Prepare Method

Applies to

TQuickReport

Declaration

```
void __fastcall NewPage(void);
```

The NewPage event can be called from an event handler during report preparation. It will force a page break after the current band is printed. Page headers and footers will be printed correctly.

QuickReport

QuickReport version 1.0 Copyright (c) 1996 QSD AS

Welcome to QuickReport, the integrated reporting solution for Borland C++Builder!

About QSD AS

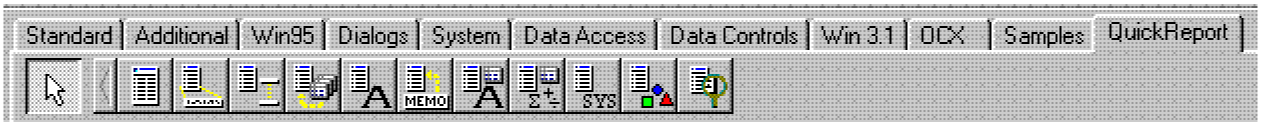
QSD AS is a Norwegian software develompent company specialising in Delphi and C++Builder development. In addition to creating QuickReport wi do both contract work and have plans for several other Delphi and C++Builder add-ons. Please feel free to contact us if you have any questions.

We can be contacted in several ways:

World Wide Web	http://www.qsd.no
Internet mail	support@qsd.no
CompuServe	101372,3471
Fax	+47 22 41 74 91
Postal Mail	QSD AS Radhusgaten 9 N-0151 Oslo Norway

QuickReport
Version 0.95c

QuickReport Component Overview



Visible components

TQuickReport	The 'controller' for each of your reports. Put one of these on your form to change it into a report
TQRBand	QuickReports are build up of printable components placed on QRBands
TQRLabel	QRLabel prints text on the report
TQRDBText	QRDBText prints database fields, including memo fields
TQRDetailLink	QRDetailLink is used to create any type of master/detail reports
TQRMemo	QRMemo prints of multiline text
TQRShape	Prints different shapes
TQRDBCalc	QRDBCalc performs calculations on your database fields
TQRPreview	Used to create your own custom preview forms
TQRGroup	Create data groups using the TQRGroup component
TQRSysData	Other components

Other Objects and Components

TQRPrinter	QRPrinter is a low level printer object with special functions
TQRCustomControl	QRCustomControl is the base class for printable QuickReport components

TQuickReport Component

Properties Events Methods

Header

QuickRep

Description

TQuickReport is the main component handling all the printing and data functionallity. This component is required to transform a form into a QuickReport. Typically you would connect the DataSource property to your data, put some QRBands on the form and put som printable controls on the bands. Finally you call the Preview or Print method to output the report.

ColumnMarginInches Property

Applies to

TQuickReport

Declaration

```
__property long LeftMarginInches;
```

The space put between the columns in a multicolumn report. The value is 1/10th inches

TQRBand Component

Properties Events

Header

QuickRep

Description

You build your reports by creating different QRBands and putting visible QuickReport controls on them. There are several different band types, like detail band, page header band, page footer band and so on. Some band types are printed automatically, while some need to be connected to a QRDetailLink or QRGroup component to be printed. The BandType property explains this more in depth.

TQRLabel Component

Properties Events

Header

QuickRep

Description

The QRLabel component is used to display text on a report like column headings. The caption property can be changed in any events during report preparation. If you want a to output text from a data field use the QRDBText component instead.

In addition the QRLabel component has all the properties from the QRCustomControl component.

TQRDBText Component

Properties Events

Header

QuickRep

Description

Use this component to display any field from a dataset. Works much like a DBText components on forms. If datafield is a memo field, the detail band will stretch to make room for all lines in the memo.

TQRMemo Component

Properties

Header

QuickRep

Description

Use the TQRMemo component when you need to print multiline text

TQRShape Component

Properties

Header

QuickRep

Description

QRShape is used to display simple lines and figures on the report.

TQRDBCalc Component

Properties

Header

QuickRep

Description

This component automates the process of suming and counting of data fields. In addition to these properties the QRDBCalc has all the properties and events it inherits from TQRDBText.

TQRPrinter

Properties Events Methods

Header

QuickRep

Description

TQRPrinter is a low level printer object with preview, save and load capabilities. An instance of this object called QRPrinter is automatically created when you application starts.

QRPrinter is like a Printer object with recording capabilities. All print commands you send to its Canvas property are recorded in a Windows Metafile object. If you use the NewPage method the current page is kept in memory and QRPrinter creates a new on. When you have created the entire printout you can choose to preview, print or save the result.

QuickReport uses this object for all its printing.

TQRPreview Component

Properties Events Methods

Header

QuickRep

Description

The TQRPreview component makes it very easy to create your own preview forms. The QRPreview component takes care of all the work involved in displaying the report on screen, including scrolling and zooming. Please look at the DEMOPRE form in the QuickReport demo project for an example on how to use the QRPReview component.

QRPreview has several methods you can call to control the behaviour during a preview. As an example you can call the ZoomToFit method in a button's OnClick event. This will make the QRPreview component zoom in/out so the entire page fits on the control.

ColumnMarginInches Property

Applies to
TQuickReport

Declaration

```
__property long ColumnMarginInches;
```

The space put between the columns in a multicolumn report. The value is 1/10th inches

TQRGroup Component

Properties Events

Header

QuickRep

Description

There are two ways of printing master/detail reports with QuickReport. One is to use a TQuery component with a join Query and QRGroups to group the data. The other method is to use the TQRDetailLink component combined with DetailGroup bands. Each method has its unique strengths.

TQRSysData Component

Properties Events

Header

QuickRep

Description

The QRSysData component is used to display certain system data. Set the Data property to select what information to print. This component also has the properties and events of the TQRLabel component

Properties

- ColumnMarginInches
- ColumnMarginMM
- Columns
- DataSource
- DisplayPrintDialog
- LeftMarginInches
- LeftMarginMM
- Orientation
- PageCount
- PageHeight
- PageNumber
- PageWidth
- RecordCount
- RecordNo
- ReportTitle
- ReportType
- RestartData
- ShowProgress
- SQLCompatible
- TitleBeforeHeader

Properties

Data

Text

AlignToBand Property

Applies to

TQRSysData

Declaration

Events

- AfterDetail
- AfterPrint
- BeforeDetail
- BeforePrint
- OnEndPage
- OnNeedData
- OnStartPage

Preview Method

Applies to

TQuickReport

Declaration

```
void __fastcall Preview(void);
```

The Preview method first prepares the report if necessary, then calls the QRPrinter.Preview method. If you haven’t assigned an event handler for the QRPrinter.OnPreview event the standard preview form is displayed. Otherwise your OnPreview event handler is called. The report is freed from memory before returning.

Methods

- NewPage
- Prepare
- Preview
- Print

Prepare Method

Applies to

TQuickReport

Declaration

```
void __fastcall Prepare(void);
```

Call the prepare method if you want to create the report without automatically printing or previewing it. Prepare create the report and sends it to the QRPrinter object.You must call the QRPrinter.CleanUp method when you want to release the report.

Properties

- BandType
- Color
- Enabled
- Font
- ForceNewPage
- Frame
- LinkBand
- Ruler

QuickReport Licensing Terms

The 32 bit version of QuickReport 1.0 is included with C++Builder. No additional license is needed to use this product

If you want to use QuickReport in your 16 bit Delphi 1.0 applications or you want the source code, QSD AS is selling a package containing the 16 bit version of QuickReport and the source for both the 16 and 32 bit version. The list price is $79.

CompuServe SWREG

QuickReport is available at CompuServe SWREG. Type GO SWREG at any CompuServe prompt. The SWREG number for QuickReport is 8644.

Credit Card

We accept VISA and MasterCard. Please send or fax us the registration form with your credit card details.

Check and Money Order

You can also registering by sending a check or International Money Order together with the registration form.

Properties

- Alignment
- AutoSize
- Caption
- AlignToBand
- Color
- Font
- Transparent

AfterPrint Event

Applies to

TQRBand

Declaration

```
typedef void __fastcall (__closure *TQRBandAfterPrintEvent)(bool BandPrinted);
__property TQRBandAfterPrintEvent AfterPrint;
```

This event is called whenever the band has been printed or tried to be printed but stopped in the BeforePrint event. The BandPrinted parameter will tell if the band was actually printed or not.

Events

AfterPrint
BeforePrint

Events

OnPrint

OnPrint Event

Applies to

TQRLabel, TQRDBCalc, TQRDBText

Declaration

```
typedef void __fastcall (__closure *TQRLabelOnPrintEvent)(System::TObject*
sender,  System::AnsiString & Value);
__property TQRLabelOnPrintEvent OnPrint;
```

Called immediately before the component is printed. The string to be printed is in the value parameter and any changes to this string will not be stored to the record. Use this event to format the data or concatenate several fields into one.

Properties

- Alignment
- AutoSize
- DataSource
- DataField
- Font
- Transparent

Master Property

Applies to

TQRDetailLink

Declaration

```
__property TQRController* Master;
```

The Master property is used to specify the master/detail relationship to be used during the report creation. In addition to using the Master property, the proper relationship must be set up between the tables involved. The master property can be set to a QuickReport component or another QRDetailLink component.

Events

OnPrint

Alignment Property

Applies to

TQRDBText

Declaration

```
__property Alignment ;
```

Control alignment of the text within the control

AutoSize Property

Applies to

TQRDBText

Declaration

```
__property bool AutoSize ;
```

If true, the control will resize to the width needed for the caption. If false, text will be clipped at the designed rectangle.

DataSource Property

Applies to

TQRDBText

Declaration

```
__property Db::TDataSource* DataSource;
```

DataSource to pick field from

DataField Property

Applies to

TQRDBText

Declaration

```
__property System::AnsiString DataField;
```

The field to display. This can be any type of field, except a Paradox formatted memo field or blob field. If it's a regular memo the band on which the control is placed on will strech to make room for the entire text. The text will not be split between pages, though.

Font Property

Applies to
TQRDBText, TQRMemo

Declaration

```
__property TFont Font;
```

Use to control the font, style and size of the text. Best results are obtained using a TrueType font. If you experience some scaling problems in the preview window change the font to one contailing the TT prefix.

Transparent Property

Applies to

TQRDBText

Declaration

```
__property bool Transparent;
```

If true, text background will be transparent, i.e. any components in the background will be visible.

PageHeight Property

Applies to

TQuickReport

Declaration

```
__property long PageHeight;
```

The current page height, taken from the QRPrinter object. The value is always given in the same values as the screen resolution.

DataSource Property

Applies to

TQuickReport, TQRDetailLink,

Declaration

```
__property Db::TDataSource* DataSource;
```

Set the DataSource property to the datasource you print the records from. QuickReport automatically advances through this dataset and prints a set of detail bands for each record. If you don’t spesify a DataSource and not a OnNeedData event handler only one set of detail bands will be printed. Also see the OnNeedData event and RestartData property.

PageWidth Property

Applies to

TQuickReport

Declaration

```
__property long PageWidth;
```

The current page width, taken from the QRPrinter object. PageWidth is read only.

ShowProgress Property

Applies to

TQuickReport

Declaration

```
__property bool ShowProgress
```

If true (Default) a progress form will appear during report preparation and printing. The progress form has a Cancel button.

Orientation Property

Applies to

TQuickReport

Declaration

```
enum TPrinterOrientation { poPortrait, poLandscape };
__property Printers::TPrinterOrientation Orientation;
```

Selects the paper orientation. Orientation can be set to poPortrait and poLandscape. Please note that the TPrinterOrientation type is declared in the Printers header file. If you want to set this property you need to include the PRINTERS.HPP file.

Alignment Property

Applies to

TQRLabel

Declaration

```
__property Alignment;
```

Sets the alignment of the text within the given rectangle or QRLabel position inside a QRBand. See AlignToBand.

AutoSize Property

Applies to
TQRLabel

Declaration
__property bool AutoSize;

If true (Default) the component will expand horizontally to make room for the entire text. If false, the size will be fixed.

Caption Property

Applies to

TQRLabel

Declaration

```
__property Caption;
```

Caption contains the actual text to be printed. This can be changed in events during report preparation.

AlignToBand Property

Applies to

TQRLabel

Declaration

```
__property bool AlignToBand;
```

Controlls the effect of the Alignment property

True	The QRLabel component will be aligned left, centered or right inside the extents of the QRBand.
False	Only the text inside the components will be aligned. (Default)

Color Property

Applies to
TQRLabel

Declaration

```
__property Color;
```

The background color of the text. See also the Transparent property.

Font Property

Applies to

TQRLabel

Declaration

```
__property TFont Font;
```

If you want to use another font than the default for the form or QRBand set the Font property.

Tip

Best results are obtained using TrueType fonts. If you experience some scaling problems in the preview window change the font to one contailing the TT prefix.

Transparent Property

Applies to

TQRLabel

Declaration

```
__property bool Transparent;
```

If Transparent is false (default) the bacground of the text will be painted before the text is drawn. If Transparent is true, the background will not be painted.

Properties

Font
Lines
WordWrap

Properties

Brush
Pen
Shape

PrinterOK Property

Applies to

TQRPrinter

Declaration

```
__property bool PrinterOK;
```

Runtime and read only
PrinterOK is true if a printer driver is installed.

Brush Property

Applies to

TQRShape

Declaration

```
__property Graphics::TBrush* Brush;
```

The brush to use when drawing the shape. See TBrush for more information.

Height Property

Applies to

TQRShape

Declaration

```
__property Height ;
```

Pen Property

Applies to

TQRShape

Declaration

```
__property Graphics::TPen* Pen;
```

The Pen to use for drawing the shape. See TPen for more information

Shape Property

Applies to

TQRShape

Declaration

```
__property TQRShapeType Shape;
```

These are the possible values of Shape:

Value	Meaning
qrsRectangle	Draw a rectangle
qrsCircle	Draw a circle
qrsVertLine	Draw a horizontal line
qrsHorLine	Draw a vertical line
qrsTopAndBottom	Draw horizontal lines at top and bottom of the control
qrsRightAndLeft	Draw vertical lines at the right and left edges

Width Property

Applies to

TQRShape

Declaration

__property Width ;

Properties

AsInteger
AsReal
Operation
PrintMask
ResetBand

Operation Property

Applies to

TQRDBCalc

Declaration

```
__property TQRCalcOperation Operation;
```

Set the operation you want to perform.

qrcSUM	Calculates the sum of the field
qrcCOUNT	Counts the field
qrcAVERAGE	Calculates the average value of the field
qrcMIN	Contains the minimum value of the field
qrcMAX	Contains the maximum value of the field

PrintMask Property

Applies to

TQRDBCalc

Declaration

```
__property System::AnsiString PrintMask;
```

Use the PrintMast to format the output of the QRDBCalc component. Look up the FormatFloat procedure in the VCL help for documentation on how the PrintMask works.

ResetGroup Property

Applies to

TQRDBCalc

Declaration

```
__property TQRBand* ResetBand;
```

The QRDBCalc will be reset after the band set in ResetBand property is printed

AsInteger Property

Applies to
TQRDBCalc

Declaration

```
__property long AsInteger
```

Returns the current value of the QRDBCalc component as a **long**. AsInteger is read only.

AsReal Property

Allpies to

TQRDBCalc

Declaration

```
__property double AsReal;
```

Returns the current value of the QRDBCalc component as a **double**. AsReal is read only.

TQRDetailLink Component

Properties Events

Header

QuickRep

Description

The QRDetailLink component is used to create any type of Master/detail report. For a simple two table master detail report you would have on QuickReport component for the master and one QRDetailLink for the Detail. You would then set the Master property of the QRDetailLink component to point to the QuickReport component. The Master property can also be set to point to another QRDetailLink component in order to create a multi level master detail report. Several QRDetailLink components can have the same master, which would print them on the same level, one after the other. You can specify three bands for the QRDetailLink component, a Detail band, header band and footer band. The detail band is printed once for each record in the dataset set by the DataSource property.

Records can be filtered using the OnFilter event. The OnNeedData event can be used if you want to print data from a datasource other than a table or query.

Properties

- Canceled
- Canvas
- EnableOpenBtn
- EnablePrintBtn
- EnableSaveBtn
- ToPage
- FromPage
- Orientation
- Page
- PageHeight
- PageCount
- PageNumber
- PageWidth
- PrinterOK
- ShowProgress
- Status
- Title
- Thumbs

PageCount Property

Applies to

TQuickReport

Declaration

```
__property long PageCount;
```

The total number of pages in the report. This property is not valid before the report is finished. PageCount is read only.

Events

OnPreview

OnPreview Event

Applies to

TQRPrinter

Declaration

```
typedef void __fastcall (__closure *TOnPreviewEvent)(void);
__property TOnPreviewEvent OnPreview;
```

Use this event if you want to display your own preview form instead of the standard included with QuickReport. Creating a custom preview is very easy using the supplied TQRCustomPreview component. See the demo program for an example on how to do this.

Canceled Property

Applies to

TQRPrinter

Declaration

```
__property bool Canceled;
```

True if the Cancel method has been called. This can be used to check if users pressed a Cancel button.

Canvas Property

Applies to

TQRPrinter

Declaration

```
__property Graphics::TMetafileCanvas* Canvas;
```

This is the canvas you use to do your printing.

EnableOpenBtn Property

Applies to

TQRPrinter

Declaration

```
__property bool EnableOpenBtn;
```

If true, the Open file button will appear on the standard preview form.

EnablePrintBtn Property

Applies to

TQRPrinter

Declaration

```
__property bool EnablePrintBtn;
```

If true, the Print button will appear on the standard preview form.

EnableSaveBtn Property

Applies to

TQRPrinter

Declaration

```
__property bool EnableSaveBtn;
```

If true, the Save file button will appear on the standard preview form.

ToPage Property

Applies to
TQRPrinter

Declaration

```
__property int ToPage;
```

Specify the last page to be printed.

FromPage Property

Applies to

TQRPrinter

Declaration

```
__property int FromPage;
```

Specify the first page to be printed.

Orientation Property

Applies to

TQRPrinter

Declaration

```
__property Printers::TPrinterOrientation Orientation;
```

Set to poPortrait or poLandscape to select printer orientation.

Page Property

Applies to

TQRPrinter

Declaration

```
__property Graphics::TMetafile* Page;
```

This is a metafile representation of the currently selected page. Status must be mpFinished for this metafile to be valid. Page is read only.

PageHeight Property

Applies to

TQRPrinter

Declaration

```
__property int PageHeight;
```

The current page height in pixels. PageHeight is read only.

PageCount Property

Applies to

TQRPrinter

Declaration

```
__property int PageCount
```

Total number of pages sent to the QRPrinter in the current print job. PageCount is read only.

PageNumber Property

Applies to

TQRPrinter

Declaration

```
__property int PageNumber;
```

Select the page you want QRPrinter to make available for viewing or copying to clipboard. The Status property must be mpFinished.

PageWidth Property

Applies to

TQRPrinter

Declaration

```
__property int PageWidth;
```

The current page width in pixels. PageWidth is read only.

ShowProgress Property

Applies to

TQRPrinter

Declaration

```
__property bool ShowProgress;
```

If true a progress form will appear indicating the progress of printing, saving and loading. The form has a Cancel button and the Cancelled property will be set to true if the user presses this button.

Status Property

Applies to
TQRPrinter

Declaration

```
__property TQRPrinterStatus Status;
```

Indicates the current QRPrinter status. It can be one of the following:

Value	Meaning
`mpReady`	QRPrinter is ready to accept a new print job
`mpBusy`	A print job is currently being recorded
`mpFinnished`	`The print job has been ended`
`mpPrinting`	QRPrinter is sending a job to the printer
`mpPreviewing`	A preview is currently active

Status is read only.

Title Property

Applies to

TQRPrinter

Declaration

```
__property System::AnsiString Title
```

The current print job title. This title will appear in the print manager and in any network print queue.

Thumbs Property

Applies to

TQRPrinter

Declaration

```
__property int Thumbs;
```

QRPrinter can print several scaled down images of pages on each sheet of paper. If you set Thumbs to two QRPrinter will print two by two images, if you set it to three, three by three images will be printed and so on.

Properties

PageNumber
Zoom

Events

Show

TQRPreview event Show

Methods

- BeginDoc
- Cancel
- Cleanup
- EndDoc
- Load
- NewPage
- Preview
- Print
- Save

BeginDoc Procedure

Applies to
TQRPrinter

Declaration

```
void __fastcall BeginDoc(void);
```

Start a new print job. This sets the Status property to mpBusy.

Cancel Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall Cancel(void);
```

Cancel the current print job. This sets the Status property to mbReady

Cleanup Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall Cleanup(void);
```

After you are finished with the current print job this command will free all pages from memory. This sets the Status property to mpReady.

EndDoc Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall EndDoc(void);
```

Tells QRPrinter that the you are finished sending commands to the canvas. This sets the Status property to mpFinished.

Load Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall Load( System::AnsiString Filename);
```

Loads a previously saved printout from a file.

NewPage Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall NewPage(void);
```

Tells QRPrinter to start printing a new page. The current page is kept in memory and a new one is created. Page number is increased by one. This requires the Status property to be mpBusy.

Preview Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall Preview(void);
```

This brings up print preview. It requires the Status property to be mbFinished. If you have assigned an event handler to the OnPreview event this gets called and nothing else is done.

Print Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall Print(void);
```

This prints the current report. It requires the Status property to be mpFinished.

Save Procedure

Applies to

TQRPrinter

Declaration

```
void __fastcall Save( System::AnsiString Filename);
```

This saves the current report to a file. The Status property must be mpFinished. The file format is a QuickReport internal format. Documentation for this will be released later.

QuickReport version 1.0 Registration Form

____
Date

____
Company

____
Contact

____
Postal address

____
zip/city

____
Country

____
E-mail address

____
Signature

Registration fee is US $79.

Fill out the above registration form and send it to :

```
QSD AS
Radhusgaten 9
N-0151 Oslo
Norway
```

Properties

- DataField
- DataSource
- FooterBand
- HeaderBand
- Level
- GroupData

Events

OnNeedData

DataSource Property

Applies to

TQRGroup

Declaration

```
__property Db::TDataSource* DataSource;
```

Select the datasource you want to connect this group to.

DataField Property

Applies to
TQRGroup

Declaration

```
__property System::AnsiString DataField;
```

Select the data field you want to connect this group to. The group will reprint when the value of Datafield change or a higher level group is reprinted.

FooterBand Property

Applies to
TQRGroup

Declaration

```
__property TQRBand* FooterBand;
```

Select a band to be printed as this groups footer band.

HeaderBand Property

Applies to

TQRGroup

Declaration

```
__property TQRBand* HeaderBand;
```

Selects a band to be printed as this groups header band.

Level Property

Applies to

TQRGroup

Declaration

```
__property int Level;
```

This group’s level. The outer (highest) level is 0, the inner (lowest) level is 9.

OnNeedData Event

Applies to

TQRGroup

Declaration

```
typedef void __fastcall (__closure *TGroupNeedDataEvent)( System::AnsiString
&Value);
__property TGroupNeedDataEvent OnNeedData;
```

The QRGroup.OnNeedData event can be used to alter the way groups break by changing the Value parameter. The current value of the connected datafield is send in this parameter and it can be changed in any ways. As an example you could connect the group to a Name field, then use the following line to make the group break every time the first letter of the name changes:

Value:=copy(value,1,1);

Methods

GroupData Property

Applies to

TQRGroup

Declaration

```
__property System::AnsiString GroupData;
```

Read only.

Events

OnPrint

OnPrint Event

Applies to

TQRSysData**Declaration**

```
__property OnPrint;
```

AutoSize Property

Applies to

TQRSysData

Declaration

```
__property AutoSize;
```

Color Property

Applies to

TQRSysData

Declaration

```
__property Color;
```

Data Property

Applies to
TQRSysData

Declaration

```
__property TQRSysDataType Data;
```

The Data property decides what information the component will display.

Value	Meaning
`qrsTime`	Current time in the format HH:MM
`qrsDate`	Current date in the format set by the global ShortDateFormat variable
`qrsDateTime`	Current date and time separated by a comma
`qrsPageNumber`	The current page number
`qrsReportTitle`	The report title as it appears in the TQuickReport.ReportTitle property
`qrsDetailCount`	Total number of records to be printed in report
`qrsDetailNo`	Current record number to be printed

Font Property

Applies to

TQRSysData

Declaration

```
__property TFont Font;
```

Text Property

Applies to

TQRSysData

Declaration

```
__property System::AnsiString Text;
```

The Text property is printed as a caption before the actual system data.

Transparent Property

Applies to

TQRSysData

Declaration

```
__property Transparent;
```

AfterPrint Event

Applies to

TQuickReport

Declaration

```
__property Classes::TNotifyEvent AfterPrint;
```

The AfterPrint event is called when the report is prepared and sent to the QRPrinter object.

BeforeDetail Event

Applies to

TQuickReport

Declaration

```
__property Classes::TNotifyEvent BeforeDetail;
```

The BeforeDetail event is called before the detail bands for a record are printed.

AfterDetail Event

Applies to

TQuickReport

Declaration

```
__property Classes::TNotifyEvent BeforeDetail;
```

The AfterDetail event is called after the detail bands for one record are printed.

Print Method

Applies to

TQuickReport

Declaration

```
void __fastcall Print(void);
```

This event prepares and prints the report.

TQRCustomControl

Properties Methods

Header

QuickRep

Description

The QRCostomControl component is the base class for all printable QuickReport components. If you want to create your own printable components you should make them children of this class. QRCustomControl has two virtual methods you might override.

BandType Property

Applies to

TQRBand

Declaration

```
__property TQRBandType BandType;
```

BandType selects when this band will be printed. Some bands are printed automatically by QuickReport, while others must be connected to a QRDetailLink or QRGroup component to be printed.

Use the 'Bring to Front' and 'Send to Back' menu items to control printing order for bands of the same type

Value	Meaning
`rbTitle`	This band is printed automcatically once at the start of the report. The QuickReport.TitleBeforeHeader property decides if the PageHeader or Title band will appear first.
`rbPageHeader`	Page header bands appear automatically on top of each page. The QuickReport.TitleBeforeHeader property decides which band will be printed first on the first page. You can have as many Page header bands you want.
`rbDetail`	The Detail band gets printed once for every record in your dataset. If you design a master/detail report using the QRDetailLink component this band will be the detail band for your master table. You can have as many detail bands you want.
`rbSubDetail`	If you create a master/detail report using the QRDetailLink component then this band will be the Detail band for your Detail Table(s). You must connect the DetailGroup band to the DetailLink component using the QRDetailLink.DetailBand property.
`rbPageFooter`	This band is printed automatically at the bottom of every page.
`rbSummary`	The Summary band is printed automatically on the last page of the report, after all detail bands and group footers.
`rbGroupHeader`	Group Header bands are used as group headers for QRGroup and QRDetailLink components. They must be connected to such a component to be printed.
`rbGroupFooter`	Group footers are used as footers for QRGroup and QRDetailLink components. They must be connected to such a component to be printed.
`rbColumnHeader`	If you print a multi column report this band will be printed on top of each column.
`rbOverlay`	If you want some static text or graphic to be printed on every page of your reports then you can use the rbOverlay band. This band will be printed once on each page and it overlay all other text and graphic printed on the page.

QuickReport Technical support

For users of the 32 bit version of QuickReport, please call the Borland technical support staff for questions. Users who have ordered the 16 bit version or source code can get technical support by contacting QSD directly.

You can also visit our WWW based QuickReport forum which you will find at http://www.qsd.no.

Color Property

Applies to

TQRBand

Declaration

```
__property Color;
```

Sets the background color of the band.

Enabled Property

Applies to

TQRBand

Declaration

```
__property bool Enabled;
```

Enables and disables the printing of this band. If this property is false the BeforePrint and AfterPrint events will be skipped. You can change this property during report progress, for instance to print different detail bands depending on the type of data to print.

Font Property

Applies to

TQRBand

Declaration

```
__property TFont Font;
```

The default font for the band. Use a True Type font to ensure that QuickReport is able to scale the font properly during preview.

ForceNewPage Property

Applies to

TQRBand

Declaration

```
__property bool ForceNewPage;
```

If you set this property to true, a page break will always be performed before this band is printed.

Frame Property

Applies to
TQRBand

Declaration

```
__property TQRFrame* Frame;
```

Sets the pen to be used as a frame around the band. The default pen with is 0, so no frame is drawn.

Ruler Property

Applies to

TQRBand

Declaration

```
__property TQRRuler Ruler;
```

This property will tell QuickReport to display a ruler in the background of the band during design time. This will help you position the controls on the bands. One of the following values can be set:

Value	Meaning
qrrNone	No ruler is displayed
qrrInchesH	Horizontal ruler lines every inch
qrrInchesV	Vertical ruler lines every inch
qrrInchesHV	Horizontal and vertical ruler lines every inch
qrrCmH	Horizontal rulers every centimetre
qrrCmV	Vertical ruler lines every centimetre
qrrCmHV	Horizontal and vertical ruler lines every centimetre

BeforePrint Event

Applies to

TQRBand

Declaration

```
typedef void __fastcall (__closure *TQRBandBeforePrintEvent)(bool &PrintBand);
__property TQRBandBeforePrintEvent BeforePrint
```

The BeforePrint event is called before the band is about to be printed. You can set the PrintBand parameter to false if you want the band to be skipped for this time.

BeforePrint Event

Applies to

TQuickReport

Declaration

```
typedef void __fastcall (__closure *TQRReportBeforePrintEvent)(bool
&PrintReport);
__property TQRReportBeforePrintEvent BeforePrint;
```

The BeforePrint event is called before the report is prepeared. Set the PrintReport parameter to false to cancel printing of the report.

OnEndPage Event

Applies to

TQuickReport

Declaration

```
__property Classes::TNotifyEvent OnEndPage;
```

The OnEndPage event is called whenever a page brake is about to be performed.

OnNeedData Event

Applies to

TQuickReport

Declaration

```
typedef void __fastcall (__closure *TReportNeedDataEvent)(bool &MoreData);
__property TReportNeedDataEvent OnNeedData
```

The OnNeedData event is used if you want to print a report of data not coming from a regular DataSource, but from arrays, variables, text files or whatever you like. If you leave the DataSource property blank and instead write an OnNeedData event handler this handler gets called whenever QuickReport is going to print a new set of detail bands. Set the MoreData parameter to false when you are printing the last set of data.

OnStartPage Event

Applies to

TQuickReport

Declaration

```
__property Classes::TNotifyEvent OnStartPage;
```

The OnStartPage event is called whenever a new page is started.

PageNumber Property

Applies to

TQuickReport

Declaration

```
__property long PageNumber;
```

The current page number. This property can be access in events during report preparation. PageNumber is read only.

RecordCount Property

Applies to

TQuickReport

Declaration

```
__property long RecordCount;
```

The total number of records in the master dataset. RecordCount is read only. See also the SQLCompatible property.

RecordNo Property

Applies to

TQuickReport

Declaration

```
__property long RecordNo;
```

The current record in the dataset. This property always reflect the record number in the master table. RecordNo is read only.

DisplayPrintDialog Property

Applies to

TQuickReport

Declaration

```
__property bool DisplayPrintDialog;
```

If DisplayPrintDialog is set to true a print dialogue will appear before the report is printed. This dialogue will let the user select printer, pages to print and print to file.Please note that the PrinterDialog is not automatically displayed if the user presses the print button in the standard preview, even if this property is set to true.

Colomns Property

Applies to

TQuickReport

Declaration

```
__property int Columns;
```

The Columns property can be used to print multi column mailing labels or phone directory style reports. Also see the QRBand.BandType=rbColumnHeader for creating column heading bands.

LeftMarginInches Property

Applies to

TQuickReport

Declaration

```
__property long LeftMarginInches;
```

The left margin in 1/10th inches, not including the space wasted by the printer

LeftMarginMM Property

Applies to

TQuickReport

Declaration

```
__property long LeftMarginMM;
```

The left margin in millimetres, not including the space wasted by the printer

ReportTitle Property

Applies to

TQuickReport

Declaration

```
__property System::AnsiString ReportTitle;
```

The ReportTitle will appear in the Print Manager and in any network printing queue. It will also printed wherever you have used a TQRSysData component set to display ReportTitle.

ReportType Property

Applies to

TQuickReport

Declaration

```
__property ReportType;
```

Set ReportType to qrMasterDetail if you want QuickReport to print the detail tables you have linked using the QRDetailLink component.

Value	Meaning
qrStandard	Standard report listing from a table or query
qrSingleDetail	??
qrMasterDetail	Master detail using QRDetailLink components.

RestartData Property

Applies to

TQuickReport

Declaration

```
__property bool RestartData;
```

If true (default) QuickReport will move to the first record in the dataset before printing starts. If false, printing will start at the current record.

SQLCompatible Property

Applies to

TQuickReport

Declaration

```
__property bool SQLCompatible;
```

Not all Databases support the RecordCount function. If you get a BDE error message saying ‘function not supported’ try setting the SQLCompatible to true.

TitleBeforeHeader Property

Applies to

TQuickReport

Declaration

```
__property bool TitleBeforeHeader;
```

If false (Default) the page header will be printed before the Title band on the first page. You can reverse this by setting the TitleBeforeHeader property to true.

Methods

Print
Stretch

Print Method

Applies to

TQRCustomControl

Declaration

```
virtual void __fastcall Print(int X, int Y);
```

The Print method prints the component to the QRPrinter object at the given position. The default print method prints the text of the protected Caption property. If you want another behaviour you will need to override this method.

TQRCustomControl Stretch

Applies to

TQRCustomControl

Declaration

```
virtual void __fastcall Stretch(int &size);
```

If your component will resize itself vertically at printing time, like a stretching memo field you must return the new component height with this method. If your component will have a static size you won't use this method.

Properties

- AutoSize
- Height
- Left
- Top
- Visible
- Width

AutoSize Property

Applies to

TQRCustomControl

Declaration

```
__property AutoSize;
```

Height Property

Applies to

TQRCustomControl

Declaration

```
__property Height;
```

The height of the control in pixels.

Left Property

Applies to

TQRCustomControl

Declaration

```
__property Left;
```

The left horizontal position of the control in pixels, counting from the left margin.

Top Property

Applies to

TQRCustomControl

Declaration

```
__property Top;
```

The vertical position of the control within the band it is printed on.

Visible Property

Applies to

TQRCustomControl

Declaration

```
__property Visible;
```

If true (default) the control is printed. If false, the control is not printed.

Width Property

Applies to

TQRCustomControl

Declaration

```
__property Width;
```

The width of the control in pixels.

Properties

DataSource
DetailBand
FooterBand
HeaderBand
Master
PrintBefore

DataSource Property

Applies to

TQRDetailLink

Declaration

```
__property Db::TDataSource* DataSource;
```

The datasource to connect this QRDetailLink to

DetailBand Property

Applies to

TQRDetailLink

Declaration

```
__property TQRBand* DetailBand;
```

The band to use as this QRDetailLink's Detail band. It is printed for every record in the detail table maching with the current master record, provided you have set up the master/detail relationship correctly of course. See your compiler manuals for help on how to do this if you are unsure.

FooterBand Property

Applies to

TQRDetailLink

Declaration

```
property FooterBand : TQRBand
```

The band to use as this QRDetailLink's footerl band. The footer band is printed after all detail records for the current master record.

HeaderBand Property

Applies to

TQRDetailLink

Declaration

```
__property TQRBand* HeaderBand;
```

The band to use as this QRDetailLink's Detail band

Events

OnFilter

OnNeedData

OnNeedData Event

Applies to

TQRDetailLink

Declaration

```
typedef void __fastcall (__closure *TReportNeedDataEvent)(bool &MoreData);
__property TReportNeedDataEvent OnNeedData;
```

If you want to print from an array or some other source except a DataSource you can use the OnNeedData event to retrieve the information. See the QuickReport.OnNeedData event for a more in depth explanation on this.

Lines Property

Applies to

TQRMemo

Declaration

```
__property Lines;
```

The lines property is used to store the text being displayed in the QRMemo control. It can be manipulated at run time through the methods of a TStringList.

Methods

ZoomToWidth
ZoomToFit

ZoomToWidth Method

Applies to

TQRPreview

Declaration

```
void __fastcall ZoomToWidth(void);
```

Zooms the preview so the page fits exacty in the component width.

ZoomToFit Method

Applies to
TQRPreview

Declaration

```
void __fastcall ZoomToFit(void);
```

Zooms the preview so the entire page fits on the preview control

PageNumber Property

Applies to

TQRLabel

Description

__property long PageNumber;

The PageNumber property is used to set what page you want to show in the preview. Changing this property changes the current preview page.

LinkBand property

Applies to

TQRBand

Declaration

```
__property Extctrls::TCustomPanel* LinkBand;
```

The LinkBand property is used make sure two or more bands always are printed on the same page. You should always connect this to the NEXT band, not to the previous. Several bands can be linked in a chain. If there isn't enough room for all the linked bands on the page, a page break will be performed.

PrintBefore Property

Applies to

TQRDetailLink

Declaration

```
__property bool PrintBefore;
```

If false (default) the detail table will be printed after the detail bands of the Master controller, if any. If set to true, the detail table will be printed before the master table's detail band.

OnFilter event

Applies to

TQRDetailLink

Declaration

```
typedef void __fastcall (__closure *TQRFilterEvent)(bool &PrintRecord);
__property TQRFilterEvent OnFilter;
```

The OnFilter event can be used for local filtering of records during report preparation. You can check the contents of the data and set the PrintRecord parameter to false if you don't want the record to be printed.

WordWrap property

Applies to

TQRMemo

Declaration

```
__property bool WordWrap;
```

If the WordWrap property is true, lines will be wraped to fit the width of the control If false, lines will not be wrapped.