

FastReport 3.0

Benutzer-Handbuch

Ausgabe 1.01

Copyright (c) 1998-2004, Fast Reports, Inc.

Inhaltsverzeichnis

Arbeitshinweise zu diesem Handbuch.....	4
Designer.....	5
Tastaturbefehle.....	7
Maus-Tasten.....	7
Werkzeugleisten.....	8
Globale Einstellungen.....	14
Entwurfs-Eigenschaften.....	14
Report-Eigenschaften.....	15
„Seite“-Eigenschaften.....	17
Report-Objekte.....	18
„Hello, World“-Report.....	18
Das „Text“-Objekt.....	21
HTML-Befehle im „Text“-Objekt.....	24
Ausdrücke mit Hilfe des „Text“-Objekts darstellen.....	25
Bänder im FastReport.....	26
„Daten“-Bänder.....	29
„Kundenlisten“-Report.....	30
Datenbank-Felder mit Hilfe des Text-Objekts anzeigen.....	32
Variablen.....	33
„Grafik“-Objekt.....	35
Report mit Bildern.....	38
Die mehrzeilige Textanzeige.....	39
Objekte von Text umhüllen.....	44
Daten in Form einer Tabelle anzeigen lassen.....	45
Etiketten drucken.....	48
„Child“-Bänder.....	50
Verschobene Objekte.....	51
Reports mit zwei Datenebenen (Master-Detail).....	53
Kopf- und Fußzeilen Daten-Bänder (Header und Footer).....	54
Report mit Gruppen.....	56
Andere Eigenschaften von Gruppen.....	58
Zeilennummerierung.....	59
Aggregat-Funktionen.....	61
Seiten- und Report-Summen.....	64
Das Einfügen von Aggregat-Funktionen.....	65
Das Aggregat-Funktionen-Aufruf-Merkmal.....	66
Werte formatieren.....	66
Inline-Formatierung.....	67
Bedingtes Hervorheben.....	69
Streifen anzeigen.....	70
Reports mit verschiedenen Seiten.....	72
Verschachtelter Report (Subreport).....	73
Nebeneinander stehende „Subreports“.....	74
Einschränkungen bei der Nutzung von „Subreports“.....	74
Die PrintOnParent-Eigenschaft.....	75
„Kreuztabellen“-Reports.....	76
Erstellen eines „Kreuztabellen“-Reports.....	78
Benutzung von Funktionen.....	81

Werte sortieren.....	82
Tabellen mit zusammengesetzten Kopfzeilen.....	82
Spaltenbreite anpassen.....	84
Schriftfarbe und Hervorhebung.....	86
Größe von Zeilen/Spalten anpassen.....	90
Eine „Kreuztabelle“ manuell füllen.....	91
Diagramme.....	93
Begrenzung der Anzahl der Diagramm-Werte.....	95
Einige nützliche Einstellungen.....	96
Diagramm mit speziellen Werten.....	97
Das „Script“.....	98
Beispiel eines „Scripts“.....	99
Struktur eines Scripts.....	101
Pascal- und C++-Scriptstruktur.....	101
Jscript- und Basic-Scriptstruktur.....	102
„Hello, World!“-Script.....	102
Objekte im „Script“ verwenden.....	103
Die Variablen aus der Variablenliste des Reports.....	104
Aufruf von Datenbank-Feldern.....	104
Nutzen von Aggregat-Funktionen im „Script“.....	104
Den Variablen-Werte in einem Report anzeigen.....	105
Ereignisse.....	105
Beispiel der Benutzung des „OnBeforePrint“-Ereignisses.....	106
Die Gesamtsumme aus dem Gruppen-Fuß im Gruppen-Kopf ausgeben.....	108
„OnAfterData“-Ereignis.....	111
weitere zur Verfügung stehende Objekte.....	112
„Engine“-Objekt.....	113
„Outline“-Objekt.....	114
Benutzung des „Engine“-Objekts.....	115
Anker (Anchors).....	117
Die Benutzung von „Outline“-Objekten.....	119
„OnManualBuild“-Ereignis der Seite.....	122
Erstellung von Objekten im „Script“.....	127
Dialog-Fenster.....	127
Steuerungs-Objekte.....	128
„Hello, World!“-Report als Dialog-Fenster.....	130
Eingabe von Parametern und Übertragung in einen Report.....	131
Zusammenspiel von Objekten.....	131

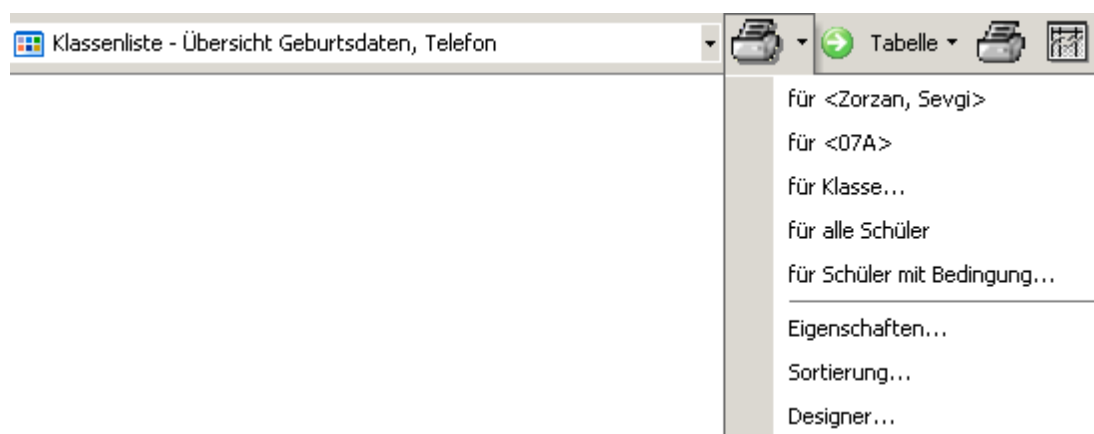
Arbeitshinweise zu diesem Handbuch

Die in diesem Handbuch verwendete Datenbank „DBDEMOS“ ist eine Beispieldatenbank. Anhand dieser Beispieldatenbank werden einzelne praktische Beispiele durchgeführt. Sie können an dieser Stelle die „Schueler“-Datenbank als Datenquelle (Dataset) anstelle „frxDBDataSet1“ verwenden.

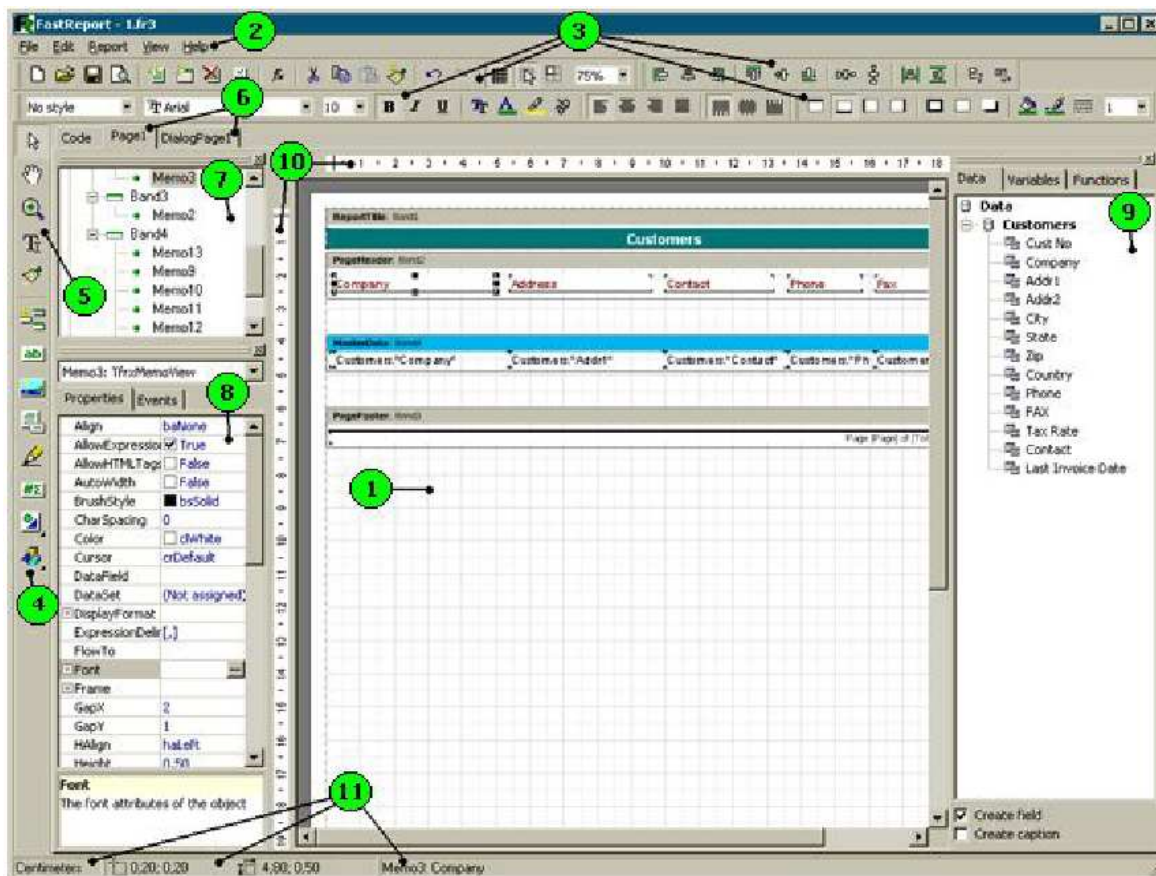
Designer

Die Komponenten werden mitgeliefert und sind schon im Designer eingebettet. Er unterstützt den Benutzer mit Werkzeugen für das Gestalten von Reports zusammen mit der Fähigkeit der gleichzeitigen Vorschau. Die Schnittstelle des Designers enthält mehrere Werkzeugleisten, die benutzt werden können, wo auch immer Sie wollen. Die Information über die Position der Werkzeugleisten wird in einer Ini-Datei gespeichert und wird jedes Mal wiederhergestellt, wenn Sie den Designer aufrufen.

Der Designer wird durch Anklicken des Befehls „Designer“ im Popup-Menü des Druckersymbols der vorgefertigten Reports aufgerufen:



Aufbau des Designers



Die Zahlen bedeuten:

- 1 - die Arbeitsfläche des Designers;
- 2 - Menüleiste;
- 3 - Werkzeugleisten;
- 4 - Objekt Toolbar;
- 5 - Designer-Mode-Toolbar;
- 6 – Reportseiten, Register;
- 7 – Fenster der Reportstruktur;
- 8 - „Objekt-Inspektor“-Fenster;
- 9 - „Datenstruktur“ Fenster. Sie können Elemente zu einer Reportseite aus diesem Fenster mit der Maus auf die Arbeitsfläche ziehen;
- 10 - Lineale;
- 11 – Status-Zeile, Beschreibung eines ausgewählten Elements auf der Arbeitsfläche

Tastaturbefehle






Tasten	Beschreibung
Strg+O	"Datei-->Öffnen..." Menü Befehl
Strg+S	"Datei-->Speichern..." Menü Befehl
Strg+P	"Datei-->Vorschau..." Menü Befehl
Strg+Z	"Bearbeiten-->Rückgängig..." Menü Befehl
Strg+Y	"Bearbeiten-->Wiederherstellen..." Menü Befehl
Strg+C	"Bearbeiten-->Kopieren..." Menü Befehl
Strg+V	"Bearbeiten-->Einfügen..." Menü Befehl
Strg+X	"Bearbeiten-->Ausschneiden..." Menü Befehl
Strg+A	"Bearbeiten-->Alles auswählen..." Menü Befehl
Pfeiltasten	Bewegen zwischen den Objekten
Entf	Löscht das ausgewählte Objekt
Enter	Ruft den Editor des selektierten Objektes auf
Umschalt+Pfeiltasten	Ändern der Größe der ausgewählten Objekte
Strg+Pfeiltasten	Bewegen der ausgewählten Objekte
Alt+Pfeiltasten	Das ausgewählte Objekt wird am nächstliegenden Objekt ausgerichtet.

Maus-Tasten

Operation	Beschreibung
Links Klick	<ul style="list-style-type: none"> - Auswählen eines Objektes - Einfügen eines neuen Objektes; - Bewegen von Objekten und Größe verändern;
Rechts Klick	Öffnet das Kontextmenü des ausgewählten Objektes
Doppelklick	Öffnet den Editor für das ausgewählte Objekte. Auf der leeren Arbeitsfläche öffnet es die Dialogbox "Seiteneinstellungen"
Mausrad	Scrollt über die Reportseite
Umschalt+Links Klick	Schaltet zusätzliche Objekte zur Auswahl hinzu
Strg+Links Klick	Markieren der Kästchen in dem Sie die gewünschten Objekte in den Rahmen umfahren.
Alt+Links Klick	Wenn das "Text"Objekt ausgewählt ist, können Sie den Text im Feld editiert.

Werkzeuggesteuen





„Designer-Mode“ Toolbar

Symbol	Name	Beschreibung
	Auswahlwerkzeug 	Standard, in dem mit dem Cursor Gegenstände auswählbar, modifizierbar ihre Größen veränderbar usw. sind; hebt die vorherige Auswahl auf.
	Hand 	Mit dem Klicken auf die Reportseite können Sie die Seite noch oben oder unten scrollen.
	Zoom 	Mit dem Klicken der linken Maustaste wird die Arbeitsfläche vergrößert dargestellt, mit der rechten Maustaste verkleinert.
	Text Editor 	Das Klicken auf ein Objekt auf der Arbeitsfläche erlaubt, seinen Inhalt direkt zu ändern.
	Format Copying 	Diese Schaltfläche übernimmt die Formatierung des selektierten Objekts. Wenn Sie nun auf ein anderes Objekt klicken, werden die Formate übernommen.

„Standard“ Toolbar






Symbol	Name	Beschreibung
	Neuer Report	Erzeugt einen neuen leeren Report.
	Report öffnen	Öffnet einen bestehenden Report aus einer gespeicherten Datei. (Tastenkürzel: "Strg+O").
	Report speichern	Speichert einen Report in einer Datei. (Tastenkürzel: „Strg+S“)
	Vorschau	Zeigt Aufbau und Vorschau des Reports. (Tastenkürzel: „Strg+P“)
	Neue Reportseite	Öffnet eine neue Seite im Report.
	Neue Dialogseite	Fügt dem Report eine neue Seite hinzu.
	Seite löschen	Löscht die geöffnete Seite.
	Seiteneinstellungen	Öffnet das Dialogfenster „Seiteneinstellungen“.
	Variablen	Ruft den Variablen-Editor auf.
	Ausschneiden	Schneidet das ausgewählte Objekt aus. (Tastenkürzel: „Strg+X“)
	Kopieren	Kopiert das ausgewählte Objekt in die Windows-Zwischenablage. (Tastenkürzel: „Strg+C“)
	Einfügen	Fügt Objekte aus der Windows-Zwischenablage ein. (Tastenkürzel: „Strg+V“)
	Format kopieren	Kopiert das Format des ausgewählten Objekts. Alle folgenden eingefügten Objekte haben dann dasselbe Format wie das Muster. Um Formateinstellungen zurückzusetzen klicken sie auf einen freien Bereich der Seite und dann diese Schaltfläche.
	Rückgängig	Macht die letzte Eingabe rückgängig. (Tastenkürzel: „Strg+Z“)
	Wiederherstellen	Stellt die zuletzt rückgängig gemachte Eingabe wieder her. (Tastenkürzel: „Strg+Y“)

Symbol	Name	Beschreibung
	Gitter anzeigen	Zeigt das Gitter auf der Seite an. Ein Gitter kann in den „Designer“-Eigenschaften eingestellt werden.
	Am Gitter ausrichten	
	Am Gitter anpassen	Modifiziert Größe/Ort der ausgewählten Objekte so, dass sie auf den Schnittpunkten des Gitters angeordnet werden.
	Zoom	Setze die Größeneinstellung der Arbeitsfläche.












„Text“ Toolbar

Symbol	Name	Beschreibung
No style	Format	Hier kann eine Formatvorlage gewählt werden.
Arial	Schriftart	Hier kann der Schrifttyp aus der Drop-Down-Liste ausgewählt werden. Nach dem Trennstrich werden die fünf zuletzt benutzten Schriften angezeigt.
8	Schriftgröße	Hier kann die Schriftgröße aus der Drop-Down-Liste ausgewählt oder manuell eingegeben werden.
B	Fett	Text wird fett geschrieben.
<i>I</i>	Kursiv	Text wird kursiv geschrieben.
<u>U</u>	Unterstreichen	Text wird unterstrichen.
	Schriftfarbe	Wählt eine Schriftfarbe aus der Drop-Down-Liste.
	Hervorheben	Zeigt das Dialogfenster mit Attributen zum Hervorheben des ausgewählten Objekts.
ab	Textdrehung	Ermöglicht es den ausgewählten Text zu drehen.
	Ausrichtung links	Text wird links ausgerichtet.
	Ausrichtung zentral	Text wird horizontal mittig ausgerichtet.
	Ausrichtung rechts	Text wird rechts ausgerichtet.
	Blocksatz	Text wird als Block ausgerichtet.

Symbol	Name	Beschreibung
	Ausrichtung oben	 Text wird nach oben ausgerichtet
	Ausrichtung Mitte	 Text wird vertikal mittig ausgerichtet
	Ausrichtung unten	Text wird nach unten ausgerichtet

„Rahmen“ Toolbar



Symbol	Beschreibung	Beschreibung
	Linie oben	Setzt eine Linie oben
	Linie unten	Setzt eine Linie unten
	Linie links	Setzt eine Linie links
	Linie rechts	Setzt eine Linie rechts
	Alle Rahmenlinien	Setzt einen Rahmen
	Kein Rahmen	Schaltet alle Linien aus
	Schatten	Setzt einen Schatten
	Hintergrundfarbe	Wählt eine Hintergrundfarbe aus der Drop-Down-Liste.
	Rahmenfarbe	Wählt eine Rahmenfarbe aus der Drop-Down-Liste.
	Rahmenstil	Wählt einen Rahmenstil aus der Drop-Down-Liste.
	Rahmenbreite	2 Wählt die Breite der Rahmenlinie aus der Drop-Down-Liste.

„Ausrichtung“-Toolbar



Diese Werkzeuge funktionieren nur, wenn mindestens zwei Objekte ausgewählt sind. Dabei ist das zuerst gewählte Objekt das Referenzobjekt, wenn nach diesem ausgerichtet werden soll.

Mehrere Objekte können nach zwei Methoden ausgewählt werden:

1. Methode

Auf der Tastatur die Taste „Strg“ drücken und gedrückt lassen; mit gedrückter linken Maustaste ein Rechteck ziehen, in dem alle auszuwählenden Objekte eingeschlossen sind; die Taste „Strg“ wieder loslassen. Diese Methode eignet sich nicht zum Ausrichten an einem Objekt, da damit kein Referenzobjekt festgelegt wird.

2. Methode

Ein Objekt mit der linken Maustaste anklicken - dies ist das Referenzobjekt; die Umschalttaste Großbuchstabe drücken und gedrückt lassen; andere Objekte mit der linken Maustaste anklicken; die Umschalttaste Großbuchstabe wieder loslassen.

Symbol	Beschreibung
	Ausrichtung links
	Ausrichtung horizontal zentriert
	Ausrichtung rechts
	Ausrichtung oben
	Ausrichtung vertikal zentriert
	Ausrichtung unten
	gleicher Abstand horizontal
	gleicher Abstand vertikal
	horizontal im "Band zentrieren
	Vertikal im Band zentrieren
	Gleiche Breite
	Gleiche Höhe

Globale Einstellungen

Entwurfs-Eigenschaften

Stellen Sie die Eigenschaften im Menü „Ansicht|Optionen...“ ein.

Entwurfsoptionen

Gitter

Typ: ☐ Zentimeter: 0,025 cm ☒ Gitter anzeigen
☐ Zoll: 0,1 Zoll ☒ Am Gitter ausrichten
☒ Pixel: 1 Punkt
 Dialogformular: 5 Punkt

Schriften

Codefenster: Courier New Größe: 10
 Memoeditor: Courier New Größe: 10
☒ Schrift des Objekts verwenden

Farben

Arbeitsbereich:
 Werkzeugfenster:
☐ LCD Gitterfarbe

Sonst

☒ Editor nach dem Einfügen anzeigen
☒ Bandbezeichnungen anzeigen
☒ Show drop-down fields list
☐ Freie Bandpositionierung
☒ Show startup screen
 Abstand zwischen Bändern: 15 Punkt

Vorgaben wiederherstellen OK Abbruch

Bereich „Gitter“

Hier kann man die benötigten Einheiten einstellen (Zentimeter, Zoll, Pixel) und die Zeilenabstände bestimmen. Man kann die Einheiten ebenso per Doppelklick auf der linken Seite der Statuszeile, auf dem die derzeitige Einstellung angezeigt wird, verändern.

Man kann ebenso einstellen, ob das Gitter angezeigt werden soll und ob daran ausgerichtet werden soll. Dies erfolgt ebenfalls auf der „Standard“-Toolbar des Designers.

Bereich „Schriften“

Man kann die Schrift für das „Codefenster“ (Pascal-Quelltext-Fenster) und den „Memoeditor“ (Objekt-Inhalt-Fenster) einstellen. Falls die Eigenschaft „Schrift des Objekts verwenden“ eingestellt ist, richtet sich die Schrift im Memoeditor nach der des editierten Objekts.

Bereich „Farben“

Falls ihnen die weiße Hintergrundfarbe der Arbeitsfläche des Designers oder die der Werkzeugleisten nicht zusagen, kann man sie über die Schaltflächen „Arbeitsbereich“ und „Werkzeugfenster“ verändern.

Die Eigenschaft „LCD Gitterfarbe“ erhöht den Kontrast der Gitterlinien etwas und verbessert ihre Sichtbarkeit auf LCD-Monitoren.

Bereich „Sonst“

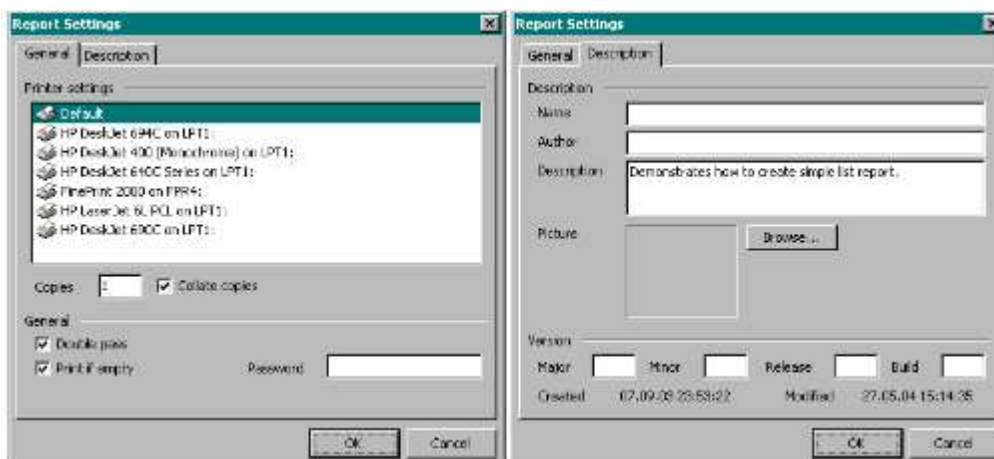
Die Eigenschaft „Editor nach dem Einfügen anzeigen“ steuert den Prozess des Einfügens neuer Objekte. Falls diese Eigenschaft eingeschaltet ist, wird der Editor nach jeder Einfügeoperation von Objekten angezeigt. Wenn man eine größere Anzahl leerer Objekte erstellt, ist es ratsam, diese Eigenschaft auszuschalten.

Mit dem Ausschalten der Eigenschaft „Bandbezeichnungen anzeigen“ kann man die Anzeige der Bezeichnungen der „Bands“ ausschalten, um Platz auf der Seite zu sparen. In diesem Fall werden die Bezeichnungen innerhalb der „Bands“ geschrieben.

Die Eigenschaft „Freie Bandpositionierung“ ermöglicht das freie Positionieren von „Bands“ auf der Seite. Diese Eigenschaft ist standardmäßig ausgeschaltet und „Bands“ werden automatisch nach ihrer Eigenschaft auf der Seite angeordnet. Der Abstand zwischen den Bändern kann im Feld „Abstand zwischen Bändern“ gesetzt werden.

Report-Eigenschaften

Ein Fenster mit Report-Einstellungen ist verfügbar im Menü „Report|Optionen...“. Der Dialog hat zwei Seiten:



Man kann einen Report mit einem der Drucker, der auf dem PC installiert ist, verbinden. Das bedeutet, der Druck wird vom ausgewählten Drucker durchgeführt. Das könnte nützlich sein, falls mehrere verschiedene Drucker an das System angeschlossen sind, z.B. können Textdokumente mit Schwarz-Weiß-Druckern verbunden werden, während Dokumente mit Grafiken mit Farbdruckern gedruckt werden können.

Es gibt das Symbol „Vorgabe“ in der Liste der Drucker. Falls dieses Symbol ausgewählt wurde, wird der Report mit keinem speziellen Drucker verbunden und es erfolgt das Drucken durch den Drucker, der im Betriebssystem als „Standard-Drucker“ eingestellt ist.

Man kann außerdem die Anzahl der Kopien einstellen und festlegen, ob die Kopien sortiert werden sollen. Der Wert, den der Benutzer im Dialog eingibt, wird im Drucker-Vorschau-Fenster angezeigt.

Falls die Eigenschaft „Zweipass“ eingestellt ist, wird die Report-Erstellung in zwei Schritten durchgeführt. Während des ersten Durchlaufs wird der Report gebildet und in Seiten eingeteilt, im zweiten Durchlauf wird die Standard-Formatierung durchgeführt, bei der die Ergebnisse in der Vorschau angezeigt werden.

Warum sind zwei Durchläufe notwendig?

Meistens wird diese Eigenschaft genutzt, wenn in einem Report die Gesamt-Seitenanzahl ausgegeben werden soll, z.B. „Seite 1 von 15“. Ein weiteres Anwendungsfeld ist die Ausgabe von Werten, die im ersten Durchlauf noch nicht vorhanden sind. Zum Beispiel über die „TotalPages“ System-Variable oder die Ausgabe einer Summe im Gruppen-Kopf, welche eigentlich erst im Gruppen-Fuß berechnet wird (Berechnungen dieser Art werden in einer in FastReport-eigenen Programmiersprache geschrieben).

Die komplette Anzahl der Seiten wird während des ersten Durchlaufs berechnet und ist dann verfügbar. Benutzt man diese Variable „TotalPages“ ohne die Eigenschaft „Zweipass“, wird nach dem Wort „von“ eine „0“ ausgegeben.

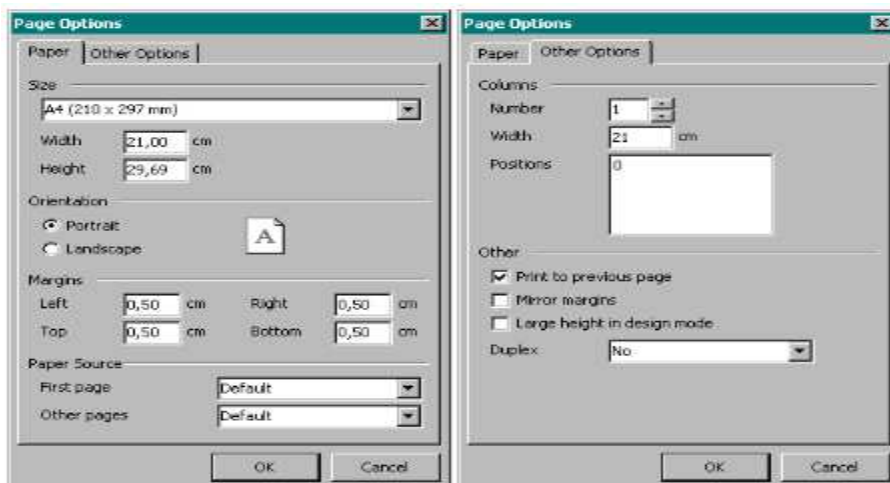
Die Eigenschaft „Drucken wenn leer“ erlaubt es, einen Report zu erstellen, der keine Datenzeilen beinhaltet. Falls diese Eigenschaft ausgeschaltet ist, können keine leeren Reports erstellt werden.

Das „Passwort“-Feld ermöglicht es ein Kennwort zu setzen, welches beim Öffnen des Reports abgefragt wird.

Die Felder auf der zweiten Seite des Dialogs ermöglichen es, eine Beschreibung des erstellten Reports einzufügen.

„Seite“-Eigenschaften

Seiten-Einstellungen sind verfügbar über das Menü „Datei|Seiteneinstellungen...“ oder mittels Doppelklick auf einen freien Platz auf der Seite. Der Dialog hat zwei Seiten:



Auf der ersten Dialog-Seite kann man die Größe und Ausrichtung des Papiers sowie die Ränder einstellen. Im Bereich „Papierzufuhr“ wählt man aus den jeweiligen Drop-Down-Listen Einstellungen für die „Erste Seite“ und für „Andere Seiten“.

Man kann die Anzahl der Spalten für mehrspaltige Druck festlegen. Die derzeitigen Einstellungen werden im Designer angezeigt.

Die Option „Auf vorige Seite drucken“ erlaubt es, neue Seiten auf dem freien Platz der vorigen Seite zu drucken. Diese Option kann genutzt werden, wenn ein Report aus mehreren Seiten besteht oder beim Druck von zusammengesetzten Reports.

Die Option „Ränder spiegeln“ überträgt linke und rechte Ränder der Seiten sowohl während der Vorschau, als auch während des Drucks.

Die Option „Seitenhöhe im Entwurfsmodus vergrößern“ steigert die Seitenhöhe von Zeit zu Zeit mehr. Die Eigenschaft ist nützlich, wenn mehrere Bänder auf der Seite angeordnet sind. Dennoch wird der Wert der Seitenhöhe während der Report-Erstellung nicht verändert.

Reports erstellen

In diesem Kapitel werden wir Eigenschaften der FastReport-Objekte betrachten, welche die Basis solcher Reports bilden. Zuerst werden wir lernen, wie man einen einfachen Report erstellt, der Daten aus Datentabellen enthält.

Report-Objekte

Ein leerer Report in FastReport wird als leeres Blatt dargestellt. Jedem Teil dieser Seite kann der Benutzer Objekte zuweisen, welche unterschiedliche Informationen anzeigen (Text und/oder Grafik). Genauso kann das Aussehen des Reports bestimmt werden. Lassen sie uns kurz die Aufgaben dieser FastReport-Objekte, die im Standardpaket enthalten sind, beschreiben:

- das „Band einfügen“-Objekt ermöglicht es einen Report-Bereich zu schaffen, welcher bestimmte Eigenschaften besitzt;
- das „Text“-Objekt zeigt eine oder mehrere Textzeilen innerhalb eines rechteckigen Bereichs an;
- das „Grafik“-Objekt fügt eine Grafik-Datei im Format „BMP“, „JPG“, „ICO“, „WMF“, oder „EMF“ ein;
- das Objekt „Linien“ fügt vertikale oder horizontale Linien ein;
- das Objekt „Systemtext“ fügt Systemvariable (date, time, page number, etc) sowie Aggregatwerte ein,;
- das Objekt „Unterreport“ ermöglicht es, einen zusätzlichen Report in den Basisreport einzufügen;
- das Objekt „Zeichnen“ fügt verschiedene geometrische Formen ein (Diagonale Linie, Rechteck, Abgerundetes Rechteck, Ellipse, Dreieck und Raute);
- das Objekt „Diagramme“ fügt Diagramme verschiedener Form (Kreisdiagramm, Histogramm usw.) ein;
- das „RichText“-Objekt fügt einen formatierten Text im „RichText“-Format (RTF) ein;
- das „Kontrollkästchen“-Objekt öffnet ein Kontrollkästchen mit entweder einem Karo oder einem Kreuz;
- das „Strichcode“-Objekt fügt Daten in Form von Strichcodes ein (es sind viele verschiedene Strichcode-Arten verfügbar);
- das „OLE“-Objekt ermöglicht es, jedes Objekt mit Hilfe der OLE-Technologie einzufügen.

Die Basisobjekte, die Sie am meisten benötigen werden, sind „Bänder einfügen“ und „Text“-Objekt. Sie werden später in diesem Kapitel näher über die Details dieser Objekte informiert.

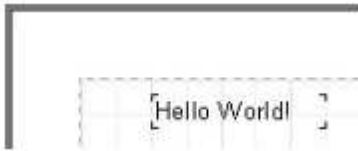
„Hello, World“-Report

Folgendermaßen erstellen wir einen neuen Report:

Wir wählen aus dem Menü „Datei“ den Befehl „Neuer Report“.




Der Report wird lediglich eine einzige Aufschrift haben („Hello World“). Klicken Sie auf das „Text“-Objekt der Designer-Mode-Toolbar. Bewegen Sie den Mauszeiger auf den gewünschten Platz auf der Seite und klicken erneut. Das Objekt wird an diese Stelle gesetzt. Das Texteditor-Fenster geöffnet sich sofort, falls dem nicht so ist (dies kann in


den Entwurfs-Optionen eingestellt werden), machen Sie einen Doppelklick auf das Objekt. Geben Sie den Text „Hello World“ ein und drücken die „OK“-Schaltfläche (grünes Häkchen).

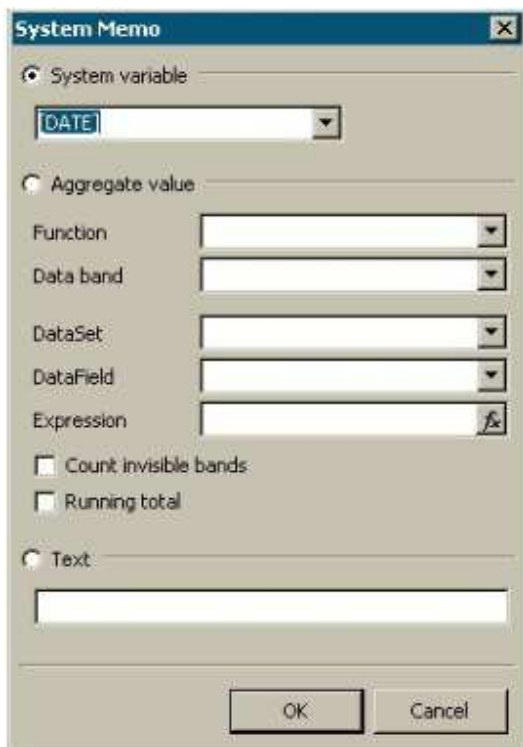


Der Report wurde erstellt. Um ihn sich in einer Vorschau anzusehen, wählen Sie entweder im Menü „Datei“ den Befehl „Vorschau“ oder drücken die entsprechende Schaltfläche auf der Werkzeugleiste. Das Vorschaufenster enthält lediglich die Reportseite auf der die Aufschrift „Hello World!“ zu lesen ist. Dieser Report kann nun ausgedruckt, in einer Datei gespeichert oder in ein unterstütztes Format exportiert werden.

Lassen Sie uns unseren ersten Report etwas erweitern. Machen Sie den Text („Hello World!“) fett, mit gelbem Hintergrund und einem Rahmen. Fügen Sie außerdem das aktuelle Datum neben dem Text ein.

Öffnen Sie noch einmal den FastReport und wählen ein Text-Objekt mit Hilfe der Maus aus. Sie werden bemerken, dass einige Schaltflächen auf der Werkzeugleiste aktiviert werden. Finden Sie die „Fett“-Schaltfläche  auf der „Text“-Werkzeugleiste und drücken Sie diese. Um den Rahmen zu erstellen, klicken Sie auf „Alle Rahmenlinien“  (zweites Symbol von rechts). Falls erforderlich, können einige Linien ausgeschaltet werden mit Hilfe der entsprechenden Schaltflächen. Der Benutzer kann die Linienfarbe, seine Breite und seinen Stil einstellen, falls dies erforderlich ist. Wir nutzen die „Hintergrundfarbe“  Schaltfläche und wählen Gelb aus der Auswahlliste.

Der einfachste Weg, um das Datum einzufügen, ist es, das „Systemtext“-Objekt  zu verwenden. Sie fügen es der Seite in derselben Weise zu, wie Sie es mit dem ersten Objekt gemacht haben. Dann wählen Sie „Systemvariable“ aus dem Editor-Fenster und „[DATE]“ aus der Auswahlliste darunter.



Schließen Sie den Editor mit der „OK“-Schaltfläche und starten Sie den Report um das Ergebnis zu sehen.

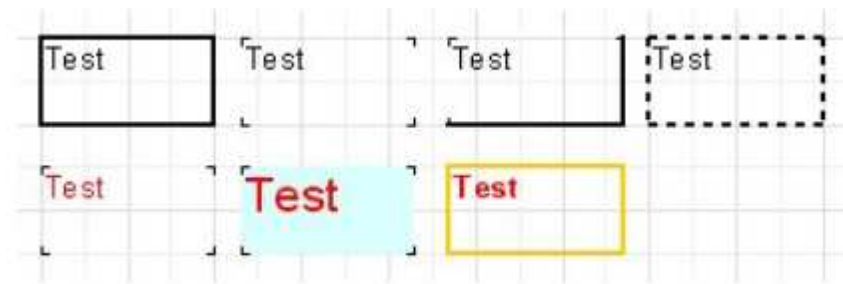


Das „Text“-Objekt

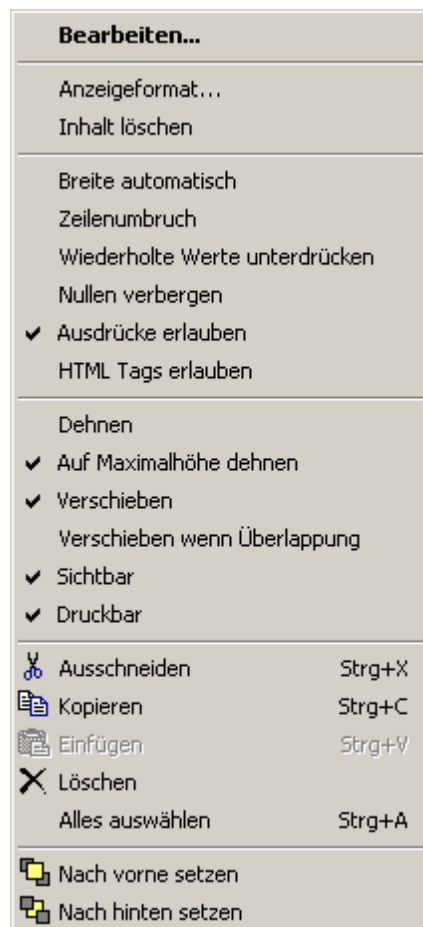
Das „Text“-Objekt besitzt vielgestaltige Eigenschaften. Nun wissen Sie, dass es möglich ist, einem Text einen Rahmen und Füllung einzufügen. Ein Text kann in jeglicher Schriftart und Größe angezeigt werden. Alle Parameter können sichtbar mit der Hilfe der Werkzeugleiste gesetzt werden:



Hier einige Beispiele zum Text-Design:



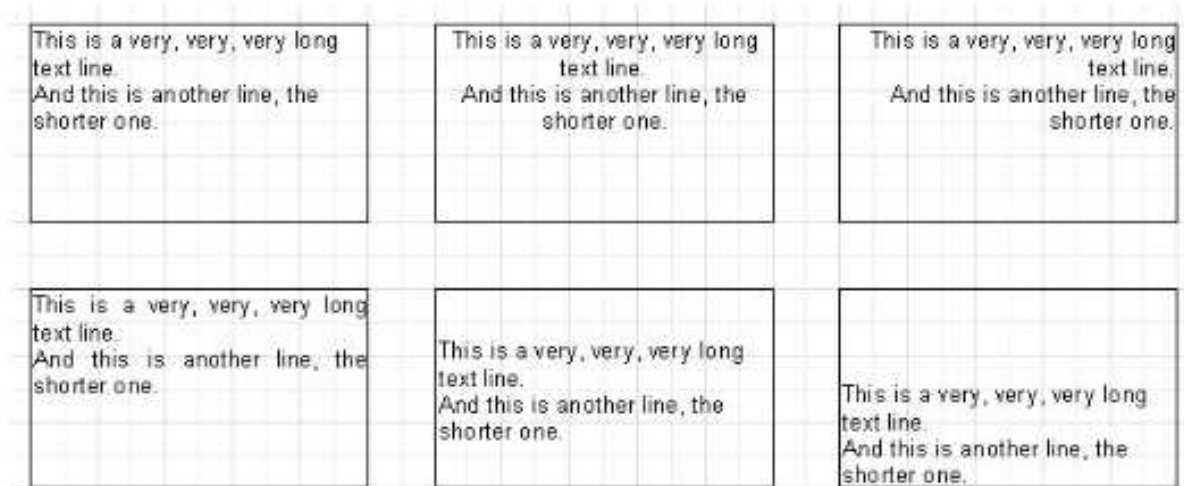
Nun werden wir uns mit anderen Eigenschaften dieses Basis-Objekts vertraut machen. Als Beispiel lassen Sie uns zwei Text-Objekte und zwei Linien einfügen:



*This is a very, very, very long text line.
And this is another line, the shorter one.*


Stellen Sie den Objekt-Rahmen ein und verändern Sie das Objekt dann mit Hilfe der Maus zu 9x3 cm. Sie sehen, dass das Objekt nicht nur einen einzeiligen Text, sondern auch einen mehrzeiligen darstellen kann. Nun reduzieren Sie die Objekt-Breite auf 5 cm. Nun ist es offensichtlich, dass lange Zeilen keinen Platz im Objekt haben und deshalb werden sie abgeschnitten. Dies passiert aufgrund der „WordWrap“- Objekt Eigenschaft. Falls es ausgeschaltet ist (entweder im Objekt-Inspektor-Fenster oder im Objekt-Kontext-Menü über die rechte Maustaste und die Eigenschaft „Zeilenumbruch“), werden zu lange Zeilen einfach abgeschnitten.

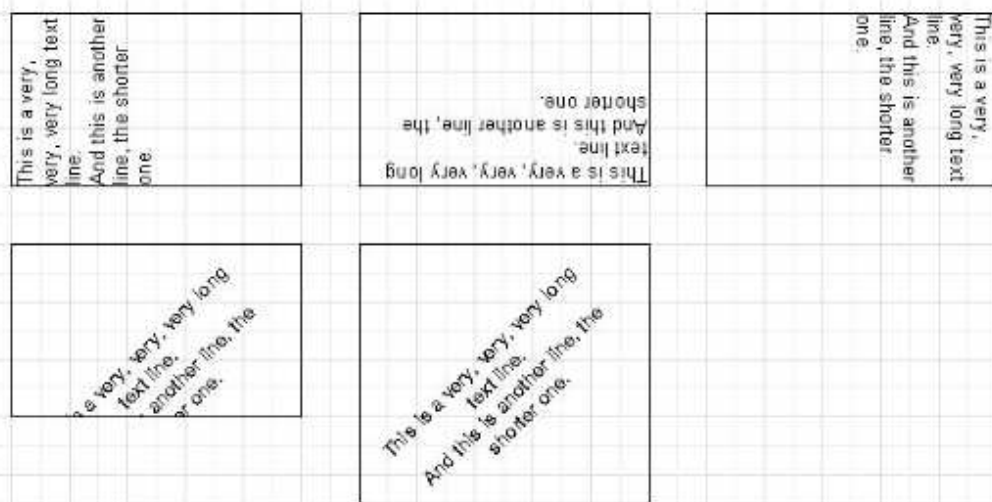
Nun lassen Sie uns ausprobieren, wie die Textausrichtung innerhalb des Objekts funktioniert. Ausrichtung-Schaltflächen sind in der „Text“-Werkzeugleiste angeordnet und ermöglichen eine horizontale oder vertikale Textausrichtung. Seien Sie vorsichtig mit dem „Blocksatz“-Schaltfläche; diese Schaltfläche ermöglicht die Ausrichtung des Textes an beiden Objekt-Kanten. Bei der Durchführung dieser Option muss die „WordWrap“-Option auf „True“ gestellt sein.



Von links oben nach rechts unten werden folgende Eigenschaften gezeigt:

- linksbündig, oben ausgerichtet
- zentriert, oben ausgerichtet
- rechtsbündig, oben ausgerichtet
- Blocksatz, oben ausgerichtet
- linksbündig, mittig ausgerichtet
- linksbündig, unten ausgerichtet

Ein ganzer Text kann in jeden Winkel gedreht werden, innerhalb der Begrenzung von 0 bis 360 Grad. Die  -Schaltfläche in der „Text“-Werkzeugleiste erlaubt es, den Text schnell um 45, 90, 180 und 270 Grad zu drehen. Wenn Sie den Text in einem anderen Winkel drehen wollen, verwenden Sie den Objekt-Inspektor. Die Eigenschaft „Drehen“ (Rotation) stellt den gewollten Winkel ein. Wenn man den Text mit einem anderen Wert als 90, 180 oder 270 dreht, kann der Text die Grenzen des Objekts überschreiten, wie in unserem Fall (siehe Abb. unten). Vergrößern Sie die Objekt-Höhe ein wenig, so dass der Text in das Objekt passt.



Lassen Sie uns kurz mit einigen „Text“-Objekt-Eigenschaften befassen, die die Erscheinung beeinflussen. Viele dieser Eigenschaften sind nur im Objekt-Inspektor verfügbar:

- BrushStyle – Art der Objekt-Füllung;
- CharSpacing – Platz zwischen Symbolen in Pixeln;
- GapX, GapY – Einrücken des Texts in der linken Seite des Objekts und der oberen Grenze (in Pixeln);
- LineSpacing – Platz zwischen Zeilen (in Pixeln);
- ParagraphGap – die erste Textzeile rückt ein (in Pixeln);

HTML-Befehle im „Text“-Objekt

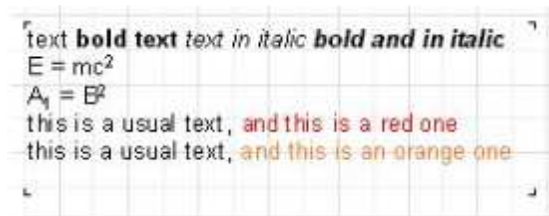
Dieses Objekt versteht einige der einfachsten HTML-Befehle. Die Befehle können innerhalb des Textes auf dem Objekt eingefügt werden. Befehle sind voreingestellt nicht möglich. Um sie zu ermöglichen wählen Sie entweder das „HTML-Befehle zulassen“-Symbol im Objekt-Kontext-Menü oder Sie stellen die „HTML-Befehle zulassen“-Eigenschaft im Objekt-Inspektor ein. Hier die Liste der unterstützten Befehle:

- – Text fett schreiben
- <i> – Text in italic schreiben
- <u> – Text unterstreichen
- <sub> – Tiefgestellt
- <sup> – Hochgestellt
- – Schriftfarbe

Vielleicht haben Sie als erstes festgestellt, dass nicht viele Befehle unterstützt werden, aber es sind annähernd genug Befehle für den Großteil der Anwendungen. Es ist unmöglich, Schriftgröße und Name zu verändern; andernfalls wird das Interpretationsprogramm in FastReport beträchtlich komplizierter.

Die folgenden Beispiele zeigen, wie man die Befehle benutzen kann.

```
text <b>bold text</b> <i>text in italic</i> <b><i>bold and in italic</b></i>
E = mc<sup>2</sup>
A<sub>1</sub> = B<sup>2</sup>
this is a usual text, <font color=red>and this is a red one</font>
this is a usual text, <font color="#FF8030">and this is an orange one</font>
```



Ausdrücke mit Hilfe des „Text“-Objekts darstellen

Eines der wichtigsten Eigenschaften dieses universellen Objekts ist nicht nur einen statischen Text darstellen zu können, sondern auch Ausdrücke. Diese können zur selben Zeit mit einem Text auf dem Objekt angeordnet werden. Probieren wir ein einfaches Beispiel aus, um zu sehen, wie es gemacht werden kann.

Im vorhergehenden Teil haben Sie bereits einen Report erstellt, der die Zeile „Hello World!“ und ein aktuelles Datum ausgibt. Um das zu erreichen, mussten Sie zwei Objekte in den Report einfügen. Demnach enthält eines dieser Objekte einen Grußtext, während das andere die „DATE“-Systemvariable enthält. Wie auch immer, um sowohl eine Zeile als auch ein Datum anzeigen zu lassen, können Sie nur ein „Text“-Objekt verwenden. Um dies zu erreichen, müssen Sie eine Zeile in das Objekt einfügen und das schaut dann ungefähr so aus:

Hello, World! Today is [DATE].

Demzufolge erscheint bei der Ausführung des Reports in etwa folgendes:

Hello, World! Today is 01.01.2004.

Wie kommt es zu so einem Ergebnis?

Während der Report-Erstellung, trifft der Interpreter im Text auf einen Ausdruck, der in eckige Klammern gesetzt ist, berechnet diesen und setzt den erhaltenen Wert wieder zurück in den Text (natürlich ohne Klammern). Das „Text“-Objekt kann unzählige Anzahlen von Ausdrücken zusammen mit einem normalen Text beinhalten. Beide einzelnen Variablen und Ausdrücke können in Klammern gesetzt werden (z.B. $[1+2*(3+4)]$). Jede Konstante, Variable, Funktion und jedes Datenbank-Feld kann als Ausdruck verwendet werden. Wir werden dies später genauer betrachten.

Demzufolge erkennt FastReport Ausdrücke in Klammern automatisch im Text. Aber was machen Sie, wenn der Text eckige Klammern besitzt, die Sie aber nicht als Ausdruck ausgegeben haben möchten? Zum Beispiel, wenn Sie folgenden Text darstellen sollen:

a[1] := 10

FastReport hält es als einen Ausdruck und zeigt folgendes:

a1 := 10

Dies ist natürlich sehr ungünstig für Sie. Eine der Möglichkeiten, um solch eine Situation zu verhindern, ist es die Ausgabe des Ausdrucks auszuschalten. Einfach die Eigenschaft „AllowExpression“ („Ausdrücke erlauben“ im Kontextmenü) ausschalten, so werden alle Ausdrücke im Text ignoriert. In unserem Beispiel würde FastReport genau das zeigen, was wir brauchen:

a[1] := 10

Manchmal ist es erforderlich, dass ein Text sowohl einen Ausdruck, als auch einen Text in eckigen Klammern wiedergibt, z.B.:

```
a[1] := [myVar]
```

Das Ausschalten von Ausdrücken erlaubt es, eckige Klammern im erforderlichen Bereich anzuzeigen, aber es verhindert die Ausgabe von Ausdrücken. In diesem Fall ermöglicht es FastReport einen neuen Satz Symbole zu verwenden, die die Ausdrücke bestimmen. Die Objekt-Eigenschaft „ExpressionDelimiters“, welche vergleichbar mit „[,]“ ist, ist verantwortlich dafür. In unserem Fall kann der Benutzer spitze Klammern für Ausdrücke verwenden, anstelle von eckigen:

```
a[1] := <myVar>
```

Gleichzeitig muss der „<,>“-Wert in der „ExpressionDelimiters“-Eigenschaft (Objekt-Inspektor) eingestellt werden. Wie Sie sehen können, trennt das Komma Öffnungs- und Schließsymbole. Es gibt eine einzige Einschränkung: das Öffnungs- und Schließsymbol darf nicht das gleiche sein, demnach würde „%,%“ nicht funktionieren. Man kann verschiedene Symbole verwenden, zum Beispiel „<%,%>“. Demnach schaut unser Beispiel nun so aus:

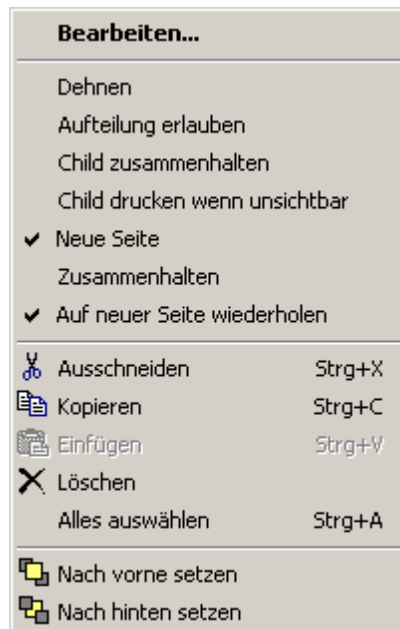
```
a[1] := <%myVar%>
```


Bänder im FastReport

Bänder werden zur logischen Objekt-Gruppierung genutzt. Demzufolge wird, wenn Sie ein Objekt auf einem Band platzieren, z.B. im „Seitenkopf“, der FastReport informiert, dass das gegebene Objekt oben auf jeder Seite eines Reports zu stehen hat. Auf dieselbe Art und Weise wird das Band „Seitenfuß“ am Ende jeder Seite stehen mit all den Objekten, die auf ihm sind.

Lassen Sie es uns mit einem Beispiel verdeutlichen. Erstellen wir einen Report, der die „Hello!“ Inschrift im Kopf der Seite stehen hat, ein aktuelles Datum rechts davon und die Seitenzahl am Ende der Seite (rechts).

Öffnen Sie den FastReport Designer und klicken Sie die Schaltfläche „Neuer Report“ in der Werkzeugleiste. Sie werden eine Reportseite sehen, die bereits drei Bänder enthält: „Report Title“, „Master Data“ and „Page Footer“. Nehmen Sie das „Master Data“-Band für eine Weile heraus (um dies zu tun klicken Sie entweder auf einen freien Platz auf dem Band oder auf seinen Kopf und löschen mit der Entfernen-Taste oder mit einem entsprechenden Befehl aus dem Kontextmenü).



Fügen Sie nun ein neues Band hinzu („Page Header“). Um dies durchführen zu können klicken Sie die Schaltfläche „Band einfügen“  und wählen „Page Header“ aus der Auswahlliste aus. Ein neues Band wurde der Seite hinzugefügt. Gleichzeitig sind die bereits bestehenden Bänder nach unten gerutscht. Der FastReport-Designer fügt neue Bänder der Seite automatisch hinzu und daraus ergibt sich, dass Kopfbänder oben, Datenbänder in der Mitte und Fußbänder unten am Seitenende platziert sind.

Fügen Sie nun die folgenden Objekte hinzu:

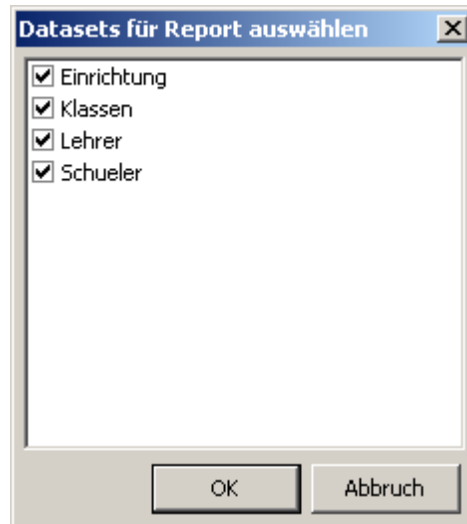
Setzen Sie ein „Systemtext“-Objekt in das „Page Header“-Band und wählen die Systemvariable „[DATE]“ im Editor aus (Sie sollten sich erinnern, dass das Datum mit Hilfe eines üblichen „Text“-Objekts angezeigt werden kann, indem man „[DATE]“ im Editor schreibt). Sie fügen ein „Text“-Objekt, welches den Text „Hello!“ beinhaltet in das „Page Header“-Band ein. Außerdem wurde das Objekt, welches die Seitenzahl angibt, wie man sehen kann, bereits in das „Page Footer“-Band eingesetzt.

ReportTitle: ReportTitle1	
[Hello!]	
PageHeader: PageHeader1	
	[DATE]
PageFooter: PageFooter1	
	[Page#]

Wenn Sie den Report durchlaufen lassen, werden Sie sehen, dass die Objekte auf der Seite so ausgegeben werden, wie Sie diese brauchen.

Hello!	27.01.04
1	

Demnach sind die Bänder dafür verantwortlich, dass Objekte an entsprechender Stelle stehen. Abhängig vom Band-Typ können wir entweder Objekte auf dem Kopf oder dem Fuß einer Seite, auf die erste oder letzte Seite setzen. Diese Basisbänder, die Sie wohl in den meisten Reports brauchen werden, arbeiten nach folgendem Prinzip:



- das „Page Header“-Band wird stets an den Anfang einer Seite gesetzt;
- das „Page Footer“-Band sitzt immer am Fuße einer Seite;
- das „Report Title“-Band steht immer auf der ersten Seite, allerdings unter dem „Seiten Kopf“-Band;
- das „Report Summary“-Band ist stets auf der letzten Seite am Ende eines Reports;

„Daten“-Bänder

In diesem Kapitel werden Sie die wohl interessanteste Eigenschaft unseres Reports ausprobieren. Die Möglichkeit, Daten aus Datenbank-Tabellen oder -Queries auszudrucken. Wofür brauchen Sie in diesem Fall Tabellen? Es ist eine gewünschte Anzahl von Zeilen (Records), von der jede eine bestimmte Anzahl an Spalten (Felder) hat. Um Informationen dieser Art zu drucken, benutzt der FastReport einen bestimmten Typ Bänder (Datenbänder). Das sind Bänder des Namens „xxx data level“. Um eine komplette Tabelle oder einige ihrer Felder zu drucken, müssen Sie einige solcher Bänder in den Report einfügen, diese mit der Tabelle verbinden und dann die Objekte mit den Feldern, die der Benutzer ausdrucken möchte, in sie einsetzen. Bei der Druckausgabe sollen genauso viele Zeilen gedruckt werden wie Datensätze (Records) in der Tabelle vorhanden sind. Sollte kein Platz mehr auf der Seite sein, werden neue Reportseiten gebildet.

Um die Daten im Report verfügbar zu machen, müssen die Datenquellen (Datasets), die Sie im Report nutzen, ausgewählt werden. Um das zu machen, wählen Sie im Menü „Report“ den Dialog „Daten...“. Anschließend wählen Sie die erforderlichen Quellen im geöffneten Fenster.

„Kundenlisten“-Report

Unser zweiter Report wird wesentlich komplizierter als der erste (er wird Datentabellen, eine Liste der Kunden einer Firma, enthalten). Um dies zu demonstrieren, benutzen wir hier die Beispieldatenbank DBDEMOS.

Öffnen Sie einen neuen Report mit der Schaltfläche „Neuer Report“, so dass FastReport automatisch ein leeres Formular mit drei Bändern öffnet („Report Title“, „Master Data“ und „Page Footer“). Um die Tabellen im FastReport sichtbar zu machen, wählen Sie dazu im Menü „Report“ den Dialog „Daten...“ und selektieren die Tabelle (es ist die einzige Tabelle in der Liste im Moment). Nachdem man das Dialogfenster geschlossen hat, werden die Tabelle und ihre Felder im Datenstruktur-Fenster sichtbar.

Machen Sie sich nun an die Formular-Erstellung. Setzen Sie das „Text“-Objekt mit dem Text „Kundenliste“ in das „Report Title“-Band. Danach verbinden Sie das „Master Data“-Band mit der Datenquelle. Dies kann auf folgende drei Arten erfolgen:

- Doppelklick auf das Band und Auswahl der Tabelle
- wählen des „Bearbeiten...“-Eintrags aus dem Kontext-Menü des Bandes (rechte Maustaste auf die Titelseite des Bandes) und Auswahl der Tabelle
- Klicken auf die „DataSet“-Eigenschaft im Objekt-Inspektor und Auswahl der Tabelle

Nun müssen Sie vier Objekte (Kundennummern - Feld CustNo, Kundennamen - Feld Company, Telefon - Feld Phone und Fax - Feld FAX) auf das Band setzen. Hier wird dies in verschiedener Art und Weise erklärt, um die Möglichkeiten des FastReport-Designers zu demonstrieren:

Setzen Sie das erste „Text“-Objekt auf das „Master Data“-Band und geben Sie „[frxDBDataSet1.“CustNo“]“ ein. Das ist der komplizierteste Weg, da der Eintrag manuell eingegeben werden muss und die Gefahr besteht, dass Sie den Text inkorrekt eingeben. Um die Eingabe solcher Texte einfacher zu gestalten, können Sie den Ausdrucks-



Designer nutzen (seine Schaltfläche **Ausdruck einfügen**) befindet sich auf der Werkzeugleiste des „Text“-Objekt_Editors).

Um unser Feld einzufügen doppelklicken Sie auf das erforderliche Element auf dem geöffnetem Dialog. Beim Klicken der „OK“-Schaltfläche schließt sich das Dialogfenster und Sie haben das Feld in das Objekt eingefügt.

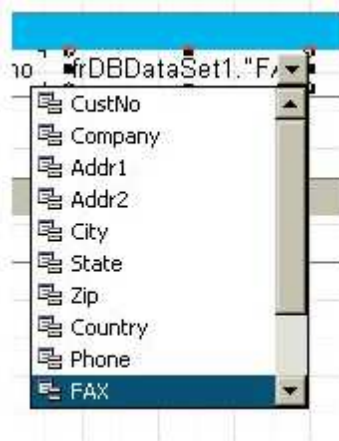
Der zweite Weg, um das Datenbank-Feld in den Report einzufügen, ist ziemlich ähnlich dem ersten. Mit Hilfe der Einstellung der Eigenschaften „DataSet“ und „DataField“ im Objekt-Inspektor wird es gemacht. Setzen Sie das zweite Objekt in das „Master Data“-Band, ohne etwas in den Editor zu schreiben. Nun setzen Sie die Eigenschaften im Objekt-Inspektor:

```
DataSet = frxDBDataSet1
DataField = 'Company'
```

Diese beiden Eigenschaften brauchen Sie nur noch mit Hilfe der Maus auswählen.

Die dritte Möglichkeit ist, das erforderliche Feld aus dem Datenstruktur-Fenster zu ziehen und es im Report abzulegen. Dies ist der einfachste und eindeutigste Weg. Nehmen Sie das „Telefon“-Feld mit Hilfe der Maus und ziehen es in das „Master Data“-Band.

Schließlich der vierte Weg. Setzen Sie ein leeres Text-Objekt in das Band und bewegen dann den Cursor auf das Objekt. Im rechten Teil des Objekts werden Sie eine Schaltfläche mit dem Pfeil nach unten sehen. Das ist die Datenbank-Feld-Liste. Drücken Sie die Schaltfläche und wählen das „FAX“-Feld aus der Liste aus. Sie können diese Möglichkeit nutzen, wenn das Band mit den Daten verbunden ist.



Jetzt ist unser Report fertig:

ReportTitle: ReportTitle1			
Customer list			
MasterData: MasterData1			
[frDBDataSet1]	[frDBDataSet1, "Company"]	[frDBDataSet1, "Pho"]	[frDBDataSet1, "FAX"]
PageFooter: PageFooter1			
[Page#]			

Drücken Sie nun die Vorschau-Schaltfläche und sehen Sie das Ergebnis:

Customer list			
1221	Kauai Dive Shoppe	808-555-0269	808-555-0278
1231	Unisco	809-555-3915	809-555-4958
1351	Sight Diver	357-6-876708	357-6-870943
1354	Cayman Divers World Unlimited	011-5-697044	011-5-697064
1356	Tom Sawyer Diving Centre	504-798-3022	504-798-7772
1380	Blue Jack Aqua Center	401-609-7623	401-609-9403
1384	VIP Divers Club	809-453-5976	809-453-5932
1510	Ocean Paradise	808-555-8231	808-555-8450
1513	Fantastique Aquatica	057-1-773434	057-1-773421
1551	Marmot Divers Club	416-698-0399	426-698-0399
1560	The Depth Charge	800-555-3798	800-555-0353
1563	Blue Sports	610-772-6704	610-772-6898
1624	Makai SCUBA Club	317-649-9098	317-649-6787
1645	Action Club	813-870-0239	813-870-0282
1651	Jamaica SCUBA Centre	011-3-697043	011-3-697043
1680	Island Finders	713-423-5675	713-423-5676

Datenbank-Felder mit Hilfe des Text-Objekts anzeigen

Wie man sehen kann, ist das „Text“-Objekt in Ergänzung zu dargestellten statischen Texten und Ausdrücken in der Lage, Daten aus Tabellen anzuzeigen. Darüber hinaus können Sie dies auf zwei Wegen machen:

- indem Sie entweder eine Verbindung zum Datenbank-Feld in das „Text“-Objekt legen oder ein Objekt mit dem erforderlichen Feld mit Hilfe der „DataSet“- und „DataField“-Eigenschaften verbinden. Der erste Weg ist vorzuziehen, wenn der Feldinhalt in ein und demselben Objekt angezeigt werden soll. Zum Beispiel:

Contact person: [frxDBDataSet1."Contact_Person"]

Wie Sie sehen können, gibt es eine bestimmte Syntax um eine Verbindung zum Datenbank-Feld zu setzen:

- data_set_name.field_name

Der Feldname (sowie der DataSet-Name) kann Leerzeichen enthalten. Leerzeichen zwischen dem „Punkt“ und dem „Quote“ Symbol sind nicht verboten.

Es kann nicht nur eine Verbindung zum Feld in das Text-Objekts gesetzt werden. Es können genauso gut verschiedene Rechenoperationen zum Feld angestellt werden:

Length (cm): [<frxDBDataSet1."Length_in"> * 2.54]
(um von Zoll auf Zentimeter umzurechnen)

Seien Sie vorsichtig bei der Benutzung von eckigen und spitzen Klammern. Erinnern Sie sich, dass eckige Klammern in der Voreinstellung genutzt werden, um Ausdrücke hervorzuheben, die im Text-Objekt enthalten sind. Im Falle des Gebrauchs können die eckigen Klammern auch mit anderen Öffnungs- und Schließ-Sequenzen ersetzt werden (Sie sehen den angezeigten Ausdruck als „Text“-Objekt Eintrag). Spitze Klammern werden innerhalb eines Ausdrucks verwendet, um die Variablen im FastReport und die Datenbank-Felder hervorzuheben. Um es logisch zu machen, sollten wir folgendes schreiben:

Contact person: [*<frxDBDataSet1."Contact_Person">*]

anstelle von:

Contact person: [*frxDBDataSet1."Contact_Person"*]

Trotzdem sind beide Schreibweisen korrekt, wenn der FastReport das Weglassen von spitzen Klammern zulässt, falls ein Ausdruck nur eine Variable oder nur ein Datenbank-Feld enthält. Falls ein Text-Objekt mehrere Datenfelder oder Ausdrücke enthält, sind die spitzen Klammern obligatorisch:

Length (cm): [*<frxDBDataSet1."Length_in"> * 2.54*]

Variablen

Im nachfolgenden Absatz ist erklärt, wie Sie die Namen für Datenbank-Felder neu bestimmen können (und nicht nur für diese). Man kann einen Datenbank-Feldnamen sowie jeden Ausdruck mit Variablen vergleichen. Um mit Variablen zu arbeiten (Aufruf des Variablen-Editors), wählen Sie im Menü „Report“ den Dialog „Variablen...“ aus und klicken auf „Variablen“ in der linken Liste des Fensters.

Die Liste der Variablen im FastReport hat eine Zwei-Ebenen-Struktur. Die erste Ebene enthält Kategorien und die zweite enthält die Variablen an sich. Die Kategorisierung von Variablen ist für den Fall erschaffen worden, dass die Liste der Variablen zu lang ist. Eine Liste muss mindestens eine Kategorie enthalten. Das bedeutet, dass eine Variable nicht in eine höhere Ebene platziert werden kann. Außerdem sind Kategorien nur für die Klassifikation logischer Variablen, demnach sind sie nicht im Report enthalten. Wenn Sie einen Namen für eine Variable setzen wollen, vergessen Sie nicht, dass dieser einzigartig sein muss; es ist unmöglich zwei Variablen mit gleichem Namen in verschiedenen Kategorien zu schaffen.

Lassen Sie uns den Gebrauch von Variablen am folgenden Beispiel illustrieren. Nehmen Sie an, dass Sie zwei Datenquellen haben:
die erste Datenquelle ist „frxDBDataSet1“ mit dem Datenfeldern „CustNo“ und „Name“ und die zweite ist „frxDBDataSet2“ mit dem „OrderNo“- und dem „Date“-Feld.

Wir können die folgende Liste von Variablen mit den Feldern vergleichen:

Clients

Client number

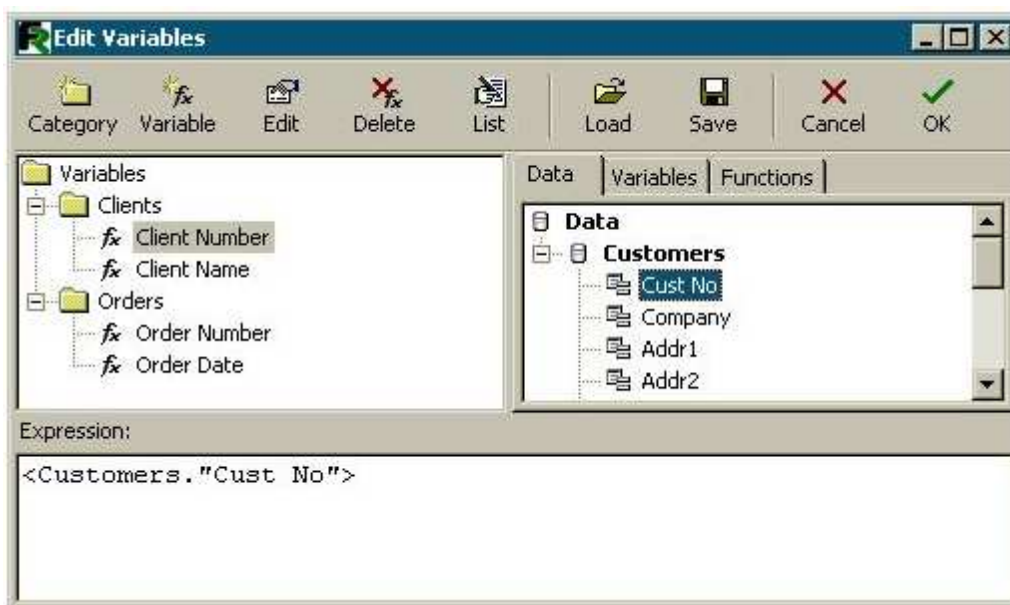
Client name

Orders

Order number

Order date

Die Bezeichnung „Clients“ und „Order“ sind zwei Kategorien. Öffnen Sie den Variablen-Editor und erstellen diese Struktur mit Hilfe den Schaltflächen „Kategorie“, „Variable“ und „Bearbeiten“. Um die Variable mit einem Datenbank-Feld zu verbinden (d.h. die Variable erhält den Wert des gewählten Datenfeldes - sie wird mit dem Feldinhalt initialisiert), wählen Sie sich eine Variable aus und machen auf dem erforderlichen Datenbank-Feld im rechten Teil des Fensters einen Doppelklick. Die Verbindung zum Datenbank-Feld wird im linken unteren Teil des Fensters angezeigt. Ein Ausdruck im linken unteren Teil des Fensters würde der Wert der Variable sein. Wenn es notwendig wäre, könnten Sie es auch manuell schreiben. Kategorien sollten nicht mit Datenfeldern initialisiert werden.



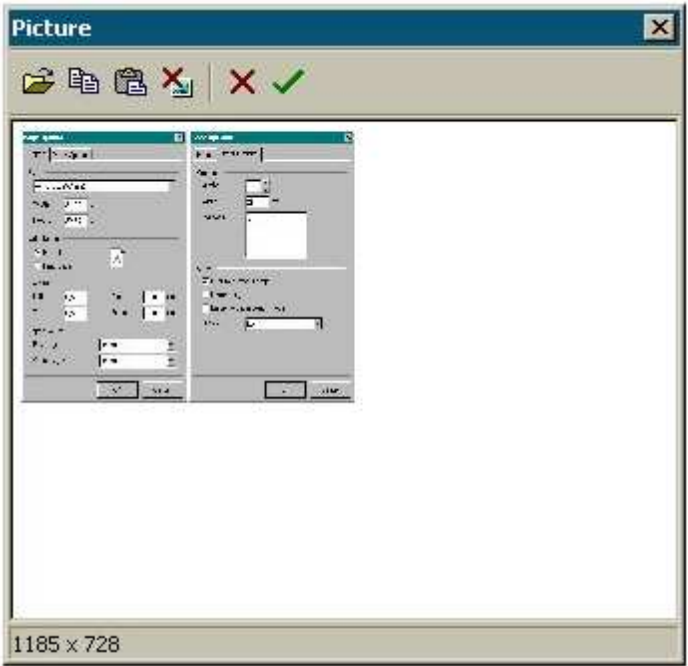
Nachdem die Liste der Variablen erstellt wurde, schließen Sie den Variablen-Editor. Jetzt müssen Sie die Variablen in den Report einfügen. Im Gegensatz zum Einsetzen von Datenbank-Feldern gibt es hier weniger Varianten. Sie können entweder eine Variable manuell in das „Text“-Objekt einfügen, indem Sie den Text „[,„Kundenzahl“] eingeben oder eine Variable aus der Variablen-Registerseite des Ausdrucks-Editors zum erforderlichen Objekt im Report ziehen:



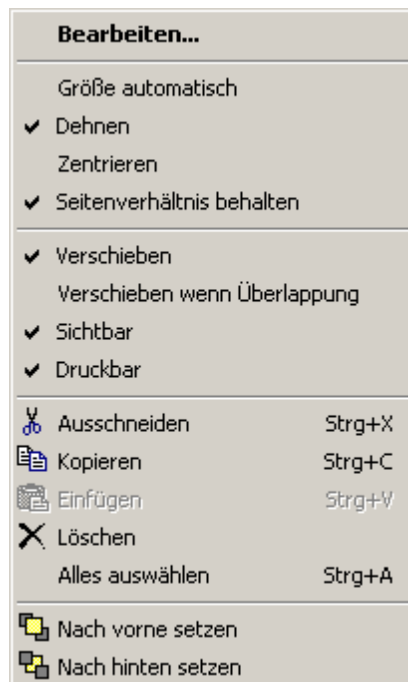
„Grafik“-Objekt

Das folgende zu untersuchende Objekt ist das „Grafik“-Objekt. Es wird ebenfalls sehr häufig in Reports verwendet. Mit Hilfe dieses Objekts können Sie ein Logo Ihres Unternehmens, ein Foto ihres Angestellten oder jegliche andere graphische Informationen einfügen. Das „Grafik“-Objekt kann Grafiken im „BMP“, „JPEG“, „ICO“, „WMF“ und „EMF“-Format anzeigen.

Lassen Sie uns die Fähigkeiten dieses Objekts ausprobieren. Erstellen Sie einen leeren Report und platzieren das „Grafik“-Objekt auf die Reportseite. Sie können ein Bild aus einer Datei laden oder ein bereits bestehendes Bild im Objekt-Editor löschen (wenn der Objekt-Editor nicht automatisch öffnet, dann klicken sie doppelt auf das Objekt). Laden Sie ein gewünschtes Bild hoch und klicken Sie „OK“.



Es gibt die folgenden Optionen im Objekt-Kontext-Menü (die Häkchen bedeuten die Voreinstellung):

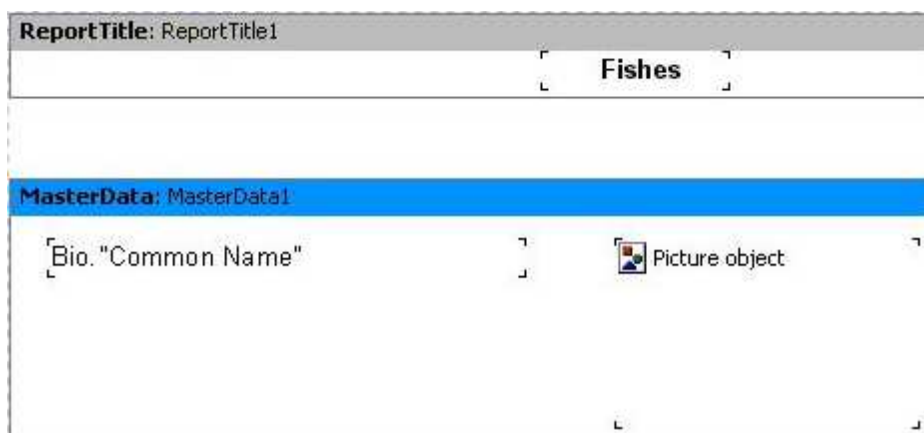


Die „Dehnen“- Option ist voreingestellt. Diese Option dehnt die Abbildung innerhalb eines Gegenstandes auf die Größe des Objekts aus. Ändern Sie die Größe des Objekts mit Hilfe der Maus, dann sehen Sie, dass die Größe der Abbildung immer im Verhältnis zur Größe des Objekts steht. Ist diese Option deaktiviert, wird die Abbildung in ihrer ursprünglichen Größe angezeigt.

Dieses Verhalten unterscheidet sich von der „Größe automatisch“-Option, weil die Objektmaße nicht entsprechend der Größe der Abbildung eingestellt werden. Dies bedeutet, dass der Gegenstand größer oder kleiner als die Abbildung sein kann.

Die „Zentrieren“-Option ermöglicht eine Ausrichtung einer Abbildung innerhalb des Gegenstandes.

Die „Seitenverhältnis behalten“-Option ist ebenfalls voreingestellt und besitzt eine sehr nützliche Eigenschaft: sie verhindert eine Verzerrung der Abbildungsmaße bei der Änderung der Objektgröße. Diese Option ist nur zusammen mit dem „Dehnen“-Kommando aktiv. Wenn nun jegliche Objektmaße geändert werden, bleibt ein gezogener Kreis. Ein Kreis, ohne zu einem Oval zu werden. Die gedehnte Abbildung nimmt dabei nicht den gesamten inneren Raum des Objekts ein sondern nur den Teil davon, der gebraucht wird, um das Bild im richtigen Höhen- / Seitenverhältnis anzuzeigen. Ist diese Option ausgeschaltet, nimmt das Bild die gesamte Größe des Objekts ein. Falls die Größe des Objekts nicht den Originalmaßen der Abbildung entspricht, wird das Bild verzerrt.



Report mit Bildern

Das „Grafik“-Objekt, sowie viele Objekte im FastReport, kann Daten aus Datenbanken anzeigen. Die Verbindung eines Objekts zu einem gewünschten Datenbank-Feld wird mit Hilfe der „DataSet“- und „DataField“-Eigenschaft im Objekt-Inspektor erreicht. Im Gegensatz zum „Text“-Objekt ist das hier die einzige Möglichkeit, um das Objekt mit Daten zu verbinden.

Lassen Sie uns das oben erwähnte in einem Report demonstrieren, der Bilder von Fischen und deren Namen enthält. Um das durchzuführen, werden Sie wieder die „DBDEMOS“ Demonstrations-Datenbank brauchen.




Erstellen Sie einen neuen Report. Verbinden Sie eine Tabelle mit dem Report im Menü „Report“, „Daten...“.

Erstellen Sie ein Report-Formular. Setzen Sie das „Text“-Objekt mit dem „Fisch“-Text auf das Band „Report Title“- . Verbinden Sie das Daten-Band mit der Datenquelle (Doppelklick auf das Band und dann „Bio“ aus der Liste auswählen). Erhöhen Sie das Band um 3cm, sodass das Bild Platz findet. Setzen Sie nun das „Text“-Objekt in das Band und verbinden es mit dem Feld „CommonName“, indem Sie die oben beschriebenen Methoden verwenden. Danach setzen Sie das „Picture“-Objekt daneben und verbinden es mit dem „Grafik“-Feld. Um dies durchzuführen stellen Sie die Eigenschaften im Objekt-Inspektor ein:

DataSet = Bio
DataField = 'Graphic'

Um Platz für die Bilder zu haben, strecken Sie das Objekt zu 4x2.5 cm.

Das ist alles. Der Report ist abgeschlossen (s.Abb. Unten):



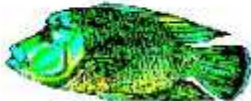
Fishes	
Clown Triggerfish	
Red Emperor	
Giant Maori Wrasse	

Die mehrzeilige Textanzeige

Gehen wir zum vorhergehenden Beispiel mit den Fischen zurück. In der „Biolife.db“-Tabelle gibt es ein „Notes“-Feld, welches eine detaillierte Beschreibung jedes Fisches enthält. Verbessern Sie unseren Report, indem Sie dieses Feld hinzufügen.

Auf den ersten Blick scheint alles einfach zu sein: fügen Sie das „Text“-Objekt zu dem Daten-Band hinzu, verbinden es mit dem „Notes“-Feld und stellen die Größe des Objekts auf (8x2.5 cm) ein.

Beim Starten des Reports werden Sie sehen, dass Sie nicht exakt das erhalten, was Sie erwartet hatten:

Fishes		
Clown Triggerfish	Also known as the big spotted triggerfish. Inhabits outer reef areas and feeds upon crustaceans and mollusks by crushing them with powerful teeth. They are voracious eaters, and divers report seeing the clown triggerfish devour beds of pearl <i>nuculans</i> .	
Red Emperor	Called seaperch in Australia. Inhabits the areas around lagoon coral reefs and sandy bottoms.	
Giant Maori Wrasse	The red emperor is a valuable food fish and considered a great sporting fish that fights with fish when hooked. The flesh of an old fish is just This is the largest of all the wrasse. It is found in dense reef areas, feeding on a wide variety of mollusks, fishes, sea urchins, crustaceans, and other invertebrates. In spite of its immense size, divers find it a very wary fish.	

Indes führt FastReport eben genau das aus, worum es gebeten wird. Das „Notes“-Feld enthält einen mehrzeiligen Text, dessen Größe sich ändern kann. Zur selben Zeit, hat das „Text“-Objekt, welches die Informationen vom Feld anzeigt, eine fixierte Größe. Und darum haben einige Zeilen keinen Platz im Objekt und werden einfach abgeschnitten. Was machen Sie in solch einer Situation?

Natürlich kann entweder die Objekt-Größe verstellt werden oder die Schriftgröße wird reduziert. Je nachdem führt dies zur unökonomischen Benutzung von Platz auf der Seite, aufgrund dessen, dass einige Fische lange Beschreibungen haben und andere kurze. Im FastReport gibt es Mittel, die helfen, dieses Problem zu lösen.

Dies betrifft die Fähigkeit des Bandes sich automatisch an die Höhe anzupassen, um ausreichend Platz für die beinhalteten Objekte zu finden. Um dies durchzuführen, müssen Sie nur die „Stretch“-Eigenschaft aktivieren. Dennoch ist das nicht alles, denn ein Objekt mit einer längeren Beschreibung sollte in der Lage sein, sich selbst zu strecken. Das „Text“-Objekt ist in der Lage, dies durchzuführen.

Das Objekt kann automatisch Höhe und Breite einstellen, um Platz für den gesamten Text zu finden, der im Datenfeld enthalten ist. Sie können die „AutoWidth“ und „StretchMode“-Eigenschaft nutzen, um dies zu schaffen. Die „AutoWidth“-Eigenschaft wählt die Objekt-Breite so aus, dass alle Zeilen Platz finden, ohne dass Wörter abgetrennt werden. Diese Methode ist günstig, wenn der Text aus lediglich einer Zeile besteht. Die „StretchMode“-Eigenschaft wählt die Objekt-Höhe so aus, dass der ganze Text Platz findet.. Die Objekt-Breite wird dabei nicht verändert. Diese Eigenschaft führt das Auflisten durch und Sie können eine der folgenden Modi im Objekt-Inspektor auswählen:



smDontStretch – streckt ein Objekt nicht (Voreinstellung);

smActualHeight – streckt ein Objekt um Platz für den gesamten Text zu finden;

smMaxHeight – streckt ein Objekt so, dass sein unterer Rand mit dem unteren Rand des Bandes zusammenfällt (auf dem das Objekt platziert ist). Wir werden dieses Beispiel etwas später ausprobieren.

Nun sind wir erst einmal an der „StretchMode“-Eigenschaft im „Text“-Objekt interessiert. Schalten Sie es im Objekt-Kontext-Menü ein oder setzen die Eigenschaft „StretchMode=smActualHeight“.

Aktivieren Sie außerdem die „Stretch“-Eigenschaft des Bandes. Starten Sie den Report und versichern Sie sich, dass alles in der richtigen Weise arbeitet.

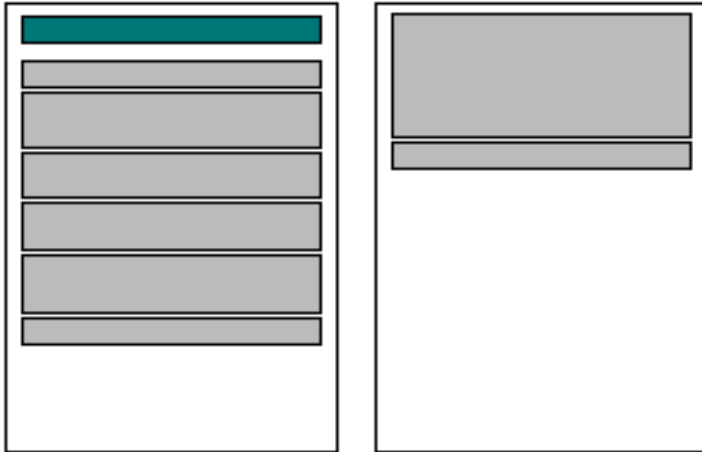
Fishes		
Clown Triggerfish	<p>Also known as the big spotted triggerfish. Inhabits outer reef areas and feeds upon crustaceans and mollusks by crushing them with powerful teeth. They are voracious eaters, and divers report seeing the clown triggerfish devour beds of pearl oysters.</p> <p>Do not eat this fish. According to an 1878 account, "the poisonous flesh acts primarily upon the nervous tissue of the stomach, occasioning violent spasms of that organ, and shortly afterwards all the muscles of the body. The frame becomes rocked with spasms, the tongue thickened, the eye fixed, the breathing laborious, and the patient expires in a paroxysm of extreme suffering."</p> <p>Not edible.</p>	
Red Emperor	<p>Range is Indo-Pacific and East Africa to Somoa. Called seaperch in Australia. Inhabits the areas around lagoon coral reefs and sandy bottoms.</p> <p>The red emperor is a valuable food fish and considered a great sporting fish that fights with fury when hooked. The flesh of an old fish is just as tender to eat as that of the very young.</p>	

Wie Sie beim Erstellen eines Reports sehen können, füllt FastReport Objekte mit Daten, streckt sie mit der eingestellten „Stretch“-Option und bildet dann die Band-Höhe, um für alle Objekte ausreichend Platz zu schaffen. Wenn die „Stretch“-Eigenschaft des Bandes ausgeschaltet ist, werden die Höheneinstellungen nicht übernommen und das Band wird mit der Höhe angezeigt, wie sie im Designer gesetzt ist. Wenn Sie versuchen diese Eigenschaft zu deaktivieren, werden Sie bemerken, dass Objekte mit längeren Texten trotzdem gestreckt werden, obwohl die Bänder bleiben wie sie sind. Das führt zur Textüberlagerung, da jeder folgende Datensatz direkt nach dem vorhergehenden angezeigt wird.

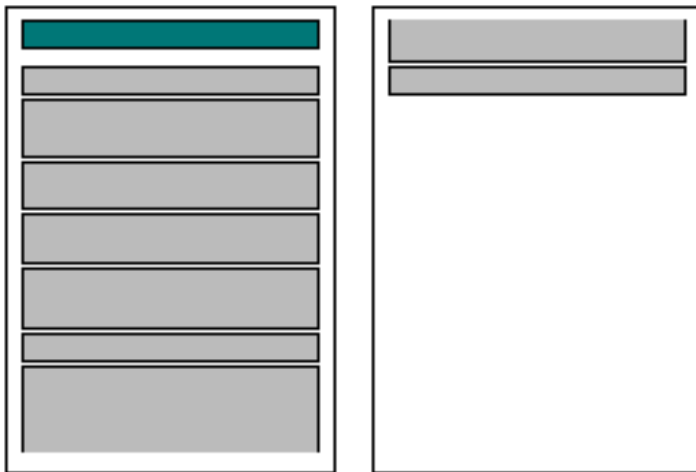
Datensplittung

Betrachten wir eine Besonderheit des Reports mit Fischen näher. Es gibt reichlich leeren Platz auf dem Ende der Seite. Wie kommt es dazu? Wenn ein Report erstellt wird, füllt der FastReport freien Platz mit Bändern. Nachdem jedes Band angezeigt wurde, bewegt sich die aktuelle Position nach unten. Wenn FastReport bemerkt, dass nicht genug Platz ist, um das nächste Band anzuzeigen, (seine Höhe größer als der freie Platz auf der Seite ist), wird eine neue Seite erstellt und die Anzeige des Band hier weitergeführt. Diese Operation wird solange durchgeführt, wie es Einträge im Datensatz gibt.

Unser Objekt beinhaltet ein Objekt mit großem Text und das, weil die Band-Höhe ebenfalls sehr groß ist. Darüber hinaus, wenn ein großes Band keinen Platz auf einer Seite hat, wird es auf die nächste Seite geleitet und der große ungenutzte Platz bleibt am Ende der vorherigen Seite. Dies wird in den folgenden Bildern dargestellt:



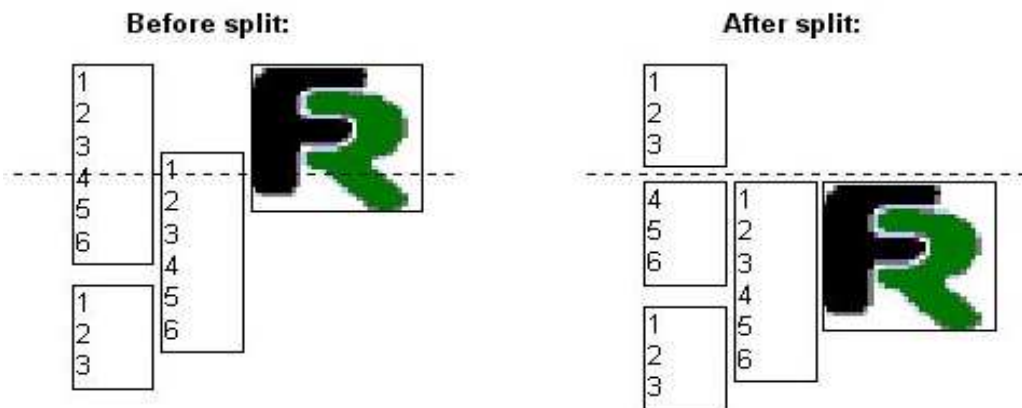
Um Papier rationeller zu nutzen, verwenden Sie ein FastReport-Merkmal, welches Absätze in den Band-Inhalt setzt. Das einzige, was Sie machen müssen, ist die „AllowSplit“-Eigenschaft aus dem Band zu aktivieren. Sie werden sehen, dass der leere Platz am Ende der Seiten weniger wird:



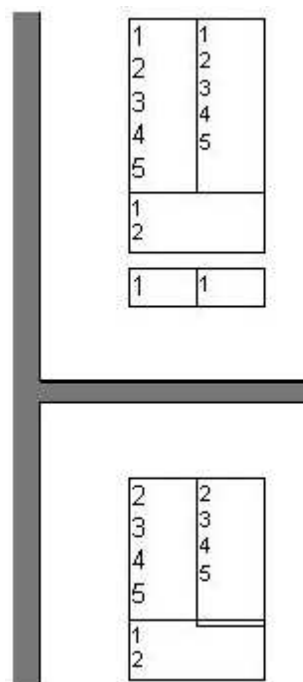
Wie funktioniert die Bandteilung? Es gibt einige Objekte im FastReport, die dieses Merkmal unterstützen. Das sind das „Text“- , „Line“- und „RichEdit“-Objekt. Diese können „gesplittet“ werden, wobei andere Objekte dies nicht können. Wenn es soweit ist, dass im FastReport die Seite geteilt werden muss, wird dies auf folgendem Weg getan:

- es werden die unteilbaren Objekte angezeigt, die Raum auf dem leeren Platz finden;
- zum Teil werden teilbare Objekte angezeigt (Text-Objekte werden so angezeigt, dass alle Zeilen im Objekt Platz finden);
- es wird eine neue Seite erstellt und die Textanzeige wird fortgesetzt;
- falls ein unteilbares Objekt keinen Platz im leeren Bereich findet, wird es auf die nächste Seite geleitet; zur selben Zeit werden alle Objekte die darunter platziert sind, aufgrund des Transfers der Objekte verschoben;
- der Prozess geht weiter bis alle Objekte des Bandes vollständig angezeigt werden.

Der Teilungs-Algorithmus wird klarer, wenn wir folgende Bilder betrachten:



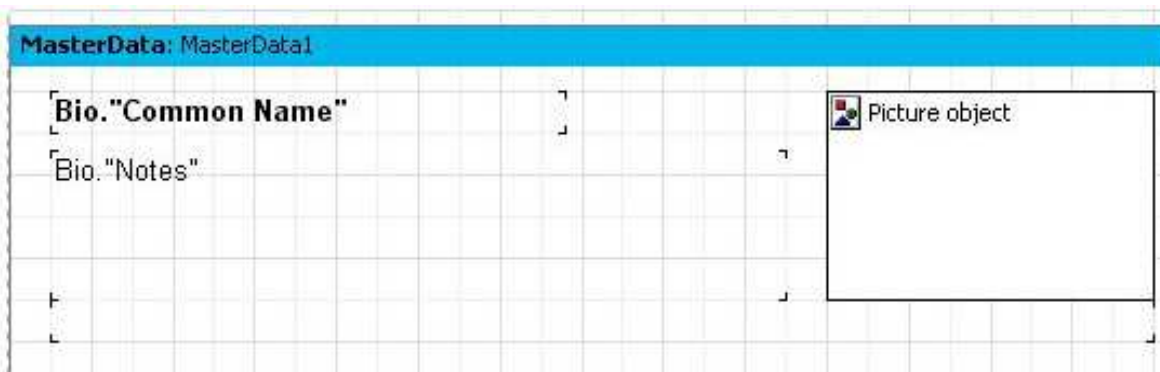
Es sollte bemerkt werden, dass der Teilungsalgorithmus keine 100%-ige Qualität für den erhaltenen Report bietet. Darum sollten Sie diese Option mit äußerster Vorsicht verwenden, falls die Objekte auf dem geteilten Band in einem sehr komplizierten Weg gruppiert sind und außerdem ihre Schriftgröße schwankt. Hier ein Beispiel von dem, was passieren kann:



Objekte von Text umhüllen

Bei der Ausgabe in Reports ist es ab und zu erforderlich, Objekte mit Text zu umhüllen (oft bei der Nutzung von Bildern). Demonstrieren wir dieses FastReport-Merkmal mit dem Beispiel der Fische.

Fügen Sie ein weiteres „Text“-Objekt zum Report hinzu und dann ordnen Sie das Objekt wie im folgenden Bild unterhalb von Text und Bild an:



Sie stellen die „StretchMode“-Eigenschaft des „Bio.Notes“-Objekt aus. Im Gegenzug dazu aktivieren Sie diese Eigenschaft für das Objekt unterhalb von Text und Bild. Um den Text des „Bio.Notes“-Objekt mit dem neuen Objekt zu verschmelzen, setzen Sie die „FlowTo“-Eigenschaft im „Bio.Notes“-Objekt. Diese Eigenschaft wird im Objekt-Inspektor gesetzt und ist eine der Objekte im gesamten Report. Der Name des Objekts, den sie zuletzt eingefügt haben, muss in dieser Liste ausgewählt werden. Das Ergebnis sieht aus, wie in den folgenden Bildern:

Clown Triggerfish

Also known as the big spotted triggerfish. Inhabits outer reef areas and feeds upon crustaceans and mollusks by crushing them with powerful teeth. They are voracious eaters, and divers report seeing the clown triggerfish devour beds of pearl oysters.



Do not eat this fish. According to an 1878 account, "the poisonous flesh acts primarily upon the nervous tissue of the stomach, occasioning violent spasms of that organ, and shortly afterwards all the muscles of the body. The frame becomes rocked with spasms, the tongue thickened, the eye fixed, the breathing laborious, and the patient expires in a paroxysm of extreme suffering."

Not edible.

Range is Indo-Pacific and East Africa to Samoa.

Red Emperor

Called seaperch in Australia. Inhabits the areas around lagoon coral reefs and sandy bottoms.



The red emperor is a valuable food fish and considered a great sporting fish that fights with fury when hooked. The flesh of an old fish is just as tender to eat as that of the very young.

Range is from the Indo-Pacific to East Africa.

Falls der Text keinen Platz im oberen Objekt findet, wird der Teil, der nicht auf die Seite passt, in das Objekt darunter geleitet. Wenn das Objekt um das Bild herum platziert wird, wird der Effekt der Umhüllung ausgeführt.

Vorsicht: Das „Text“-Objekt sollte in den Report eingefügt werden, bevor das verbundene eingefügt wird. Ansonsten funktioniert eventuell die Umhüllung nicht korrekt! Falls der Report nicht richtig funktioniert, wählen Sie das verbundene Objekt aus und dann führen Sie es nach vorn mit Hilfe des Befehls „Nach vorne setzen“ aus dem Kontext-Menü.

Daten in Form einer Tabelle anzeigen lassen

Manchmal ist es nötig, eine Datei in der Form einer Tabelle mit einem Rahmen anzeigen zu lassen.

Ein Beispiel eines solchen Reports ist jede Preisliste. Um so einen Report in FastReport zu erstellen, sollte der Benutzer lediglich die Rahmen-Eigenschaft für die Objekte, die im „Master Data“-Band platziert sind, einschalten.

Wir wollen mit dem Test-Report verschiedene Arten von Rahmen zeigen. Erstellen Sie einen Report der folgenden Art:

PageHeader: PageHeader1		
MasterData: MasterData1		
[Bio."Specie"]	[Bio."Common Name"]	[Bio."Length"]
ReportSummary: ReportSummary1		

Der erste und einfachste Typ einer Tabelle ist eine mit vollem Rahmen. Um dies durchzuführen, müssen Sie bei allen Objekten des „Master Data“-Bandes alle Rahmen-Linien erlauben:

90020	Clown Triggerfish	50
90030	Red Emperor	60
90050	Giant Maori Wrasse	229
90070	Blue Angelfish	30
90080	Lunartail Rockcod	80

Der nächste Rahmen-Typ soll nur vertikale oder horizontale Linien anzeigen. Das wird in genau derselben Weise durchgeführt. Nur horizontale oder vertikale Rahmen-Linien werden in den Objekten aktiviert (im Beispiel nur horizontale).

90020	Clown Triggerfish	50
90030	Red Emperor	60
90050	Giant Maori Wrasse	229
90070	Blue Angelfish	30
90080	Lunartail Rockcod	80

Letztendlich, um nur einen äußeren Rahmen zu schaffen, muss der Report ein wenig modifiziert werden:

PageHeader: PageHeader1		
MasterData: MasterData1		
[Bio."Specie"]	[Bio."Common Name"]	[Bio."Length"]
ReportSummary: ReportSummary1		

Wie Sie sehen können, haben Sie zwei „Text“-Objekte hinzugefügt und Rahmenlinien für das Objekt entlang der Kanten der Bänder eingestellt. Als Ergebnis wird in der Vorschau in etwa wie folgt aussehen:

90020	Clown Triggerfish	50
90030	Red Emperor	60
90050	Giant Maori Wrasse	229
90070	Blue Angelfish	30
90080	Lunartail Rockcod	80
90090	Firefish	38
90100	Ornate Butterflyfish	19
90110	Swell Shark	102
90120	Bat Ray	66
90130	California Moray	160
90140	Lingcod	160
90150	Cabezon	99
90160	Atlantic Spadefish	90
90170	Nurse Shark	400
90180	Spotted Eagle Ray	200
90190	Yellowtail Snapper	75
90200	Redband Parrotfish	28
90210	Great Barracuda	160
90220	French Grunt	30
90230	Dog Snapper	90
90240	Nassau Grouper	91
90250	Bluehead Wrasse	15
90260	Yellow Jack	90
90270	Redtail Surperch	40
90280	White Sea Bass	160
90290	Rock Greenling	60
90300	Senorita	25
90310	Surt Smelt	25

Alle vorherigen Beispiele enthalten Bänder mit fixierter Größe. Aber wie ist es möglich, Tabellen anzuzeigen, wenn sich das Band streckt? Lassen Sie uns das mit Hilfe des Beispiels unten erklären. Fügen Sie ein neues Feld (ein mehrzeiliger Text aus „Bio.Notes“) in unseren Report. Wie Sie bereits wissen, muss die „StretchMode“-Eigenschaft im Objekt und die „Stretched“-Eigenschaft im Band eingestellt sein, in dem das Objekt platziert ist. In diesem Falle ist die Höhe des Bandes abhängig vom Text im „Text“-Objekt. Demnach erhalten Sie einen Report der folgenden Art:

90020	Crown Triggerfish	50	Also known as the big spotted triggerfish. Inhabits outer reef areas and feeds upon crustaceans and mollusks by crushing them with powerful teeth. They are voracious eaters, and divers report seeing the crown triggerfish devour beds of pearl oysters. Do not eat this fish. According to an 1878 account, "the poisonous flesh acts primarily upon the nervous tissue of the stomach, occasioning violent spasms of that organ, and shortly afterwards all the muscles of the body. The frame becomes racked with spasms, the tongue thickened, the eye fixed, the breathing laborious, and the patient expires in a paroxysm of extreme suffering." Not edible. Range is Indo-Pacific and East Africa to Samoa.
90030	Red Emperor	60	Called seaperch in Australia. Inhabits the areas around lagoon coral reefs and sandy bottoms.

Er ist etwas anders als der, den Sie brauchen; es sollte so sein, dass sich der Rahmen der benachbarten Objekte ebenfalls streckt. FastReport ermöglicht es, dieses Problem einfach zu lösen. Um so einen Report zu erstellen, reicht es, die Eigenschaft `StretchMode=smMaxHeight` im Objektinspektor für alle Objekte einzustellen, die gestreckt werden sollen. So ermittelt FastReport zunächst einmal die maximale Höhe des Bandes, dann streckt es die Objekte mit der eingestellten Option. Auf Grund dessen, dass sich der Objekt-Rahmen zusammen mit dem Objekt streckt, verändert sich die Ausgabe:

90020	Crown Triggerfish	50	Also known as the big spotted triggerfish. Inhabits outer reef areas and feeds upon crustaceans and mollusks by crushing them with powerful teeth. They are voracious eaters, and divers report seeing the crown triggerfish devour beds of pearl oysters. Do not eat this fish. According to an 1878 account, "the poisonous flesh acts primarily upon the nervous tissue of the stomach, occasioning violent spasms of that organ, and shortly afterwards all the muscles of the body. The frame becomes racked with spasms, the tongue thickened, the eye fixed, the breathing laborious, and the patient expires in a paroxysm of extreme suffering." Not edible. Range is Indo-Pacific and East Africa to Samoa.
90030	Red Emperor	60	Called seaperch in Australia. Inhabits the areas around lagoon coral reefs and sandy bottoms.

Etiketten drucken

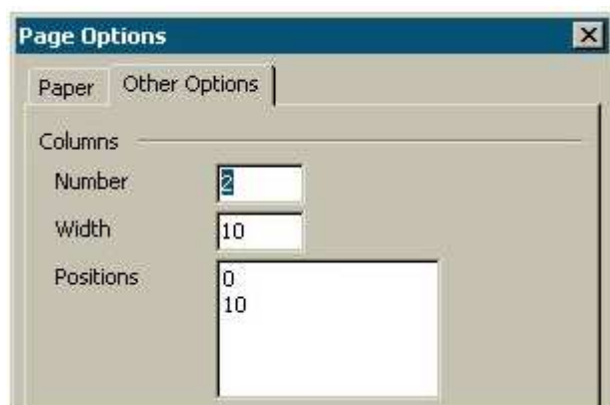
MasterData: Band3	
Number:	[Bio."Species No"]
Category:	[Bio."Category"]
Name:	[Bio."Common Name"]
Length, cm:	[Bio."Length (cm)"]

Im Gegensatz zu Reports als Tabellen, sind Daten bei Etiketten untereinander angelegt. Probieren wir ein Beispiel solch eines Reportes aus, der Daten von Fischen aufzeigt (siehe vorhergehendes Beispiel). Der Report wird in Form eines Etiketts dargestellt und besitzt folgende Struktur:

Wenn Sie den Report starten erhalten Sie folgendes Bild:

Number:	90020
Category:	Triggerfish
Name:	Clown Triggerfish
Length, cm:	50
Number:	90030
Category:	Snapper
Name:	Red Emperor
Length, cm:	60

Wie Sie sehen können, gibt es reichlich leeren Platz auf dem rechten Teil der Seite. Um die gesamte Seite zu füllen, kann die Anzahl der Spalten, in denen die Daten angezeigt werden, in den Report-Einstellungen gesetzt werden. Um dies durchzuführen, sollten Sie entweder auf dem freien Platz der Seite doppelklicken oder im Menü „Datei“ den Dialog „Seiteneinstellungen...“ aufrufen.



Auf der Registerseite „Andere Optionen“ kann man solche Spalten-Parameter wie die Spaltenzahl, Breite und Position einstellen. In unserem Fall sind 2 Spalten genug, die restlichen Parameter setzt FastReport automatisch.

Der Spalten-Rahmen ist im Designer als eine dünne vertikale Zeile dargestellt:



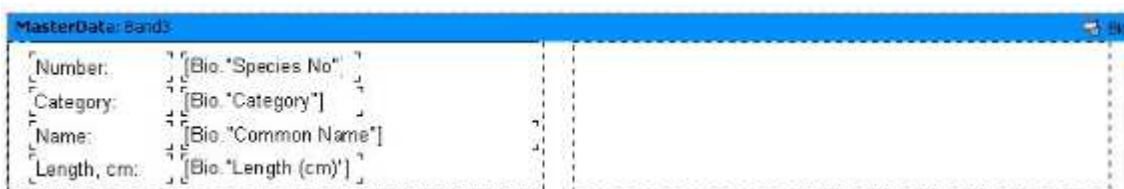
Auf folgende Weise werden die Daten ausgegeben. FastReport zeigt die Daten aus dem „Daten“-Band solange an, bis kein leerer Platz mehr auf der Seite ist. Danach entsteht eine neue Spalte auf derselben Seite (im Gegensatz zu einem einfachen Report, in dem in diesem Fall eine neue Seite erstellt werden würde). Das wird fortgeführt bis alle Spalten angezeigt werden. Danach erstellt FastReport erst eine neue Seite und fährt mit dem Darstellen der ersten Spalte fort.

Ihr Report mit zwei Spalten sieht folgendermaßen aus:

Number:	90020	Number:	90140
Category:	Triggerfish	Category:	Cod
Name:	Clown Triggerfish	Name:	Lingcod
Length, cm:	50	Length, cm:	150
Number:	90030	Number:	90150
Category:	Snapper	Category:	Sculpin
Name:	Red Emperor	Name:	Cabezon
Length, cm:	60	Length, cm:	99

Die „Columns“-Eigenschaft (in allen Bändern verfügbar) ist ein weiterer Weg, um die Anzahl der Spalten zu setzen. Es ermöglicht, die Anzahl der Spalten für ein einzelnes Band zu setzen und nicht für alle insgesamt (wie es im vorhergehenden Beispiel der Fall war). Damit ist die prinzipielle Darstellung von Daten nicht „von oben nach unten und von links nach rechts“, sondern „von links nach rechts und dann von oben nach unten“.

Lassen Sie uns die Spalten in der Seite deaktivieren (setzen sie die Spaltenzahl auf 1) und geben Sie „2“ in die „Columns“-Eigenschaft des Bandes ein. FastReport zeigt die Spaltenlinien als gestrichelte Linie an. Verändern Sie die Spaltenbreite, indem Sie die „ColumnWidth“-Eigenschaft eintragen:



Der auf diese Weise erstellte Report unterscheidet sich vom vorhergehenden nur in der Anordnung der Daten „von links nach rechts und dann von oben nach unten“.

„Child“-Bänder

Probieren Sie, wenn eine der Zeilen in einem „Etiketten“-Report eine unterschiedliche Größe hat. Um diese Situation zu simulieren verwenden Sie unser Beispiel und reduzieren die Breite des „Bio.Common Name“-Objekts auf 2.5cm und erlauben die „StretchMode“-Eigenschaft. Stellen Sie die „Stretch“-Eigenschaft auch für das „Master Data“-Band ein. Stellen Sie alle Rahmen-Linien für alle Objekte ein, so dass das Prinzip der Streck-Funktion deutlicher wird. Sie erhalten einen Report der folgenden Art:

Number:	90020
Category:	Triggerfish
Name:	Clown
Length, cm:	Triggerfish
	50
Number:	90030
Category:	Snapper
Name:	Red Emperor
Length, cm:	60

Sie können sehen, dass das Objekt im ersten Fall einen längeren Text besitzt, weil er auf zwei Zeilen gestreckt wurde. So rückt das Objekt, welches unter ihm platziert ist, runter. Dies geschieht, da alle Objekte die „Verschieben“-Eigenschaft voreingestellt haben (oder ShiftMode=smAlways im Objekt-Inspektor).

Solche Objekte rücken nach unten, wenn es streckbare Objekte über ihnen gibt (das „Text“-Objekt mit der eingestellten „Stretch“-Eigenschaft). Der Wert, um den das Objekt verrückt wird, hängt von der Streckung des Objekts über ihm ab.

Dennoch ist es in unserem Fall unakzeptabel, da Sie das Objekt mit dem Text „Length, cm:“ ebenso verschieben wollen. Um dies durchzuführen, gibt es einen bestimmten Band-Typ im FastReport, das „Child-Band“. Es ist mit dem Basis-Band verbunden (und wird nach ihm angezeigt). Erneuern Sie Ihren Report:

MasterData: MasterData1	
Number:	Bio."Species No"
Category:	Bio."Category"
Name:	Bio."Common"
Child: Child1	
Length, cm:	Bio."Length (cm)"

Um das „Master Data“-Band mit dem „Child“-Band zu verbinden, müssen Sie die „Child=Child1“-Eigenschaft im Objekt-Inspektor für das „Master Data“-Band einstellen. Nun wird bei jedem Ausdruck des Basis-Bandes auch das „Child“-Band gezeigt:

Number:	90020
Category:	Triggerfish
Name:	Clown Triggerfish
Length, cm:	50
Number:	90030
Category:	Snapper
Name:	Red Emperor
Length, cm:	60

Wie Sie sehen können, ist alles genau dort, wo er sein soll.

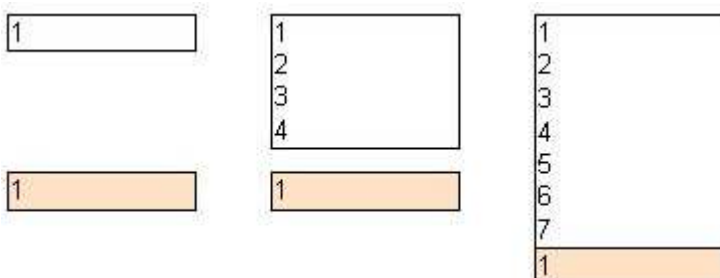
Um die Weiterleitung des „Child“-Bandes auf die nächste Seite zu verhindern (was hauptsächlich bedeutet, dass es vom Basis-Band getrennt wird), stellen Sie die „KeepChild“-Eigenschaft für das „Master Data“-Band („KeepChild“ im Objekt-Inspektor).

Verschobene Objekte

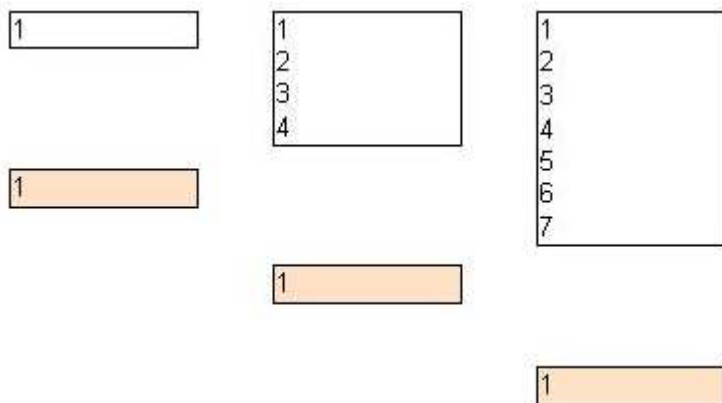
Sie haben bereits gesehen, wie die „ShiftMode“-Eigenschaft funktioniert. Probieren Sie nun die nächste Art der Verschiebung:

„Verschieben bei Überlappung“.

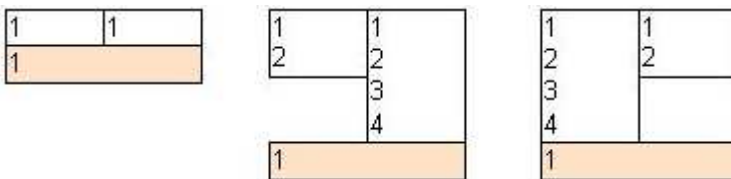
Die „ShiftMode=smWhenOverlapped“-Eigenschaft im Objekt-Inspektor korrespondiert mit dieser Art. Demnach wird die Verschiebung von Objekten in diesem Fall so ausgeführt, dass das untere Objekt verschoben wird, wenn das obere Objekt bei der Streckung das untere überlappt. Drei Fälle sind in den Bildern unten dargestellt. Wie Sie sehen, bewegt sich das untere Objekt mit der eingeschalteten „smWhenOverlapped“-Eigenschaft nur im letzten Fall, d.h. wenn es viel Text im oberen Objekt gibt und es das untere überlappt.



Falls die „ShiftMode“-Eigenschaft auf „smAlways“ eingestellt ist, verrückt das untere Objekt sowieso:



In einigen Fällen ist eine ziemlich komplizierte Logik des Objekt-Designs realisierbar, besonders wenn ein Objekt auf mehrere andere Objekte zur gleichen Zeit übergreift. So im folgenden Beispiel, beide der höheren Objekte beinhalten streckbare Texte und das untere hat die „smDontShift“-Eigenschaft aktiviert. Das untere Objekt wird immer nah zu dem Objekt angezeigt, welches mehr Text hat, ungeachtet der Text-Größe im oberen Objekt:



In diesem Beispiel wird das untere Objekt doppelt verrückt, wenn die „smDontShift“-Option eingeschaltet ist, sobald es unter zwei Objekten platziert wird und so entsteht eine unnötige Lücke.

Reports mit zwei Datenebenen (Master-Detail)

Bis jetzt haben Sie nur Reports erstellt, die lediglich ein Daten-Band zeigten. Das ermöglichte das Anzeigen von Daten aus einer Datenbank-Tabelle. FastReport ermöglicht es Reports zu erstellen, die bis zu sechs Datenebenen enthalten (man kann auch mehr Ebenen schaffen per „Unterreport“-Objekt; dieses Merkmal werden Sie später genauer untersuchen). Normalerweise werden Reports mit vielen Datenebenen selten erstellt. In der Regel gelten 1-3 Ebenen als Limit.

Probieren Sie den Zwei-Ebenen Report. Er wird Daten aus der „DBDEMOS“ Tabelle beinhalten: „Customer.db“ und „Orders.db“. Die erste Tabelle ist die Liste der Kunden, die zweite ist die Liste der Bestellungen, die von den Kunden getätigt wurden. Die Tabellen enthalten folgende Daten:

Customer:

CustNo	Company
1221	Kauai Dive Shoppe
1231	Unisco
1351	Sight Diver
....	

Orders:

OrderNo	CustNo	SaleDate
1003	1351	12.04.1988
1023	1221	01.07.1988
1052	1351	06.01.1989
1055	1351	04.02.1989
1060	1231	28.02.1989
1123	1221	24.08.1993
....		

Wie Sie sehen können, beinhaltet die zweite Tabelle eine Liste aller Bestellungen, die von allen Gesellschaften getätigt wurden. Um die Liste der Bestellungen der einzelnen Gesellschaften zu erhalten, sollten die Einträge, in denen die „CustNo“-Felder die Nummer der ausgewählten Gesellschaften enthalten, in der Tabelle ausgewählt werden. Der Report, der auf solchen Daten basiert, sieht folgendermaßen aus:

1221	Kauai Dive Shoppe
1023	01.07.1988
1123	24.08.1993
1231	Unisco

Erstellen Sie einen neuen Report.

1060	28.02.1989
1351	Sight Diver
1003	12.04.1988
1052	06.01.1989
1055	04.02.1989

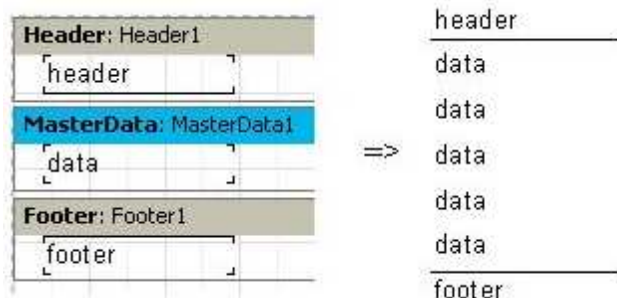
Im Designer verbinden Sie unsere Datenquellen im Menü „Report“, „Daten...“. Setzen Sie das „Master Data“- und das „Detail Data“-Band auf die Seite:

MasterData: MasterData1		Customers
[Customers."CustNo"]	[Customers."Company"]	
DetailData: DetailData1		Orders
[Orders."CustNo"]	[Orders."OrderNo"]	[Orders."SaleDate"]

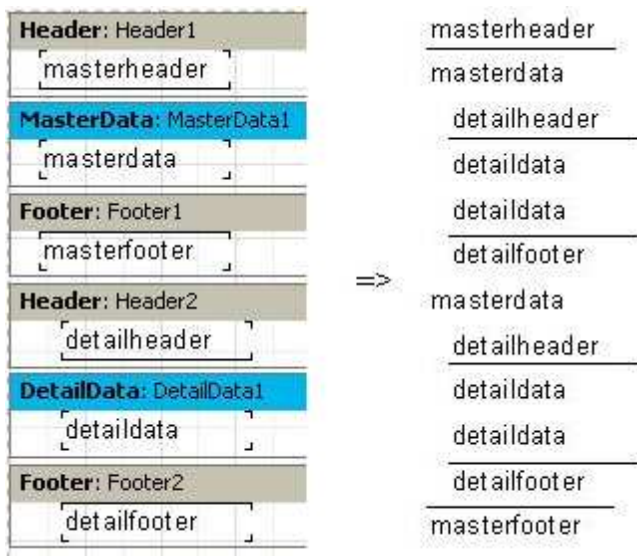
Beachten Sie, dass das „Master Data“-Band über das „Detail Data“-Band angebracht werden muss! Wenn es darunter gesetzt wird, informiert Sie FastReport über eine Fehlermeldung beim Starten des Programms.

Kopf- und Fußzeilen Daten-Bänder (Header und Footer)

Jedes Daten-Band kann eine Kopf-(Header) und eine Fußzeile(Footer) haben. Die Kopfzeile wird vor dem Ausdruck des Daten-Bandes angezeigt, die Fußzeile wird angezeigt nachdem alle Datenzeilen gedruckt sind. Hier am Beispiel eines einfachen Reports, wie Kopf- und Fußzeilen arbeiten:



Hier ein etwas komplizierteres Beispiel, das „Master-Detail“-Bänder verwendet:



Wie Sie sehen können, wird die Kopfzeile (Header) vor allen Datenzeilen ausgedruckt. So wird die „Master Data“-Kopfzeile einmal am Beginn des Reports ausgedruckt, „Detail Data“-Kopfzeilen werden vor jeder Gruppe von Detail-Bändern ausgedruckt, die zu den „Master Data“-Daten gehören. Die Fußzeile wird nach allen Datenzeilen ausgedruckt.

Bei der Benutzung der „FooterAfterEach“-Eigenschaft des „Master Data“-Bands überschreiben Sie eventuell dieses Verhalten. Setzen Sie diese Eigenschaft auf True (Sie verwenden vielleicht auch das Kontext-Menü des „Master Data“-Bands, den „Kopfzeile nach jeder Reihe“-Eintrag), wird jede Fußzeile nach jeder Datenzeile ausgedruckt. Es könnte nützlich sein beim Gestalten eines Master-Detail-Reports. Das vorangegangene Beispiel mit der „FooterAfterEach“-Eigenschaft des „Master Data“-Bands auf True gesetzt sieht so aus:

```

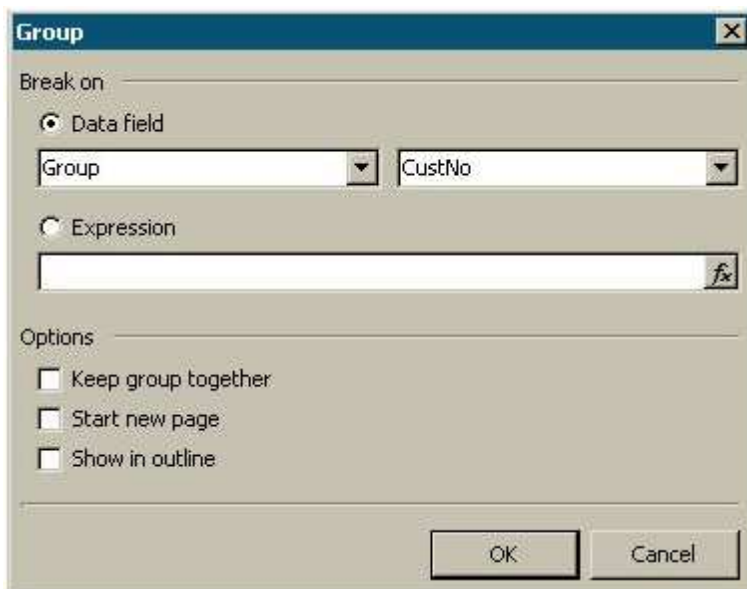
masterheader
masterdata
  detailheader
  detaildata
  detaildata
  detailfooter
masterfooter
masterdata
  detailheader
  detaildata
  detaildata
  detailfooter
masterfooter
  
```

Report mit Gruppen

In dem Beispiel oben erstellten Sie einen Zwei-Ebenen-Report auf der Basis der Daten aus zwei Tabellen. FastReport erlaubt analoge Reports auf der Grundlage von Datensätzen, die speziell aufbereitet sind.

Wie kann ein Mehrebenen-Report auf der Basis dieser Daten erstellt werden? Im FastReport gibt es ein spezielles „Gruppen Kopf“-Band (Group Header). Eine spezielle Bedingung für das Band muss angegeben sein (Datenbank-Feldwert oder ein Ausdruck). Das Band wird angezeigt, sobald sich der Feldwert ändert. Das folgende Beispiel verdeutlicht dieses.

Erstellen Sie einen neuen Report und verbinden unsere Datenquellen mit dem Report. Danach fügen Sie das „Gruppen Kopf“- und das „Master Data“-Band zum Report hinzu. Setzen Sie eine Bedingung (in diesem Fall ist es das Datenfeld „Group.CustNo“) im „Gruppen Kopf“-Band-Editor:



Verbinden Sie das „MasterData“-Band mit der „Group“-Datenquelle und platzieren die Objekte auf folgende Weise (beachten Sie, dass das „Gruppen Kopf“-Band vor dem „Master Data“-Band gesetzt werden muss):

GroupHeader: GroupHeader1		Group."CustNo"	
[Group."CustNo"]	[Group."Company"]		
MasterData: MasterData1		Group	
[Group."OrderNo"]	[Group."SaleDate"]		

Beim Starten erhalten Sie einen Ausdruck:

1221	Kauai Dive Shoppe
1269	16.12.94
1023	01.07.88
1176	26.07.94
1076	16.12.94
1123	24.08.93
1169	06.07.94
1231	Unisco
1173	16.07.94
1178	02.08.94

Wie Sie sehen, wird der „Gruppenkopf“ nur angezeigt, wenn das Feld, mit dem es verbunden ist, seinen Wert ändert. Sonst wird das „Master Data“-Band, welches die Daten für diese Gruppe enthält, angezeigt. Wenn Sie diesen Report mit dem Master-Detail-Report vergleichen, den wir vorher erstellt hatten, scheint es offensichtlich, dass die Bestellnummern nicht in der Reihenfolge der Bestellung sortiert wurden.

Ebenso können Reports mit verschachtelten Gruppen erstellt werden. Gleichzeitig ist die Anzahl der Unterteilungen (Ebenen) in diesem Report unbegrenzt. So haben die Reports mit Gruppen einige Vorteile gegenüber den Master-Detail-Reports:

- der gesamte Report benötigt lediglich eine Tabelle (Query);
- die Anzahl der Daten-Ebenen ist unbegrenzt;
- die zusätzliche Eigenschaft der Sortierung;
- eine rationellere Benutzung der Datenbank-Ressourcen (die Query gibt nur die Daten zurück, die ausgedruckt werden sollen, ohne dass die Daten gefiltert werden müssen).

Der einzige Nachteil ist, dass man die Abfragen in der SQL-Sprache schreiben muss. Jedoch, das Wissen der SQL-Grundlage ist obligatorisch für einen Anwender, der mit Datenbanken arbeitet.

Andere Eigenschaften von Gruppen

Lenken Sie Ihre Aufmerksamkeit darauf, wie die Gruppen auf die nächste Seite übertragen werden:

1380	Blue Jack Aqua Center
1006	06.11.94
1079	03.05.89
1106	23.09.92
1153	16.04.94
1253	26.11.94
1384	VIP Divers Club
1007	01.05.88
1027	07.07.88

Wenn Sie sich den Ausdruck anschauen, scheint es nicht klar zu sein, zu welchem Kunden sich die Liste der Bestellungen ganz oben auf der zweiten Seite bezieht. FastReport ermöglicht das Wiederholen von „Gruppen Kopf“-Bändern (welche in unserem Fall Informationen über die Kunden enthalten) auf der nächsten Seite. Um dies durchzuführen, muss der „Auf neuer Seite wiederholen“-Eintrag im Kontext-Menü (oder die „ReprintOnNewPage“-Eigenschaft im Objekt-Inspektor) im „Gruppen Kopf“-Band aktiviert werden. Dann sieht der Report folgendermaßen aus:

1380	Blue Jack Aqua Center
1006	06.11.94
1380	Blue Jack Aqua Center
1079	03.05.89
1106	23.09.92
1153	16.04.94
1253	26.11.94
1384	VIP Divers Club
1007	01.05.88

Es gibt einen anderen Weg, der das Auseinanderreißen der Gruppen verhindert. Dazu muss die Eigenschaft „Zusammenhalten“ im „Gruppen Kopf“-Band (oder „KeepTogether“ im Objekt-Inspektor) eingestellt sein. Demnach wird die gesamte Gruppe, falls sie nicht genügend Platz auf der Seite hat, auf eine neue Seite gesetzt. In unserem Beispiel sieht das folgendermaßen aus:

1356	Tom Sawyer Diving Centre
1005	20.04.88
1059	24.02.89
1072	11.04.89
1080	05.05.89
1105	21.07.92
1180	06.08.94
1266	15.12.94
1280	26.12.94
1305	20.01.95
1380	Blue Jack Aqua Center
1006	06.11.94
1079	03.05.89

Demzufolge entsteht viel leerer Platz auf einigen Seiten, aber die Gruppen werden als ein Ganzes auf einer Seite angezeigt.

Als Schlussfolgerung, ermöglicht es die Eigenschaft „StartNewPage“ im „Gruppen Kopf“-Band, jede Gruppe auf einer eigenen Seite zu zeigen. Das führt möglicherweise zu hohem Papierverbrauch, dennoch könnte es in einigen Fällen nützlich sein.

Zeilennummerierung

Nutzen Sie unser Beispiel, um zu testen, wie man Zeilen in Gruppen nummeriert. Um dies durchzuführen, fügen Sie ein „Systemtext“-Objekt mit der Variablen [Line] zu beiden Bändern hinzu (es geht einfacher, indem man von der Registerseite „Variablen“ des „Datenstruktur“ Fenster das Objekt „Line“ auf das Formular zieht).

GroupHeader: GroupHeader1		
[Line]	[Group."Cust"]	[Group."Company"]
MasterData: MasterData1		
[Line]	[Group."Orde"]	[Group."SaleDate"]

Beim Starten des Reports können wir sehen, dass beide Datenebenen ihre Nummerierung haben:

1	1221	Kauai Dive Shoppe
1	1023	01.07.88
2	1076	16.12.94
3	1123	24.08.93
4	1169	06.07.94
5	1176	26.07.94
6	1269	16.12.94
2	1231	Unisco
1	1060	28.02.89
2	1073	15.04.89
3	1102	06.06.92
4	1160	01.06.94

In einigen Reports brauchen Sie eventuell eine fortlaufende Nummerierung der zweiten Datenebene. Um dies durchzuführen, sollten Sie die „Line#“-Variable anstelle der „Line“-Variablen benutzen. Das Ergebnis wird wie folgt sein:

1	1221	Kauai Dive Shoppe
1	1023	01.07.88
2	1076	16.12.94
3	1123	24.08.93
4	1169	06.07.94
5	1176	26.07.94
6	1269	16.12.94
2	1231	Unisco
7	1060	28.02.89
8	1073	15.04.89
9	1102	06.06.92

Aggregat-Funktionen

In den meisten Fällen sollten Reports mit Gruppen auch einige Rechenergebnisse anzeigen (wie: „Gesamtsumme der Gruppe“ oder „Anzahl der Gruppen-Elemente“ etc.). Es gibt dafür die so genannten „Aggregat-Funktionen im FastReport, die für diesen Zweck vorhanden sind. Nachfolgend die Liste der Aggregat-Funktionen:

SUM	Gibt die Gesamtsumme von Ausdrücken wieder
MIN	Gibt den Minimalwert von Ausdrücken wieder
MAX	Gibt den Maximalwert von Ausdrücken wieder
AVG	Gibt den Durchschnittswert von Ausdrücken wieder
COUNT	Gibt die Anzahl von Datensätzen wieder

Die Syntax aller Aggregat-Funktionen (außer COUNT) ist die folgende (probieren Sie es mittels der „SUM“-Funktion aus):

SUM(Ausdruck, Band, Flags)

SUM(Ausdruck, Band)

SUM(Ausdruck)

Die Parameter bedeuten:

Ausdruck	– ein Datenbank-Feld oder ein Ausdruck
Band	– der Name des Bandes, mit dem die Rechenoperation durchgeführt wird
Flag	– ein Zahlen-Feld, das die folgenden Werte und ihre Kombinationen enthalten kann
	1 – berücksichtigt die unsichtbaren Bänder
	2 – sammelt den Wert von „Ausdruck“ über alle Daten (der Wert wird bei der nächsten Anzeige des Bandes nicht zurückgesetzt)

Wie Sie sehen, ist „Ausdruck“ der einzige obligatorische Parameter; der Rest kann übersprungen werden. Es wird trotzdem empfohlen, immer den Namen des Bandes anzugeben, um Fehlern vorzubeugen.

Die „COUNT“-Funktion hat folgende Syntax:

COUNT (Band, Flag)

COUNT (Band)

Die Parameterangabe ist gleich der, die oben beschreiben wurde.

Es gibt eine generelle Regel für die Funktionen:

Eine Funktion kann nur für ein Daten-Band angewendet werden und nur in dem dazugehörigen Fuß-Band angezeigt werden (die folgenden Bänder: Fuß, Seitenfuß, Gruppenfuß, Spaltenfuß und Reportfuß).

Wie arbeiten Aggregat-Funktionen? Probieren Sie unser Beispiel mit den Gruppen einfach aus. Fügen Sie neue Elemente zum Report hinzu:

GroupHeader: GroupHeader1		
[Group."Cust"]	[Group."Company"]	
MasterData: MasterData1		
[Group."Orde"]	[Group."SaleDate"]	[Group."ItemsTotal"]
GroupFooter: GroupFooter1		
[SUM(<Group."ItemsTotal">,MasterData1)]		

Das Feld „ItemTotal“ der Tabelle „Group“ auf dem „MasterData“-Band zeigt die Gesamtsumme der Bestellung in einem Datensatz an. Sie platzieren das „Text“-Objekt, welches die SUM-Funktion beinhaltet, in das „Gruppen Fuß“-Band. Es wird die Gesamtsumme aller Bestellungen anzeigen, die von den gegebenen Kunden getätigt wurden. Wenn Sie den Report starten, sehen Sie folgendes Ergebnis:

1221	Kauai Dive Shoppe	
1023	01.07.88	\$4 674,00
1076	16.12.94	\$17 781,00
1123	24.08.93	\$13 945,00
1169	06.07.94	\$9 471,95
1176	26.07.94	\$4 178,85
1269	16.12.94	\$1 400,00
		51450,8

Wie arbeiten Aggregat-Funktionen nun? Vor dem Erstellen eines Reports sucht FastReport den Inhalt des „Text“-Objekts nach Aggregat-Funktionen ab. Die gefundenen Funktionen werden zu den entsprechenden Daten-Bändern verankert (in unserem Beispiel die „SUM“-Funktion mit dem „MasterDaten1“-Band). Während der Erstellung eines Reports (wenn das Daten-Band angezeigt wird) wird der Wert der mit dem Band verbundenen Aggregat-Funktionen zusammengezählt. In unserem Falle werden die „ItemsTotal“-Feldwerte angesammelt. Nach der Ausgabe der Fußzeile, in der der Wert der Aggregat-Funktion angezeigt wird, wird der Funktionswert zurückgesetzt und der Kreislauf wiederholt sich bei der nächsten Gruppe.

Jetzt sollten Sie den Zweck des „Flags“-Parameters in Aggregat-Funktionen erfahren. In manchen Reports werden einige Daten-Bänder (oder auch alle) verborgen, dennoch müssen wir vielleicht trotzdem den Wert der Aggregat-Funktionen zusammenrechnen, der alle Daten-Bänder umfasst. Also, in unserem Beispiel kann die „Visible“-Eigenschaft des „Master Data“-Bands eingestellt werden; danach werden keine Datenzeilen mehr angezeigt.

Um auf dem verborgenen Daten-Band eine Gesamtsumme auszurechnen, fügen Sie dem Aufruf der SUM-Funktion einen dritten Parameter hinzu:

```
[SUM(<Group."ItemsTotal">,MasterData1,1)]
```

Es wird Ihnen folgender Report ausgegeben:

1221	Kauai Dive Shoppe	51450,8
1231	Unisco	85643,6
1351	Sight Diver	261575,8

Der „Flag=2“-Parameterwert erlaubt es, das Rücksetzen der ermittelten Funktionswerte direkt nachdem sie angezeigt wurden, zu vermeiden. Dies ermöglicht es, die so genannte „Gesamtsumme“ zu erhalten. Erneuern Sie den Aufruf der SUM-Funktion:

```
[SUM(<Group."ItemsTotal">,MasterData1,3)]
```

Der „3er“ Wert ist eine Kombination aus „1“ und „2“, was bedeutet, dass die unsichtbaren Bänder beachtet werden und ohne die Summenwerte zurückzusetzen. Infolge dessen haben Sie:

1221	Kauai Dive Shoppe	51450,8
1231	Unisco	137094,4
1351	Sight Diver	398670,2

Seiten- und Report-Summen

Ziemlich oft müssen Sie die Gesamtsumme der Seite oder des gesamten Reports anzeigen lassen. Dies kann genau so gut mit den Aggregat-Funktionen durchgeführt werden. Probieren Sie es mit der Hilfe unseres Beispiels aus.

GroupHeader: GroupHeader1		
[Group."Cust"]	[Group."Company"]	
MasterData: MasterData1		
[Group."Orde"]	[Group."SaleDate"]	[Group."ItemsTotal"]
GroupFooter: GroupFooter1		
[SUM(<Group."ItemsTotal">,MasterData1)]		
ReportSummary: ReportSummary1		
[Total: [SUM(<Group."ItemsTotal">,MasterData1)]]		
PageFooter: PageFooter1		
[Total this page: [SUM(<Group."ItemsTotal">,MasterData1)]]		

Wie Sie sehen können, wurde hier das „Report Summary“-Band und das „Page Footer“-Band mit dem „Text“-Objekt für die Summe hinzugefügt. Das ist alles, was Sie brauchen.

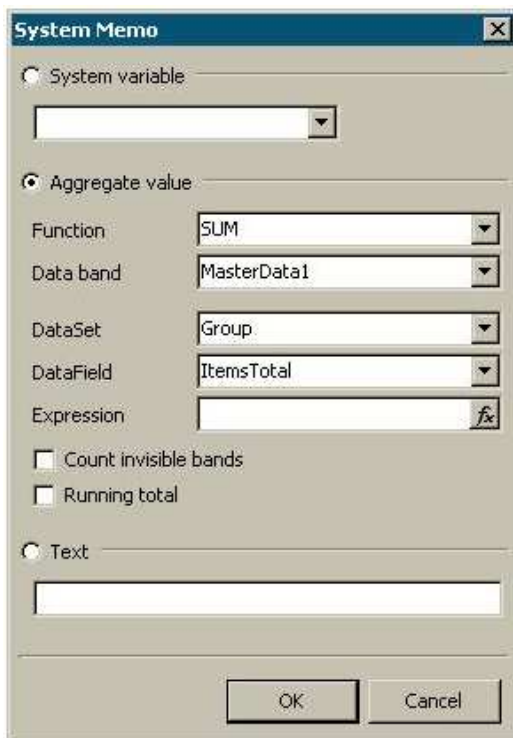
9841	Neptune's Trident Supply	
1045	16.10.88	\$ 787,80
1049	13.12.88	\$ 1 809,85
1145	17.01.94	\$ 4 229,80
1149	14.03.94	\$ 12 900,75
		19728,2
Total: 2922666,1		

Total this page: 101298,95		
----------------------------	--	--


Das Einfügen von Aggregat-Funktionen

Bis jetzt haben Sie die Aggregat-Funktionen manuell in das „Text“-Objekt eingefügt. Probieren Sie nun, ein paar günstigere Wege Aggregat-Funktionen einzufügen.

Zuerst können Sie das „Systemtext“-Objekt für die Ausgabe der Aggregat-Funktionen nutzen. Eigentlich ist es dasselbe „Text“-Objekt, aber eins mit einem speziellen Editor für ein einfacheres Einfügen von System-Variablen oder Aggregat-Funktionen.



Sie sollten Schritt für Schritt alle Werte für die drei Bereiche „Systemvariable“, „Aggregatwert“ und „Text“ ausprobieren. Andererseits können Sie auch das „Unsichtbare Bänder zählen“- und „Kumulative Summe“-Flag markieren.

Der zweite Weg ist das Benutzen des „Text“-Objekts und der  -Schaltfläche seines Editors. Danach erscheint ein zusätzliches Fenster, ähnlich dem oben beschriebenen. Durch Klicken der „OK“-Schaltfläche wird der Aufruf der Aggregat-Funktion in das „Text“-Objekt eingefügt.

Das Aggregat-Funktionen-Aufruf-Merkmal

In diesem Kapitel werden wir betrachten, wann eckige und spitze Klammern innerhalb von Ausdrücken in einem „Text“-Objekt genutzt werden. Erinnern Sie sich, dass alle Ausdrücke, die nicht standardmäßig in der Programmiersprache Pascal interpretiert wurden (die für die Berechnung Ausdrucks-Werten verwendet wird), durch spitze Klammern umschlossen sein sollten. Die Datenbank-Felder (über die Konstruktion *Table.name.“Fieldname“*), sowie die Variablen aus der Variablen-Liste sind in diese Gruppe mit eingeschlossen.

Es ist nötig, spitze Klammern zu benutzen, um die Aggregat-Funktionen in Ausdrücken auszurufen. Demnach ist die folgende Angabe nicht korrekt:

```
[SUM(<Group."ItemsTotal">,MasterData1) * 2]
```

und so ist es richtig:

```
[<SUM(<Group."ItemsTotal">,MasterData1)> * 2]
```

Erinnern Sie sich ebenfalls daran, dass FastReport es erlaubt, die spitzen Klammern auszulassen, falls der Ausdruck aus einem Element besteht, was bedeutet, dass beide Angaben

```
[<SUM(<Group."ItemsTotal">,MasterData1)>]  
[SUM(<Group."ItemsTotal">,MasterData1)]
```

identisch sind.

Werte formatieren

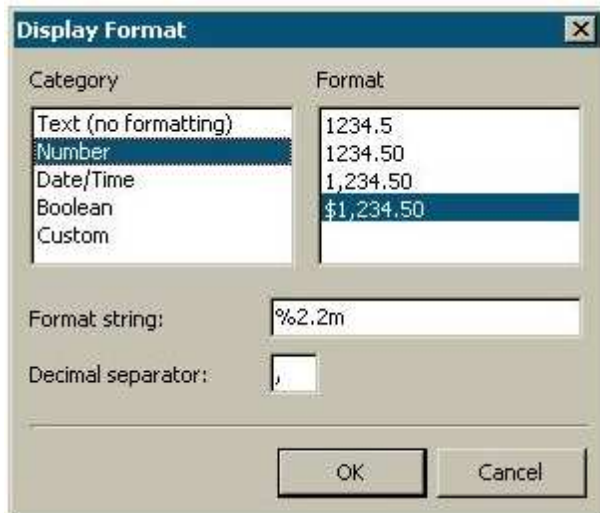
Lenken Sie Ihre Aufmerksamkeit auf eine Besonderheit des Gebrauchs von Aggregat-Funktionen:

Numerische Werte, die zurückgeben werden, sind nicht formatiert. Dies wird offensichtlich, wenn Sie sich auf das erste Beispiel mit der „SUM“-Funktion beziehen:

1176	26.07.94	\$4 178,85
1269	16.12.94	\$1 400,00
		51450,8

Das geschieht, weil in der Regel die Datenfelder einen formatierten Wert zurückgeben, der durch das „Text“-Objekt ohne Änderungen einfach angezeigt wird. Um eine Formatierung der Ergebnisse der „SUM“-Funktion zu bekommen, benutzen Sie die Formatierungswerkzeuge im FastReport.

Wählen Sie das Objekt mit der Summe und rufen Sie im Kontext-Menü den Eintrag „Anzeigeformat“ auf. Der Format-Editor kann ebenfalls durch die Benutzung der „Formatierung einfügen“-Schaltfläche im „Text-Objekt“-Editor oder mit der Eigenschaft „DisplayFormat“ im Objekt-Inspektor aufgerufen werden.



Wie Sie sehen, ist die Liste der Kategorien links und die Liste der ausgewählten Formate rechts platziert. Wählen Sie die „Number“-Kategorie (Zahlenformat) und das „\$1,234,50“-Format. Danach wird der Format-String entsprechend dem gewählten Format unten angezeigt. Man kann einen Format-String genau wie das Dezimal-Trennzeichen verändern.

Nach dem Klicken der „OK“-Schaltfläche und dem Erstellen des Reports sehen Sie, wie die Gesamtsumme im Report nun korrekt ist:

1176	26.07.94	\$4 178,85
1269	16.12.94	\$1 400,00
		<hr/>
		\$51450,80

Inline-Formatierung

Der dargestellte Weg der Formatierung kann auf jeden Ausdruck angewandt werden, der im Objekt platziert ist. In Ihrem Fall funktioniert alles, da es lediglich ein Ausdruck im Objekt ist. Was, wenn Sie zwei Ausdrücke haben und die außerdem noch von unterschiedlichen Typen sind?

Versuchen Sie folgenden Fall:

die Gesamtsumme und Anzahl der Bestellungen werden in einem Objekt angezeigt. Um dies durchzuführen, muss folgender Text im Objekt eingetragen werden:

Total: [SUM(<Group."ItemsTotal">,MasterData1)]

Number: [COUNT(MasterData1)]

Beim Starten können Sie überprüfen, dass beide Werte im Währungsformat angezeigt werden (welches wir im vorigen Beispiel eingestellt hatten), was eher inkorrekt ist:

1269	16.12.94	\$1 400,00
		Total: \$51 450,80
		Number: \$6,00

Um die Werte richtig anzeigen zu lassen, muss jeder von ihnen individuell formatiert werden. Um dies durchzuführen, gibt es die so genannten Formatierungsmarkierungen. Sie werden vor den eckigen schließenden Klammern des Ausdrucks hinzugefügt. In unserem Beispiel stellen Sie die Objekt-Formatierung dar (wählen Sie die „Text“-Kategorie im Format-Editor). Nun müssen Sie das Format der ersten Variable modifizieren, da die zweite definitiv korrekt angezeigt wird (ohne Formatierung, d.h. als Zahl und das ist genau das, was wir brauchen). Um dies durchzuführen, verändern Sie den Text in folgender Weise:

Sum: [SUM(<Group."ItemsTotal">,MasterData1) #n%2,2m]
 Number: [COUNT(MasterData1)]

Überzeugen Sie sich, dass der Report nun korrekt arbeitet:

1269	16.12.94	\$1 400,00
		Total: \$51 450,80
		Number: 6

Legen Sie nun Ihre Aufmerksamkeit auf die Benutzung von Markierungen (tags). Die generelle Syntax ist folgende:

[Ausdruck #tag]

Zwischen dem Ausdruck und dem „#“-Zeichen steht ein Leerzeichen! Die Markierung selbst könnte wie folgt aussehen:

#nFormat-Zeile– numerisches Format

#dFormat-Zeile– Datum/Zeit Format

#bFalse,True– Boolesches Format

„Format-Zeile“ ist in jedem Fall ein Argument für die Funktion, mit deren Hilfe das Formatieren durchgeführt wird. Demnach würde das Argument für Datum/Zeit das FormatDateTime-Argument sein. Es folgen verschiedene Werte, die im FastReport genutzt werden:

für das numerische Formatieren:

%g – eine Zahl mit der minimalen Zeichen-Anzahl hinter dem Dezimalpunkt

%2.2f – eine Zahl mit einer festgelegten Anzahl von Zeichen nach dem Dezimalpunkt

%2.2n – eine Zahl mit einem Bit-Begrenzungszeichen

%2.2m – Währungs-Format, das in Windows OS akzeptiert wird, abhängig von den regionalen Einstellungen in der Systemeinstellung

für das Datum/Zeit Format:

tt.mm.jjjj – Datum der Formatierung 23.11.2003

tt mmm jjjj – Datum der Formatierung 23 Nov. 2003

tt mmmm jjjj – Datum der Formatierung 23 November 2003

hh:mm – Zeit der Formatierung 23:12


hh:mm:ss – Zeit der Formatierung 23:12:00

tt mmmm jjjj, hh:mm – Zeit und Datum der Formatierung 23 November 2003, 23:12

Es ist gültig, ein Komma oder einen Gedankenstrich anstelle des Punktes im numerischen Format zu setzen. In diesem Fall wird das Symbol als Trennung zwischen dem ganzzahligen und dem gebrochenen Teil der Zahl. Die Benutzung anderer Trennzeichen ist nicht gültig.


Beim Formatieren des „#b“-Typs (Boolescher Wert) wird die „Format-Zeile“ mit zwei Werten, getrennt durch ein Komma, angegeben. Der erste Wert bestimmt die Formatierung, wenn der Wert „False“ ist und der zweite den Wert „True“.

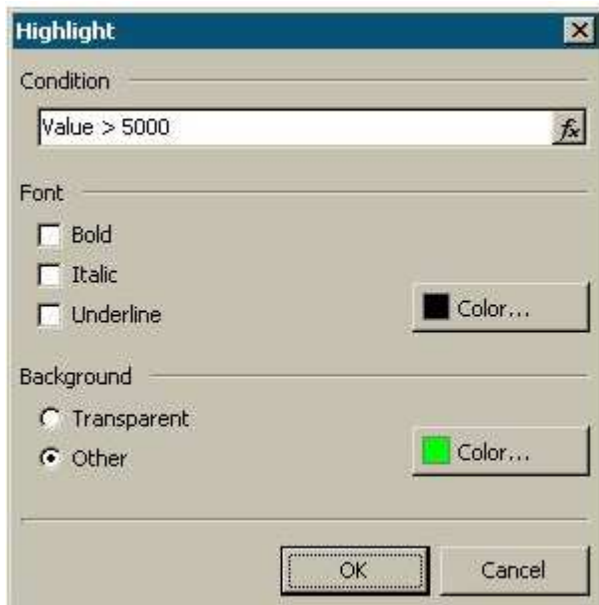
Um diese Argumente und all ihre Bedeutungen nicht merken zu müssen, gibt es eine einfache Quelle für das Formatieren im „Text“-Objekt-Editor. Beim klicken auf die

Schaltfläche  wird der Format-Editor (den wir bereits behandelt haben) aufgerufen. Nachdem man ein Format ausgesucht hat, wird es in den Text eingefügt. Falls der Cursor vor bzw. nach den eckigen schließenden Klammern gesetzt ist, ist das Format korrekt eingefügt.

Bedingtes Hervorheben

Diese Eigenschaft des „Text“-Objekts erlaubt es, ein Objekt farblich hervorzuheben, das sich auf eine spezielle Bedingung bezieht. Jeder Ausdruck kann eine Bedingung sein. Erläutern wir das farbliche Hervorheben mit dem Beispiel der Gruppen. Färben Sie die Gesamt-Bestellungen, die größer als 5000 sind grün ein. Um dies mit dem „Group.ItemTotal“-Feld durchzuführen, drücken Sie die „Bedingung

hervorheben“-Schaltfläche  in der Werkzeugleiste (oder wählen im Objekt-Inspektor bei der Eigenschaft „Highlight“ die Zeile „Condition“). Im Editor geben Sie nun eine Bedingung ein, nach der das Objekt hervorgehoben werden soll und wählen Sie ein Farb-Attribute aus (Schrift und Hintergrundfarbe), so wie im nächsten Fenster gezeigt.



Das Ergebnis sieht folgendermaßen aus:

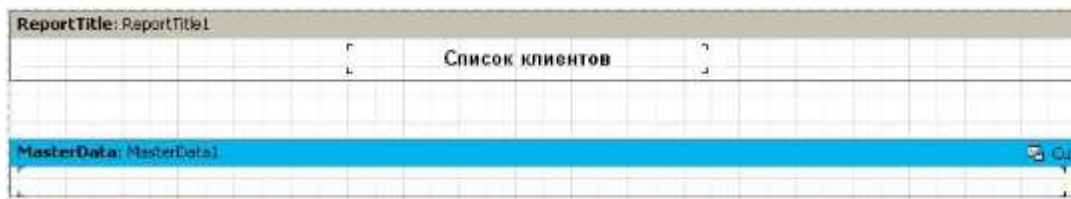
1221	Kauai Dive Shoppe	
1023	01.07.88	\$4 674,00
1076	16.12.94	\$17 781,00
1123	24.08.93	\$13 945,00
1169	06.07.94	\$9 471,95
1176	26.07.94	\$4 178,85
1269	16.12.94	\$1 400,00
		Total: \$51 450,80

Vorsicht mit der Bedingung, die wir festgelegt haben (Wert>5000). Der Wert ist der Datenbank-Feldwert mit dem das Objekt verbunden ist. Sie können die Bedingung „<Group.“ItemsTotal“> > 5000“ auch so schreiben. Allgemein kann jeder Ausdruck im FastReport als Bedingung geschrieben werden.

Streifen anzeigen

Mit Hilfe der „Bedingung hervorheben“-Eigenschaft ist es einfach den Report anspruchsvoller aussehen zu lassen, indem man jede zweite Datenzeile einfärbt. Probieren Sie es mit dem Report aus, den Sie im vorigen Kapitel erstellt hatten.

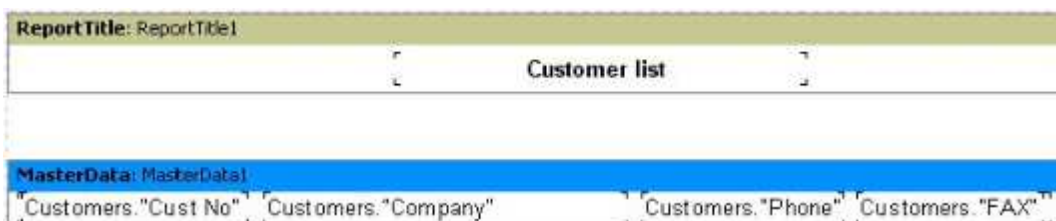
Zuerst platzieren Sie den „Report Titel“ und das „Master Data“-Band auf den Report. Setzen Sie ein „Text“-Objekt auf das Daten-Band und strecken Sie es, so dass es praktisch den ganzen Platz des Bandes ausfüllt:



Dieses Objekt stellt die Fläche dar, die abhängig von der Zeilennummer die Farbe wechseln soll. Danach rufen Sie den Bedingungseditor auf und setzen folgende Bedingung:

$\langle \text{Line} \rangle \bmod 2 = 1$

Wählen Sie grau zum Hervorheben aus, aber nicht zu gesättigt (eher heller). Nun können weitere Objekte auf das Daten-Band gesetzt werden:




Wenn viele Objekte über der Farbwechselfläche liegen, kommt es leicht zum Übersehen dieser. Beim Starten des Reports sehen Sie folgendes:

Customer list			
1645	Action Club	813-870-0239	813-870-0282
3158	Action Diver Supply	22-44-500211	22-44-500596
1984	Adventure Undersea	011-34-09054	011-34-09064
3053	American SCUBA Supply	213-654-0092	213-654-0095
6312	Aquatic Drama	613-442-7654	613-442-7678
3984	Blue Glass Happiness	213-555-1984	213-555-1995
1380	Blue Jack Aqua Center	401-609-7623	401-609-9403
1563	Blue Sports	610-772-6704	610-772-6898
2118	Blue Sports Club	612-897-0342	612-897-0348
3054	Catamaran Dive Club	213-223-0941	213-223-2324
1354	Cayman Divers World Unlimited	011-5-697044	011-5-697064
5151	Central Underwater Supplies	27-11-4432458	27-11-4433259
2156	Davy Jones' Locker	803-509-0112	803-509-0553
3055	Diver's Grotto	213-432-0093	213-432-4821


Reports mit verschiedenen Seiten

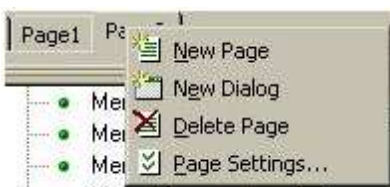
Ein Report kann aus verschiedenen Seiten bestehen. Sie können Parameter wie Größe und Hoch- oder Querformat auf jeder Seite einrichten, genauso wie Sie verschiedene Objekte und Bänder auf jeder Seite platzieren können.

Beim Erstellen eines Reports werden alle Bänder der ersten Seite, dann die der zweiten usw. angezeigt.

Wenn ein Benutzer einen neuen Report erstellt, enthält dieser bereits eine Seite als Voreinstellung. Sie können eine neue Seite hinzufügen, indem Sie auf die Schaltfläche  in der Werkzeugleiste klicken oder im Menü „Datei“ den Befehl „Neue Seite“ auswählen. Dann können Sie sehen, dass eine neue Registerseite im Designer erscheint:



Sie können einfach zwischen den beiden Seiten hin und her wechseln, indem Sie auf die erforderliche Registerseite klicken. Darüber hinaus können Registerseiten gezogen werden („drag&drop“), so können Sie die Reihenfolge der Seiten einfach verändern. Eine unnötige Seite kann mit der Hilfe der  -Schaltfläche in der Werkzeugleiste oder im Menü „Bearbeiten“ mit dem Befehl „Seite löschen“ gelöscht werden. Sie können außerdem das Kontext-Menü mit einem Klick mit der rechten Maustaste auf die Bezeichnung der Seite (Register) aufrufen:



Die Anzahl der Seiten in einem Report ist unbegrenzt. In der Regel werden zusätzliche Seiten als Titelseiten oder in komplizierteren Reports mit Daten aus vielen Quellen benutzt.

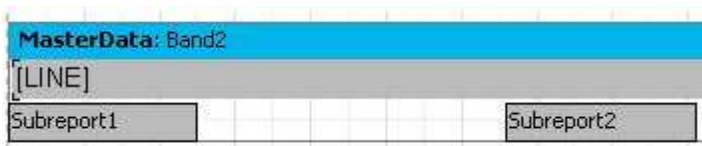
Versuchen Sie ein einfaches Beispiel mit einer Titelseite. Nutzen Sie den Report, den Sie vorher erstellt hatten. Fügen Sie eine neue Seite hinzu und so wird diese zur zweiten Seite. Um sie an den Anfang des Reports zu setzen, ziehen Sie die Registerseite mit Hilfe der Maus in die Nähe der ersten Seite und lassen es los. Es ändert sich nun die Reihenfolge der Seiten. Wechseln Sie zu der neuen Seite und platzieren ein „Text“-Objekt (mit dem Text „Our Report“ auf ihm) in der Mitte der Seite.

Sie können außerdem das „Subreport“-Objekt auf einen verschachtelten Report platzieren, dabei kommt es zu einer Erhöhung der Verschachtelungsebenen.

Es sollte deutlich sein, dass die Fähigkeit von FastReport einen „Subreport“ zu erstellen, die Verschachtelungs-Ebene von Daten erhöht. Denken Sie daran, dass die Anzahl von Ebenen im FastReport begrenzt ist, wenn Sie nicht das „Subreport“-Objekt nutzen (nicht mehr als sechs).

Nebeneinander stehende „Subreports“

Sie können zwei oder mehr „Subreport“-Objekte nebeneinander in ein Band einfügen:



So können Sie Reports erschaffen, die auf keine andere Art erstellt werden können (d.h. wenn Listen verschiedener Länge in jedem verschachtelten Report angezeigt werden):



Wie Sie sehen, fährt FastReport mit dem Basis-Report an der Stelle fort, an der die Ausgabe der längeren Liste endet.

Einschränkungen bei der Nutzung von „Subreports“

Ab der Erstellung eines verschachtelten Reports auf dem Basis-Report ist es nicht mehr möglich, folgende Bänder einzufügen:

„ReportTitle/ReportFooter“,
 „PageTitle/PageFooter/PageBackground“ und
 „ColumnTitle/ColumnFooter“.

Es ist möglich, diese Bänder auf den verschachtelten Report zu setzen, aber sie werden nicht verarbeitet. Aus dem selben Grund ist es nicht sinnvoll, die Einstellungen der verschachtelten Reportseite zu modifizieren, sofern die Einstellungen der Seite des Basis-Reports während der Erstellung eines Reports genutzt wurden.

Es ist unmöglich, Objekte hinter ein „Subreport“-Objekt zu setzen:



Beim Anzeigen eines verschachtelten Reports, wird das Objekt des verschachtelten Report alles überlappen, was nach ihm platziert wurde und der Nutzer erhält etwa folgendes:



Um das Objekt trotzdem nach dem verschachtelten Report anzeigen zu lassen, nutzen Sie das „Child“-Band:



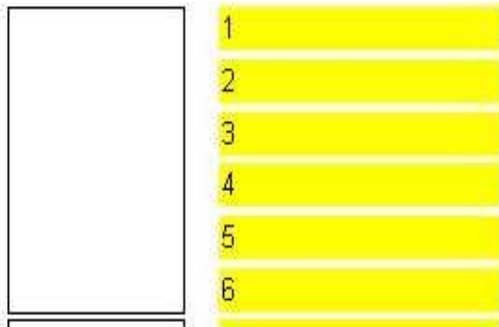
Es gibt auch Fälle, in denen es notwendig ist, verschiedene verschachtelte Reports untereinander anzeigen zu lassen.

Die PrintOnParent-Eigenschaft

Das „Subreport“-Objekt enthält die „PrintOnParent“-Eigenschaft, die in einigen Fällen recht hilfreich sein kann. Die Eigenschaft ist als Voreinstellung deaktiviert (False).

Normalerweise wird der Subreport als eine Zusammenstellung von Bändern auf dem Basis-Report gedruckt. In diesem Fall ist das Band, welches ein „Subreport“-Objekt beinhaltet, nicht abhängig vom Subreport-Band, d.h. es kann sich nicht strecken.

Falls die „PrintOnParent“-Eigenschaft eingestellt ist (True) - man kann dies im Objekt-Inspektor oder im Kontext-Menü einstellen -, werden die Objekte des Subreports



physikalisch auf dem Band, welches das „Subreport“-Objekt enthält, ausgedruckt. Man kann dieses Band streckbar machen und gestreckte Objekte draufsetzen:

„Kreuztabellen“-Reports

Diese Art von Reports hat die Struktur einer Tabelle, was bedeutet, dass er aus Spalten und Zeilen besteht. Gleichzeitig ist es vorneweg nicht klar, wie viele Zeilen und Spalten so eine Tabelle besitzen wird. Das ist deshalb so, weil ein Report nicht nur nach unten wächst (wie die Reports, die weiter oben beschrieben wurden) sondern auch zur Seite. Ein typisches Beispiel eines solchen Reports wird unten gezeigt.

Betrachten Sie die Elemente der Tabelle:

	1	2	3	4
a	a1	a2	a3	a4
b	b1	b2	b3	b4

In der Abbildung sehen Sie eine Tabelle mit zwei Zeilen und vier Spalten, in der „a“ und „b“ *Zeilentitel*, „1“, „2“, „3“ und „4“ *Spaltentitel* und „a1“..„a4“, „b1“..„b4“ *Zellen* sind. Um einen Report wie diesen zu erstellen, brauchen Sie lediglich einen Datenbestand (eine Abfrage oder eine Tabelle), der drei Felder hat und folgende Daten enthält:

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 Sie sehen können, enthält das erste Feld eine Zeilenbezeichnung, das zweite eine Spaltennummer und das dritte enthält den Inhalt als Schnittpunkt der Tabelle mit den ausgewählten Zahlen. Beim Erstellen eines Reports erstellt FastReport eine Tabelle im Datenspeicher und füllt sie mit Daten. Demzufolge expandiert die Tabelle dynamisch, falls eine Zeile oder eine Spalte mit einer bestimmten Zahl nicht existiert.

Die Titel können aus mehr als einer Ebene bestehen. Betrachten Sie folgendes Beispiel:

	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 Zahl oder der *Index* der Spalten zusammengesetzt, d.h. er besteht aus zwei Werten. Dieser Report verlangt folgende 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

```

In diesem Beispiel enthält das erste Feld die Zeilenbezeichnung, so wie es vorher war; das zweite und das dritte Feld enthalten Spaltenindexe. Das letzte Feld enthält den Zellenwert. Schauen Sie sich folgende Abbildung an, um zu verdeutlichen, wie FastReport eine Tabelle mit komplexen Titeln erstellt:

	10	10	20	20
	1	2	1	2
a	a10.1	a10.2	a20.1	a20.2
b	b10.1	b10.2	b20.1	b20.2

Wenn die Bearbeitung abgeschlossen ist, sollte unsere Tabelle wie die in der Abbildung aussehen. Während der Verarbeitung vereinigt FastReport die Titelzellen mit gleichen Werten, die in einer Ebene eingefügt werden.

Das nächste Tabellen-Element, welches in der folgenden Abbildung gezeigt wird, zeigt Zwischen- und Gesamtsummen an:

	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 Report ist erstellt worden, indem die selben Daten verwendet wurden wie im vorhergehenden. Die Spalten, im Bild hellgrün hervorgehoben, werden automatisch berechnet und sind nicht im anfänglichen Datenbestand enthalten.

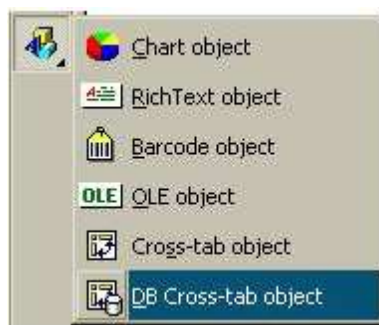
Erstellen eines „Kreuztabellen“-Reports

Gehen wir nun von der Theorie zur Praxis. Erstellen Sie einen einfachen „Kreuztabellen“-Report, der das Gehalt von Angestellten im Zeitraum von vier Jahren darstellt. Um dies durchzuführen, brauchen Sie die „simplecross.db“-Tabelle. Die Tabelle enthält Daten folgender Art:

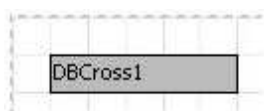
Name	Year	Salary
Ann	1999	3300
Ben	2002	2000
....		

Zum Erstellen von „Kreuztabellen“-Reports muss Sie die „TfrxCrossObject“-Komponente auf der Ausgangs-Form enthalten sein.

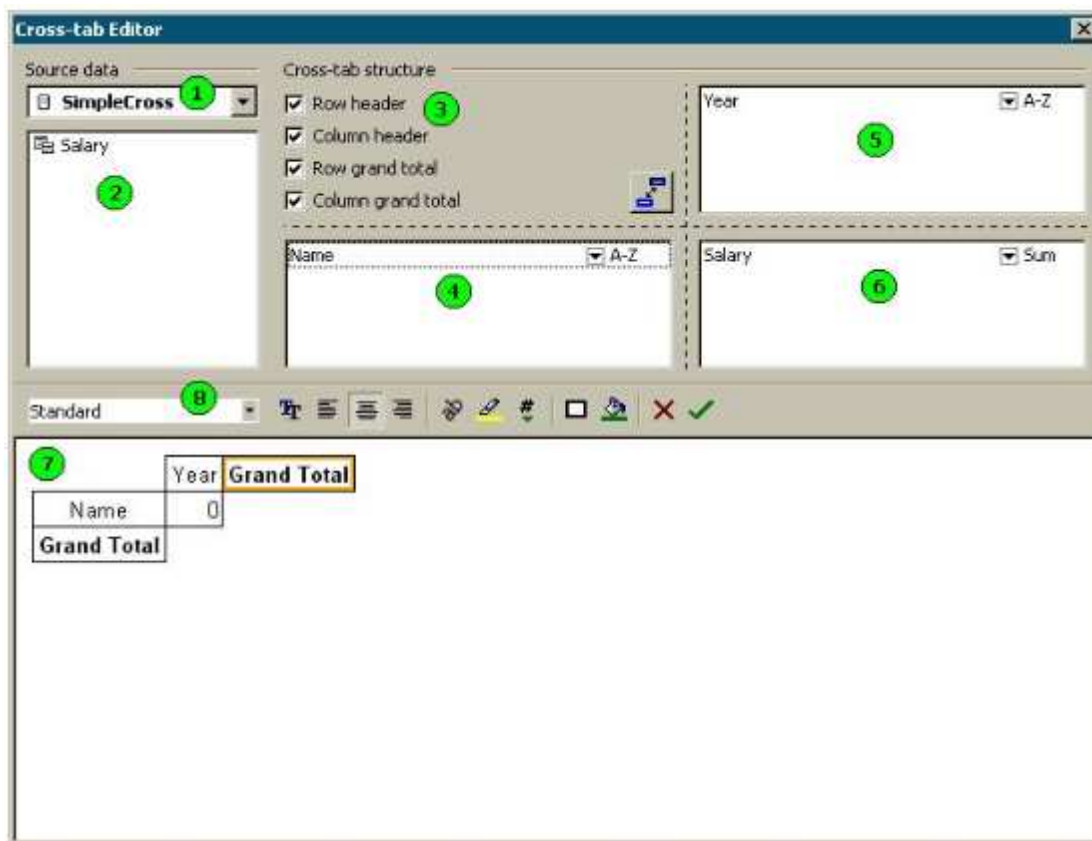
Wie üblich, erstellen Sie einen leeren Report. Gehen Sie dazu in den Report-Designer. Zuerst verbinden Sie unsere Datenquelle mit dem Menüdialog „Report“, „Daten...“. Setzen Sie das „DB Kreuztabellen“-Objekt auf den Report:



Auf der Seite sieht das Objekt einfach so aus:



Alle Einstellungen werden mit der Hilfe des Objekteditors gemacht. Rufen Sie ihn durch einen Doppelklick auf das Objekt auf:



Die folgenden mit Zahlen beschrifteten Elemente bedeuten:

- 1 – die Auswahlliste mit verfügbaren Datenquellen;
- 2 – die Liste der Felder, in der ausgewählten Datenquelle. Die Felder dieser Liste können in Listen 4, 5 und 6 gezogen werden;
- 3 – hier können Sie das Ein- bzw. Ausschalten der Anzeige von Titel und Gesamtsumme einstellen
- 4 – die Liste der Felder, die eine Zeilen-Überschrift erzeugen;
- 5 – die Liste der Felder, die eine Spalten-Überschrift erzeugen;
- 6 – die Liste der Felder, die eine Tabellen-Zelle erzeugen;
- 7 – hier gibt es eine Vorschau der zukünftigen Struktur der Tabelle; alle Elemente der Tabelle sind anklickbar;
- 8 – Werkzeugleiste um das Aussehen der Tabelle zu ändern;

Standard - Auswahl der Tabellen-Art;


- Textausrichtung;

- Text-Drehung;

- bedingtes Hervorheben;

- Zelle formatieren;


- Zellen-Rahmen und -füllung;

Wie Sie sehen, ist es möglich, hier nur mit der Maus vorzugehen. In unserem Fall reicht es zu, Felder von der Liste „2“ in die Listen „4“, „5“ und „6“ zu ziehen, wie es in der Abbildung gezeigt ist. Machen Sie erstmal noch nichts. Schließen Sie den Editor, indem Sie die „OK“-Schaltfläche  drücken. Wenn Sie nun den Report starten, erhalten Sie eine Tabelle wie unten:


	1999	2000	2001	2002	Grand Total
Ann	3300	2700	3100	1700	10800
Ben	4300	2400		2000	8700
Catherine	6100	3200			9300
Den		3999	8100		12099
Grand Total	13700	12299	11200	3700	40899

Das ist genau das, was Sie erhalten wollten. Probieren Sie das Objekt weiter aus. Rufen Sie den Objekteditor ein weiteres Mal auf. Das erste, was Sie tun werden, ist die Farbe der Überschriften zu verändern und anstelle von „Grand Total“ soll nun „Total“ angezeigt werden. Es ist recht einfach dies zu verrichten, wenn man das untere Editor-Feld verwendet (Nummer 7 in der Abbildung). Hier wird die Struktur der Kreuztabelle angezeigt und alles kann genauso gut mit der Maus eingestellt werden. Die aktive Zelle wird mit einem orangen Rahmen dargestellt:

	Year	Grand Total
Name	0	
Grand Total		

Um die Farbe der Überschriften in grau zu ändern, klicken Sie einfach auf die Objekte „Name“, „Year“ und „Grand Total“, eins nach dem anderen und wählen dann die gewünschte Farbe durch Anklicken der  Schaltfläche in der Werkzeugleiste. Um die „Grand Total“-Beschriftung zu verändern, klicken Sie doppelt auf die Zelle, damit sich der Texteditor öffnet, in dem man dann „Total“ eintippt. Danach wird Ihr Report wie folgt aussehen:

	1999	2000	2001	2002	Total
Ann	3300	2700	3100	1700	10800
Ben	4300	2400		2000	8700
Catherine	6100	3200			9300
Den		3999	8100		12099
Total	13700	12299	11200	3700	40899

Es bleibt nur noch, das Format einzustellen, in dem die Geldwerte angezeigt werden. Um dies im Kreuz-Objekteditor durchzuführen, klicken Sie auf das „Total“-Objekt und das Objekt, welches eine Zelle darstellt (mit „0“ Text) nacheinander und wählen sie das erforderliche Format, indem Sie auf die  Schaltfläche in der Werkzeugleiste klicken. Sie werden folgendes Ergebnis erhalten:

	1999	2000	2001	2002	Total
Ann	\$3 300,00	\$2 700,00	\$3 100,00	\$1 700,00	\$10 800,00
Ben	\$4 300,00	\$2 400,00		\$2 000,00	\$8 700,00
Catherine	\$6 100,00	\$3 200,00			\$9 300,00
Den		\$3 999,00	\$8 100,00		\$12 099,00
Total	\$13 700,00	\$12 299,00	\$11 200,00	\$3 700,00	\$40 899,00

Benutzung von Funktionen

In unserem Beispiel wird die Gesamtsumme jedes Angestelltengehalts während der letzten vier Jahre in der „Total“-Zeile angezeigt. Sie können folgende Funktionen verwenden:

SUM – Summe der Werte

MIN – Minimalwert

MAX – Maximalwert

AVG – Durchschnittswert

COUNT – Anzahl der Werte

Probieren Sie die „MIN“-Funktion in unserem Beispiel einmal aus. Um dies durchzuführen, öffnen Sie den Kreuz-Objekteditor und klicken auf das „Salary“-Feld in dem Bereich mit dem Symbol, wo ein Pfeil nach unten zeigt.



Wählen Sie die „MIN“-Funktion im Menü aus.

Nun können Sie einen Text in der Zelle der Gesamtsumme von „Total“ zu „Minimum“ ändern.

Der fertige Report sieht in etwa so aus:

	1999	2000	2001	2002	Minimum
Ann	3300	2700	3100	1700	1700
Ben	4300	2400		2000	2000
Catherine	6100	3200			3200
Den		3999	8100		3999
Minimum	3300	2400	3100	1700	1700

Werte sortieren

Zeilen und Spalten sind in aufsteigender Reihenfolge angeordnet. Demnach werden sie, falls die Werte vom Typ „Zahl“ sind, nach ihrem Wert sortiert und wenn sie vom Typ „Zeichen“ sind, werden sie alphabetisch geordnet.

Folgende Möglichkeiten sind verfügbar:

„Anordnung in aufsteigender Reihenfolge“,
 „Anordnung in absteigender Reihenfolge“ und
 „ohne Sortierung“.

Im letzteren Fall werden die Werte in Zeilen/Spalten in der Reihenfolge, in der sie eingetragen werden, angezeigt.

Verändern Sie die Sortierung der Spalten in unserem Beispiel. Ordnen Sie die Jahre in absteigender Reihenfolge an. Um dies durchzuführen, nutzen wir den Kreuz-Objekteditor und wählen das Element „Year“-Spalte. Um die Anordnung zu verändern, klicken Sie auf den Bereich des Symbols mit dem Pfeil nach unten und wählen absteigend (Z-A):



Danach schließen Sie den Editor und starten den Report, der folgendes zeigt:

	2002	2001	2000	1999	Total
Ann	1700	3100	2700	3300	10800
Ben	2000		2400	4300	8700
Catherine			3200	6100	9300
Den		8100	3999		12099
Total	3700	11200	12299	13700	40899

Tabellen mit zusammengesetzten Kopfzeilen

Unser vorheriges Beispiel enthält eine Überschrift pro Zeile und Spalte. Versuchen Sie

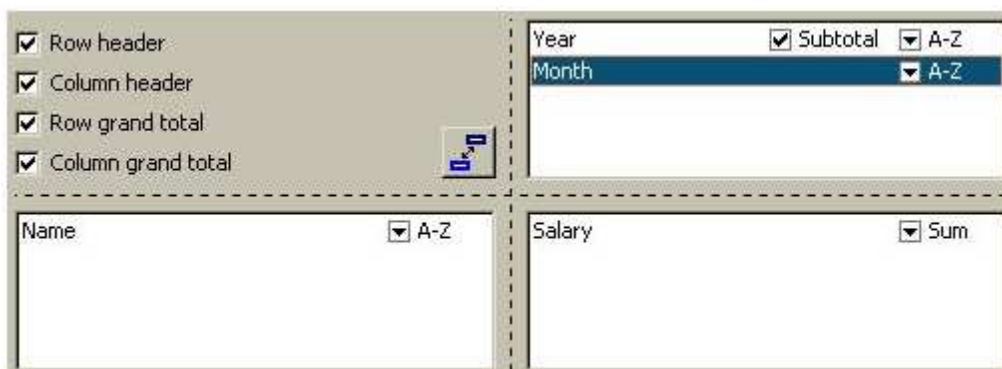
Name	Year	Month	Days	Salary
Ann	1999	2	3	1000
Ben	2002	1	5	2000
....				

nachfolgend eine Tabelle mit einer komplexen Kopfzeile zu erstellen, was bedeutet, dass sie zwei oder mehr Werte enthält. Um dies durchzuführen, werden Sie die „cross.db“-Tabelle brauchen. Die Tabelle enthält Daten mit folgendem Aussehen:

Wie Sie sehen, ist diese Tabelle ähnlich der „simplecross.db“-Tabelle, die Sie im vorhergehenden Beispiel verwendet haben. Wir haben die „Month“- und „Days“-Felder hinzugefügt, welche die Anzahl der Monate und Tage, die gearbeitet wurden, anzeigen. Man kann verschiedene Reports auf der Basis dieser Daten erstellen, z.B. das Gehalt aller Angestellten über alle Jahre hinweg in Monate aufgeteilt.

Erstellen Sie einen neuen Report. Im Designer machen Sie die üblichen Dinge: verbinden die Datenquelle, indem Sie das Menüdialog „Report“, „Daten...“ benutzen und anschließend das „DB Kreuztabelle“-Objekt in den Report einfügen. Um das „Kreuztabellen“-Objekt einzustellen, starten Sie seinen Editor mit einem Doppelklick auf das Objekt.

Welche Art Report müssen wir auswählen? Er muss dem Report im vorigen Beispiel ähneln, aber zur selben Zeit muss er auch in Monate aufgeteilt sein. Daraus folgt, dass das „Kreuztabellen“-Objekt in der selben Weise eingestellt werden muss, aber Sie müssen zusätzlich das „Month“-Feld in die Spaltenliste hinzufügen:



So wird im unteren Teil des Editors die zukünftige Reportstruktur angezeigt:

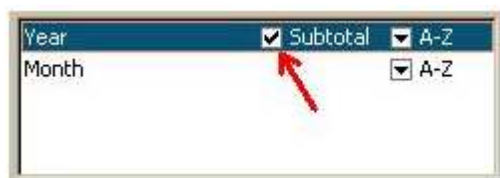
		Year	Grand Total
	Month	Total	
Name			
Grand Total			

Als Ergebnis erhalten Sie folgende Ausgabe:

	1999					2000				2001				2002		Grand Total
	2	10	11	12	Total	1	2	3	Total	1	2	3	Total	1	Total	
Ann	1000		1100	1200	3300	1300	1400		2700		1500	1600	3100	1700	1700	10800
Ben		2100	2200		4300		2400		2400				0	2000	2000	8700
Catherine		3000	3100		6100			3200	3200				0		0	9300
Den					0	3999			3999	4000	4100		8100		0	12099
Grand Total	1000	5100	6400	1200	13700	5299	3800	3200	12299	4000	5600	1600	11200	3700	3700	40899

Sie werden bemerken, dass FastReport automatisch eine Spalte hinzugefügt hat, mit der Teilsumme, die nach jedem Jahr angezeigt wird. Diese Eigenschaft kann im Kreuzobjekteditor eingestellt werden:

Es reicht, wenn Sie das „Year“-Spalten-Element auswählen und die „Subtotal“-Markierung ausschalten:



Zusätzlich bemerken Sie, dass es keine Teilsummen in den untersten Spalten-Elementen gibt (das selbe trifft zu, wenn dieses Element das Einzige ist). Derzeit brauchen Sie in unserem Beispiel keine Teilsumme für jeden Monat.

Probieren Sie, ein weiteres Beispiel die Teilsummen betreffend, aus. In unserem Beispiel ist es wünschenswert, „Gesamt für 1999“ anstelle der „Total“-Beschriftung anzuzeigen. Es ist einfach, dies durchzuführen. Rufen Sie den Kreuz-Objekteditor auf, wählen das erforderliche Objekt in dem unteren Teil aus und geben folgenden Text ein:

Total for [Value]

Während der Erstellung wird der „Value“-Ausdruck durch den Wert der Tabellenüberschrift ersetzt, die darüber angeordnet ist:

	1999				
	2	10	11	12	Total for 1999
Ann	1000		1100	1200	3300
Ben		2100	2200		4300
Catherine		3000	3100		6100
Den					0
Grand Total	1000	5100	6400	1200	13700

Spaltenbreite anpassen

Beim Schauen auf die obere Abbildung wird offensichtlich, dass FastReport die Spaltenbreite automatisch anpasst, in einer Weise, die es ermöglicht, die längeren Zeilen der Zelle anzupassen. Dies ist in einigen Fällen nicht wünschenswert, da die Tabellen mit sehr langen Zeilen nicht gut aussehen. Was kann in solch einer Situation getan werden?

Der einfachste Weg ist es, Zeilen im Text des Objekts mit Teilsummen zu teilen, d.h. eine Zeile hinzuzufügen:

*Total
for [Value]*

Sie sehen, dass die Tabelle nun besser aussieht:

	1999				
	2	10	11	12	Total for 1999
Ann	1000		1100	1200	3300
Ben		2100	2200		4300
Catherine		3000	3100		6100
Den					0
Grand Total	1000	5100	6400	1200	13700

Dennoch können solche Methoden nicht immer angewandt werden. Falls der Zeilen-/Spalten-Wert eher lang ist, kann dieser nicht durch manuelles Teilen korrigiert werden. Das kommt daher, dass das „Kreuztabellen“-Objekt die „MinWidth“- und „MaxWidth“-Eigenschaft hat (minimale und maximale Spaltenbreite). Beide Eigenschaften sind über den Kreuz-Objekteditor zugänglich.

Der „Minimalbreite“-Wert ist „0“ und der „Maximalbreite“-Wert ist „200“ als Voreinstellung. Das ist in den meisten Fällen genug. Sie können ihre Werte entsprechend den Erfordernissen Ihrer Tabelle setzen.

Demnach können Sie in unserem Beispiel folgendes einstellen:

MinWidth=MaxWidth=50. Das setzt die Spaltenbreite auf 50 Pixel fest. Falls eine Zelle kleiner ist, wird sie der „Minimalbreite“ angepasst, falls sie größer ist, wird die Breite entsprechend der „Maximalbreite“ eingestellt und der Text in der Zelle wird geteilt. In unserem Beispiel würde das folgendermaßen aussehen:


	1999				
	2	10	11	12	Total for 1999
Ann	1000		1100	1200	3300
Ben		2100	2200		4300
Catherine		3000	3100		6100
Den					0
Grand Total	1000	5100	6400	1200	13700

Schriftfarbe und Hervorhebung

Manchmal ist es nötig, Werte hervorzuheben und/oder die Schriftfarbe zu ändern. Sie haben dies in dem Report mit Gruppen ausgetestet. Da haben Sie die Hervorhebungseigenschaft im „Text“-Objekt genutzt, die für uns jetzt genauso hilfreich sein kann.

Führen Sie den Prozess des Hervorhebens durch, indem Sie unser Beispiel nehmen. Nehmen Sie an, dass Sie die Schriftfarbe der Werte verändern müssen, die größer als 3000 sind. Um dies durchzuführen, öffnen Sie den Kreuz-Objekteditor und klicken im unteren Teil des Editorfensters auf das Objekt, welches die Objektzelle repräsentiert:

	Year		Grand Total
	Month	Total for Year	
Name		0	
Grand Total			

Um Parameter hervorzuheben klicken Sie auf die Schaltfläche  in der Werkzeugleiste. Das bereits bekannte Fenster zum Hervorheben wird geöffnet, in dem man die folgende Bedingung eingibt:

Value>3000

Das ist alles, was Sie brauchen. Schließen Sie den Editor mit der Hilfe der „OK“-Schaltfläche und starten Sie Ihren Report:

	1999					2000			
	2	10	11	12	Total for 1999	1	2	3	Total for 2000
Ann	1000		1100	1200	3300	1300	1400		2700
Ben		2100	2200		4300		2400		2400
Catherine		3000	3100		6100			3200	3200
Den					0	3999			3999
Grand Total	1000	5100	6400	1200	13700	5299	3800	3200	12299

In der gleichen Weise ist der Anwender in der Lage, die Werte von allen Spalten und Zeilen hervorzuheben, falls dies nötig ist.

Handhabung einer „Kreuztabelle“ mit Hilfe eines Scripts

Um eine Tabelle zu erstellen, sind die visuellen Objekte manchmal nicht genug, Sie können das sogenannte „Script“ für detaillierte Einstellungen des Aussehens der Tabelle nutzen.

Das „Kreuztabellen“-Objekt hat folgende Ereignisse:

Ereignis	Beschreibung
OnAfterPrint	Ereignis wird aufgerufen, nachdem die Tabelle ausgedruckt wurde.
OnBeforePrint	Ereignis wird aufgerufen, bevor die Tabelle ausgedruckt wird.
OnCalcHeight	Ereignis wird aufgerufen, bevor die Höhe einer Zeile der Tabelle berechnet wird. Ein „Script“ kann die gewünschte Höhe übergeben oder „0“, wenn die Zeile weg gelassen werden soll.
OnCalcWidth	Ereignis wird aufgerufen, bevor die Spaltenbreite in der Tabelle berechnet wird. Ein „Script“ kann die gewünschte Höhe übergeben oder „0“, wenn die Spalte weg gelassen werden soll.
OnPrintCell	Ereignis wird aufgerufen, bevor eine Tabellen-Zelle angezeigt wird. Ein „Script“ kann das Aussehen oder den Inhalt der Zelle verändern.
OnPrintColumnHeader	Ereignis wird aufgerufen, bevor die Überschrift der Tabellenspalte angezeigt wird. Ein „Script“ kann das Aussehen oder den Inhalt der Überschriften-Zelle verändern.
OnPrintRowHeader	Ereignis wird aufgerufen, bevor die Überschrift der Tabellen-Zeile angezeigt wird. Ein „Script“ kann das Aussehen oder den Inhalt der Überschriften-Zelle verändern.

Es ist hilfreich, die folgenden Funktionen des „Kreuztabellen“-Objekts in der Ereignis-Routine zu benutzen:

Methode	Beschreibung
function ColCount: Integer	Gibt die Anzahl der Spalten (Datenfelder) in einer Tabelle wieder.
function RowCount: Integer	Gibt die Anzahl der Zeilen (Datensätze) in einer Tabelle wieder.
function IsGrandTotalColumn (Index: Integer): Boolean	Gibt „True“ zurück, wenn die angegebene Spalte die Gesamtsumme ist.
function IsGrandTotalRow (Index: Integer): Boolean	Gibt „True“ zurück, wenn die angegebene Zeile die Gesamtsumme ist.
function IsTotalColumn (Index: Integer): Boolean	Gibt „True“ zurück, wenn die angegebene Spalte eine Spalte mit einer Zwischensumme ist.
function IsTotalRow (Index: Integer): Boolean	Gibt „True“ zurück, wenn die angegebene Zeile eine Zeile mit einer Zwischensumme ist.
Procedure AddValue(const Rows, Columns, Cells: array of Variant)	Fügt der Tabelle einen Wert in der angegebenen Zeile (Rows) und Spalte (Columns) hinzu.

Erörtern wir, wie man die dritte Spalte hervorheben kann (in unserem Beispiel ist es die Spalte mit dem „November“-Datum). Um dies durchzuführen, wählen Sie die „Kreuztabelle“ und erstellen das OnPrintCell-„Script“:

```
procedure Cross1OnPrintCell(Memo: TfrxMemoView;
 RowIndex, ColumnIndex, CellIndex: Integer;
  RowValues, ColumnValues, Value: Variant);
begin
  if ColumnIndex = 2 then
    Memo.Color := clRed;
end;
```

Sie bekommen folgendes Ergebnis:

	1999				
	2	10	11	12	Total
Ann	1000		1100	1200	3300
Ben		2100	2200		4300
Catherine		3000	3100		6100
Den					0
Grand Total	1000	5100	6400	1200	13700

Um zusätzlich die Spalten-Überschrift hervorzuheben, erstellen Sie ein „OnPrintColumnHeader“-„Script“:

```
procedure Cross1OnPrintColumnHeader(Memo: TfrxMemoView;
  HeaderIndexes, HeaderValues, Value: Variant);
begin
  if (VarToStr(HeaderValues[0]) = '1999') and
    (VarToStr(HeaderValues[1]) = '11') then
    Memo.Color := clRed;
end;
```

Das Ergebnis wird folgendermaßen aussehen:

	1999				
	2	10	11	12	Total
Ann	1000		1100	1200	3300
Ben		2100	2200		4300
Catherine		3000	3100		6100
Den					0
Grand Total	1000	5100	6400	1200	13700

Wie funktioniert ein „Script“? Das „OnPrintCell“-„Script“ wird aufgerufen, bevor eine Zelle im Tabellen-Körper ausdruckt wird (beim Ausdrucken von Zellen der Tabellen-Überschrift wird entweder das „OnPrintColumnHeader“- oder das „OnPrintRowHeader“-„Script“ aufgerufen). Zugleich wird eine Verbindung zum „Text“-Objekt hergestellt, welches eine Tabellen-Zelle („Memo“-Parameter) und eine Zellen-Adresse repräsentiert in zwei Varianten:

1. der Nummer der Zeile,
der Nummer der Spalte und
der Nummer der Zelle (das letzte ist relevant, wenn die Tabellen-Zelle über mehrere Ebenen geht)
in den „RowIndex“- „ColumnIndex“- und „CellIndex“-Parametern,
die in das „OnPrintCell“-„Script“ übertragen werden.
2. Die „RowValues“ und „ColumnValues“-Parameter sind die zweite Variante der „Adresse“ einer Zelle. Der „Values“-Parameter ist der Inhalt der Zelle.

Um eine Zellen-„Adresse“ anzugeben, können Sie die zweite Variante nutzen (RowValues, ColumnValues). Sie ist im gegebenen Fall günstiger - genauso gut können Sie die erste Variante verwenden (RowIndex, ColumnIndex). In unserem Fall wollen wir die dritte Spalte hervorheben; so wäre es günstiger die erste Variante zu betrachten. Da die Nummerierung der Spalten und Reihen mit „0“ beginnt, erlaubt uns der „ColumnIndex=2“-Vergleich die dritte Spalte zu definieren. Sie können dies auf verschiedene Weise tun, d.h. auch durch Vergleich der gewünschten Spalte anhand unserer Daten (wir brauchen den 11. Monat von 1999):

```
procedure CrosslOnPrintCell(Memo: TfrxMemoView;
  RowIndex, ColumnIndex, CellIndex: Integer;
  RowValues, ColumnValues, Value: Variant);
begin
  if (VarToStr(ColumnValues[0]) = '1999') and
    (VarToStr(ColumnValues[1]) = '11') then
    Memo.Color := clRed;
end;
```

Werte, die in die „RowValues“- und „ColumnValues“-Parameter übertragen werden, sind Arrays des Types „Variant“ mit der Basis Null. Das Null-Element ist ein Wert der höchsten Ebene der Tabellen-Überschrift; das erste ist ein Wert der nächsten Ebene, usw. In unserem Fall enthält „ColumnValues[0]“ die Jahre und „ColumnValues[1]“ die Monate.

Warum ist eine „VarToStr“-Umwandlung nötig? Dies garantiert das Ausbleiben von Fehlern während der Typ-Umwandlung.

Das „OnPrintColumnHeader“-„Script“ wird während des Schreibens in die Zelle der Spaltenüberschrift aufgerufen. Die Einstellung der Parameter ist ähnlich den Parametern des „OnPrintCell“-„Script“, obwohl in diesem Fall die Zellen-„Adresse“ (die „HeaderIndex“- und „HeaderValues“-Parameter) in einer anderen Weise übertragen werden. Der „HeaderValues“-Parameter gibt den gleichen Wert wie die „ColumnValues“- und „RowValues“-Parameter im „OnPrintCell“-„Script“ zurück. Der „HeaderIndex“-Parameter ist ebenfalls ein Array von Werten des -Types „Variant“, welches eine Adresse der Überschriften-Zelle in einer anderen Form beinhaltet.

Das Null-Element ist eine laufende Nummer der höchsten Ebene der Tabellenüberschriften, das Eins-Element ist eine Zahl der nächsten Ebene, usw. Um die Prinzipien der Nummerierung der Zellen zu verdeutlichen, sehen Sie auf die Abbildung unten:

	0					1				2			
	0	1	2	3	4	0	1	2	3	0	1	2	3
0	1000		1100	1200	3300	1300	1400		2700		1500	1600	3100
1		2100	2200		4300		2400		2400				0
2		3000	3100		6100			3200	3200				0
3					0	3999			3999	4000	4100		8100
4	1000	5100	6400	1200	13700	5299	3800	3200	12299	4000	5600	1600	11200

In unserem Fall ist es günstiger, den „HeaderValues“-Wert zu betrachten, aber Sie können auch folgendes „Script“ schreiben:

```
procedure Cross1OnPrintColumnHeader(Memo: TfrxMemoView;
  HeaderIndexes, HeaderValues, Value: Variant);
begin
  if (HeaderIndexes[0] = 0) and (HeaderIndexes[1] = 2) then
    Memo.Color := clRed;
end;
```

Größe von Zeilen/Spalten anpassen

Der Benutzer kann Breite und Höhe der Tabellenzeilen und -spalten mit Hilfe der „OnCalcWidth“- und „OnCalcHeight“-Ereignis-Routine einstellen. Wir wollen sehen, wie Sie die Breite der Spalte, die mit dem 11. Monat von 1999 zusammenhängt, im folgenden Beispiel vergrößern. Um dies durchzuführen, erstellen Sie ein „OnCalcWidth“-„Script“:

```
procedure Cross1OnCalcWidth(ColumnIndex: Integer;
  ColumnValues: Variant; var Width: Extended);
begin
  if (VarToStr(ColumnValues[0]) = '1999') and
    (VarToStr(ColumnValues[1]) = '11') then
    Width := 100;
end;
```

Und das Ergebnis sieht wie folgt aus:

	1999				
	2	10	11	12	Total
Ann	1000		1100	1200	3300
Ben		2100	2200		4300
Catherine		3000	3100		6100
Den					0
Grand Total	1000	5100	6400	1200	13700

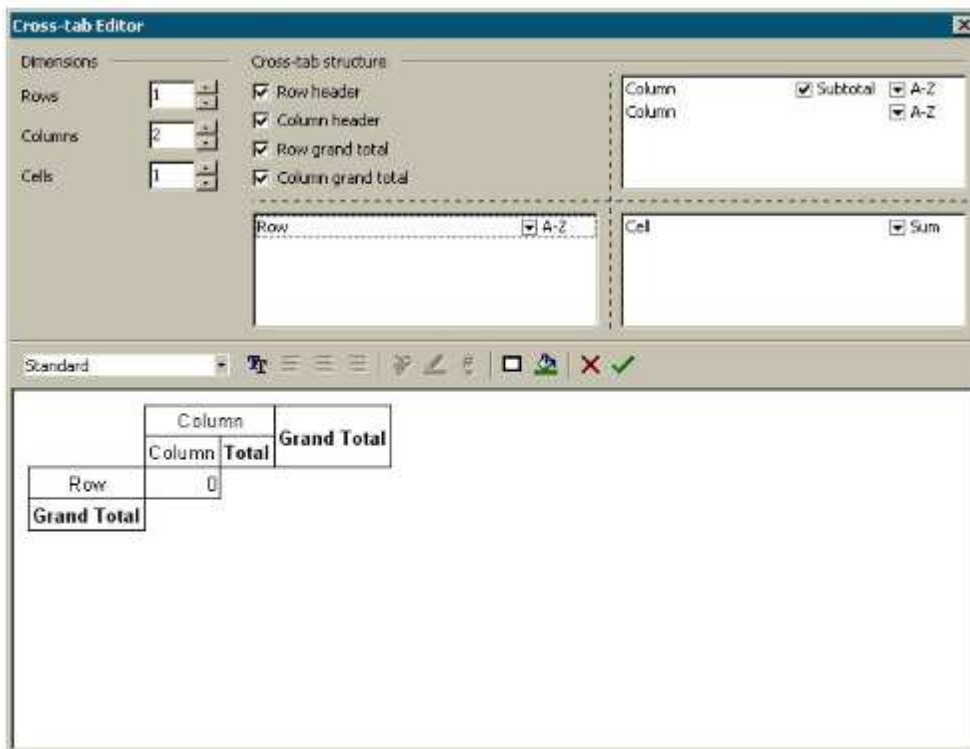
In unserem Beispiel reicht es, die Breite=0 zu setzen, um eine Spalte wegzulassen. Aber aufpassen, da die Summen zur selben Zeit nicht neu berechnet werden, da die Tabelle zu diesem Zeitpunkt bereits mit Werten gefüllt ist.

Eine „Kreuztabelle“ manuell füllen

Wie Sie bereits wissen, gibt es zwei Varianten der „Kreuztabelle“: die „DB Kreuztabelle“ und die „Kreuztabelle“.

Sie haben die ganze Zeit mit der ersten Variante gearbeitet, die, verbunden mit den Daten einer Datenbank-Tabelle, sich automatisch füllt, sobald der Report läuft. Probieren Sie das zweite Objekt die „Kreuztabelle“ aus.

Dieses Objekt ist nicht an die Daten einer Datenbank angeschlossen. So müssen Sie die Tabelle manuell mit Daten ausfüllen. Dieses Objekt besitzt einen ähnlichen Editor, aber Sie müssen die Ausmaße der Tabellenüberschriften und -zellen angeben anstatt der Auswahl von Datenbankfeldern:



Demonstrieren wir das Arbeiten mit dem „Kreuztabellen“-Objekt an einem Beispiel. Setzen Sie das Objekt auf einen Reports und zwar so, wie es im vorherigen Bild dargestellt ist: die Anzahl der Ebenen bei den Zeilenüberschriften ist „1“, bei den Spaltenüberschriften „2“, bei den Zellen „1“.

Um die Tabelle mit Daten zu füllen, nutzen Sie die „OnBeforePrint“-Ereignis-Routine dieses Objekts:

```
procedure Cross1OnBeforePrint(Sender: TfrxComponent);
begin
  with Cross1 do
    begin
      AddValue(['Ann'], [2001, 2], [1500]);
      AddValue(['Ann'], [2001, 3], [1600]);
      AddValue(['Ann'], [2002, 1], [1700]);

      AddValue(['Ben'], [2002, 1], [2000]);

      AddValue(['Den'], [2001, 1], [4000]);
      AddValue(['Den'], [2001, 2], [4100]);
    end;
  end;
end;
```

Es ist nötig, im „Script“ die erforderlichen Daten mit der Methode „TfrxCrossView.AddValue“ in die Tabelle einzufügen. Diese Methode hat drei Parameter; jeder von ihnen ist ein Array von Werten des Types „Variant“. Der erste Parameter ist der Zeilenwert, der zweite ist der Spaltenwert und der dritte enthält die Zellenwerte. Beachten Sie, dass die Anzahl der Werte in jedem Array den Objekt-Einstellungen entsprechen muss! In unserem Beispiel hat das Objekt eine Ebene in der Zeilen-Überschrift, zwei Ebenen in der Spalten-Überschrift und eine Ebene bei den Zellen. So übertragen Sie mit „AddValue“ einen Wert für die Zeilen, zwei für die Spalten und einen für die Zellen.

Beim Ausführen des Reports sehen Sie folgendes:

	2001				2002		Grand Total
	1	2	3	Total	1	Total	
Ann		1500	1600	3100	1700	1700	4800
Ben				0	2000	2000	2000
Den	4000	4100		8100		0	8100
Grand Total	4000	5600	1600	11200	3700	3700	14900

Sie können die „AddValue“-Methode ebenso gut für die „DB Kreuztabelle“ verwenden. Dies macht es möglich, Daten in die „Kreuztabelle“ einzufügen (welche nicht in der Datenquelle vorhanden sind, die an das Objekt angebunden ist). Andererseits, wenn es solche Daten gibt, werden sie mit den Daten aus der Tabelle zusammengerechnet.

Diagramme

FastReport ermöglicht es, Diagramme in den Report einzufügen.

Wir zeigen eine einfache Erstellung eines Diagramms und nutzen folgendes Beispiel. Um dies durchzuführen, brauchen Sie die „country.db“-Tabelle aus dem „DBDEMOS“-Demo-Daten. Die Tabelle enthält Daten über Länder, ihre Fläche und Bevölkerung:

Name	Area	Population
Argentina	2 777 815	32 300 003
Bolivia	1 098 575	7 300 000

....

Erstellen Sie einen neuen leeren Report und verbinden Sie die Datenquelle im Designer. Danach setzen Sie das „Diagramm“-Objekt auf die Reportseite:

Table1:

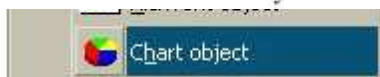
DatabaseName = 'DBDEMOS'

TableName = 'country.db'

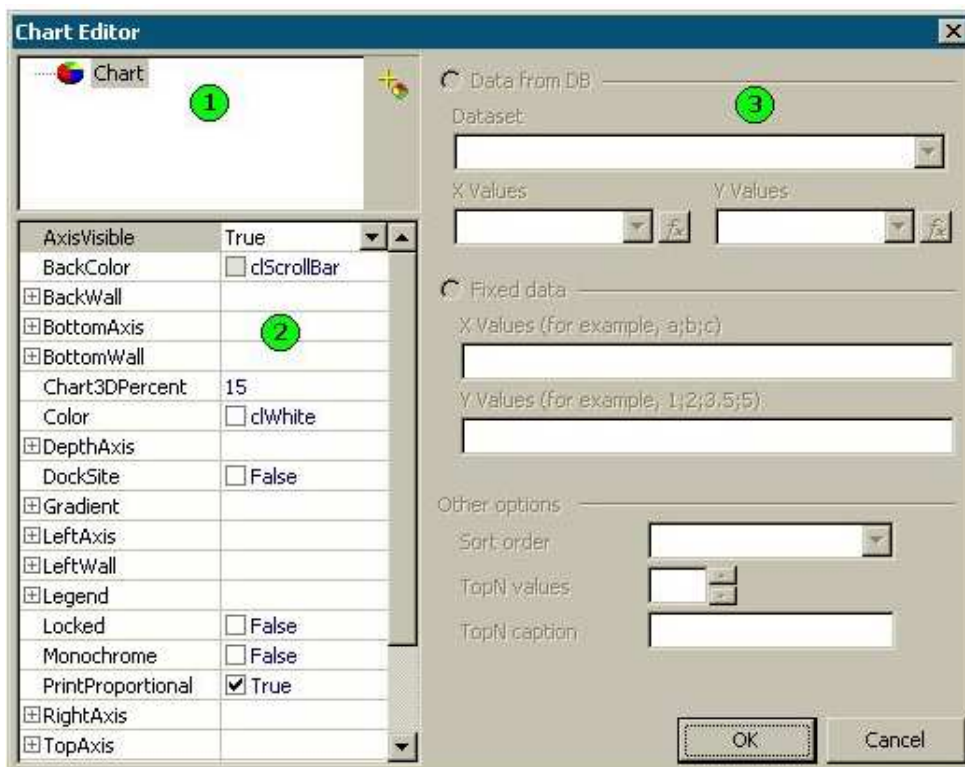
frxDBDataSet1:

DataSet = Table1

UserName = 'Country'




Setzen Sie die Größe des Objekts auf 18x8 cm. Um das Objekt anzupassen, rufen Sie seinen Editor per Doppelklick auf das Objekt auf.



Die mit Zahlen beschrifteten Elemente bedeuten:

- 1 – Diagramm-Struktur. Das Diagramm kann entweder eine oder mehrere Arten beinhalten;
- 2 – Objekt-Inspektor, der die Eigenschaften der gewählten Objekte im Fenster anzeigt. So können Sie die Eigenschaften des Diagramms optimieren.
- 3 – Werkzeugleiste zum Verbinden von Diagrammarten mit Daten; sie wird aktiviert, sobald die Art im Fenster 1 ausgewählt wird.

Beim ersten Aufruf erscheint im Editor-Fenster ein Bild (1), wie in der Abbildung gezeigt. Zuerst müssen eine oder mehrere Arten hinzugefügt werden (in unserem Beispiel eine

Art). Um dies durchzuführen, klicken Sie auf die Schaltfläche  und wählen das „Kuchen“-Diagramm im Menü aus:

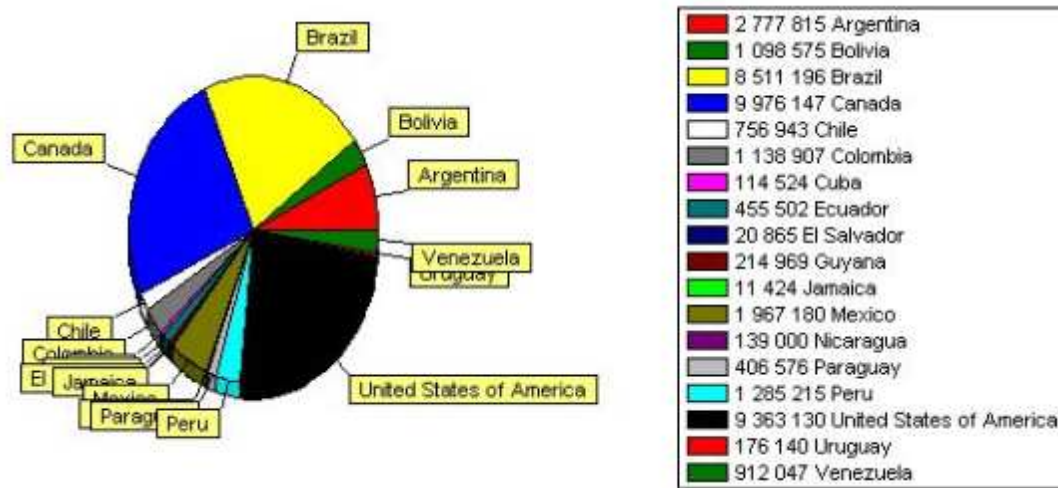


Wie Sie sehen, sind 8 verschiedene Arten verfügbar. Nachdem Sie die Art hinzugefügt haben, wird die Werkzeugleiste (3) aktiv. Hier müssen Sie angeben, welche Daten für die Darstellung genutzt werden. Zuerst wählen Sie die Tabelle aus der „Dataset“-Liste. Füllen Sie anschließend die „X“- und „Y“-Werte mit Daten-Feldern nachfolgend gezeigt (die Felder können auch aus der aufklappbaren Liste ausgewählt werden):



Es ist nötig, hier zu beachten, dass die „X“-Werte von jedem Typ sein können (Zeichen zum Beispiel). Die „Y“-Werte müssen aus Zahlen bestehen. In unserem Fall (mit dem Kreisdiagramm) werden die „X“-Werte für die Beschriftung genutzt, während die „Y“-Werte für die Diagramm-Erstellung verwendet werden.

Schließen Sie die Einstellungen (drücke Sie „OK“ um den Editor zu schließen) und dann starten Sie den Report:



Was kann in diesem Report verbessert werden? Zuerst wäre es gut, die Werte in absteigender Reihenfolge zu sortieren. Öffnen Sie wieder den Diagramm-Editor und wählen die Art im oberen Teil des Fensters aus. Nun wählen Sie die gewünschte Sortierung in der Werkzeugeiste aus:

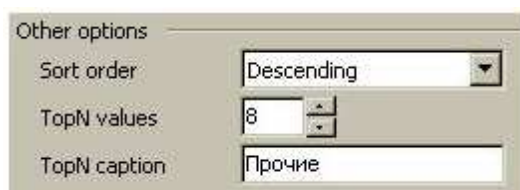


Wenn Sie nun den Report starten, werden Sie sehen, dass die Daten in der Legende geordnet sind.

Begrenzung der Anzahl der Diagramm-Werte

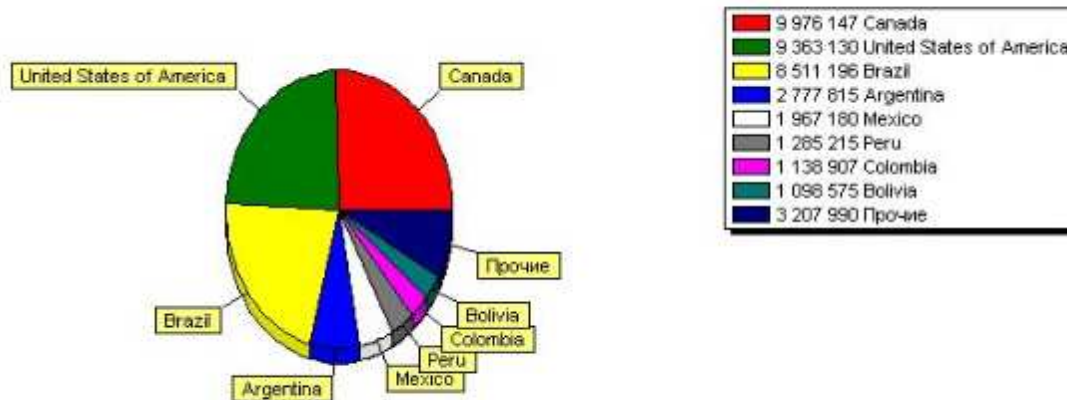
Unser Diagramm sieht eher überladen aus, da zu viele kleine Werte, die sowieso nicht sichtbar sind, im Diagramm enthalten sind. FastReport erlaubt das Begrenzen der Anzahl von Werten in einem Diagramm durch einen vordefinierten Wert. Demnach werden alle Werte, die nicht der Begrenzung entsprechen, als einen Einzelwert angezeigt, die die Summe der Werte darstellt, die nicht ins Diagramm gepasst haben.

In unserem Beispiel hat das Diagramm 18 Werte und nur 8 von ihnen können angezeigt werden. Öffnen Sie den Editor und setzen eine Begrenzung:



Die Begrenzung funktioniert, wenn der „TopN“-Wert nicht Null ist. Der Name ist die „TopN“-Beschriftung, welcher zum Summenwert angezeigt wird. Er sollte angegeben werden. Die Sortierung ist nicht bedeutend; die Werte werden bereits als Voreinstellung sortiert.

Das Ergebnis des Reports sieht folgendermaßen aus:



Einige nützliche Einstellungen

Probieren Sie verschiedene Einstellungen aus, die nützlich für das Aussehen des Diagramms sein können. Diese Einstellungen können nur im Objekt-Inspektor gesetzt werden.

Beim Auswählen des Diagramms aus der Liste links oben sind folgenden Basis-Eigenschaften wichtig:



- Gradient – Einstellung für die Hintergrund-Füllung des Gradienten. Aktivieren Sie die „Gradient.Visible“-Eigenschaft, um den Gradienten anzuzeigen.
- Legende – Einstellungen für das Aussehen einer erklärenden Tabelle. Die Tabelle können Sie mit Hilfe der „Legend.Visible“-Eigenschaft ausblenden (False). Die Tabellenposition wird mit der „Legend.Alignment“-Eigenschaft gesetzt.


Die folgenden Eigenschaften sind wichtig beim Auswählen einer Art:

- ColorEachPoint – färbt jeden Wert mit einer anderen Farbe.
- ExplodeBiggest – wählt den größten Wert aus (nur für die Art „Kreisdiagramm“).
- Marks - Einstellungen für die Erscheinung der erklärenden Bemerkungen.
- ValueFormat – die Zeile zum Formatieren von Werten.

Diagramm mit speziellen Werten

Im vorherigen Beispiel haben wir ein Diagramm auf der Basis der Daten von Datenbanktabellen entworfen. Es gibt einen weiteren Weg, um ein Diagramm zu erstellen:

Man gibt die nötigen Daten manuell ein. Dieser Weg ist günstig, wenn man kleine Diagramme erstellen möchte.

Wir zeigen an einem einfachen Beispiel, wie das funktioniert. Setzen Sie ein Diagramm auf die Reportseite und öffnen dessen Editor. Fügen Sie die Art „Balkendiagramm“  hinzu und stellen folgende Eigenschaften ein:

☒ Fixed data

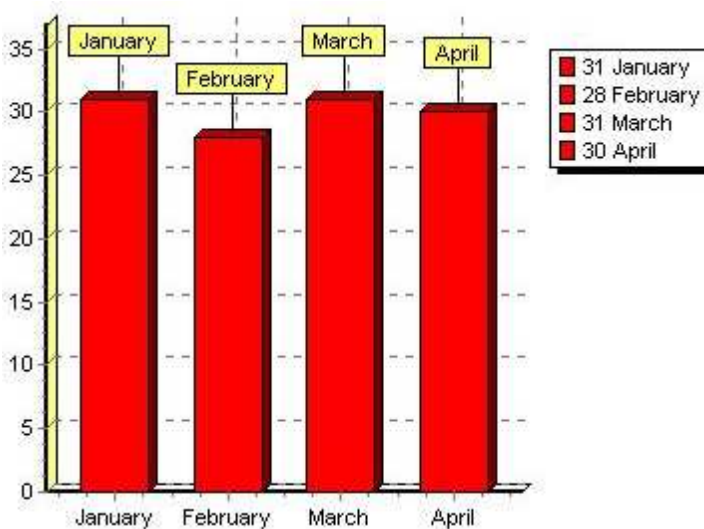
X Values (for example, a;b;c)

January;February;March;April

Y Values (for example, 1;2;3.5;5)

31;28;31;30

Das Ergebnis kann ebenso gut im Designer angesehen werden, ohne den Report zu starten:



Das „Script“

Ein „Script“ ist ein Programm, das in einer Programmiersprache geschrieben wird, die ein Teil eines Reports ist. Wenn ein Report gestartet wird, wird es ebenfalls geöffnet. Ein „Script“ ist in der Lage, Daten-Bearbeitungen durchzuführen, die über übliche Mittel des FastReports nicht durchzuführen sind. Zum Beispiel nutzlose Daten gemäß einer vordefinierten Bedingung zu verbergen. Das „Script“ wird außerdem genutzt, um Eigenschaften eines Dialog-Formulars, das eine Komponente des Reports ist, zu steuern.

Das „Script“ kann in einer der folgende Sprachen geschrieben werden:

- *Pascal*
- *C++*
- *Basic*
- *JScript*

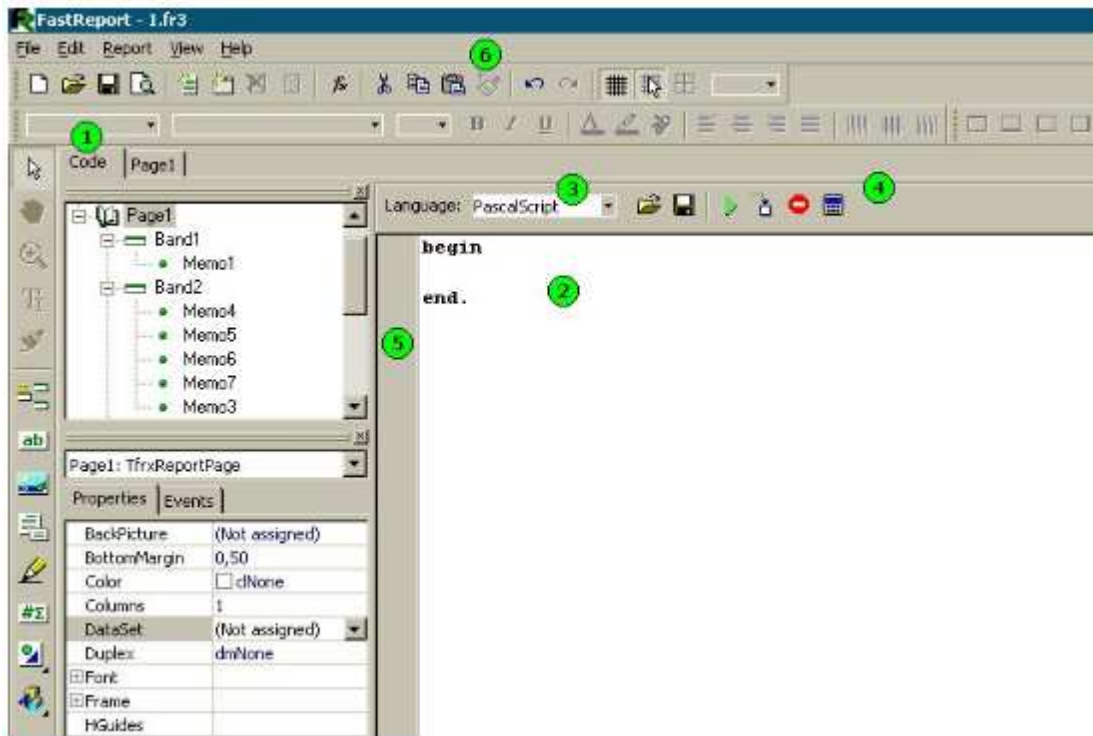
Die folgenden „Script“-Eigenschaften sind verfügbar:

- Standardsprachensatz:
Variablen, Konstanten, procedure, function (mit Verschachtelung) mit Variablen, Konstanten, voreingestellten Parametern, allen Standard-Operatoren (wie case, try, finally, except, with), Typen (Ganzzahl, Gleitkomma, logisch, Zeichen, Zeichenketten, mehrdimensionale Felder, set, variant), Klassen (mit Methoden, Ereignissen, Eigenschaften, Indexen, und voreingestellte Eigenschaften);
- Deklarationen der folgenden Typen fehlen:
records, Klassen im Script, Zeiger, sets (jedoch ist die Benutzung des 'IN' - „a in ['a'..'c'..'d']“-Operators möglich), shortstring-Typen, keine Sprünge (GOTO);
- Prüfen der Vereinbarkeit von Typen;
- Fähigkeit, auf das Objekt jedes Reports zuzugreifen.

Sie können „Scripts“ im FastReport-Designer entwerfen (Registerseite „Code“), der den „Script“-Editor mit der Syntax-Hervorhebung enthält. Außerdem gibt es einen eingebauten Debugger, der die folgenden Funktionen besitzt: „schrittweises Abarbeiten“, „Haltepunkt“, „arbeiten bis zur Cursorposition“ und „Auswerten“.





Beispiel eines „Scripts“

Werkzeuge, um mit dem „Script“ zu arbeiten, sind auf der „Code“-Registerseite des Reports angeordnet. Wenn Sie zu dieser Registerseite wechseln, erscheint der Designer wie folgt:







In der Abbildung oben bedeuten die Zahlen:

- 1 - „Code“-Registerseite;
- 2 – Fenster des „Script“-Editors;
- 3 – eine Auswahl der Sprachen, in der das „Script“ geschrieben werden soll;
- 4 – die Debugger-Werkzeuggestreife:

-  - startet den Report im Debuggermodus;
-  - führt die regelmäßige Codezeile aus (schrittweises Arbeiten);
-  - unterbricht die Arbeit des „Scripts“ (Haltepunkt);
-  - Anzeige der Auswertung von Ausdrücken (Auswerten).

- 5 – Lesezeichen und Haltepunkte werden in diesem Feld angezeigt; zusätzlich besitzen die Zeilen den ablauffähigen Code der hervorgehoben ist;
- 6 – Werkzeugleiste des Reports, die Sie ebenfalls im „Script“ verwenden können:

-  - Schneidet markierten Text aus;
-  - Kopiert markierten Text in die Windows-Zwischenablage;
-  - Fügt Text aus der Windows-Zwischenablage ein;
-  - macht den vorhergehenden Schritt rückgängig.

Tasten und -kombinationen, die Sie im „Script“ nutzen können.

Tasten	Bedeutung
Pfeiltasten	Bewegt den Mauszeiger
Bild hoch, Bild runter	Zur vorhergehenden / nächsten Seite springen
Strg+Bild hoch	Zum Anfang des Textes springen
Strg+Bild runter	Zum Ende des Textes springen
Pos1	Zum Anfang der Zeile springen
Ende	Zum Ende der Zeile springen
Enter	Zeilenumbruch
Entf	Löscht das Symbol neben der Zeigerposition; löscht den markierten Text
Backspace	Löscht das Symbol auf der linken Seite des Zeigers
Strg+Y	Löscht die aktuelle Zeile
Strg+Z	Macht die letzte Aktion rückgängig (bis zu 32 Ereignisse)
Shift+Pfeiltasten	Wählt einen Textblock aus
Strg+A	Wählt den gesamten Text aus
Strg+U	Bewegt den ausgewählten Block zwei Symbole nach links
Strg+I	Bewegt den ausgewählten Block zwei Symbole nach rechts
Strg+C, Strg+Einf	Kopiert den ausgewählten Block in die Windows-Zwischenablage
Strg+V, Shift+Einf	Fügt den Text aus der Windows-Zwischenablage ein
Str+X, Shift+Entf	Schneidet den ausgewählten Block aus und speichert ihn in der Windows-Zwischenablage
Strg+Shift+<Zahl>	Setzt ein Lesezeichen mit den Zahlen 0..9 in der aktuellen Zeile
Strg+<Zahl>	Springt zu den gesetzten Lesezeichen
Strg+F	Sucht einen Ausdruck
Strg+R	Ersetzt einen Ausdruck
F3	Wiederholt suchen / ersetzen von der Cursorposition aus
F4 oder F5	Setzt Haltepunkte und Punkte für „Arbeiten bis zum Cursor“
Strg+F2	Setzt das Programm zurück
Strg+F7	Gibt eine Vorschau der Variablenwerte (Auswerten)
F9	Startet das „Script“
F7 oder F8	Führt Codezeilen aus (schrittweise Abarbeitung)

Struktur eines Scripts

Die Scriptstruktur hängt von der Sprache ab, die Sie verwenden; auch wenn es übereinstimmende Elemente gibt, das sind der Scripttitel, Körper und die Hauptprozedur, die ausgeführt werden, wenn der Report startet. Unten stehen Beispiele des Scripts für alle vier unterstützten Sprachen:

Pascal- und C++-Scriptstruktur

```
#language PascalScript // optional
program MyProgram;      // optional

// the "uses" chapter should be located before any other chapter
uses 'unit1.pas', 'unit2.pas';

var                      // the "variables" chapter can be placed anywhere
  i, j: Integer;

const                   // "constants" chapter
  pi = 3.14159;

procedure p1;           // procedures and functions
var
  i: Integer;

  procedure p2;         // nested procedure
  begin
  end;

begin
end;

begin                   // main procedure.
end.
```

C++Script's structure:

```
#language C++Script    // optional

// the "include" chapter should be placed before any other chapter
#include "unit1.cpp", "unit2.cpp"

int i, j = 0;           // the "variables" chapter can be placed anywhere

#define pi = 3.14159    // "constants" chapter

void p1()               // functions
{                       // no nested procedures
}

{                       // main procedure.
}
```

Jscript- und Basic-Scriptstruktur

```
#language JScript      // optionally

// the "import" chapter should be before any other chapter
import "unit1.js", "unit2.js"

var i, j = 0;           // the "variables" chapter can be located
anywhere

function pl()           // functions
{                       //
}

                        // main procedure.
pl();
for (i = 0; i < 10; i++) j++;
```

BasicScript's structure:

```
#language BasicScript // optionally

// the "imports" chapter should be located before any other chapter
imports "unit1.vb", "unit2.vb"

dim i, j = 0            // the "variables" chapter can be placed anywhere

function pl()           // functions
{                       //
}

                        // main procedure.
for i = 0 to 10
    pl()
next
```

Später werden Sie Beispiele testen, die in der „Pascal“-Sprache geschrieben sind. Sobald ein neuer Report erstellt wird, ist diese Sprache als Voreinstellung ausgewählt.

„Hello, World!“-Script

Sie haben bereits ein Beispiel des „Hello, World!“-Reports getestet; schauen Sie sich nun an, wie man einen einfachen Report erstellt, der ein Fenster mit einer Begrüßung anzeigt.

Erstellen Sie einen leeren Report. Öffnen Sie den Designer und klicken auf die Schaltfläche „Neuer Report“, um in FastReport automatisch eine leere Dokumentenvorlage zu erstellen. Wechseln Sie zur „Code“-Registerseite und schreiben folgendes „Script“:

```
begin
    ShowMessage('Hello, World!');
end.
```

Danach starten Sie den Report. Wie wir erwartet haben, zeigt FastReport ein kleines Fenster mit einer Begrüßung:

```
begin
  ShowMessage('Hello, World!');
end.
```



Lassen Sie uns einige Details erklären. Sie erstellten ein „Script“, bestehend aus einem einzigen „begin..end“-Block. Demnach hat unser „Script“ eine sehr einfache Struktur; es besteht nur aus einer Hauptprozedur (siehe das Kapitel „Struktur eines Scripts“). Die Hauptprozedur wird durchgeführt, sobald der Report startet. In unserem Fall wird ein Begrüßungsfenster angezeigt; die Prozedur endet gleich nachdem das Fenster geschlossen wurde. Nachdem die Hauptprozedur beendet ist, beginnt die Reporterstellung.

Objekte im „Script“ verwenden

Sie können jedes Report-Objekt aus dem „Script“ ansteuern. Wenn es z.B. die „Page1“-Seite und das „Memo1“-Objekt gibt, können Sie diese im „Script“ nutzen, indem Sie ihre Namen aufrufen zum Beispiel:

```
Memo1.Color := clRed
```

Die Liste der Report-Objekte, die aus dem „Script“ verfügbar sind, werden im Fenster der „Report“-Struktur als Baumstruktur angezeigt. Welche Objekt-Eigenschaften sind im „Script“ verfügbar? Die Antwort ist einfach: Die im Objekt-Inspektor sichtbar sind. Zur selben Zeit gibt es am Fuß des Inspektors ein Hinweis, der sich auf die ausgewählte Eigenschaft bezieht. Beide Fenster (sowohl „Report“-Struktur, als auch Inspektor) sind während der Arbeit mit dem „Script“ verfügbar.

Demonstrieren wir das an einem Beispiel. Setzen Sie das „Text“-Objekt mit dem Namen „MyTextObject“ und dem Text „Test“ auf die Reportseite. Dann schreiben Sie in das Script:

```
begin
  MyTextObject.Color := clRed
end.
```

Starten Sie den Report und Sie sehen, dass unser Objekt rot geworden ist.

Die Variablen aus der Variablenliste des Reports

Sie können jede Variable, die in der Liste der Report-Variablen enthalten ist, aus dem „Script“ aufrufen (auch über den Menüdialog „Report“, „Variablen...“). Der Variablenname sollte demnach in spitze Klammern eingeschlossen werden:

```
if <my variable> = 10 then ...
```

Eine Alternative ist es, die „Get“-Funktion zu nutzen:

```
if Get('my variable') = 10 then ...
```

Veränderungen solcher Variablen-Werte können nur über die „Set“-Prozedur gemacht werden:

```
Set('my variable', 10);
```

Sie sollten die System-Variablen, wie „Page#“, in genau der selben Weise verwenden:

```
if <Page#> = 1 then ...
```

Aufruf von Datenbank-Feldern

Genau wie im Falle der Variablen müssen Sie spitze Klammern nutzen, um die Datenbank-Felder aufzurufen:

```
if <Table1."Field1"> = Null then...
```

Genauso gut können Sie auch die „Get“-Funktion nutzen (tatsächlich wird diese Funktion immer implizit genutzt, um Ausdrücke zu berechnen, die in spitze Klammern eingeschlossen sind).

Nutzen von Aggregat-Funktionen im „Script“

Im Gegensatz zu anderen Funktionen muss man die Aggregat-Funktionen („SUM“, „MIN“, „MAX“, „AVG“ und „COUNT“) entweder aufrufen, indem man spitze Klammern nutzt oder über die „Get“-Funktion (s. Kapitel „Eigenschaften des Aufrufs der Aggregat-Funktion“):

```
if <Sum(<Table1."Field1">, MasterData1)> > 10 then...
```

Eine weitere Eigenschaft der Aggregat-Funktionen ist, dass sie innerhalb des „Text“-Objekt benutzt werden müssen; Sie können sie im „Script“ anschließend aufrufen. Wenn die Aggregat-Funktion nur im „Script“ genutzt wird (ohne sie im „Text“-Objekt zu verwenden), erscheint eine Fehlermeldung. Das kommt daher, dass eine Aggregat-Funktion mit einem bestimmten Band verbunden sein muss und nur dann funktioniert sie richtig.

Den Variablen-Werte in einem Report anzeigen

Um den Inhalt einer „Script“-Variablen im Report anzuzeigen, müssen Sie diese Variable definieren und ihr einen Wert zuweisen. Hier ein einfaches Beispiel des „Scripts“:

```
var
  MyVariable: String;
begin
  MyVariable := 'Hello!';
end.
```

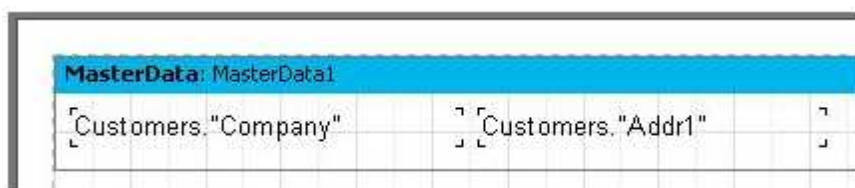
Der Variablen-Wert kann im „Text“-Objekt angezeigt werden, z.B. indem Sie die *[MyVariable]*-Zeile in ihm platzieren.

Ein Variablenname muss im Report einzigartig sein, was bedeutet, dass er nicht mit Report-Objektnamen, Standardfunktionen und Konstanten übereinstimmen darf. Falls es einen Fehler im „Script“ gibt, wird eine Meldung angezeigt und die Ausgabe des Reports wird angehalten.

Ereignisse

Wir haben bereits „Scripts“ mit nur einer Prozedur (Hauptprozedur) ausprobiert, die ausgeführt wird, wenn ein Report gestartet wird. In der Hauptprozedur können Sie jegliche Einstellungen durchführen, genauso wie Sie Variablen initialisieren können. Dennoch ist dies nicht genug, um eine Steuerung über die Report-Gestaltung zu haben. Um einen Report so gut wie möglich zu steuern, hat jedes Report-Objekt verschiedene Ereignisse, in denen ein Steuerungsprogramm eingreifen kann (d.h. eine Prozedur aus dem „Script“). Zum Beispiel im Steuerungsprogramm, das mit dem Daten-Band verbunden ist, können Sie Daten filtern, was bedeutet, dass z.B. das Band weg gelassen wird oder nur unter einer bestimmten Bedingung angezeigt wird.

Zeigen wir die Erstellung eines Reports und der Ereignisse, die während dieses Prozesses generiert werden, an dem Beispiel eines einfachen Reports, der eine Seite, ein „MasterData“-Band und zwei „Text“-Objekte auf dem Band enthält:



MasterData: MasterData1	
Customers."Company"	Customers."Addr1"

Wie gesagt, wird die Hauptprozedur am Anfang eines Reports aufgerufen. Danach beginnt der wesentliche Prozess der Report-Erstellung. Zum Start des Reports wird das „OnStartReport“-Ereignis des „Report“-Objekts aufgerufen. Bevor die Seite erstellt wird, wird das „OnBeforePrint“-Ereignis aufgerufen. Dieses Ereignis wird immer vor jeder Seite aufgerufen (es sollte nicht mit den Seiten eines bereits fertigem Report durcheinander gebracht werden!). In unserem Fall wird das Ereignis einmal aufgerufen, wenn das Report-Formular aus einer Seite besteht, ungeachtet der Seitenzahl bei der Ausgabe des Reports.

Dann beginnt die Ausgabe der Daten-Bänder. Es wird auf folgende Weise durchgeführt:

1. wird das „OnBeforePrint“- Ereignis aufgerufen;
2. werden die „OnBeforePrint“-Ereignisse aller Objekte, die zu einem Band gehören, aufgerufen;
3. werden alle Objekte mit Daten gefüllt (in unserem Fall mit Werten der „Company“ und der „Addr1“ Datenbank-Felder), danach wird das „OnAfterData“-Ereignis aller Objekte aufgerufen;
4. Aktionen wie die Positionierung der Objekte auf dem Band (wenn es streckbar ist) werden durchgeführt;
5. das „OnAfterCalcHeight“- Ereignis des Bandes wird aufgerufen;
6. eine neue Seite wird erstellt, falls das Band nicht genügend Platz findet;
7. das Band und all seine Objekte werden auf der beendeten Reportseite angezeigt;
8. das „OnAfterPrint“-Ereignis aller Objekte des Bandes wird aufgerufen;
9. das „OnAfterPrint“-Ereignis des Bandes selber wird aufgerufen.

Bänder werden solange ausgedruckt, wie Daten vorhanden sind, mit denen sie verbunden sind. Danach wird in unserem Fall das Erstellen des Reports angehalten; das Ereignis „OnAfterPrint“ der Reportseiten und letztendlich das „OnStopReport“-Ereignis des „Report“-Objekts werden aufgerufen.

Demnach kann man über das Verwenden von Ereignissen verschiedener Objekte praktisch jeden Moment der Erstellung des Reports steuern. Ein Schlüssel, um die Ereignisse korrekt zu verwenden, ist das vollständige Verständnis des Prozesses der Band-Ausgabe, welches in den nächsten Kapiteln beschrieben wird. So können die meisten Aktionen durchgeführt werden, indem man einzig das „OnBeforePrint“-Ereignis des Bandes nutzt; jegliche Veränderung, die am Objekt gemacht werden, wird simultan angezeigt. Trotzdem ist es in diesem Ereignis unmöglich, zu bestimmen, auf welcher Seite das Band ausgedruckt wird, wenn es streckbar ist, da die Berechnung der Höhe des Bandes im 4. Schritt durchgeführt wird. Dies kann über das „OnAfterCalcHeight“-Ereignis im 5. Schritt oder das „OnAfterPrint“-Ereignis im Schritt 8 durchgeführt werden, wobei im letzteren Fall das Band bereits ausgedruckt ist, wenn keine Operation mit Objekten etwas verändert. In einem Satz, Sie sollten deutlich sehen, zu welchem Zeitpunkt jedes der Ereignisse aufgerufen werden sollte und Sie sollten die Ereignisse nutzen, die zu den entsprechenden Objekten gehören.

Beispiel der Benutzung des „OnBeforePrint“-Ereignisses

Zeigen wir das eben gesagte in der Praxis. Erstellen Sie einen Report, der eine Liste der Kunden aufzeigt. Dieser Report soll nur Firmen enthalten, deren Namen mit dem Buchstaben „A“ beginnen.

Öffnen Sie den Report-Editor und erstellen einen Report mit folgendem Aussehen:

Table1:

DatabaseName = 'DBDEMOS'

TableName = 'customer.db'

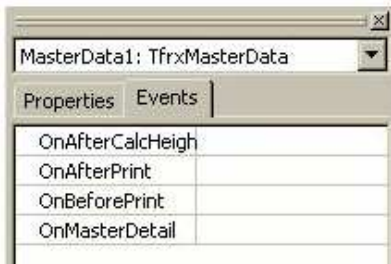
frxDBDataSet1:

DataSet = Table1

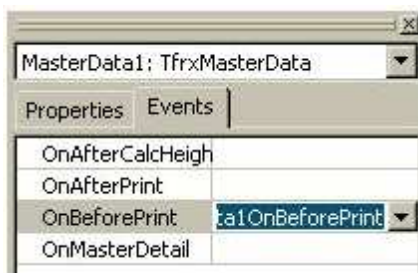
UserName = 'Customers'

MasterData: MasterData1		
[Customers."Company"]	[Customers."Contact"]	[Customers."Phone"]

Wählen Sie das Daten-Band aus und wechseln Sie zur Registerseite „Ereignisse“ im Objekt-Inspektor:



Um ein „Script“ für ein „OnBeforePrint“-Ereignis zu erstellen (das ist genau das, was Sie meistens benötigen), machen Sie einen Doppelklick auf das freie Feld rechts neben dem Ereignis-Namen:



Zur gleichen Zeit wird eine leere Prozedur zum „Script“ hinzugefügt und der Designer springt zur „Code“-Registerseite:

```

procedure MasterData1OnBeforePrint(Sender: TfrxComponent);
begin
  |
end;

begin
end.

```

Das einzigste, was Sie danach tun sollten, ist folgenden Quelltext in die Prozedur zu schreiben:

```

if Copy(<Customers."Company">, 1, 1) = 'A' then
  MasterData1.Visible := True else
  MasterData1.Visible := False;

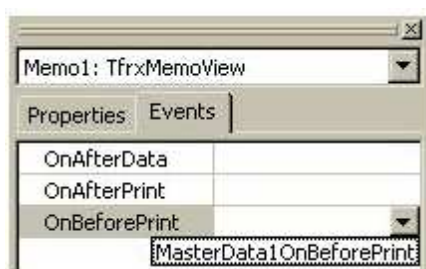
```

Öffnen Sie den Report und Sie werden sehen, dass das „Script“ korrekt funktioniert:

Action Club	Michael Spurling	813-870-0239
Action Diver Supply	Marianne Miles	22-44-500211
Adventure Undersea	Gloria Gonzales	011-34-09054
American SCUBA Supply	Lynn Cinciripini	213-654-0092
Aquatic Drama	Gillian Owen	613-442-7654

Lassen Sie uns verschiedenen Details erklären.

Sie können jede Ereignis-Routine zu verschiedenen Ereignissen verschiedener Objekte zuordnen. In diesem Fall definiert der Parameter „Sender“ das Objekt, welches das Ereignis initiiert hat. Um den Namen einer bereits existierenden Routine einem Ereignis zuzuordnen, kann man den Namen entweder manuell im Objekt-Inspektor eingeben oder ihn in der Auswahlliste aussuchen:



Die Zuweisung einer Routine kann einfach gelöscht werden. Um dies durchzuführen, wählen Sie die gewünschte Eigenschaft aus und drücken die „Entf“-Taste.

Die Gesamtsumme aus dem Gruppen-Fuß im Gruppen-Kopf ausgeben


Diese sehr häufig genutzte Methode erfordert die Benutzung des „Scripts“, da der Gesamtwert in einem herkömmlichen Report nur verfügbar ist, nachdem alle Gruppen-Daten ausgegeben wurden. Um eine Summe im Gruppen-Kopf anzuzeigen (bevor die Gruppe ausgegeben wurde) wird der folgende Algorithmus genutzt:

- die Eigenschaft „Zweipass“ des Reports wird eingestellt (Menü „Report“, „Optionen...“);
- im ersten Schritt wird die Summe jeder Gruppe berechnet und in einem Feld (Array) gespeichert;
- im zweiten Durchgang werden die Werte aus dem Feld genommen und im Gruppen-Kopf ausgegeben.

Es gibt zwei Wege, wie diese Aufgabe durchgeführt werden kann.

Zuerst erstellen Sie einen leeren Report, öffnen den Designer und verbinden die Datenquelle mit dem Report. Aktivieren Sie die Eigenschaft „Zweipass“ in den Report-Einstellungen (Menü „Report|Optionen...“). Fügen Sie zwei Bänder zum Report hinzu: „Group Header“ und „Master Data“. Stellen Sie im Editor des „Group Header“-Bandes die Bedingungen ein („Group.CustNo“-Datenfeld).

Verbinden Sie das Daten-Band mit der „Group“-Datenquelle und dann ordnen Sie die Objekte folgendermaßen an:

GroupHeader: GroupHeader1		
[Group."CustNo"]	[Group."Company"]	
MasterData: MasterData1		
[Group."OrderNo"]	[Group."SaleDate"]	[Group."ItemsTotal"]
GroupFooter: GroupFooter1		
[SUM(<Group."ItemsTotal">,MasterData1)]		

Die Summenwerte sollen in das mit Pfeil gekennzeichnete Objekt in der Abbildung (in unserem Beispiel heißt es „Memo8“) ausgegeben werden.

Der erste Weg

Sie nutzen die „TStringList“-Klasse als ein Array für die Speicherung der Summe. Die Werte werden in Form von Zeilen abgelegt. Gleichzeitig entspricht die erste Zeile der Liste der ersten Gruppe, usw. Die Integer-Variable „i“ (die jeweils erhöht wird, nachdem eine Gruppe ausgedruckt wurde) werden für die Berechnung der Anzahl der Gruppen genutzt.

Demnach sieht unser „Script“ wie folgt aus:

```
var
  List: TStringList;
  i: Integer;

procedure frReport1OnStartReport(Sender: TfrxComponent);
begin
  List := TStringList.Create;
end;

procedure frReport1OnStopReport(Sender: TfrxComponent);
begin
  List.Free;
end;

procedure Page1OnBeforePrint(Sender: TfrxComponent);
begin
  i := 0;
end;

procedure GroupHeader1OnBeforePrint(Sender: TfrxComponent);
begin
  if Engine.FinalPass then
    Memo8.Text := 'Sum: ' + List[i];
end;

procedure GroupFooter1OnBeforePrint(Sender: TfrxComponent);
begin
  List.Add(FloatToStr(<SUM(<Group."ItemsTotal">,MasterData1)>));
  Inc(i);
end;

begin
end.
```

Wenn Sie auf die Namen der Prozeduren schauen, können Sie leicht herausfinden, welche Ereignisse Sie verwendet haben. Es sind: "ReportOnStart", "Report.OnStopReport", "Page1.OnBeforePrint", "GroupHeader1.OnBeforePrint" und "GroupFooter1.OnBeforePrint". Die ersten beiden Ereignissen werden aufgerufen, wie bereits vorher beschrieben, am Anfang bzw. am Ende des Reports. Um „Scripts“ für diese Ereignisse zu erstellen, müssen Sie das "Report"-Objekt auswählen, um seine Eigenschaften im Objekt-Inspektor anzeigen zu lassen. Danach verfahren Sie wie bekannt: Klicken auf die Registerseite „Ereignis“ und erstellen die „Scripts“.

Warum haben wir nicht die Hauptprozedur für die Erstellung der Liste „List“ genutzt und sie im "OnStartReport"-Ereignis ausgeführt? Das haben wir gemacht, weil das erstellte Objekt, nachdem der Report beendet ist, entfernt werden sollte. Es ist logischer, Objekte im "OnStartReport"-Ereignis zu erstellen und sie über das "OnStopReport"-Ereignis zu entfernen. In anderen Fällen (wenn der Speicher nicht geleert werden muss) können Sie die Hauptprozedur verwenden, um Variablen zu initialisieren.

Alles, was die Erstellung und Entfernung des "List"-Objekts betrifft, scheint ziemlich offensichtlich zu sein. Probieren Sie nun die Arbeit des „Scripts“ aus. Am Anfang der Seite wird der Zähler der aktuellen Gruppe (die "i"-Variable) auf "0" zurückgesetzt und nach dem Ausdruck jeder Gruppe (im "GroupFooter1.OnBeforePrint"-Ereignis) um „1“ erhöht (inc(i);) und der berechnete Summenwert in die Liste „List“ hinzugefügt. Das "GroupHeader1.OnBeforePrint"-Ereignis wird während des ersten Durchlaufs nicht aufgerufen (die "Engine.FinalPass"-Abfrage). Während des zweiten Durchgangs (wenn die „List“-Liste mit Werten gefüllt ist) wird der Wert, der zur aktuellen Gruppe gehört, in diesem Ereignis in das Objekt „Memo8“ gespeichert, welches die Gesamtsumme der Gruppe anzeigt. In einem fertigen Report sieht es folgendermaßen aus:

1221	Kauai Dive Shoppe	Sum: 51450,8
1023	01.07.88	4 674,00
1076	16.12.94	17 781,00
1123	24.08.93	13 945,00
1169	06.07.94	9 471,95
1176	26.07.94	4 178,85
1269	16.12.94	1 400,00
		51 450,80

Wie Sie sehen, ist der Algorithmus ziemlich einfach. Trotzdem geht es noch einfacher.

Der zweite Weg

Sie nutzen die Liste der Report-Variablen als ein Array zum Ablegen der Summen. Wie Sie sich erinnern, bekommt man einen Bezug zu solchen Objekten über die „Get“ und „Set“-Funktionen. Das schützt Sie vor Schwierigkeiten, ein Objekt zu erstellen und wieder freizugeben, wenn es überflüssig wird. Unser „Script“ sieht wie folgt aus:

```
procedure GroupHeader1OnBeforePrint(Sender: TfrxComponent);
begin
  if Engine.FinalPass then
    Memo8.Text := 'Sum: ' + Get(<Group."CustNo">);
end;

procedure GroupFooter1OnBeforePrint(Sender: TfrxComponent);
begin
  Set(<Group."CustNo">,
    FloatToStr(<SUM(<Group."ItemsTotal">,MasterData1)>));
end;

begin
end.
```

Wie Sie sehen, wurde das „Script“ etwas vereinfacht. Eine Anweisung in der „GroupFooter1.OnBeforePrint“-Routine setzt einen Variablen-Wert mit einem Namen, der gleich der Kunden-Nummer ist (<Group."CustNo">). Sie können jeglichen Namen benutzen, der eindeutig den Kunden identifiziert, z.B. seinen Namen <Group."Company">. Wenn es solch eine Variable nicht gibt, wird sie erstellt; wenn es sie gibt, wird ihr Wert verändert. In der „GroupHeader1.OnBeforePrint“-Routine wird der Variablen-Wert mit der aktuellen Gruppen-Nummer im Objekt „Memo8“ gespeichert.

„OnAfterData“-Ereignis

Dieses Ereignis wird aufgerufen, nachdem das Report-Objekt mit den Daten gefüllt ist, mit denen es verbunden ist. Es ist günstig, dieses Ereignis zu nutzen, um entweder den Wert eines Datenbankfeld zu erhalten oder den Wert des Ausdrucks, der im Objekt steht. Es ist so, dass dieser Wert an die Variable „Value“ übergeben wird, die als einzigste in diesem Ereignis verfügbar ist.

Wenn Sie zwei „Text“-Objekte mit dem Inhalt [Table1.“Field1“] und [<Table2.“Field1“>+10] haben, ist es vorteilhaft, die Variable „Value“ auszuwerten:

```
if Value > 3000 then
  Memo1.Color := clRed
```

anstatt Folgendes zu schreiben:

```
if <Table1.“Field1“> > 3000 then
  Memo1.Color := clRed
```

Außerdem erhalten Sie durch das Verwenden der Variablen „Value“ anstelle eines Ausdrucks, die Möglichkeit eine mehrfach nutzbare Routine zu schreiben und das Ereignis „OnAfterData“ mit verschiedenen Objekten zu verbinden.

Noch etwas ist zu beachten. Wenn es verschiedene Ausdrücke in einem Objekt gibt (z.B. [expr1] [expr2]), wird der Wert des letzten Ausdrucks in die „Value“-Variable übertragen.

weitere zur Verfügung stehende Objekte

Zusätzlich zu den Objekten, die im Report enthalten sind (Seiten, Bänder, „Text“ und andere Objekte), sind einige Objekte im „Script“ verfügbar, die eventuell beim Steuern der Report-Erstellung von Nutzen sein können. Das „Engine“-Objekt, welches Sie im vorhergehenden Kapitel genutzt hatten, bezieht sich auf diese Art Objekte. Die Liste dieser Objekte ist unten aufgeführt:

- Report – das „Report“-Objekt;
- Engine – die Verbindung zur Report-Vorschau;
- Outline – die Verbindung zur „Report-Baumstruktur“-Element in einem Vorschau-Fenster.

Probieren Sie jedes dieser Objekte aus.

„Report“-Objekt

Es stellt eine Verbindung zum aktuellen Report dar. Die Eigenschaften dieses Objekts kann man sehen, wenn man das „Report“-Element im „Reportstruktur“-Fenster auswählt hat.

Methoden:

Methode	Beschreibung
function Calc (const Expr: String): Variant	Gibt den „Expr“-Wert zurück, z.B. wird bei Report.Calc('1+2') „3“ zurückgegeben. Jeder Ausdruck, der im Sinne des FastReports korrekt ist, kann einen Wert berechnen.
Function GetDataSet (const Alias: String): TfrxDataSet	Gibt ein „Dataset“ (z.B. eine Tabelle) mit dem angegeben „Alias“ zurück. Das „Dataset“ sollte in die Liste der Report-Daten eingefügt werden (Menüdialog „Report Data...“).

„Engine“-Objekt

Das ist das wohl nützlichste und interessanteste Objekt, welches eine Verbindung zur Report-Vorschau darstellt (FastReport-Kern, der die Report-Erstellung verwaltet). Mit der Benutzung von „Engine“-Eigenschaften und -Methoden kann man wirklich exotische Reports erstellen. Probieren Sie Methoden und Eigenschaften dieses Objekts aus.

Eigenschaften	Typ	Beschreibung
CurColumn	Integer	Die Anzahl der aktuellen Spalten in einem mehrspaltigen Report. Dieser Eigenschaft kann ein Wert zugewiesen werden.
CurX	Extended	Der Wert der Koordinate auf der X-Achse. Dieser Eigenschaft kann ein Wert zugewiesen werden.
CurY	Extended	Der Wert der Koordinate auf der Y-Achse. Dieser Eigenschaft kann ein Wert zugewiesen werden.
DoublePass	Boolean	Gleich „True“, falls der Report ein „Zweipass“-Report ist. Analog zu Report.EngineOptions.DoublePass.
FinalPass	Boolean	Gleich „True“, wenn der letzte Durchgang des „Zweipass“-Reports durchgeführt wird.
PageHeight	Extended	Höhe der ausdruckbaren Fläche in Pixel.
PageWidth	Extended	Breite der ausdruckbaren Fläche in Pixel.
StartDate	TDateTime	Datum der Report-Erstellung. Das Gegenstück zur <Date>-Systemvariablen
StartTime	TDateTime	Zeit der Report-Erstellung. Das Gegenstück zur <Time>-Systemvariablen
TotalPages	Integer	Die Anzahl der Seiten in einem Report. Ein Gegenstück zur <TotalPages>-Variable. Der Report muss ein „Zweipass“-Report sein, wenn diese Variable genutzt werden soll.

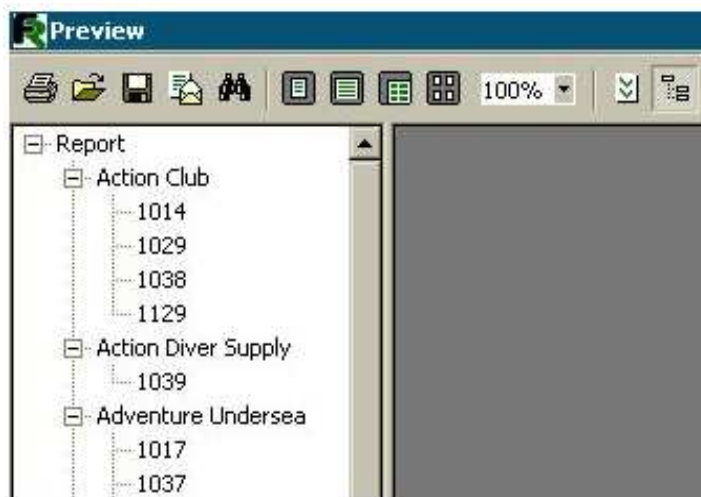
Methoden:

Methode	Beschreibung
Procedure AddAnchor (const Text: String)	Es fügt einen „Festpunkt“ in die Liste der Festpunkte hinzu. Mehr dazu weiter unten.
Procedure NewColumn	Erstellt eine neue Spalte in einem mehrspaltigem Report. Nach der letzten Spalte wird ein Seitenumbruch automatisch eingeführt.
Procedure NewPage	Erstellt eine neue Seite (Seitenumbruch).
Procedure ShowBand(Band: TfrxBand)	Zeigt ein Band mit dem spezifizierten Namen an. Nachdem ein Band angezeigt wurde, springt die „CurY“-Position automatisch weiter.
Function FreeSpace: Extended	Liefert den Wert des freien Platzes auf der Seite in Pixel.

Methode	Beschreibung
Function GetAnchorPage(const Text: String): Integer	Setzt die Nummer der Seite zurück, in der der spezielle „Festpunkt“ gesetzt wurde.

„Outline“-Objekt

Dieses Objekt stellt das „Reportstruktur“-Element in einem Vorschau-Fenster dar.



Dieses Element zeigt eine baumartige Struktur eines fertigen Reports. Wenn Sie auf ein Element klicken, springen Sie zu der Seite, die mit dieser Markierung verbunden ist.



Um den Baum anzeigen zu lassen, müssen Sie ihn aber erst einmal über die -Schaltfläche in der Werkzeugleiste erstellen oder ihn über die Eigenschaft „Report.PreviewOption.OutlineVisible=True“ angeben. Die Breite des Element in Pixel kann hier ebenfalls gesetzt werden: Report.PreviewOptions.OutlineWidth.

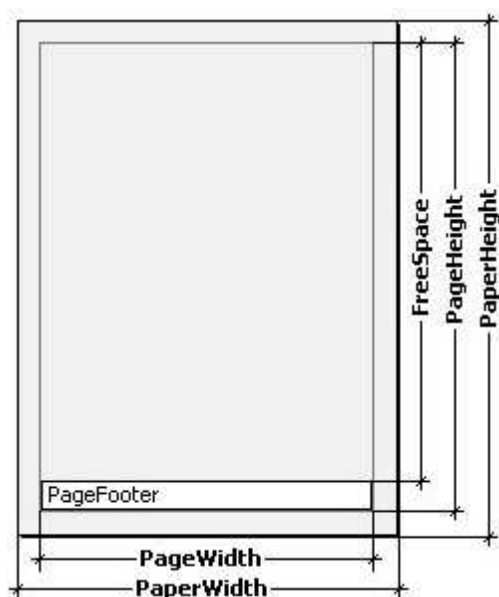
Probieren Sie die Methoden dieses Objekts aus.

Methode	Beschreibung
Procedure AddItem(const Text:String)	Fügt ein Element mit dem Namen „Text“ an die aktuelle Position der „Reportstruktur“. Die derzeitige Reportseite und die aktuelle Position auf der Seite wird mit dem Element verbunden.
Procedure LevelRoot	Setzt die aktuelle Position der „Reportstruktur“ auf die Haupt-Ebene.
Procedure Levelup	Setzt die aktuelle Position der „Reportstruktur“ eine Ebene höher.

Benutzung des „Engine“-Objekts

Wir haben bereits erwähnt, dass das „Engine“-Objekt den Verwalter darstellt, der die Report-Erstellung steuert. Mit der Benutzung der Eigenschaften und Methoden können Sie den Prozess der Anordnung der Bänder auf einer Seite verwalten. Gehen wir erstmal theoretisch ran.

Die Abbildung unten zeigt die Reportseite und die Namen der Eigenschaften, die verschiedene Werte annehmen.



Die Seite hat die „Pagewidth“- und „PageHeight“-Parameter als physikalische Größen. Diese Werte entsprechen den Eigenschaften der Seite mit den selben Namen im Objekt-Inspektor bei der Auswahl einer Seite. Die Größe einer A4-Seite ist 210X297 mm.

Die „PageWidth“- und „PageHeight“-Parameter bestimmen die Ausmaße der bedruckbaren Fläche, die meist geringer als die physikalische Größe der Seite ist. Die Größe der bedruckbaren Fläche wird durch die Seitenränder bestimmt, die von den Eigenschaften Report-Seite wie „LeftMargin“, „TopMargin“, „RightMargin“, „BottomMargin“ abhängig sind. Die Größe der bedruckbaren Fläche wird durch die „Engine.PageWidth“- und „Engine.PageHeight“-Eigenschaften gesetzt.

Letztendlich bestimmt der „FreeSpace“-Parameter die Höhe freien Platzes auf der Seite. Wenn es ein „Page Footer“-Band auf der Seite gibt, wird seine Höhe beim Berechnen des freien Platzes berücksichtigt. Dieser Parameter wird durch die „Engine.FreeSpace“-Funktion in Pixel angegeben. Sie werden bemerken, dass nachdem das nächste Band auf der Seite angezeigt wird, sich der freie Platz auf der Seite reduziert und das ist das, was während der Berechnung des Freiraums in Frage kommt.

Wie formatieren Sie fertige Report-Seiten? FastReport druckt Bänder auf der Seite solange Platz ist. Wenn kein Platz mehr ist, wird das „Page Footer“-Band ausgedruckt (falls es verfügbar ist) und eine neue leere Seite wird erstellt. Wie bereits gesagt, verringert sich die Höhe des freien Platzes, wenn das nächste Band angezeigt wird. Außerdem beginnt das Anzeigen des nächsten Bandes von der aktuellen Position an, die durch Koordinaten der X- und Y-Achse bestimmt ist. Diese Position wird entsprechend der „Engine.CurX“- und „Engine.CurY“-Eigenschaften gesetzt. Nach dem Ausdrucken des nächsten Bandes steigt die CurY-Position automatisch mit der Höhe der ausgedruckten Bänder. Nachdem eine neue Seite erstellt wurde, ist der „CurY“-Wert gleich „0“. Die „CurX“-Position wird verändert, wenn Sie einen mehrspaltigen Report ausdrucken.

Die „Engine.CurX“- und „Engine.CurY“-Eigenschaften sind nicht nur zum Lesen verfügbar, sondern auch zum Ändern. Das bedeutet, dass Bänder manuell verschoben werden können, indem man eines der dazugehörigen Ereignisse nimmt. Zum Beispiel, wenn Sie einen Report wie im Beispiel unten haben,

MasterData: MasterData1		
[Customers."Company"]	[Customers."Contact"]	[Customers."Phone"]

kann er wie folgt ausgedruckt werden:

Action Club	Michael Spurling	813-870-0239
Action Diver Supply	Marianne Miles	22-44-500211
Adventure Undersea	Gloria Gonzales	011-34-09054
American SCUBA Supply	Lynn Cinciripini	213-654-0092
Aquatic Drama	Gillian Owen	613-442-7654
Blue Glass Happiness	Christine Taylor	213-555-1984

Das ist das Ergebnis eines „Scripts“ im „OnBeforePrint“-Ereignis des Bandes:

```
procedure MasterData1OnBeforePrint(Sender: TfrxComponent);
begin
    Engine.CurX := Engine.CurX + 5;
end;
```

Durch Manipulation der „CurY“-Eigenschaft kann man Bänder zum Beispiel ineinander ausdrucken:

Action Club	Michael Spurling	813-870-0239
Action Diver Supply	Marianne Miles	227-44-5802
Adventure Undersea	Gloria Gonzales	611-34-0906
American SCUBA Supply	Lynn Cincipini	213-654-0092
Aquatic Drama	Gillian Owen	613-445-7884
Blue Glass Happiness	Christine Taylor	401-609-7623
Blue Jack Aqua Center	Ernest Barratt	401-609-7623
Blue Sports Club	Theresa Kuncic	619-667-8604

Das dazugehörige „Script“:

```
procedure MasterData1OnBeforePrint(Sender: TfrxComponent);
begin
    Engine.CurY := Engine.CurY - 15;
end;
```

Die „Engine.NewPage“-Methode ermöglicht es, dass die Seiten an beliebiger Stelle des Reports umbrechen. Gleichzeitig wird das Ausdrucken einer neuen Seite fortgeführt. Demnach kann man in unserem Beispiel einen Seitenumbruch nach dem Ausdruck der zweiten Zeile einlegen:

```
procedure MasterData1OnAfterPrint(Sender: TfrxComponent);
begin
    if <Line> = 2 then
        Engine.NewPage;
end;
```

Beachten Sie, dass wir es nun im „OnAfterPrint“-Ereignis durchführen (das ist so zu sagen nachdem das Band bereits gedruckt wurde). Wir möchten Sie darauf aufmerksam machen, dass die „Line“-Variable auf der neuen Seite zurücksetzt wird.

Die „Engine.NewColumn“-Methode bricht eine Spalte in einen mehrspaltigen Report um. Sobald keine Spalte mehr übrig ist, erstellt diese Methode eine neue Seite.

Anker (Anchors)

„Anchor“ ist eins der Elemente, die es ermöglicht, zu einem Element zu springen, das mit dem fertigen Report-Objekt verbunden ist, indem Sie darauf klicken (im Vorschau-Fenster).

„Anchor“ wird über die „Engine.AddAnchor“-Methode gesetzt. „Anchor“ hat einen Namen, der der Position der Seitenzahl auf der Seite entspricht. Um zu einem „Anchor“ mit einem speziellen Namen zu springen setzen Sie folgende Zeile in die URL-Eigenschaft eines Report-Objekts:

```
#AnchorName  
or  
#[AnchorName]
```

Im letzteren Fall wird FastReport die eckigen Klammern des Ausdrucks erweitern.

Ein Klick auf dieses Objekt führt einen Sprung zum Teil des Reports, auf dem der „Anchor“ hinzugefügt wurde.

Es ist üblich, Anchors bei der Erstellung des „Contents“(Inhalt)-Kapitel mit Verbindungen zu entsprechenden Kapiteln zu nutzen. Verdeutlichen Sie sich das durch folgendes Beispiel. Um dies durchzuführen, benötigen Sie die „Customer.db“-Tabelle, die Ihnen ja bereits bekannt ist.

Ihr Report soll ein zweiseitiger Report sein (der zwei Seiten im Designer erfordert). Sie platzieren das „Contents“-Kapitel auf der ersten Seite und die Kundenliste auf der Zweiten. Mit dem Klick auf eine Zeile des „Contents“-Kapitel wird ein Sprung zu dem entsprechenden Report-Element ausgeführt.

Die erste Seite:



Setzen Sie folgende Zeile in die URL-Eigenschaft des „Text“-Objekts, welches zum Daten-Band gehört

```
#[Customers. "Company"]
```

und setzen die Eigenschaft der Schrift: blaue Farbe und unterstrichen, um die „Hyperlink“-Darstellung zu simulieren.

Die zweite Seite:

ReportTitle: ReportTitle2			
Customers			
PageHeader: PageHeader1			
Company	Address	Contact	Phone
MasterData: MasterData2			
Customers."Company"	Customers."Addr1"	Customers."Contact"	Customers."Ph

Um einen „Anchor“ hinzuzufügen, tippen wir „MasterData2.OnBeforePrint“ in das „Script“ des Bandes:

```
procedure MasterData2OnBeforePrint(Sender: TfrxComponent);
begin
    Engine.AddAnchor(<Customers."Company">);
end;
```

Das ist alles, was Sie benötigen. Beim Starten des Reports sollten Sie prüfen, ob unsere „Hyperlinks“ funktionieren.

Das Letzte, was erwähnt werden sollte, ist die „Engine.GetAnchorPage“-Funktion. Diese Funktion liefert die Nummer der Seite, auf der der entsprechende „Anchor“ hinzugefügt wurde. Diese Funktion ist ebenfalls nützlich beim Erstellen des „Contents“-Kapitels. Der Report muss ein „Zweipass“-Report sein; ansonsten kann die Funktion nicht genutzt werden.

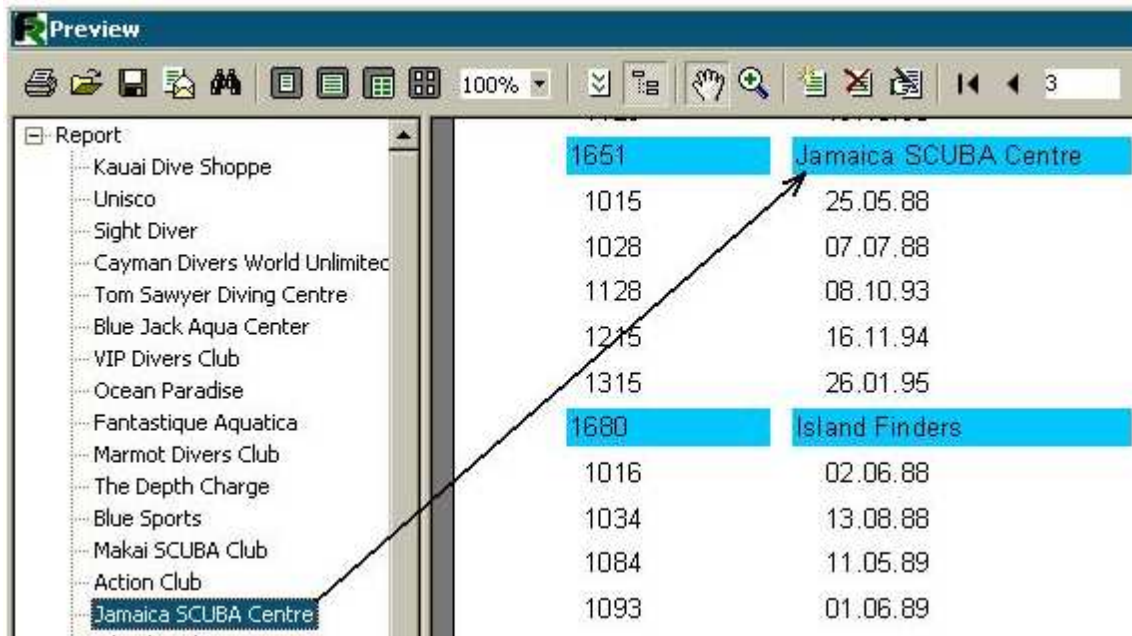
Die Benutzung von „Outline“-Objekten

Das „Outline“-Objekt stellt, wie bereits gesagt, eine Baumstruktur des Reports dar, der in einem Vorschau-Fenster angezeigt werden kann. Durch Klicken auf eines der Elemente des Baumes springen Sie zu der Reportseite, welche mit dem Element verbunden ist. Es ist nicht nötig das „Script“ zu nutzen, um mit dem „Outline“-Objekt zu arbeiten, da einige Bänder einen Mechanismus haben, der es ermöglicht, automatisch einen Baum zu erstellen. Probieren Sie zwei Beispiele, wie das „Outline“-Objekt mit der Hilfe von Bändern und dem „Script“ genutzt werden kann.

Fast alle Bänder haben die „OutlineText“-Eigenschaft, in die ein Ausdruck gesetzt werden kann und das hilft, automatisch einen neuen Baum zu erstellen. Der Ausdruck wird berechnet, wenn man einen Report erstellt und sein Wert wird dem Baum hinzugefügt, wenn das Band ausgedruckt wird. Demnach ist die Hierarchie der Elemente im Baum ähnlich der Hierarchie der Bänder in einem Report. Das bedeutet, dass es im Baum Haupt- und Unter-Elemente gibt, die den Haupt- und Detail-Bändern im Report entsprechen (ein Report mit zwei Datenebenen oder mit Gruppen kann den Punkt vereinfachen). Wir stellen den Prozess der Operation mit einem Baum am Beispiel des Reports mit Gruppen dar, den wir es in einem vorangegangenen Kapitel beschrieben hatten.

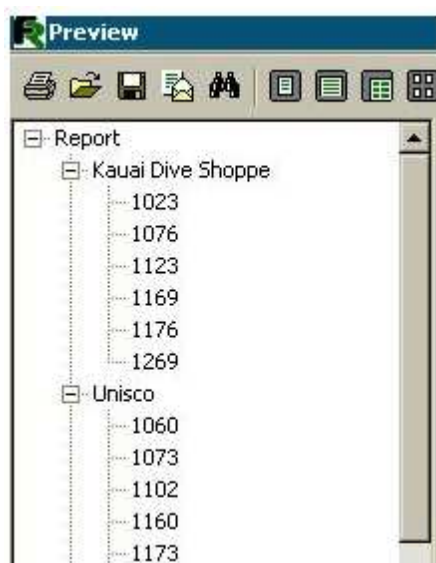
GroupHeader: GroupHeader1		Group."CustNo"	
Group."CustNo"	Group."Company"		
MasterData: MasterData1		Group	
Group."OrderNo"	Group."SaleDate"		

Setzen Sie einen Wert für die Eigenschaft „GroupHeader1.OutlineText“ des Bandes wie z.B. „<Group.\"Company\">“ fest. Um den Baum automatisch öffnen zu lassen, sobald sich das Vorschau-Fenster öffnet, müssen Sie die Eigenschaft „Report.PreviewOptions.OutlineVisible=True“ einstellen. Beim Starten des Reports werden Sie folgendes sehen:



Mit dem Klick auf ein Element des Baumes wird ein Sprung zur entsprechenden Reportseite ausgeführt und als Ergebnis erscheint das ausgewählte Element oben im Fenster.

Fügen Sie die zweite Ebene zum Report-Baum hinzu. Um dies durchzuführen, reicht es die Eigenschaft „MasterData.OutlineText“ des Bandes auf „<Group.\"OrderNo\">“ zu setzen. Dann sieht der Baum wie folgt aus:



Wie Sie vielleicht bemerkt haben, ist die Navigation ebenfalls in der Bestell-Nummer möglich und die Hierarchie der Baum-Elemente ähnelt der Hierarchie des Reports.

Zeigen wir nun, wie Sie einen analogen Baum über das „Script“ erstellen, ohne die „OutlineText“-Eigenschaft zu nutzen. In unserem Report müssen sie die „OutlineText“-Eigenschaften beider Bänder leeren und zwei Prozeduren erstellen: „GroupHeader1.OnBeforePrint“ und „MasterData1.OnBeforePrint“:

```
procedure GroupHeader1OnBeforePrint(Sender: TfrxComponent);
begin
    Outline.LevelRoot;
    Outline.AddItem(<Group."Company">);
end;

procedure MasterData1OnBeforePrint(Sender: TfrxComponent);
begin
    Outline.AddItem(<Group."OrderNo">);
    Outline.LevelUp;
end;

begin
end.
```

Überzeugen Sie sich beim Starten eines Reports, dass er auf die selben Weise funktioniert, wie der vorige, in dem der Baum automatisch erstellt wurde. Untersuchen wir, wie ein Baum erstellt wird.

Die „Outline.AddItem“-Methode fügt einen Unter-Eintrag in den aktuellen Baum ein und macht dann den Unter-Eintrag zum Aktuellen. Demnach würden Sie, falls „AddItem“ mehrere Male in einer Reihe aufgerufen wird, eine „Leiter“ wie unten gezeigt erhalten:

```
Item1
  Item2
    Item3
    ...
```

Die Methoden „LevelUp“ und „LevelRoot“ des „Outline“-Objekts werden genutzt, um das aktuelle Element zu steuern. Die Erste bewegt den Cursor zu einem Element, welches auf einer höheren Ebene platziert ist. So erstellt das „Script“

```
Outline.AddItem('Item1');
Outline.AddItem('Item2');
Outline.AddItem('Item3');
Outline.LevelUp;
Outline.AddItem('Item4');
```

einen Baum wie diesen:

```
Item1
  Item2
    Item3
    Item4
```

Das bedeutet, dass das „Item4“-Element ein Unter-Element im Vergleich zum „Item2“-Element ist. Die „LevelRoot“-Methode schiebt das aktuelle Element auf die oberste

Ebene des Baumes. Das „Script“

```
Outline.AddItem('Item1');
Outline.AddItem('Item2');
Outline.AddItem('Item3');
Outline.LevelRoot;
Outline.AddItem('Item4');
```

erstellt einen Baum, wie den unten:

```
Item1
  Item2
    Item3
  Item4
```

Dank dieser Erklärung wird es klar, wie unser Report arbeitet. Jedes Mal wenn ein „Group“-Element ausdruckt wird, wird die oberste Ebene des Baumes das aktuelle Element, wenn ein Firmen-Name hinzugefügt wird. Danach wird die Liste der Bestellungen ausgegeben und jede Bestellung wird als Unter-Element der Firmen hinzugefügt. Um die Anzahl der Bestellungen auf einer Ebene zu platzieren und sie nicht als „Leiter“ anzeigen zu lassen, wird der Übergang zu einer höheren Ebene über die „Outline.LevelUp“-Methode im „Script“ durchgeführt.

„OnManualBuild“-Ereignis der Seite

Der FastReport-Kern ist eigentlich für die Erstellung des Reports zuständig. Er zeigt Report-Bänder in einer definierten Reihenfolgen an, so oft wie es Daten gibt, um so einen fertigen Report zu erstellen. Manchmal ist es nötig, einen Report in der Form darzustellen, die der FastReport-Kern nicht erstellen kann. In diesem Fall kann man eine Möglichkeit verwenden, um den Report über das „OnManualBuild“-Ereignis, das auf der Reportseite enthalten ist, manuell zu erstellen. Wenn die Prozedur für dieses Ereignis erzeugt wurde, dann wird der FastReport-Kern beim Erstellen eines Reports die Steuerung an diese übergeben. Zugleich zeigt der FastReport-Kern auch automatisch die Bänder, die auf der Seite platziert sind, wie den „Report Title“, „Page Title“, „Column Title“, „Report Footer“, „Page Footer“, „Column Footer“ und „Background“. Der Kern steuert außerdem die Erstellung neuer Seiten und Spalten. Die Aufgabe der Prozedur im Ereignis „OnManualBuild“ ist es, Daten-Bänder und ihre Titel- und Fußzeilen in bestimmter Reihenfolge anzuzeigen.

Das heißt, die Prozedur im „OnManualBuild“-Ereignis gibt einen Befehl für das Anzeigen von bestimmten Bändern an den FastReport-Kern. Der Kern macht den Rest von selbst: er erstellt eine neue Seite, sobald kein freier Platz mehr auf der jetzigen Seite ist; führt die Scripts“ aus, die dem Ereignis hinzugefügt wurden usw.

Verdeutlichen wir die Prozedur an einem einfachen Beispiel. Im Report gibt es zwei „Master Data“-Bänder, die nicht mit Daten verbunden sind:

ReportTitle: ReportTitle1	
[OnManualBuild test]	
MasterData: MasterData1	
MasterData1	
MasterData: MasterData2	
MasterData2	
PageFooter: PageFooter1	

Die Prozedur soll diese Bänder in abwechselnder Reihenfolge anzeigen (jedes sechs Mal). Nachdem sechs Bänder erstellt wurden, soll eine kleine Lücke eingefügt werden.

```
procedure Page1OnManualBuild(Sender: TfrxComponent);
var
  i: Integer;
begin
  for i := 1 to 6 do
  begin
    { displaying bands one by one }
    Engine.ShowBand(MasterData1);
    Engine.ShowBand(MasterData2);
    { insert a small gap }
    if i = 3 then
      Engine.CurY := Engine.CurY + 10;
  end;
end;
```

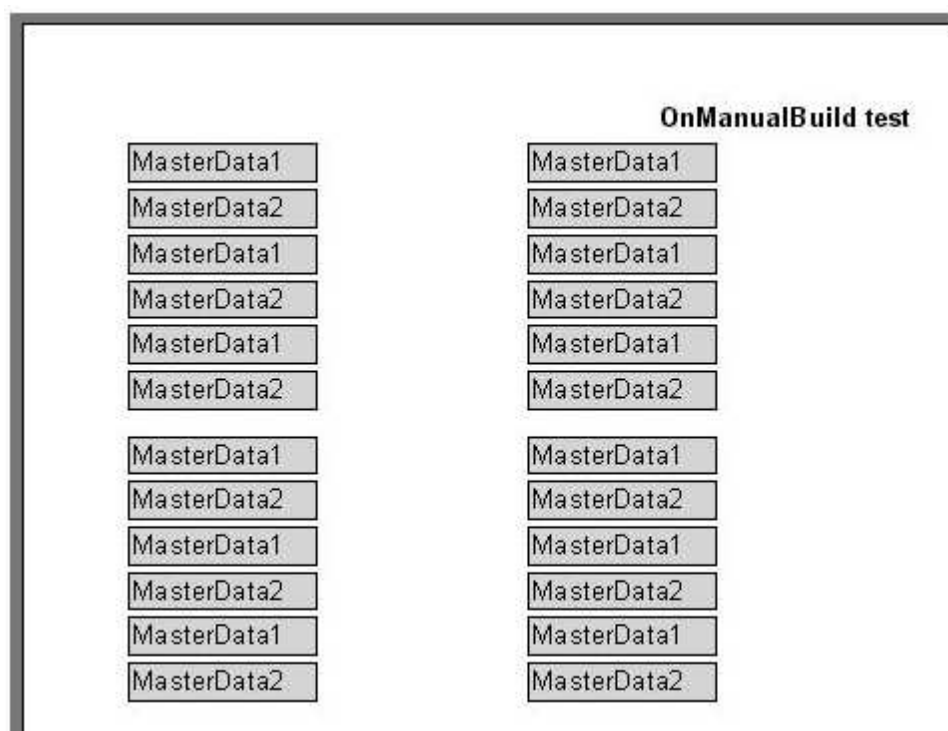
OnManualBuild test	
MasterData1	
MasterData2	
MasterData1	
MasterData2	
MasterData1	
MasterData2	
MasterData1	
MasterData2	
MasterData1	
MasterData2	
MasterData1	
MasterData2	

Das folgende Beispiel zeigt zwei Gruppen von Bändern, die nahe beieinander liegen.

```

procedure Page1OnManualBuild(Sender: TfrxComponent);
var
    i, j: Integer;
    SaveY: Extended;
begin
    SaveY := Engine.CurY;
    for j := 1 to 2 do
        begin
            for i := 1 to 6 do
                begin
                    Engine.ShowBand(MasterData1);
                    Engine.ShowBand(MasterData2);
                    if i = 3 then
                        Engine.CurY := Engine.CurY + 10;
                    end;
                    Engine.CurY := SaveY;
                    Engine.CurX := Engine.CurX + 200;
                end;
            end;
        end;
end;

```



Wie Sie sehen, haben wir in diesen Beispielen lediglich das Ausgeben der Daten-Bänder gesteuert. Alle restlichen Bänder (wie es bei unserem der „Report Title“ war) werden automatisch ausgedruckt.

Zum Schluss wollen wir zeigen, wie Sie einen Report des Typs „Kundenliste“ (wie wir ihn mehrere Male in diesem Buch erstellt haben) über das „OnManualBuild“-Ereignis erstellen. Verbinden Sie in unserem Beispiel das Daten-Band mit der Datenquelle.

ReportTitle: Band1		
Customers		
PageHeader: Band2		
[Company]	[Contact]	[Phone]
MasterData: MasterData1		
[Customers."Company"]	[Customers."Contact"]	[Customers."Ph"]
PageFooter: Band3		

Das „Script“ lautet folgendermaßen:

```
procedure Page1OnManualBuild(Sender: TfrxComponent);
var
  DataSet: TfrxDataSet;
begin
  DataSet := MasterData1.DataSet;
  DataSet.First;
  while not DataSet.Eof do
  begin
    Engine.ShowBand(MasterData1);
    DataSet.Next;
  end;
end;
```

Beim Starten des Reports überzeugen Sie sich, dass sich das Resultat der Arbeit des „Scripts“ nicht vom Standard-Report unterscheidet. Wir verweisen auf die Stelle, wie man eine Verbindung zum „Dataset“ bekommt; in unserem Beispiel haben wir ein Band mit der Datenquelle verbunden. Die Zeile

```
DataSet := MasterData1.DataSet;
```

stellt eine Verbindung zur Datenquelle her. Wenn ein Band nicht mit einer Datenquelle verbunden ist, kann die Verbindung zur erforderliche Datenquelle wie folgt gemacht werden:

```
DataSet := Report.GetDataSet('Customers');
```

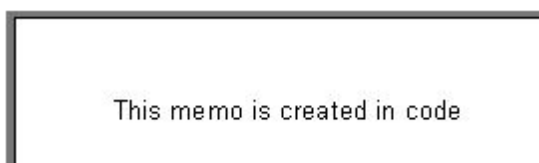
Natürlich muss die Datenquelle, an der Sie interessiert sind, zum Report im „Report|Data...“-Dialog hinzugefügt werden.

Erstellung von Objekten im „Script“

Sie können neue Objekte in den Report hinzufügen, indem Sie das „Script“ nutzen. Wir zeigen mit Hilfe eines einfachen Beispiels, wie es gemacht wird. Erstellen Sie einen leeren Report und dann schreiben Sie in die Haupt-Prozedur:


```
var
  Band: TfrxReportTitle;
  Memo: TfrxMemoView;
begin
  Band := TfrxReportTitle.Create(Page1);
  Band.Height := 20;
  Memo := TfrxMemoView.Create(Band);
  Memo.SetBounds(10, 0, 100, 20);
  Memo.Text := 'This memo is created in code';
end.
```

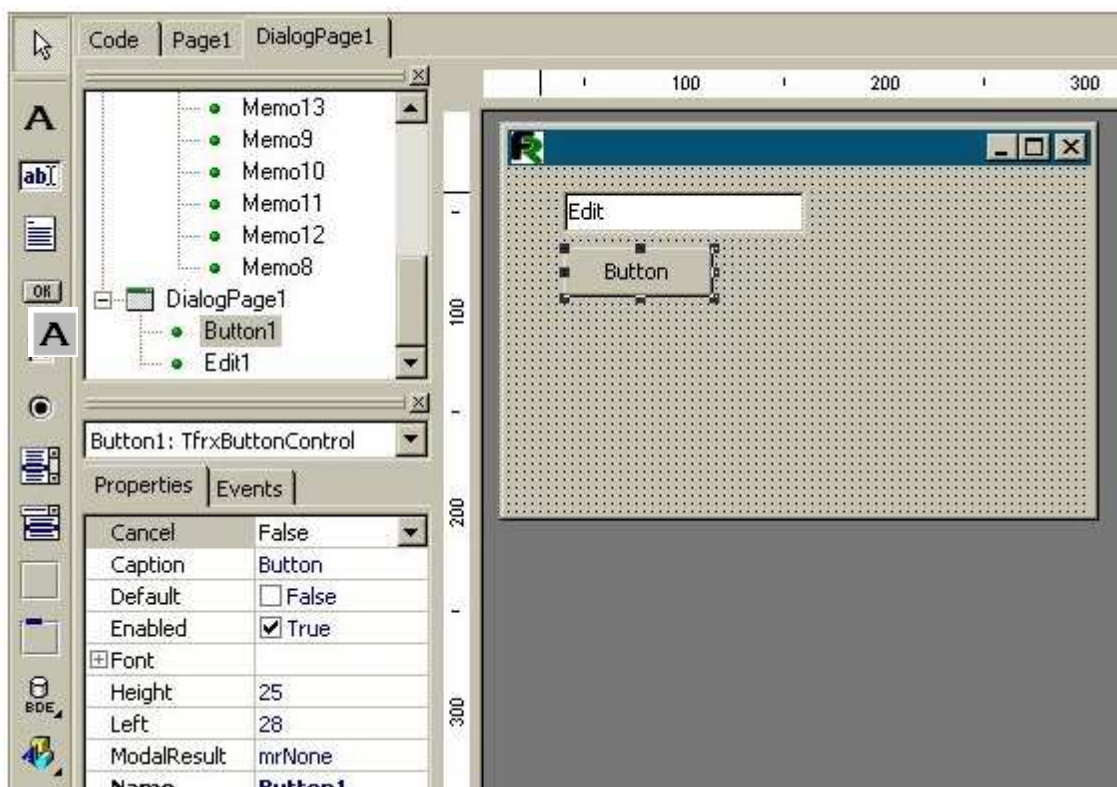
Starten Sie den Report:



Beachten Sie, dass wir das in diesem Beispiel erstellte Objekt nie entfernen. Es ist nicht erforderlich, da die Objekte automatisch entfernt werden, nachdem der Report fertig ist.







Dialog-Fenster











Zusätzlich zu den üblichen Reportseiten können Sie auch verschiedene Dialog-Fenster in einem Report verwenden. Für die Erstellung von Dialog-Fenstern wird der selbe Designer wie für Reportseiten genutzt. Die Schaltfläche  in der Werkzeugleiste des Designers wird für das Erstellen eines neuen Formulars verwendet. Sie fügt dem Report eine neue Seite hinzu. Wenn Sie zu der Dialog-Formular-Seite wechseln, verändert sich die Arbeitsfläche des Designers, so entsteht ein Formular, auf dem Objekte platziert werden können:



Steuerungs-Objekte

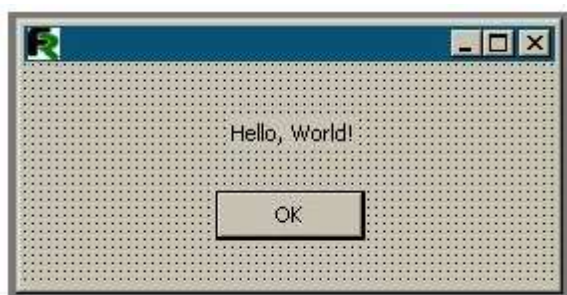
Folgende Objekte sind verfügbar:

Element	Name	Beschreibung
	TfrxLabelControl	Dieses Objekt wird genutzt, um erklärenden Text auf dem Dialog-Formular anzuzeigen.
	TfrxEditControl	Dieses Objekt wird genutzt, um eine Textzeile mit Hilfe der Tastatur einzugeben.
	TfrxMemoControl	Dieses Objekt wird genutzt, um verschiedene Textzeilen mit Hilfe der Tastatur einzugeben.
	TfrxButtonControl	Dieses Objekt stellt eine Schaltfläche dar.
	TfrxCheckBoxControl	Das Objekt stellt ein Element dar, das zwei Zustände annehmen kann: gewählt und nicht gewählt. Nahe dem Kreuz wird ein Text angezeigt.
	TfrxRadioButtonControl	Das Objekt stellt einen Schalter dar, der das Gegenstück zum „Radiobutton“ in Delphi ist. Er kann nicht einzeln genutzt werden.
	TfrxListBoxControl	Dieses Objekt stellt eine Liste dar, aus der Zeilen ausgewählt werden können.

Element	Name	Beschreibung
	TfrxComboBoxControl	Das Objekt stellt ein Eingabe-Objekt mit einer Auswahlliste dar, in der man Zeilen ausgewählt.
	TfrxDateEditControl	Das Objekt stellt ein Datumsfeld mit einem Kalender, in dem man das Datum auswählen kann, dar.
	TfrxGroupBoxControl	Das Objekt stellt eine Fläche mit Textbeschreibung dar, auf die andere Objekte gesetzt werden können.
	TfrxPanelControl	Das Objekt stellt eine Fläche ohne Textbeschreibung dar, auf die andere Objekte gesetzt werden können.
	TfrxBitBtnControl	Das Objekt stellt eine Schaltfläche mit Bild und Text dar.
	TfrxSpeedButtonControl	Das Objekt stellt eine Schaltfläche nur mit Bild dar.
	TfrxMaskEditControl	Das Objekt stellt ein Eingabefeld für eine formatierte Eingabe dar.
 	TfrxCheckListBoxControl	Das Objekt stellt eine Liste von TfrxCheckBoxControl-Elementen dar.
	TfrxBevelControl	Das Objekt stellt ein Linien-Objekt.
	TfrxImageControl	Das Objekt stellt ein Bild im „BMP“--, „ICO“--, „WMF“--, oder „EMF“-Format dar.

„Hello, World!“-Report als Dialog-Fenster

Nun wollen wir einen Report erstellen, der mit der Hilfe eines Dialog-Formulars ein Begrüßungs-Fenster zeigt. Erstellen Sie einen neuen leeren Report und fügen Sie ein Dialog-Formular in den Report ein. Setzen Sie ein „TfrxLabelControl“ und ein „TfrxButtonControl“-Objekt auf das Formular:



Stellen Sie die folgende Objekt-Eigenschaften ein:

TfrxLabelControl:
Caption = 'Hello, World!'

TfrxButtonControl:
Caption = 'OK'
Default = True
ModalResult = mrOk

Setzen Sie im Formular die Eigenschaft „BorderStyle = bsDialog“.

Sobald die Einstellungen des Dialog-Formulars fertig sind, kehren Sie zur Reportseite zurück und platzieren ein „Text“-Objekt mit irgendeinen Text auf ihm. Starten Sie den Report und Sie werden das Formular sehen:

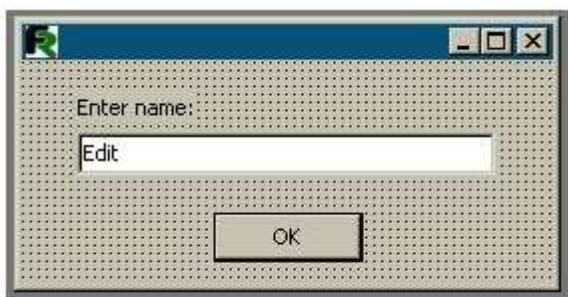


Beim Klicken auf die „OK“-Schaltfläche wird ein Report erstellt und angezeigt. Wenn Sie das Formular über die „X“-Schaltfläche des Fensters schließen, wird kein Report erstellt. Das ist ein Mechanismus, mit dem FastReport arbeitet:

wenn es Dialog-Fenster in einem Report gibt, wird er nur erstellt, wenn jedes Formular mit der „OK“-Schaltfläche geschlossen wurde, d.h. es basiert auf der Eigenschaft ModalResult = mrOk. Darum wird die „ModalResult“-Eigenschaft auf „mrOk“ gesetzt.

Eingabe von Parametern und Übertragung in einen Report

Machen wir das vorherige Beispiel etwas komplizierter, um zu zeigen, wie man Werte, die Sie in das Dialog-Fenster eingegeben haben, in den Report überträgt. Um dies durchzuführen, verändern Sie das Formular auf folgende Weise:



Platzieren Sie das „Text“-Objekt mit folgendem Inhalt auf der Seite:

*You entered:
[Edit1.Text]*

Starten Sie den Report und überzeugen Sie sich, dass der Parameter, den Sie eingegeben haben, erfolgreich angezeigt wird. Auf die selbe Weise können Sie mit anderen Objekten des Dialog-Fensters vorgehen. Da jedes Objekt einen Namen hat, der nur einmal innerhalb des gesamten Reports vorkommt, kann er im ganzen Report genutzt werden.

Zusammenspiel von Objekten

Über die Benutzung des „Scripts“ kann man leicht die Logik der Dialog-Fenster verdeutlichen, z.B. das Zusammenspiel der Steuerung. Veranschaulichen wir das mit einem einfachen Beispiel. Ändern Sie das Formular folgendermaßen:



Klicken Sie doppelt auf das „CheckBox“-Objekt, so dass eine Routine für das „OnClick“-Ereignis erstellt wird und schreiben Sie dann folgendes Script:

```
procedure CheckBox1OnClick(Sender: TfrxComponent);
begin
  Button1.Enabled := not CheckBox1.Checked;
end;
```

Wenn Sie den Report starten, werden Sie sehen, dass die OK-Schaltfläche nach dem Klicken auf die „CheckBox“-Objekt nicht mehr reagiert.

Stichwortverzeichnis

Aufruf von Funktionen.....	
[]- und < >- Klammern.....	66
Ausdrücke in „Text“-Objekten.....	25
Das „Script“.....	
Anker (Anchors).....	117
Aufruf von Datenbank-Feldern.....	104
Beispiel der Benutzung des „OnBeforePrint“-Ereignisses.....	106
Benutzung des „Engine“-Objekts.....	115
C++.....	101
Der erste Weg.....	109
Die Benutzung von „Outline“-Objekten.....	119
Die Gesamtsumme aus dem Gruppen-Fuß im Gruppen-Kopf ausgeben.....	108
Die Variablen aus der Variablenliste des Reports.....	104
Ereignisse.....	105
Erstellung von Objekten im „Script“.....	127
JScript.....	102
Nutzen von Aggregat-Funktionen im „Script“.....	104
Objekte im „Script“ verwenden.....	103
Pascal.....	101
weitere zur Verfügung stehende Objekte.....	112
„Engine“-Objekt.....	113
„Hello, World!“-Script.....	102
„OnAfterData“-Ereignis.....	111
„OnManualBuild“-Ereignis der Seite.....	122
„Outline“-Objekt.....	114
Designer.....	
Allgemeines.....	5
Maus-Tasten.....	7
Tastaturbefehle.....	7
Werkzeugeleisten.....	8
Diagramme.....	
Begrenzung der Anzahl der Diagramm-Werte.....	95
Einige nützliche Einstellungen.....	96
Einstellungen.....	
Entwurfs-Eigenschaften.....	14
Entwurfs-Optionen.....	19
Kreuztabellen-Reports.....	
Benutzung von Funktionen.....	81
Eine „Kreuztabelle“ manuell füllen.....	91
Erstellen eines „Kreuztabellen“-Reports.....	78
Größe von Zeilen/Spalten anpassen.....	90
Spaltenbreite anpassen.....	84
Werte sortieren.....	82
Master-Detail-Reports.....	
Kopf-und Fußzeilen Daten-Bänder.....	54
Mehrere Objekte auswählen.....	13
Report-Eigenschaften.....	

Anzahl der Kopien.....	16
Beschreibung.....	16
Drucken wenn leer.....	16
Drucker.....	16
Passwort.....	16
Zweipass.....	16
Report-Objekte.....	
HTML-Befehle.....	24
„Daten“-Bänder.....	29
„Text“-Objekt.....	21
Reports erstellen.....	
Bedingtes Hervorheben.....	69
Daten in Form einer Tabelle anzeigen.....	45
Eigenschaften von Gruppen.....	58
Einfügen von Aggregat-Funktionen.....	65
Etiketten drucken.....	48
Inline-Formatierung.....	67
Master-Detail-Reports.....	53
mehrzeilige Textanzeige.....	39
Nebeneinander stehende „Subreports“.....	74
Report-Objekte.....	18
Seiten- und Report-Summen.....	64
Streifen anzeigen.....	70
Verschachtelter Report (Subreport).....	73
Verschobene Objekte.....	51
Werte formatieren.....	66
Zeilennummerierung.....	59
Zusammenspiel von Objekten.....	131
„Child“-Bänder.....	50
Seiten-Eigenschaften.....	
Auf vorige Seite drucken.....	17
Ausrichtung der Seite.....	17
Papiergröße.....	17
Papierzufuhr.....	17
Ränder der Seite.....	17
Ränder spiegeln.....	17
Seitenhöhe im Entwurfsmodus vergrößern.....	17
Spalten.....	17
Subreports erstellen.....	
Einschränkungen bei der Nutzung von „Subreports“.....	74
PrintOnParent-Eigenschaft.....	75
Werkzeugeleisten.....	
„Ausrichtung“-Toolbar.....	13
„Designer-Mode“ Toolbar.....	8
„Rahmen“ Toolbar.....	12
„Standard“ Toolbar.....	9
„Text“ Toolbar.....	10