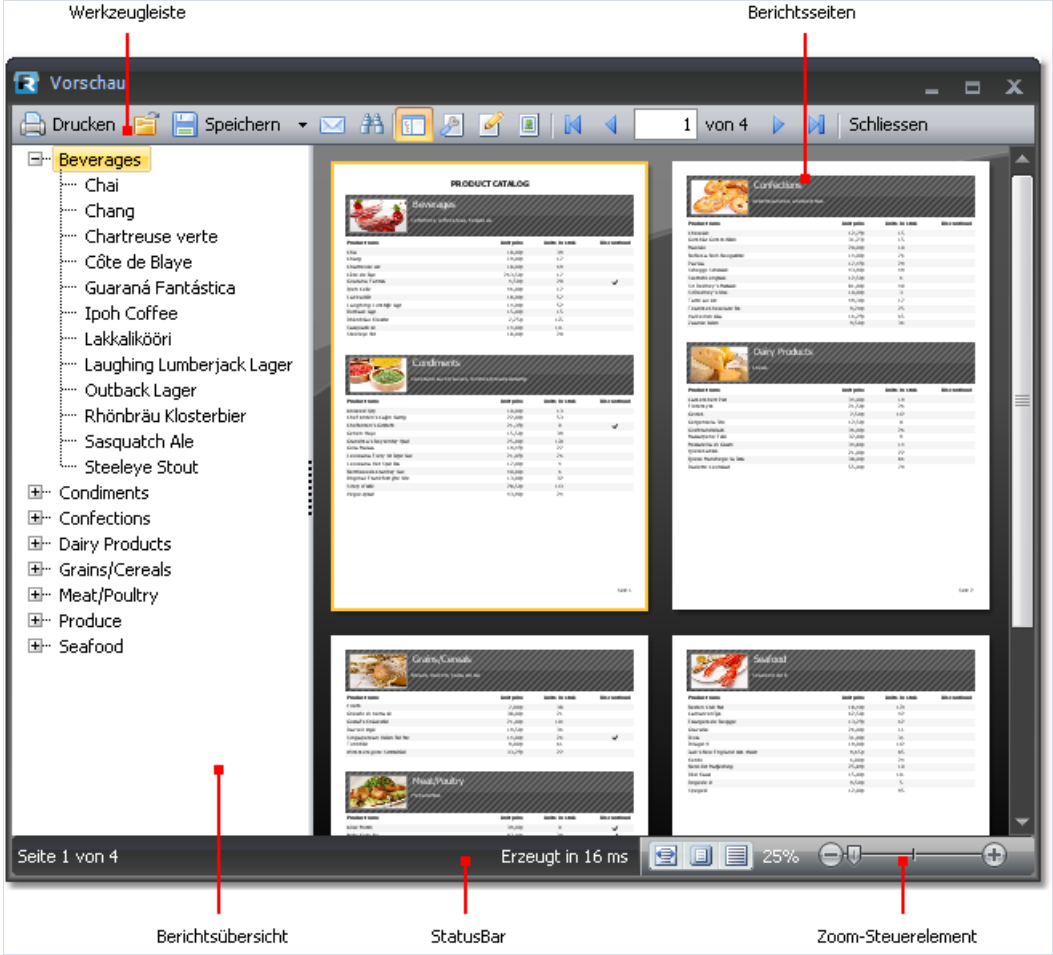
PRODUCT CATALOG

Product name	Category name	Unit price
Camembert Pierrot	Dairy Products	34,00
Chai	Beverages	18,00
Chang	Beverages	19,00
Chartreuse verte	Beverages	18,00
Côte de Blaye	Beverages	263,50
Fløtemysost	Dairy Products	21,50
Geitost	Dairy Products	2,50
Gorgonzola Telino	Dairy Products	12,50
Guaraná Fantástica	Beverages	4,50
Gudbrandsdalsost	Dairy Products	36,00
Ipoh Coffee	Beverages	46,00
Lakkalikööri	Beverages	18,00

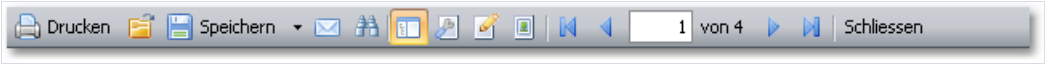
Wie sich erkennen lässt werden lediglich Produkte angezeigt, die sich in der ausgewählten Kategorie befinden.

Vorschau, Druck, Export

Ein ausgeführter Bericht kann auf dem Bildschirm angezeigt, auf dem Drucker gedruckt oder in eines der unterstützten Formate exportiert werden. All dies kann im Vorschaufenster geschehen:



Es befinden sich die folgenden Schaltflächen auf der Symbolleiste:



Schaltfläche	Beschreibung
	Bericht drucken
	Öffnen der erstellten Berichtsdatei im FXP-Format.
	Speichern des Berichts in einem der unterstützten Formate.
	Senden des Berichts per E-Mail.
	Textsuche im Bericht.

Schaltfläche	Beschreibung
	Anzeige oder Verbergen der Berichtsgliederung.
	Seiteneinstellungen.
	Bearbeiten der aktuellen Berichtsseite.
	Einstellungen des Wasserzeichens.
	Navigation zur ersten Seite.
	Navigation zur nächsten Seite.
1	Navigation zur angegebenen Seite. Eingabe der Seitenzahl und drücken von Eingabe.
	Navigation zur nächsten Seite.
	Navigation zur letzten Seite.

Die folgenden Tastatursystemsteuerungsoptionen können genutzt werden:

Taste	Beschreibung
Ctrl+P	Bericht drucken.
Ctrl+F	Textsuche.
Pfeiltasten	Die Vorschau scrollen.
BildAuf, BildAb	Bild auf/ab.
Startseite	Navigation zur ersten Seite.
Ende	Navigation zur letzten Seite.
Esc	Schließen des Vorschaufensters.

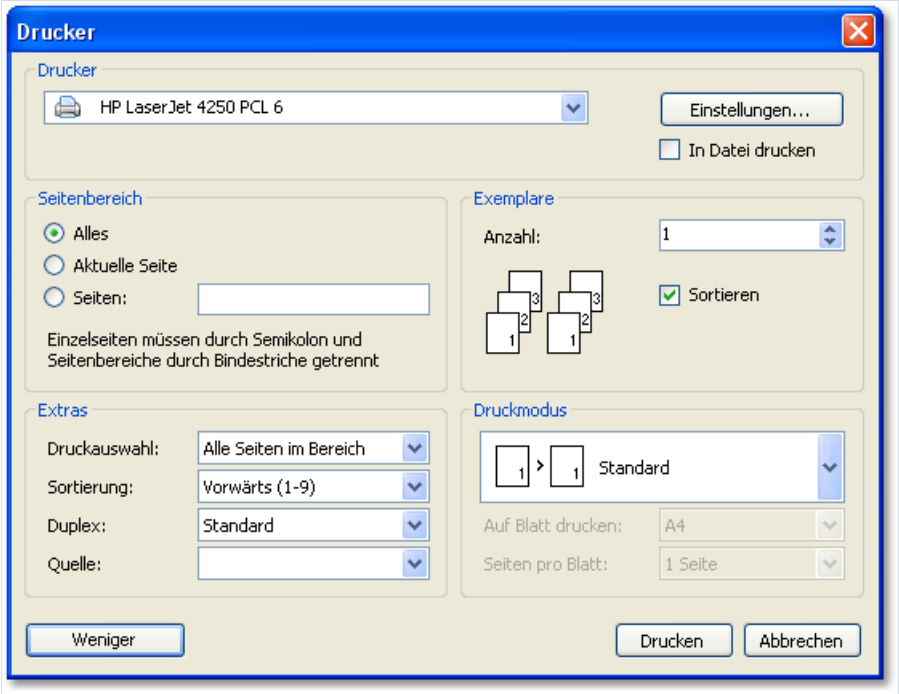
Bericht bearbeiten

Zum Bearbeiten einer Seite des erstellten Berichts wird auf die Schaltfläche im Vorschaufenster geklickt. Während dieses Vorgangs wird die aktuelle Seite in den Berich Designer geladen. Hier kann aus einer Vielzahl an Bearbeitungsmöglichkeiten ausgewählt werden:

Zum Bearbeiten wird der Designer geschlossen. Hier erscheint eine Aufforderung die Änderungen im Bericht zu speichern.

Druck des Berichts

Zum Druck eines Berichts wird die Schaltfläche (oder die Tastenkombination Ctrl+P) gedruckt. Das Dialogfeld Drucker wird angezeigt:



Die über diesen Dialog zugänglichen Einstellungen sind zu erkennen. Die Schaltfläche "Mehr/Weniger" ermöglicht sowohl die Anzeige des gesamten Dialogfeldes, als auch lediglich der grundsätzlichen Einstellungen. Das Dialogfeld wird standardmäßig in einfacher Form angezeigt.

Gruppe "Drucker": Hier kann der Drucker ausgewählt, dessen Einstellungen (Schaltfläche "Einstellungen...") geändert werden und ausgewählt werden zur Datei zu drucken.

Gruppe "Seitenbereich": Hier können die Seiten, die gedruckt werden sollen (alle, aktuelle oder angegebene Seitentzahl) ausgewählt werden.

Gruppe "Exemplare": Hier kann die Anzahl der Kopien festgelegt und die Reihenfolge der Seiten in der Kopie ("Sortieren") ausgewählt werden.

Gruppe "Extras": Hier können die zu druckenden Seiten (alle, gerade, ungerade) ausgewählt, die Druckreihenfolge (direkt, umgekehrt) geändert, der Duplex-Druck festgelegt (falls dies vom Drucker unterstützt wird) und die Papierquelle ausgewählt werden.

Die Gruppe "Druckmodus" ermöglicht die Auswahl einer der folgenden Druckmodi:

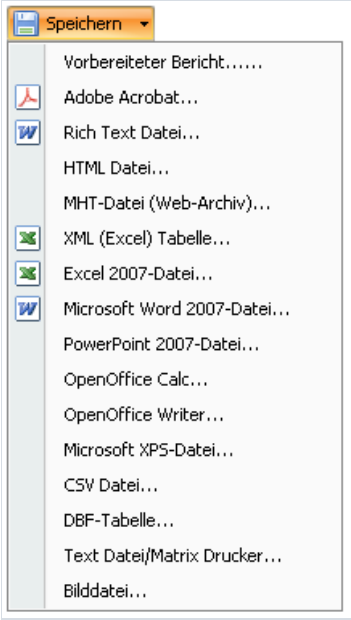
Modus	Beschreibung
	Der Drucker druckt auf das im Bericht angegebene Papier. Eine Berichtsseite entspricht einer gedruckten Seite.
	Dieser Modus wird zum Druck eines Berichts im Format A3 auf ein Papier des Formats A4 genutzt. Eine Berichtsseite druckt zwei Seiten. Zur Nutzung dieses Modus muss das Papierformat aus der Liste "Print on sheet" ausgewählt werden.

Modus	Beschreibung
	Dieser Modus wird zum Druck eines Berichts des Formats A4 auf ein Papier des Formats A4 genutzt. Es können 1, 2, 4 oder 8 Berichtsseiten auf eine Druckseite gedruckt werden. Bei der Nutzung dieses Modus muss das Papierformat, auf dem gedruckt werden soll aus der Liste "Printing on sheet" ausgewählt werden und die Anzahl der Seiten in der Liste "Pages in sheet" angegeben werden.

Der Druck des Berichts beginnt nach drücken der Schaltfläche "Drucken". Wird die Markierung "In Datei drucken" gewählt, wird der Name des Datei angefordert und der Bericht wird in dieser Datei (Datei mit einer PRN-Erweiterung) gespeichert.

Export des Berichts

FastReport erlaubt den Export des erstellten Berichts in verschiedene Formate. Derzeit werden 11 Formate unterstützt: PDF, RTF, HTML, MHT, Excel (XML), Excel 2007, PowerPoint 2007, OpenOffice Calc, CSV, TXT, image file. Hierzu wird die Schaltfläche "Speichern" im Vorschaufenster gedrückt und Export ausgewählt:



Speichern im FPX-Format

Das FPX-Format ist das systemeigene Format von FastReport. Dieses Format bietet die folgenden Vorteile:

· Speichern des Berichts ohne Qualitätsverluste. Durch Öffnen einer bereits gespeicherten Datei können alle Operationen, wie drucken, exportieren und bearbeiteten, ausgeführt werden;

· kompaktes, auf XMl basierendes, Format mit Hilfe von ZIP;

· die Berichtsdatei kann bei Bedarf von jedem Archivierungsprogramm, das das ZIP-Format unterstützt, ausgepackt und mit jedem Texteditor händisch korrigiert werden.

Zur Ansicht der Datei muss jedoch FastReport.Net auf dem entsprechenden Computer installiert sein.

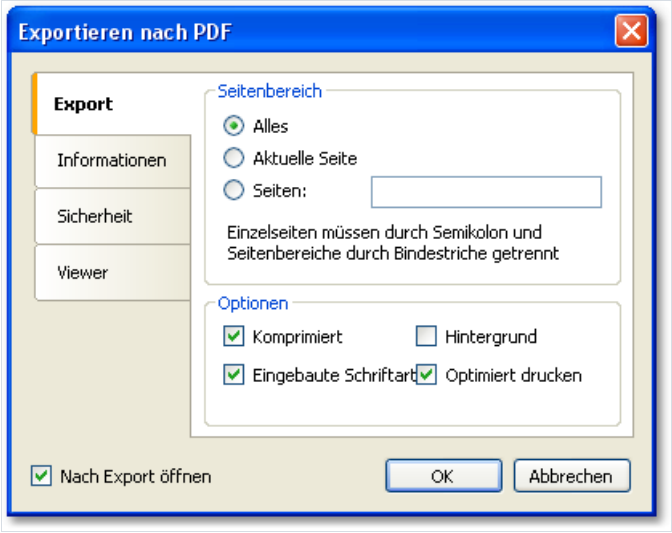
Zur Speicherung in ein FPX-Format wird die Schaltfläche "Speichern" im Vorschaufenster gedrückt und der Dateityp "Vorbereiteter Bericht" ausgewählt. Zum Öffnen einer gespeicherten Datei wird die Schaltfläche "Öffnen" gedrückt.

Export zu Adobe Acrobat (PDF)

PDF (Portable Document Format) ist ein von Adobe Systems erstelltes plattform-unabhängiges Format für elektronische Dokumente. Zur Ansicht wird das kostenlose Acrobat Reader Paket verwendet. Dieses Format ist flexible; Es erlaubt notwendige Schriftarten, Vektoren und Bitmap-Bilder einzuschließen. Es erlaubt auch Dokumente, die angezeigt oder gedruckt werden sollen, zu übermitteln und zu speichern.

Die Exportmethode ist überlagert.

Beim Export zu Adobe Acrobat erscheint ein Dialogfeld zur Einstellung von Parameters von Ausgabedateien:



Export Parameter:

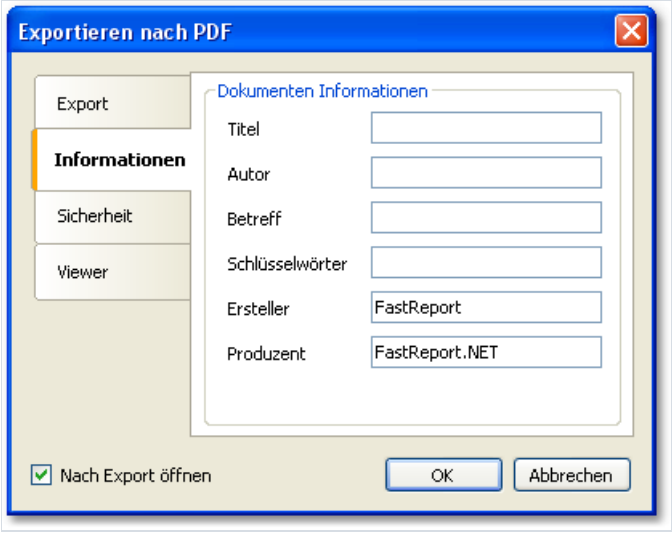
· "Komprimiert" - Die Ausgabedetai ist komprimiert. Dies reduziert die Dateigröße, verlängert jedoch die Zeit zum Export;

· "Eingebaute Schriftart" - Alle im Bericht genutzten Schriftarten werden in die PDF-Datei eingeschlossen. Dies vergrößert die Dateigröße erheblich.

· "Hintergrund" - Das Wasserzeichen der Seite wird als Bild exportiert. Dies vergrößert die Dateigröße erheblich.

· "Optimiert drucken" - Ausgabe aller Graphikobjekte (wie Bilder, Diagramme) in hochauflösend für den weiteren Druck.

Auf der Registerkarte "Informationen" können die Dokumenteninformationsfelder ausgefüllt werden:



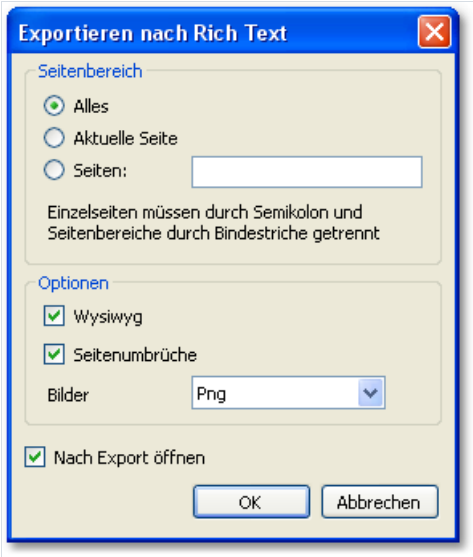
Auf der Registerkarte "Viewer" können einige Optionen die sich auf den Adobe Acrobat Dokumentviewer beziehen, festgelegt werden:

Export zu Word (RTF)

Das RTF (Rich Text Format) wurde von Microsoft als Standardformat zum Austausch von Textinformationen entwickelt. Derzeit sind RTF-Dokumente kompatible mit vielen neuen Text Editor und Betriebssysteme.

Exportmethode: tabellarisch.

Beim Export zu RTF erscheint ein Dialogfeld zur Einstellung von Parameters von Ausgabedateien:



Export Parameter:

· "Wysiwyg" - das Ergebnis ist dem Bericht so ähnlich wie möglich. Ist diese Option deaktiviert wird FastReport die Anzahl der resultierenden Zeilen und Spalten reduzieren;

· "Seitenumbrüche" - ermöglicht Seitenumbrüche in der RTF-Datei;

· "Bilder" - wählt das Format der Bilder im RTF-Format aus. Bitte beachten: Das "Metadatei"-Format eignet sich am besten zur Anzeige von Berichtsobjekten, wie MSChartObject und ShapeObject.

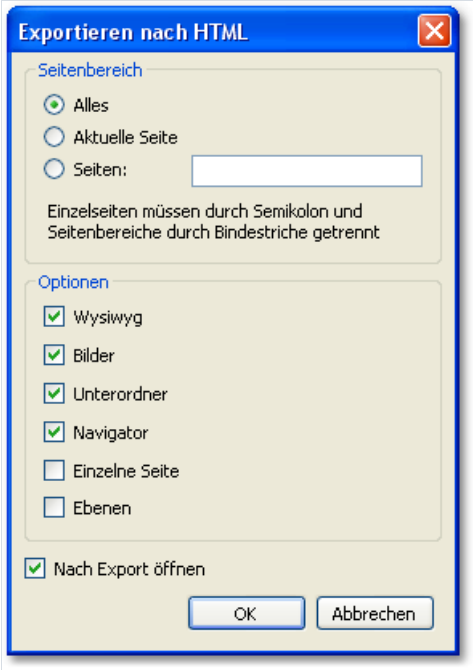
> Das Aussehen und die Größe der resultierenden Datei hängt von der Berichtsvorlage (mehr hierzu im Abschnitt "Empfehlungen zur Berichtsentwicklung") ab.

Export zu HTML

HTML (HyperText Markup Language) ist die vorherrschende Markupsprache für Webseiten. Es bietet eine Methode zur Beschreibung der Struktur der textbasierten Information in einem Dokument, indem es bestimmte Texte als Verknüpfungen, Überschriften, Paragraphen, Listen und so weiter bezeichnet und den Text durch interaktive Formulare, eingebettete Bilder oder andere Objekte erweitert.

Exportmethode: tabellarisch.

Beim Export zu HTML wird ein Dialogfenster zur Einstellung der Parameter angeboten:



Export Parameter:

· "Wysiwyg" - das Ergebnis ist dem Bericht so ähnlich wie möglich;

· "Bilder" - ermöglicht zu den Exportbildern;

· "Unterordner" - alle zusätzlichen Dateien werden in einem separaten Ordner mit dem Namen ".files" gespeichert;

· "Navigator" - erstellt einen speziellen Navigator zur Navigation auf den Seiten;

· "Einzelne Seite" - alle Seiten werden in einer Datei gespeichert.

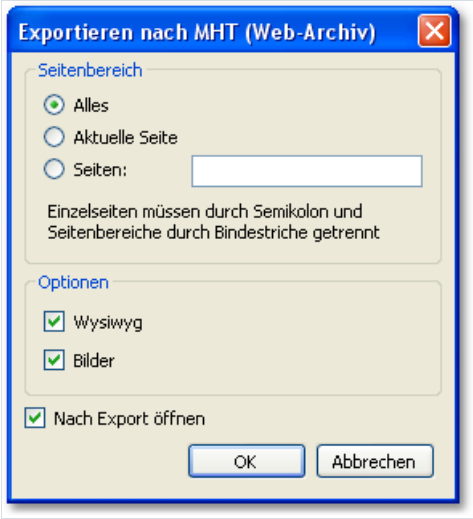
> Das Aussehen und die Größe der resultierenden Datei hängt von der Berichtsvorlage (mehr hierzu im Abschnitt "Empfehlungen zur Berichtsentwicklung") ab.

Export zu MHT (Webarchiv)

MHT, eine Abkürzung für MIME HTML, ist eine Webseitenarchivformat, das zur Bindung von Ressourcen, die üblicherweise durch externe Verknüpfungen repräsentiert werden (wie Bilder, Flashanimationen, Java Applets und Audiodateien) zusammen mit HTML-Code in eine einzelne Datei. Der Inhalt der MHT-Datei ist encodiert als wäre es eine HTML E-Mail Nachricht und nutzt den MIME TYpus mehrteilig/zugehörig.

Exportmethode: tabellarisch.

Beim Export zu MHT wird ein Dialogfenster zur Einstellung der Parameter angeboten:



Export Parameter:

· "Wysiwyg" - das Ergebnis ist dem Bericht so ähnlich wie möglich;

· "Bilder" - ermöglicht zu den Exportbildern;

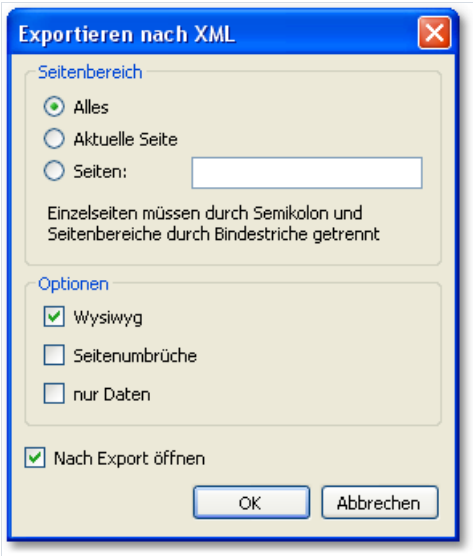
> Das Aussehen und die Größe der resultierenden Datei hängt von der Berichtsvorlage (mehr hierzu im Abschnitt "Empfehlungen zum Design des Berichts") ab.

Export zu Excel (XML)

Excel ist eine Applikation zur Arbeit an elektronischen Tabellenblättern. Es ist in ein Teil der Microsoft Office Familie.

Exportmethode: tabellarisch.

Beim Export zu Excel erscheint ein Dialogfeld zur Einstellung von Parameters von Ausgabedateien:



Export Parameter:

· "Wysiwyg" - das Ergebnis ist dem Bericht so ähnlich wie möglich. Ist diese Option deaktiviert wird FastReport die Anzahl der Zeilen und Spalten in der resultierenden Datei reduzieren;

· "Seitenumbrüche" - ermöglicht Seitenumbrüche in der resultierenden Datei.

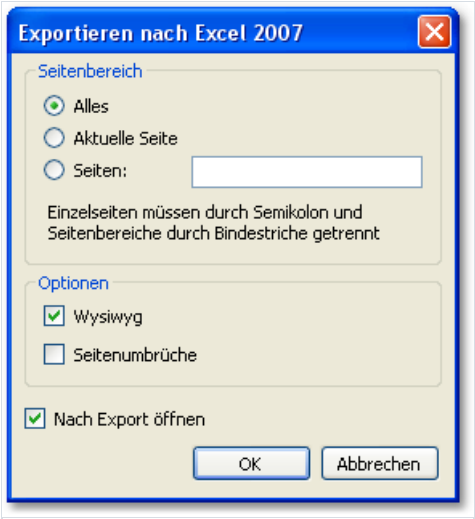
> Das Aussehen und die Größe der resultierenden Datei hängt von der Berichtsvorlage (mehr hierzu im Abschnitt "Empfehlungen zur Berichtsentwicklung") ab.

Export zu Excel 2007

Excel 2007 ist eine Applikation zur Arbeit an elektronischen Tabellenblättern. Es ist in ein Teil der Microsoft Office 2007 Familie.

Exportmethode: tabellarisch.

Beim Export zu Excel erscheint ein Dialogfeld zur Einstellung von Parameters von Ausgabedateien:



Export Parameter:

· "Wysiwyg" - das Ergebnis ist dem Bericht so ähnlich wie möglich. Ist diese Option deaktiviert wird FastReport die Anzahl der Zeilen und Spalten in der resultierenden Datei reduzieren;

· "Seitenumbrüche" - ermöglicht Seitenumbrüche in der resultierenden Datei.

> Das Aussehen und die Größe der resultierenden Datei hängt von der Berichtsvorlage (mehr hierzu im Abschnitt "Empfehlungen zur Berichtsentwicklung") ab.

Export zu PowerPoint 2007

PowerPoint 2007 ist eine Applikation zur Arbeit an elektronischen Präsentationen. Es ist in ein Teil der Microsoft Office 2007 Familie.

Die Exportmethode ist überlagert.

Beim Export zu PowerPoint erscheint ein Dialogfeld zur Einstellung von Parameters der Ausgabedatei:

Export Parameter:

· "Bilder" - wählt das Format der Bilder in der resultierenden Datei.

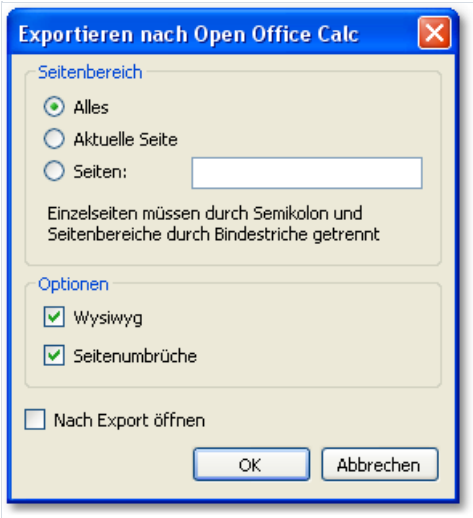
Export zu OpenOffice Calc

Das OpenDocument Format (ODF, OASIS Open Document Format für die Office Anwendung) wurde von OASIS entwickelt und basiert auf dem XML-Format, dass in OpenOffice genutzt wird.

FastReport unterstützt den Export zu einer Tabelle (.ods-Datei). Diese Dateien können in OpenOffice geöffnet werden.

Exportmethode: tabellarisch.

Beim Export zu OpenOfffice Calc erscheint ein Dialogfeld zur Einstellung von Parametern der Ausgabedatei:



Export Parameter:

· "Wysiwyg" - das Ergebnis ist dem Bericht so ähnlich wie möglich. Ist diese Option deaktiviert wird FastReport die Anzahl der Zeilen und Spalten in der resultierenden Datei reduzieren;

· "Seitenumbrüche" - ermöglicht Seitenumbrüche in der resultierenden Datei.

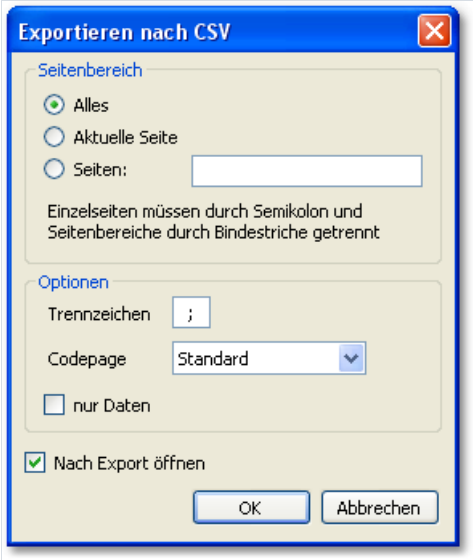
> Das Aussehen und die Größe der resultierenden Datei hängt von der Berichtsvorlage (mehr hierzu im Abschnitt "Empfehlungen zur Berichtsentwicklung") ab.

Export zu CSV

Die CSV-Datei wird zur digitalen Speicherung von Daten, die in Tabellen in Form von Listen strukturiert sind, genutzt Jede Zeile in der CSV-Datei korrespondiert mit einer Zeile in der Tabelle. Innerhalb einer Zeile werden die Felder durch Kommata getrennt; jedes Feld gehört zu einer Tabellenspalte. CSV-Dateien werden häufig zur Verschiebung von tabellarischen Daten zwischen zwei unterschiedlichen Computerprogrammen genutzt, z.B. zwischen einem Datenbankprogramm und einem Programm zu Tabellenkalkulation.

Exportmethode: tabellarisch.

Beim Export zu CSV erscheint ein Dialogfeld zur Einstellung von Parametern einer Ausgabedatei:



Export Parameter:

· "Trennzeichen" - das Trennzeichen des Feldes;

· "Codepage" - die Codepage die zur Endkodierung des Textes in der resultierenden Datei genutzt wird. Die "Default"-Codepage bezieht sich auf die Windows Standardcodepage. Bitte beachten: Excel unterstützt keine unicode Codepages;

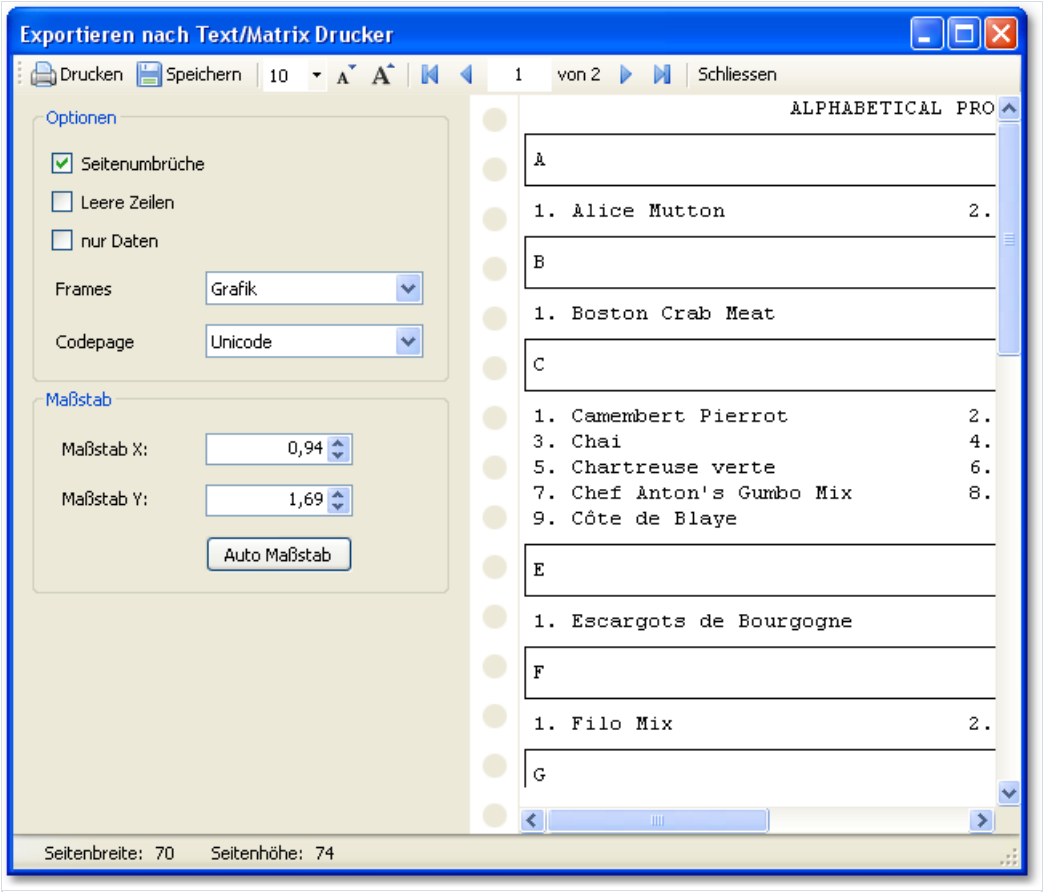
· "nur Daten" - ermöglicht diesem Kontrollkästchen Objekte, die nur auf der Datenleiste liegen, zu exportieren.

Export zu TXT

TXT ist eine reguläre Textdatei, die in jedem Texteditor geöffnet oder auf einem Nadeldrucker gedruckt werden kann.

Exportmethode: tabellarisch.

Beim Export zu TXT erscheint ein Dialogfeld zur Einstellung von Parametern einer Ausgabedatei:



Export Parameter:

· "Seitenumbrüche" - ermöglicht Seitenumbrüche in der resultierenden Datei;

· "Leere Zeilen" - ermöglicht leere Zeilen in der resultierenden Datei;

· "Data only" - ermöglicht diesem Kontrollkästchen Objekte, die nur auf der Datenleiste liegen, zu exportieren;

· "Frames" - Art des Objektrahmens. Sollen keine Rahmen exportiert werden, wird "None" ausgewählt;

· "Codepage" - die Codepage die zur Endkodierung des Textes in der resultierenden Datei genutzt wird;

· "Maßstab X" - horizontale Skala;

· "Maßstab Y"- vertikale Skala;

· "Auto Maßstab" - berechnet automatisch Skala X und Skala Y zur Vermeidung von Datenverlust.

Export zu Bild

FastReport ermöglicht den Export von Informationen in eine der folgenden graphischen Formate:

· BMP

· PNG

· JPG

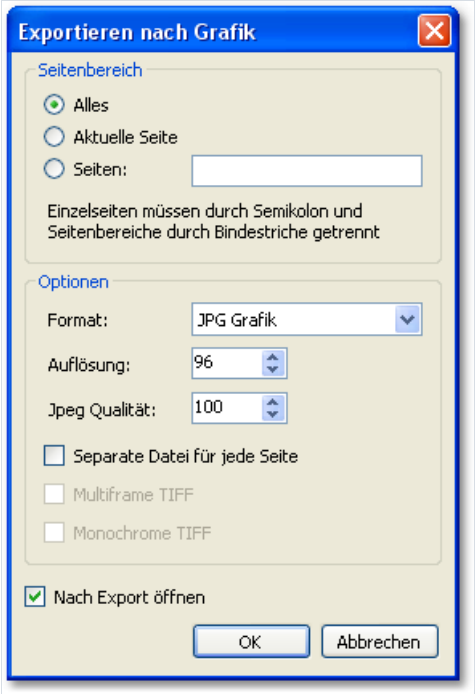
· GIF

· TIFF

· Windows Metafile (EMF,WMF)

Exportmethode: Zeichnung.

Beim Export zu einer Bilddatei wird ein Dialogfenster zur Einstellung der Parameter angeboten:



Export Parameter:

· "Auflösung" - Auflösung des graphischen Bildes. Nutzung von 96dpi zur Anzeige und 300dpi zum Druck. Beim Export zum TIFF-Fomat können Werte für horizontale und vertikale Auflösungen separat festgelegt werden;

· "Jpeg qualität" - JPG-Datei Kompressionsstufe. Diese Option wird lediglich dann genutzt, wenn in ein Jpeg-Format exportiert wird;

· "Separate Datei für jede Seite" - Aktivierung dieser Option exportiert jede Berichtsseite in eine separate Datei; Der Dateiname wird auf der Grundlage der ausgewählten Seite mit der angegebenen Zahl, geformt;

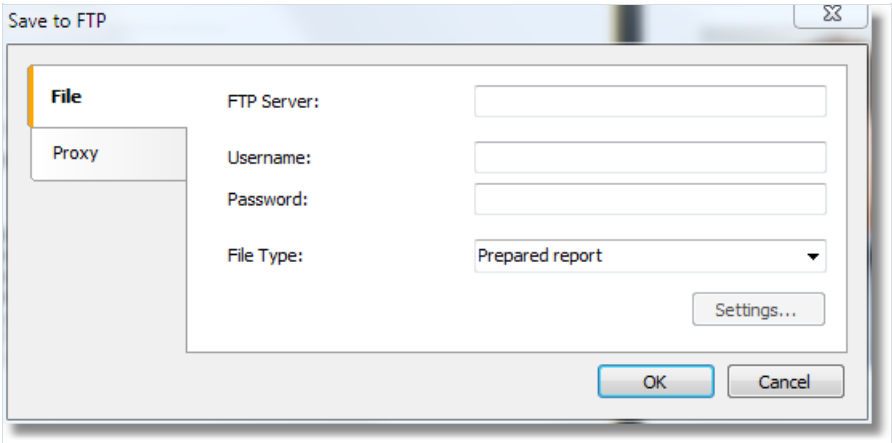
· "Multi-frame TIFF" - diese Option erstellt eine multiframe TIFF-Datei. Diese Option wird lediglich dann genutzt, wenn in ein TIFF-Format exportiert wird;

· "Monochrome TIFF" - diese Option erstellt eine monochrome TIFF-Datei. Diese Option wird lediglich dann genutzt, wenn in ein TIFF-Format exportiert wird.

Beim Export mehrerer Seiten in eine Datei (bei Deaktivierung der Option "Separate Datei für jede Seite") nimmt der Export einen Großteil der CPU/Speicherressourcen in Anspruch.

Export zu FTP

In der Vorschau von FastReport .NET gibt es die Möglichkeit, einen fertigen Bericht auf einen FTP-Server hochzuladen. Außerdem kann man den Bericht in einen der unterstützten Formate exportieren und dann auf den FTP hochladen. Dazu wählen Sie den Menüpunkt "Speichern" und wählen anschließend "FTP..." aus. Es erscheint ein Dialog zum Speichern auf einem FTP-Server:

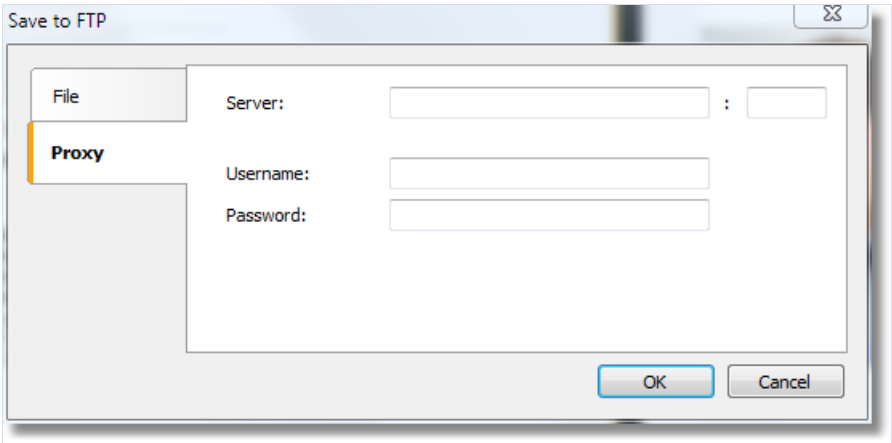


Im Reiter "Datei" gibt es folgende Felder:

- FTP-Server. Hier geben Sie die URL-Adresse des FTP-Servers an, auf den Sie die Datei speichern möchten.
- Benutzername und Passwort. Sie müssen Ihren Benutzernamen und Passwort auf diesem FTP-Server angeben.
- Dateityp. Eine Dropdown-Liste, in dem Sie das Format der zu speichernden Datei (fertigen Bericht oder eine der Exportdateien) angeben.

Bei Exportdateien wird der Button "Optionen..." aktiviert. Durch den Druck auf diesen Button können Sie zum Fenster der Einstellungen für diese Exportdatei wechseln.

Wenn Sie einen Proxy-Server verwenden, können Sie im Reiter "Proxy" die URL-Adresse des Proxyservers, den Port, sowie Login- und Passwortinformationen angeben:



Nachdem Sie die Einstellungen vorgenommen und das Format ausgewählt haben, brauchen Sie nurn och OK zu drücken. Dann wird die Datei auf den FTP-Server hochgeladen.

Export zu Dropbox

In der Vorschau von FastReport .NET besteht die Möglichkeit, einen fertigen Bericht in Dropbox zu speichern. Außerdem kann man den Bericht vorher in jedes der unterstützten Formate exportieren.

Bevor Sie diese Möglichkeit nutzen, müssen Sie erst eine App in Ihrem Dropbox-Konto erstellen. Dafür müssen Sie sich bei Dropbox einloggen und folgende Schritte durchführen:

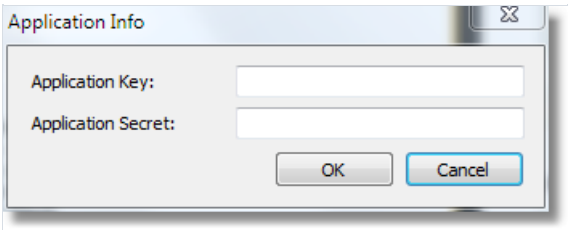
- Klicken Sie auf den Button "Mehr (More)". Er befindet sich unten auf der Seite von Dropbox.
- Im aufklappbaren Menü wählen Sie "Entwickler (Developers)". Dann gelangen Sie auf die Entwicklerseite.
- Klicken Sie auf "App Console". Dann wird eine Liste der Apps aufgelistet.
- Klicken Sie auf "Create App". Dropbox wird Ihre E-Mail Adresse überprüfen wollen. Klicken Sie auf "Send Email".

Auf Ihre Mail-Adresse wird eine Mail geschickt, in der Sie einen Bestätigungslink für Ihre Mailadresse anklicken müssen.

Anschließend kommen Sie auf die Seite "Create a new Dropbox Platform app". Hier wählen Sie "Dropbox API app" und auf die Frage "What type of data does your app need to store on Dropbox?" wählen Sie die Antwort "Files and datastores". Auf die Frage "Can your app be limited to its own, private folder?" können Sie eine der beiden Antworten wählen. Als letztes müssen Sie auf der Seite den Namen der App angeben, dieser kann beliebig sein. Nach dem Klick auf den Button "Create app" wird noch geprüft, ob der Appname schon vergeben ist, und wird die App sonst anlegen.

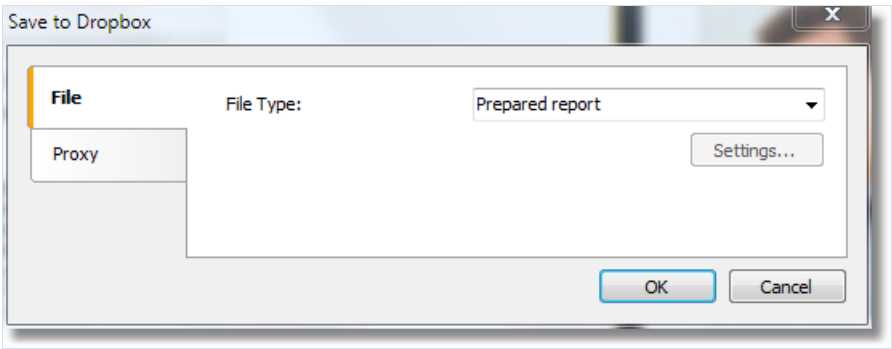
Am Ende gelangen Sie auf die Seite der App-Einstellungen. Hier brauchen Sie die Einträge "App key" und "App secret", die werden Sie für den Export nach Dropbox benötigen.

Nun können Sie zur Vorschau von FastReport .NET wechseln und die Datei nach Dropbox exportieren. Hierzu wählen Sie "Speichern" und dann den Punkt "Dropbox...". Beim ersten Versuch, nach Dropbox zu exportieren, erscheint das Fenster "Information über die App":



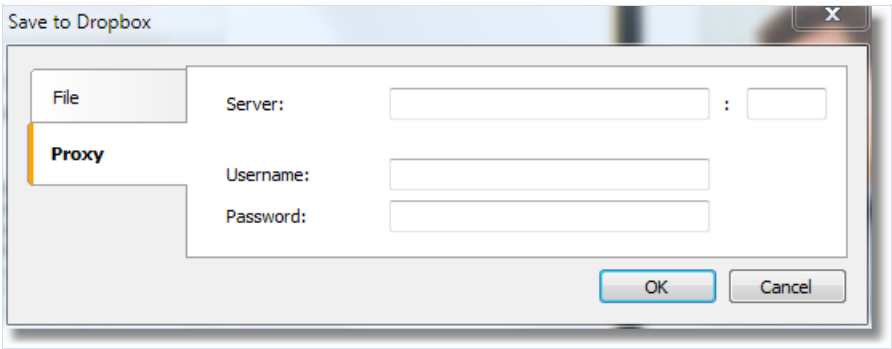
Hier geben Sie die vorher erhaltenen "App Key" und "App Secret" ein. Wenn Sie auf OK klicken, wird FastReport diese Werte speichern und sie müssen nicht nochmals eingegeben werden.

Es erscheint ein Dialog zum Speichern in Dropbox:



Auf dem Reiter "Datei" können Sie den Typ der zu speichernden Datei wählen (einen fertigen Bericht, oder eine der Exportdateien). Bei Exportdateien wird der Button "Optionen..." aktiviert. Durch den Druck auf diesen Button können Sie zum Fenster der Einstellungen für diese Exportdatei wechseln.

Wenn Sie einen Proxy-Server verwenden, können Sie im Reiter "Proxy" die URL-Adresse des Proxyservers, den Port, sowie Login- und Passwortinformationen angeben:



Nachdem Sie die Einstellungen angegeben und das Format gewählt haben, klicken Sie auf "OK", und die Datei wird in Dropbox gespeichert.

Export zu Google Drive

In der Vorschau von FastReport.NET besteht die Möglichkeit, einen fertigen Bericht nach GoogleDrive zu speichern. Außerdem kann man den Bericht vorher in jedes der unterstützten Formate exportieren.

Zunächst müssen Sie eine App in GoogleDrive anlegen. Dafür rufen Sie die Adresse https://code.google.com/apis/console auf.

Dort müssen Sie die Lizenzbedingungen annehmen.

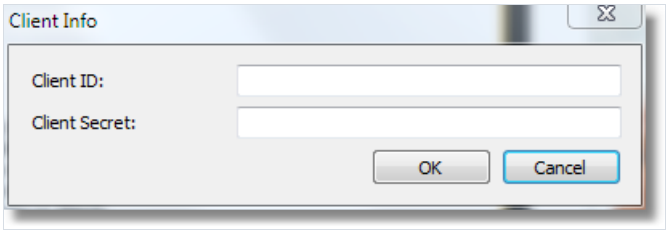
Anschließend kommen Sie auf die Seite der Appverwaltung. Hier klicken Sie auf den Reiter "Services" und aktivieren die Drive API. Dann wechseln Sie auf den Reiter "API Access" und klicken Sie auf "Create an OAuth 2.0 client ID". Im Abschnitt "Branding Information" geben Sie den Namen der App ein und klicken auf "Next". Im Abschnitt "Client ID Settings" wählen Sie folgende Angaben:

- "Installed application" für Application type.
- "Other" für Installed application type.

Klicken Sie auf "Create Client ID".

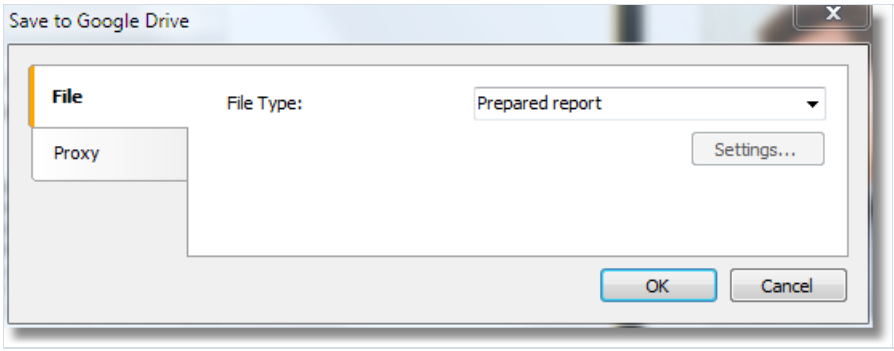
Dann erscheint auf der Seite "API Access page" der Abschnitt "Client ID for installed applications", in dem Sie die Werte Client ID und Client secret sehen. Diese beiden Werte werden Sie später benötigen.

Nun können Sie in die Vorschau von FastReport .NET wechseln und die Datei nach GoogleDrive speichern. Dafür wählen Sie den Button "Speichern" und klicken Sie auf "GoogleDrive...". Beim ersten Versuch, nach GoogleDrive zu exportieren, erscheint das Fenster "Client Information":



Hier müssen Sie die vorher gemerkten Client ID und Client Secret eingeben. Nach dem Klick auf "OK" wird FastReport diese Einstellungen speichern und Sie müssen sie nicht nochmals eingeben.

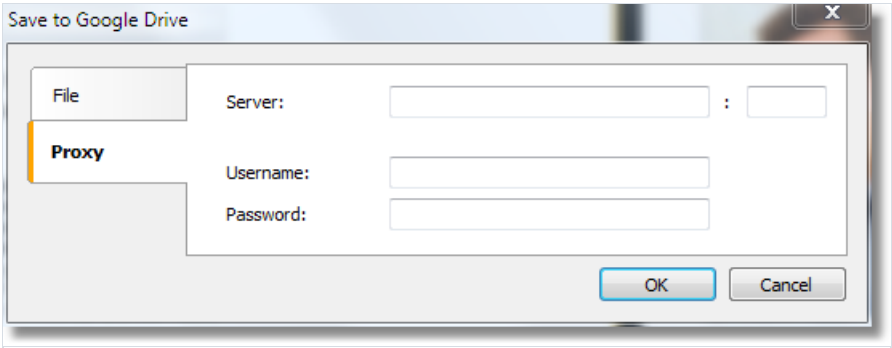
Es erscheint das Fenster zum Speichern in GoogleDrive:



Auf dem Reiter "Datei" können Sie den Typ der zu speichernden Datei wählen (einen fertigen Bericht, oder eine der Exportdateien). Bei Exportdateien wird der Button "Optionen..." aktiviert. Durch den Druck auf diesen Button können Sie zum Fenster der Einstellungen für diese Exportdatei wechseln.

Wenn Sie einen Proxy-Server verwenden, können Sie im Reiter "Proxy" die URL-Adresse des Proxyservers, den

Port, sowie Login- und Passwortinformationen angeben:



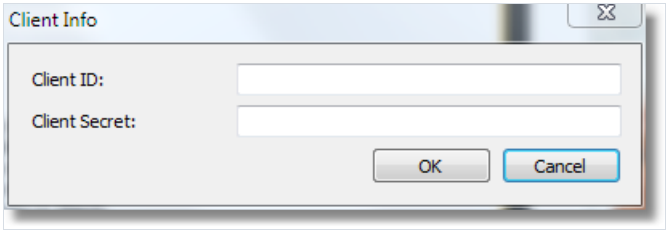
Nachdem Sie die Einstellungen angegeben und das Format gewählt haben, klicken Sie auf "OK", und die Datei wird in GoogleDrive gespeichert.

Export zu SkyDrive

In der Vorschau von FastReport.NET besteht die Möglichkeit, einen fertigen Bericht nach SkyDrive zu speichern. Außerdem kann man den Bericht vorher in jedes der unterstützten Formate exportieren.

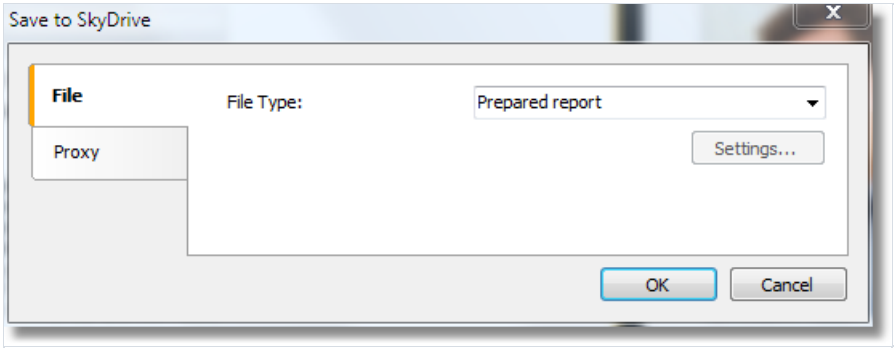
Zunächst müssen Sie eine App in SkyDrive anlegen. Dafür gehen Sie auf die Seite von SkyDrive und klicken auf den Link „Entwickler". Auf der Übersichtsseite klicken Sie auf „Meine Apps". Dann klicken Sie auf „App erstellen". Geben Sie den Namen der App und die gewünschte Sprache an. Lesen Sie die Nutzungsbedingungen sowie „Datenschutz und Cookies" durch und klicken Sie auf „Ich stimme zu". Dann gelangen Sie zur Seite der Einstellungen der App. Hier sehen Sie „Client-ID" sowie „Geheimer Clientschlüssel": diese Informationen brauchen Sie. Sie müssen nur noch eine Umleitungsdomäne eintragen und auf „Speichern" klicken.

Nun können Sie in die Vorschau von FastReport .NET wechseln und die Datei nach SkyDrive speichern. Dafür wählen Sie den Button „Speichern" und klicken Sie auf „SkyDrive...". Beim ersten Versuch, nach SkyDrive zu exportieren, erscheint das Fenster „Client Information":



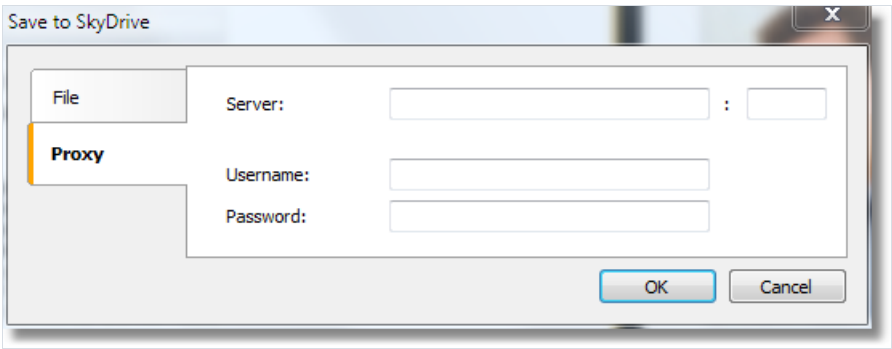
Hier müssen Sie die zuvor generierten Client-ID sowie den Geheimen Clientschlüssel (Client Secret) eingeben. Nach dem Klick auf „OK" wird FastReport diese Einstellungen speichern und Sie müssen sie nicht nochmals eingeben.

Es erscheint ein Dialog zum Speichern in SkyDrive:



Auf dem Reiter „Datei" können Sie den Typ der zu speichernden Datei wählen (einen fertigen Bericht, oder eine der Exportdateien). Bei Exportdateien wird der Button „Optionen..." aktiviert. Durch den Druck auf diesen Button können Sie zum Fenster der Einstellungen für diese Exportdatei wechseln.

Wenn Sie einen Proxy-Server verwenden, können Sie im Reiter „Proxy" die URL-Adresse des Proxyservers, den Port, sowie Login- und Passwortinformationen angeben:



Nachdem Sie die Einstellungen angegeben und das Format gewählt haben, klicken Sie auf "OK", und die Datei wird in SkyDrive gespeichert.

Empfehlungen zum Design des Berichts

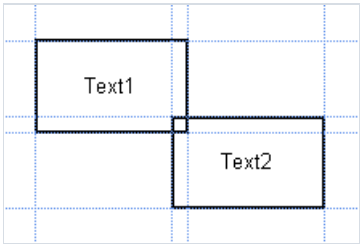
Dieses Kapitel erläutert spezielle Designanforderungen von Berichten, die für den Export in andere Datenformate bestimmt sind.

FastReport ermöglicht eine Vielzahl an Möglichkeiten zur Manipulation von Objekten während der Berichterstellung. Dies bietet den Vorteil einer schnellen Entwicklung eines jeden Berichts und im weiteren dessen Druck. Ein gedrucktes Dokument schaut genauso wie dessen Anzeige aus. Und dies ist das Hauptanliegen bei der Nutzung des FastReport Berichtsgenerators.

Ein Nachteil einer solchen Freiheit in der Entwicklung liegt in der Komplexität des Exports von FastReport Dokumenten zu verschiedenen Datenformaten. Alle diese haben ihre eigenen Beschränkungen und Anforderungen an die Präsentation der Information und können beizeiten recht komplex sein. Viele Formate, wie HTML, XLS oder RTF, nutzen zur Präsentation Tabellenblätter. Diese Formate ermöglichen jedoch keine Überkreuzungen von Zellen oder Layeranordnungen bei der Tabellenmarkierung.

Export der Filter sollte diese Anforderungen grundsätzlich in Betracht ziehen. Dies wird durch spezielle Algorithmen, die die Objektkreuzung in Betracht ziehen, und diese auf optimale Weise positioniert, ausgeführt. Bei der Objektüberkreuzung erschienen neue Spalten und Zeilen in der resultierenden Ausgabetabelle. Dies ist zur maximalen Ähnlichkeit zwischen dem resultierenden und dem Originalbericht notwendig. Eine hohe Anzahl an gekreuzten Objekten im Bericht Design resultiert in einer gesteigerten Anzahl an Spalten und Zeilen in der resultierenden Tabelle. Dies beeinflusst die Dateigröße und dessen Komplexität.

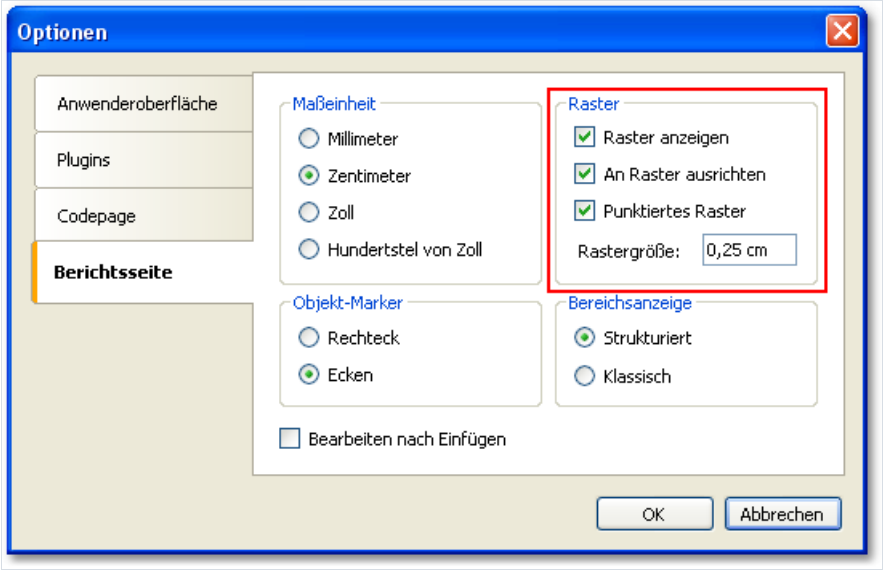
Die Qualität des Exports basiert im wesentlichen auf einem kompetenten Design des Anfangsberichts. Das folgende Beispiel soll dies erläutern:



Hier lässt sich eine minimale Kreuzung der beiden Objekte (untereinander) auf derselben Leiste erkennen. Die Anzahl der Berichte zur Berichtserstellung lag bei 150. Beim Export zu RTF werden 450 Zeilen erstellt (150 Zeilen für jedes Objekt und 150 für die Überkreuzung). Wird die Überkreuzung entfernt resultieren lediglich 300 Zeilen in der Tabelle. Bei umfangreichen Berichten mit einer hohen Anzahl an Objekten ist der Unterschied noch erheblicher. Dies hat natürlich auch einen Effekt auf die Ausgabegröße der Datei.

Bei der Erstellung von Tabellen in Berichten sollte die benachbarten Zellenränder stets im Auge behalten werden. Es ist wichtig, dass Zellen sich nicht überkreuzen oder in Ebenen angeordnet sind. Der Algorithmus des Exportfilter schneidet Zellen zwar ab, aber das Exportergebnis entspricht möglicherweise nicht den Anforderungen (sie erhalten nicht das gewünschte Ergebnis). Objekte sollten so arrangiert werden, dass sie sowohl vertikal, wie auch horizontal in einer Linie angeordnet sind. Hier können Ausrichtungslinien helfen.

Auch die Gitterausrichtung, kann für den Fall überlappender Zellen, hilfreich sein. Die Gitterausrichtung sollte in den Optionen des Designers aktiviert werden. Um die Ausrichtung zu vereinfachen kann das die Gitterdichte erweitert werden. Die Einstellungen für die Gitterdichte, und -ausrichtung kann im Menü "Ansicht|Optionen.." gefunden werden:



Zur Rahmung eines Textes ist es angebrachter den Rahmen des Textobjektes anstelle einer einzelnen graphischen Objektes, wie z.B. Linien, Rechtecke etc., zu nutzen. Hintergrundobjekte sollten nicht unter transparenten Objekten genutzt werden.

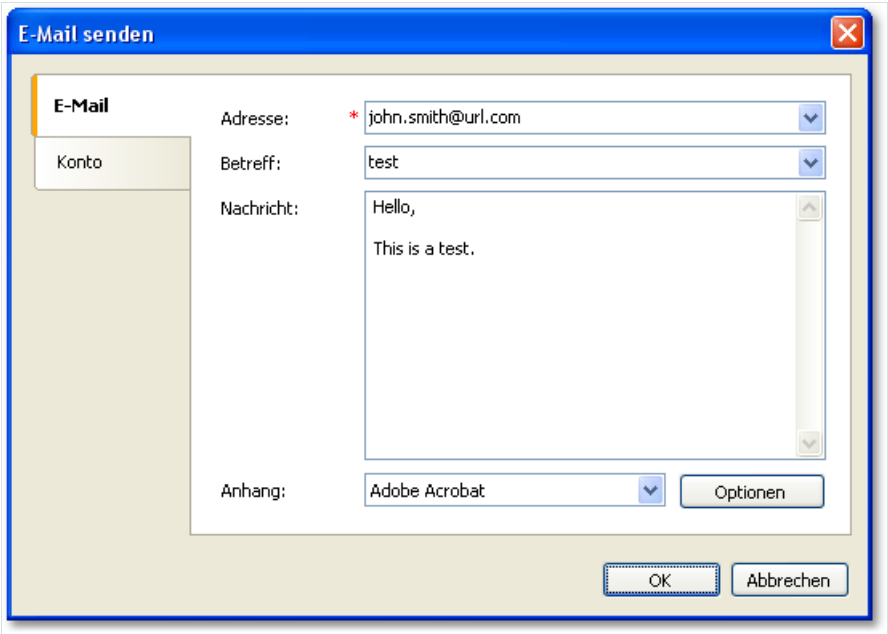
Die Anwendung dieser einfachen Regeln hilft bei der Erstellung eines perfekt aussehenden Berichts, nachdem dieser zu einem beliebigen textbasierten Format exportiert wurde.

Senden des Berichts per E-Mail

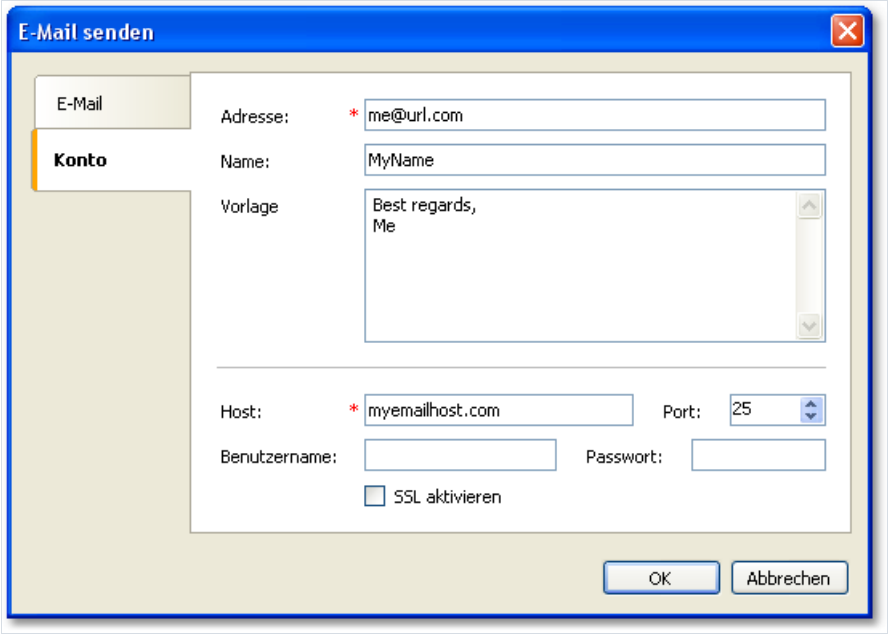
FastReport ermöglicht den Versand des erstellten Berichts per E-Mail. Dies kann auf zwei Arten stattfinden:

Modus	Beschreibung
SMTP	Dies ist der Standardmodus. Zum Versand einer E-Mail muss kein externes Programm genutzt werden.
MAPI	Dieser Modus kann programmatisch angestellt werden. Um dies zu tun, wird Config.EmailSettings.UseMAPI = true, oder, falls die Komponente EnvironmentSettings component genutz wird, wird dies folgendermaßen eingestellt EnvironmentSettings.EmailSettings.UseMAPI = true.Zum Versand einer E-Mail kann ein Standrad E-Mail-Client, z.B. Outlook Express, genutzt werden. Dieser Client muss das MAPI-Protokoll unterstützen.

Zum Versand einer E-Mail muss die E-Mail-Adresse des Empfängers angegeben werden. Auch der Betreff und der Text sollte eingegeben werden (dies ist jedoch nicht zwingend notwendig). Am unteren Rand des Dialogfeldes wird das Format des Berichts ausgewählt. Dieser wird dann an die Nachricht angehängt:



Bei der Nutzung des SMTP-Modus wird die Einrichtung eines Kontos erforderlich. Dieses muss lediglich einmal eingerichtet werden. Ist es eingerichtet speichert FastReport die Parameter in der Konfigurationsdatei. Die Parameter können auf der Registerkarte "Konto" gefunden werden. Pflichtfelder sind mit einem roten Stern gekennzeichnet.



Verlangt der Hostserver nach einer Authentifizierung muss der "Benutzername" und das "Passwort" eingegeben werden.

Minimale Systemanforderungen

Minimale Systemvoraussetzungen für die Installation und Nutzung von FastReport .NET:

- Betriebssystem MS Windows 7-11, Windows Server 2012-2019;
- CPU: 1 GHz;
- RAM: 512 MB;
- Sie benötigen außerdem das installierte .NET Framework Version 4.6.2 oder höher.

Kontakte und technische Unterstützung

Sie können Fragen zur Verwendung des Produkts jederzeit per E-Mail stellen, oder über das Formular auf der Website.

Wir begrüßen auch Ihre Vorschläge zur Verbesserung unseres Produkts.