

# Contents

## Creating Reports in RPTwin

[What Can You Do With RPTwin?](#)

[Elements of an RPTwin Report](#)

[The RPTwin Design Window](#)

[The DataSet Columns List](#)

[The ToolBox](#)

[The Toolbar](#)

[The Status Bar](#)

[The Rulers](#)

[The Grid](#)

[RPTwin Menus](#)

[Creating a New Report](#)

[Using an Existing Report](#)

[Working With Bands](#)

[Data Fields](#)

[Using Live Data in Your Report Layout](#)

[Text Fields](#)

[Working With Fields](#)

[OLE Objects](#)

[Working With OLE Objects](#)

[Working With Headers and Footers](#)

[Page Layout Options and Preferences](#)

[Aligning Fields](#)

[Sorting and Grouping](#)

[Changing Field Properties](#)

[Changing Band Properties](#)

[Changing OLE Object Properties](#)

[Previewing a Report](#)

[Printing a Report](#)

[Changing the DataSet](#)

[Creating Guided Reports](#)

[Using Sum, Average, and Other Aggregate Functions](#)

[Filtering: Using Subsets of Data](#)

[Understanding Formulas](#)

[Changing Report Defaults](#)

## **RPTwin Functions**

[Overview](#)

[Abs](#)

[Age](#)

[Avg](#)

[Cos](#)

[Count](#)

[Date](#)

[DateTime](#)

[DayName](#)

[DayNameAbr](#)

[DayOfMonth](#)

[DayOfWeek](#)

[DayOfYear](#)

[DaysBetween](#)

[GroupAvg](#)

[GroupCount](#)

[GroupMax](#)

[GroupMin](#)

[GroupSum](#)

Hour

If...Then...Else

InitCap

LCase

Left

LTrim

MakeDate

MakeMoney

MakeTime

Max

Mid

Min

Minute

Mod

Month

MonthName

MonthNameAbr

PageNum

Quarter

RecNum

Replace

ReportAvg

ReportCount

ReportCumAvg

ReportCumMax

ReportCumMin

ReportCumSum

ReportMax

ReportMin

ReportSum

Right

Round

RTrim

Second

Sign

Sin

Sum

Tan

Time

ToDate

ToNumber

ToText

Trim

Trunc

UCase

Week

Year

YearsBetween

# What Can You Do With RPTwin?

After you define a report in [ERwin](#) or [BPwin](#) and launch RPTwin, you can create and modify the report layout in RPTwin's graphic design environment.

RPTwin lets you create almost any kind of report layout. You can start with a predefined ERwin or BPwin report, use the RPTwin Quick Report or Guided Report layouts, or you can build a report completely to your own specifications.

This section explains how to use RPTwin to design attractive and meaningful reports, including how to:

- ◆ Use the RPTwin Toolbox, Toolbar, and menus
- ◆ Highlight important information using different fonts and styles
- ◆ Place data wherever you want it on the page
- ◆ Reorganize the data by sorting or grouping (one level or many levels)
- ◆ Add titles, text, lines, borders, and fill patterns to part or all of the data displayed
- ◆ Add headers and footers for groups of data, for pages, or for the entire report
- ◆ Use RPTwin functions to automatically calculate values and format information
- ◆ Embed and link OLE (Object Link and Embedding) objects

## Elements of an RPTwin Report

The sample Subject Area report shown below (which is just a portion of one page) identifies the fundamental sections of a report.

These sections are called **bands**, and will be discussed in more detail later. They include:

Report Header - Appears once, at the beginning of the report. In the sample, "Overdue Notice Subject Area" has been placed in the Report Header.

Page Header - Appears at the top of each page of the report. In the sample, the Page Header includes the page number and the date.

Group Header - Appears at the beginning of each group of data that you have specified. This report has been grouped by Entity Name (e.g., CUSTOMER and OVERDUE NOTICE).

Detail Band - Appears once for every row of data in the DataSet. In the sample, the actual data (which comes from an ERwin data model) appears in the Detail band. Every report has at least a Detail Band, even if it has nothing else.

Group Footer - Appears at the end of each group of data that you have specified. In the sample, the Group Footer displays the total attributes in each entity.

Page Footer - Appears at the bottom of every page (not shown in sample).

Report Footer - Appears once at the end of the report (not shown in sample).

It is not necessary to use all the sections in every report. All bands except the Detail Band are optional. A simple report, may have only one section, the Detail Band, without any header or footer sections at all.

## The RPTwin Design Window

The RPTwin [Design window](#) contains the tools you need to define a report.

Notice that in the Design window, the different sections of the report are laid out in horizontal bands. In this example, there is a [Report Header Band](#), a [Page Header Band](#), and so on. You build or change a report by adding elements (text, data, lines, etc.) to the appropriate bands in your report layout.

Note that each band has a border at the top which contains its name (e.g. Report Header), and it has a band layout area (immediately beneath the band border) where data or text can be inserted. Bands are discussed in detail in sections titled [Working With Bands](#) and [Changing Band Properties](#).

The following elements can be placed in a band:

**Data Fields** - contain the actual data in your report. This comes directly from an [RPTwin report data file](#).

**Text Fields** - are used for placing additional text (titles, labels, etc.) in your report.

**Formula Fields** - are used to place calculations that are based on other field values.

**Special Fields** - are common fields such as: Date, Time, Page Number, Record Number and Record Count.

**OLE Objects** - (Object Link and Embedding) are usually graphical objects such as bitmaps that can be placed and linked to an available OLE server application (PC Paintbrush, MS Excel, MS Word, etc.).

## The DataSet Columns List

The [DataSet Columns list](#) shows all the columns in your current selected DataSet. Data columns are simply dragged from this list to the spot where you want them to appear in your report. To place a column name, click and drag the desired item from the DataSet Columns list to the desired report location.

**Note:** The DataSet Columns List is a floating list; it may be moved anywhere on the screen. To move it, click on the title of the list (DataSet Columns), keep the mouse button down, and drag the list to its new location. To close (or open) the list, click on "DataSet Columns List" in the View menu.

## The ToolBox

---



The ToolBox is a floating palette of tools. Click on one of the icon buttons in the ToolBox to choose the type of element you want to add to your report. The cursor changes appropriately. Then, position the cursor where you want it in your layout, and click again. The chosen element is added to the report layout at the cursor position.

**Note:** The ToolBox is a floating palette of tools that may be moved anywhere on the screen. To move it, click on the top bar of the ToolBox, keep the mouse button down, and drag it. To close the ToolBox, double-click on the close box at the upper left corner of the ToolBox. To open it again, use the View menu and click on ToolBox.



**Selection** - Use this tool to select or move existing elements of your report layout. This is the default tool (it becomes the active tool automatically after you place a report element using one of the other tools).



**Add Text Field** - Adds a text field to the report layout wherever you click. RPTwin then opens the Text Properties dialog so you can type in the desired text.



**Add Formula** - Adds a [data field](#) to the report layout wherever you click. RPTwin then opens the Formula Editor so you can define a formula for that data field.



**Add Page Break** - Adds a page break to the report layout wherever you click.



**Add Time** - Adds a special data field to the report layout wherever you click, containing a formula that displays the time of day when the report is printed.



**Add Page Number** - Adds a special data field to the report layout wherever you click, containing a formula that displays the number of the page when the report is printed.



**Add Date** - Adds a special data field to the report layout wherever you click, containing a formula that displays the date when the report is printed.



**Add Record Number** - Adds a special data field to the report layout wherever you click, containing a formula that displays the current record number.



**Add Count of Records** - Adds a special data field to the report layout wherever you click, containing a formula that displays the count of records processed in the current group. If there are no groups defined, then the Count [function](#) shows the number of records processed in the entire report.



**Add OLE Object** - Adds an OLE object to the report wherever you click. Upon object placement, you can define the specific object server application being used. Most Windows application can act as OLE object servers. For example, a Microsoft Word document can be generated as part of the report in any location you wish. The new document is embedded in the report, but is attached to Word as a Word formatted document.

## The Toolbar

The Toolbar is located near the top of the [Design window](#) under the Menu bar. The Toolbar provides a quick way to access many of the common functions of the Menu bar. To make a Toolbar button active, simply click

on it.



**New Report** - Same as selecting New from the File menu (starts a new report). Pick an [RPTwin report data file](#) first (an .LWD file), then a report layout (e.g., columnar or vertical).



**Open Report** - Same as selecting Open from the File menu (opens an existing [RPTwin report definition file](#) (an .LWR file).



**Save Report** - Same as selecting Save from the File menu (saves your current report layout, asking you to name it if you have not saved it before).



**Print Report** - Same as selecting Print from the File menu (prints the report on your current printer).



**Preview Report** - Same as selecting Print Preview from the File menu (brings up a preview of what your report would look like when printed).



**Snap to Grid** - Click this button to align objects to a fixed grid spacing. Click again to move objects freely. (Same as selecting Snap to Grid from the Layout menu.)



**Font** - Specifies the font to be used in the currently selected text or [data field](#).



**Font Size** - Specifies the size of the type to be used in the currently selected text or data field.



**Bold** - Click this button to bold the selected text (or the value in a data field). Click again to turn bold off.



**Italic** - Click this button to italicize the selected text (or the value in a data field). Click again to turn italics off.



**Underline** - Click this button to underline the selected text (or the value in a data field). Click again to turn underline off.



**Double Underline** - Click this button to double underline the selected text (or the value in a data field). Click again to turn underline off. Useful for totals in financial reports.



**Left Justify** - Click this button to left justify the selected text (or the value in a data field).



**Center Justify** - Click this button to center justify the selected text (or the value in a data field).



**Right Justify** - Click this button to right justify the selected text (or the value in a data field).



**Full Justify** - Click on this button to fully justify the selected text (or the value in a data field).

## The Status Bar

The Status Bar, found at the bottom of the RPTwin [Design window](#), provides additional information on items you select from the Menu bar, Toolbar, or ToolBox. For instance, if you click and hold the mouse button on the Open item on the File menu, the message in the Status Bar is "Open an existing report." If you select Cascade from the Window menu, the message is "Arrange windows so they overlap."



The Status Bar can be turned on or off (to provide more screen space) by clicking on the Status Bar item in the View menu. A checkmark appears next to the Status Bar item in the menu when it is turned on.

## The Rulers

The Rulers appear at the left and top of the [Design window](#) to help you align and size items in the report layout. The measurement units for the rulers is either inches or centimeters. You can set the default measurement unit using the Preferences item in the Options menu. To use different units for a particular report (keeping your general Preference as it is), use the Current Layout item in the Options menu.

Notice that there is a different ruler for each [band](#) in your report. Also, notice that when you click the mouse button in a band or move a field, your position is highlighted on both the top and side rulers for easy reference. Rulers are especially handy if you know the exact dimensions of the report layout you are trying to produce, or if you need to place fields in precise locations. Rulers indicate dimensions inside the margins of your report, not paper size.

The Rulers can be turned on or off by clicking on the Ruler item in the View menu. A checkmark appears next to the Ruler item in the menu when the Rulers are on.

## The Grid

The [Grid](#) is an optional array of dots in the [Design window](#) that helps you align the elements of your report. If Snap To Grid is turned on (either from the Toolbar or from the Layout menu), elements placed or moved on the screen automatically align their edges with the dotted grid lines. Snap -to-Grid works whether or not the grid is shown or hidden. In either case, the dots of the grid do not appear in a previewed or printed report.

## RPTwin Menus

The following chart summarizes the purpose of each RPTwin menu option.

Menu	Option	Select This Option to
FILE	<b>New</b>	Create a new report from a .LWD report data file.
	<b>Open</b>	Open a previously saved .LWR report definition file.
	<b>Close</b>	Close the current report.
	<b>Save</b>	Save the current report with the same name and in the same location.
	<b>Save As</b>	Save the current report with a different name and/or in a different location.
	<b>Print Preview</b>	Preview a WYSIWIG version of the current report online.
	<b>Print Setup</b>	Open the Print Setup dialog to set the printing options.
	<b>Print</b>	Send the current report to the printer.
	<b>Exit</b>	Close any open reports (RPTwin prompts you to save any report that has changed since it was last saved) and exit RPTwin.
	EDIT	<b>Undo</b>
<b>Cut</b>		Remove the currently selected report object and place it in the clipboard.
<b>Copy</b>		Place a copy of the currently selected text or object into the clipboard.
<b>Paste</b>		Put the current contents of the clipboard back into the report.
<b>Paste Special</b>		Put the current contents of the clipboard into the report while allowing you to control the format of the pasted information and to establish a link with the information source (for example, a Microsoft Excel worksheet).
<b>Select All</b>		Select all the objects in the report (RPTwin displays a selected object in a box with "grabber handles" at its four corners).
<b>Delete</b>		Deletes the selected object from the report.
<b>Reset Object Size</b>		Resets the size and orientation of an OLE object to its original dimensions.
<b>Links</b>		Displays the path to a linked OLE object inserted in a report (only when you insert an OLE object as a link rather than as embedded).
<b>Object</b>		Lets you edit an OLE object inserted in a report. OLE object types vary, since they are provided by OLE servers that handles that kind of object (e.g., Microsoft Paintbrush).
VIEW	<a href="#"><u>DataSet Columns List</u></a>	Toggle on/off to show/hide a floating list box that includes all the columns for the current report.
	<b>ToolBox</b>	Toggle on/off to show/hide a floating palette that includes all the tools for adding text and objects to the report.
	<b>Toolbar</b>	Toggle on/off to show/hide the Toolbar at the top of the <a href="#"><u>Design window</u></a> .
	<b>Status Bar</b>	Toggle on/off to show/hide the Status Bar at the bottom of the Design window.

	<b>Ruler</b>	Toggle on/off to show/hide the ruler at the top of the Design window.
<b>INSERT</b>	<b>DataSet Column</b>	Choose a DataSet column for placement on the report.
	<b>Text Field</b>	Create a free form text field in the report.
	<b>Formula Field</b>	Place calculations usually based on other field values.
	<b>Page Break</b>	Insert a page break.
	<b>Special Field</b>	Insert special fields such as: Date, Time, Page Number, Record Number and Record Count.
	<b>OLE Object</b>	Insert an OLE (Object Link and Embedding) object in a report that is linked to an available OLE server application (PC Paintbrush, MS Excel, MS Word, etc.).
<b>LAYOUT</b>	<b>Sorting and Grouping</b>	Open the Sorting/Grouping dialog that lets you control how RPTwin sorts and groups the information in the report.
	<b>Report Header</b>	Toggle on/off to add or remove a Report Header <b>band</b> . When you remove a band, RPTwin deletes all objects in the band.
	<b>Report Footer</b>	Toggle on/off to add or remove a Report Footer. When you remove a band, RPTwin deletes all objects in the band.
	<b>Page Header</b>	Toggle on/off to add or remove a Page Header band. When you remove a band, RPTwin deletes all objects in the band.
	<b>Page Footer</b>	Toggle on/off to add or remove a Page Footer band. When you remove a band, RPTwin deletes all objects in the band.
	<b>Snap to Grip</b>	Toggle on/off to automatically align objects with the <b>grid</b> in the Design window.
	<b>Align</b>	Position selected objects to the left, right, top, bottom, or at the baseline within the field area (RPTwin displays a selected object in a box with "grabber handles" at its four corners).
<b>LAYOUT</b>	<b>Band Properties</b>	Open the Band Properties dialog to define properties such as height, width, patterns, and printing breaks for the current report band.
	<b>Field Properties</b>	Open the Field Properties dialog to define properties such as height and width, formulas, format, and position for the current field.
	<b>Page Layout</b>	Open the Page Layout dialog to define page orientation, margin settings, and number of columns for the current report.
<b>OPTIONS</b>	<b>Current Layout</b>	Open the Current Layout dialog to define grid properties, measurement units, and other design elements for the current report layout.
	<b>Current DataSet</b>	Open the Current DataSet dialog to verify and/or select the DataSet linked with the current report.
	<b>Preferences</b>	Open the Preferences dialog to set the default preferences for the design environment such as data formats, margin settings, measurement units, and other design elements for all new reports.
	<b>Filter</b>	Open the Filter dialog to define a subset of columns that you want to include in the current report.

<b>WINDOW</b>	<b>Cascade</b>	Display all open reports in a cascade arrangement (overlapping windows with title bars showing).
	<b>Tile</b>	Display all open reports in a tile arrangement (reduced size windows that fit next to each other in the RPTwin desktop).
	<b>Arrange Icons</b>	Arrange icons of minimized report windows.
<b>Help</b>	<b>Help Topics</b>	Display a list of the RPTwin online help contents from RPTwin online help system.
	<b>Using Help</b>	Open the How to Use Help section of the Windows online Help system.
	<b>About Logic Works RPTwin</b>	Open a dialog to view RPTwin version and serial number information.

## Creating a New Report

There are two ways to launch RPTwin and start a new report. [ERwin](#) and [BPwin](#) automatically open RPTwin when you save a [RPTwin report data file](#) (.LWD). You can also double-click on the RPTwin icon in the Windows Program Manager and select the New option on the RPTwin File menu. In either case, RPTwin displays the New Report dialog as shown below.

The New Report dialog lets you choose the type of report you want to create. There are two classifications of possible reports:

- [Quick Reports](#)
- [Guided Reports](#)

**Quick Reports** automatically creates standard report layouts for you.

There are three Quick Reports:

<b>Columnar</b>	For simple tabular reports where data is laid out in a series of columns across the page. RPTwin automatically creates a default <a href="#">columnar report</a> with all data file objects. You can then make any desired changes.
<b>Vertical</b>	For single column reports where each record of data is laid out vertically, one after the other. Example uses are contact sheets, names and phone numbers, etc. RPTwin automatically creates a default vertical report with all data file objects. You can then make any desired changes.
<b><a href="#">Blank Report</a></b>	To design your report from scratch with a completely blank design canvas (no default placed objects). In many cases, you may want to start out with one of the other report formats and make your modifications from there.

**Guided Reports** lead you through a simple, step-by-step process to create reports with many custom features. See [Creating Guided Reports](#) for detailed instructions.

There are two Guided Reports:

<b>Group/Totals</b>	For columnar reports with automatic <a href="#">grouping</a> and <a href="#">sorting</a> , complete with headers, footers, and totals.
<b>Vertical</b>	For Vertical reports where you control which data should appear, in what order, and with optional sorting.

To choose one of these report styles, click the appropriate button in the New Report dialog.

## Using an Existing Report

At any time, you can call up a report layout template (\*.LWR) that you have previously used and saved. Click the Open button on the Toolbar, or choose "Open" on the File menu. A dialog appears, allowing you to choose from a list of existing report layouts. The report opens in Design mode where you can Print it, Preview it on the screen, or make layout changes.

Every report file saved on disk (as a .LWR file) contains a reference to a corresponding [DataSet](#) (.LWD file). This relationship is established when you first create the report. Each time you enter report design mode, RPTwin looks for the matching DataSet and uses its data for report design. See

[Changing the DataSet](#) for more information on using a report layout with a different DataSet.

If RPTwin cannot find the DataSet associated with your chosen report layout, RPTwin prompts you to locate the correct DataSet on your disk. RPTwin warns you if you choose a DataSet that does not correspond with your original one (non-matching column names and data types, etc.). If you proceed with that DataSet anyway, you need to adjust your Data Field's formulas so that they refer to appropriate columns in your new DataSet. Otherwise when you print or preview the report, you may see the words Bad [Formula](#) appearing wherever formulas no longer make sense with the new DataSet. See

[Changing the DataSet](#) for more information on using different DataSets.

## Working With Bands

Bands are horizontal sections on your report layout with physically related information. They are a useful tool to visually lay out the different parts of your report on the screen. Bands (identified by titles such as [Report Header](#) or [Page Footer](#)) are report areas where you insert text and data that usually belong together. For example, report header bands usually contain a report title, current date, and perhaps a current time. Likewise, [report footer](#) bands usually contain page numbers. All reports **must** have at least a [Detail Band](#) since it contains the actual data being reported. All other bands are optional (Report Header and Footer, Group Headers and Footers, [Page Header](#) and Footer).

### Related Topics:

[Creating and Removing Bands](#)

[Understanding Bands](#)

[Expanding or Shrinking Bands](#)

[Modifying Properties of Bands](#)

## Creating and Removing Bands

You create new bands using the Layout menu. Click on the [band](#) type you want to create. A checkmark appears next to the selected band to indicate it is included in the report layout. Click again to remove it from your layout. Group Headers and Footers are dealt with in a different way. See [Sorting and Grouping](#) for more information.

## Understanding Bands

Bands have a shaded border at the top (containing the [Band](#) title, e.g., [Report Header](#)), and a layout area below where data and text can be inserted

Bands are often used in pairs (although they don't have to be). In the sample report, the Report Header is paired with the [Report Footer](#), and the [Page Header](#) is paired with the [Page Footer](#).

Notice that each Band border contains a bold horizontal line at the left. The horizontal lines for each pair of bands match in length to indicate that they are on the same organization level in your report. The length of the line shows the relative location of the band in the report organization hierarchy. For example, Report Header and Report Footer (which have the shortest lines) are at the top of the hierarchy. Likewise, the [Detail band](#) contains the detail data for your report and is always found at the innermost level of the hierarchy (with the longest lines).

The lines at the top of the Page Header and Page Footer are longer than those of the Report Header and Report Footer, indicating that these elements are deeper in the hierarchy. As you add nested bands (such as Groups and Subgroups, discussed later in this section), you can check these lines to see where you are in the report hierarchy.

## Expanding or Shrinking Bands

You can expand or shrink a [band](#) by clicking on the very bottom of the band's layout area, holding the mouse button down, and dragging the band up and down like a window shade. When you grab the band, the mouse cursor changes to .

If you have a band with no information in it and you want to hide it in your printed report, you can completely shrink the band so that its border touches the band above it. You cannot shrink a band completely to cover up any information already contained in the band.

## **Modifying Properties of Bands**

Bands have various properties such as height, borders, patterns, and print options. Each of these properties can be changed in the [Band Properties](#) dialogs. See [Changing Band Properties](#) for more information.

## Data Fields

Data Fields contain the actual data in your report. This data can come from the columns in your RPTwin [DataSet](#) or from formulas and calculations (totals, averages, etc.) that you create. Data Fields and [Text Fields](#) are the two most fundamental field types found in RPTwin. There are three general categories of Data Fields that you can place in your report:

**Simple Data Fields** Displays the name of a column from your DataSet. For example, ENTITY NAME and ATTRIBUTE NAME are DataSet Columns.

**Special Functions** Add special types of data to your report, like date, time, or page number. These functions can be accessed from the ToolBox.

**Formulas** Let you create calculations or manipulate your data in many ways. You can access the [Formula](#) Editor by using the X+Y button located in the ToolBox (to make a new field), or by double clicking on an existing [data field](#) in your report.

Data Fields are most commonly placed in the [Detail Band](#) of the report, but they can also be placed in headers and/or footers. For example, it is very common to place formulas like Sums and Averages in Group Footers and the [Report Footer](#). There are three ways to add a Data Field to your report. You can add a DataSet column name, a special [function](#) such date or time, or a formula.

See [Changing Field Properties](#) about setting Data Field properties.

**Note:** OLE objects are not considered data fields in RPTwin. See [OLE Objects](#) later in this section.

### Related Topics:

[To add a Simple Data Field to a report](#)

[To add a Special Function to a report](#)

[To add a Formula to a report](#)

[Changing a Data Field/Editing Formulas](#)

**To add a Simple Data Field to a report**

1. Click [DataSet Columns List](#) on the View menu to display the list of DataSet Columns.
2. Click the mouse on the desired DataSet Column, hold the button down, drag the column to the desired position in your report, and release the mouse button.

### **To add a Special Function to a report**

1. Click ToolBox on the View menu to display the ToolBox.
2. Click the ToolBox icon for the desired special [function](#) (Page number, time, etc.) and release the mouse button.
3. Move the cursor to the desired spot in the report, and click the mouse button again to place the field.

### To add a Formula to a report

1. Click the ToolBox option from the View menu to display the ToolBox.
2. Click on the  button in the ToolBox. Move the cursor to the desired location and click the mouse. The [Formula](#) Editor displays.
3. Complete the Formula Editor dialog with a formula and click the OK button. The Data [Field Properties](#) dialog displays.
4. Click OK from the Data Field Properties dialog.

**Note:** Using the Formula Editor, you can create or modify the formulas in Data Fields. Formulas can be composed of DataSet Columns, standard functions (such as Sum, Average, Max, Min, and a great many more), and/or custom expressions. See [Understanding Formulas](#) for more information.

## Changing a Data Field/Editing Formulas

You create and edit formulas in RPTwin using the [Formula](#) Editor. There are two ways to access the Formula Editor. To create a new formula, click the X+Y button in the ToolBox. To change or add a formula in an existing [Data Field](#) in your report, double-click on the Data Field itself. RPTwin opens the [Data Field Properties](#) dialog. Click the Formula Editor button in the upper right corner of the Properties dialog to open the Formula Editor.

The top Formula box is where you build your formula. The rest of the dialog contains tools that help you quickly create a wide variety of formulas.

If you enter the Formula Editor by double-clicking in an existing Data Field, the current contents of that Data Field appear in the Formula box. If you use the X+Y button in the ToolBox to create a new formula, the following message appears in the Formula box:

Enter Formula Here

You build up your formula by adding a combination of [DataSet](#) Columns, Functions, and/or Operators. You can also use the numeric keypad and edit buttons provided to insert numbers and use cut, copy, and paste techniques to create the formula.

To add a DataSet Column, [Function](#), or Operator, double-click on the desired item in the appropriate list. The chosen item is immediately placed in the formula window at the top of the Formula Editor.

Some basic hints for creating formulas are described below. See [Understanding Formulas](#) for more information on formulas. See [Overview](#) for a complete list of functions you can use in a formula.

### Related Topics:

[Concatenation](#)

[Adding Text to a Formula](#)

## **Concatenation**

You can concatenate (combine) data items by using the ampersand (&) key on the [Formula](#) Editor keypad, or you can type the ampersand from the keyboard.

For example, suppose you have a report with the [DataSet](#) Columns CUST\_NAME and CUST\_IDENT, and you want to concatenate them into a single field. To do this, double-click on the CUST\_NAME DataSet Column, click once on the & key, then double-click on the CUST\_IDENT DataSet Column. This produces the formula:

**CUST\_NAME & CUST\_IDENT**

When printed, your report concatenates the data:

**Anthony12345**

**Miller12350**

**Turner12355**

If you want to leave a blank space (or some other separator) between concatenated values, insert a space character between a set of double quotes as shown in the following formula:

**FIRST\_NAME & " " & LAST\_NAME**

This report output looks like this:

**Ralph Bunker**

**Dale Larson**

**Pete Harlan**

The keypad in the Formula Editor provides a handy tool for concatenating data items; it automatically inserts a pair of quotation marks ("" ) and places the cursor between the quotes for easy insertion of new text.

**Note:** Keep in mind that RPTwin is normally used to report on [ERwin](#) and [BPwin](#) model information, not the data contained in their respective databases. The formula examples convey database examples for the purpose of clarity.

## ***Adding Text to a Formula***

---

To add text to a [Data Field formula](#), you also must use quotation marks. For example, if you want to add the following text

**Salesperson:**

to the LAST\_NAME data, type the following (including the quotes and ampersand):

**"Salesperson: " & LAST\_NAME**

Be sure to include a space before the second quotation mark. Anything that is enclosed in quotes is applied literally to the formula in the Data Field. This formula generates report output that looks like this:

**Salesperson: Langbehn**

If you want quotation marks to actually appear in your report, use the Escape Character (the backslash: \) in front of each quotation mark. For example, the formula:

**"The \"Ace\" Salesperson: " & LAST\_NAME**

generates the following output:

**The "Ace" Salesperson: Smith**

See [Understanding Formulas](#) for more information.

## Using Live Data in Your Report Layout

When you add Data Fields to your report layout, the [formula](#) for each field appears on your layout screen. By default, RPTwin displays the column name or formula you set up for each field. You can, however, choose to lay out your report with actual sample data from the current [DataSet](#). Using sample data makes it easier to visualize how the final report should look, and can give you a better idea of spacing as you lay out the report.

To work with live data, select the Current Layout option on the Options menu. In the Current Layout dialog, click in the Show Data Instead of Formulas option so that an X appears in its checkbox. To show formulas instead of live data, click again to uncheck the box.

**Note:** The live data is taken from the first row in your DataSet. If you have previewed or printed the report during this work session, then the live data is taken from the last row that was previewed or printed.

## Text Fields

---

[Text Fields](#) display titles, labels, and other identifying information to organize and clarify your report. A Text Field can consist of characters, numbers, or symbols. Text is anything that does not come from your [DataSet](#). For example, if you want the text **747** in your report you must enter it as text field because it does not come from your DataSet.

### Related Topics:

[Adding Text to the Report](#)

[Editing a Text Field](#)

## Adding Text to the Report



To add text to your report, use the Text tool in the ToolBox. When you click on the Text button, the cursor changes to a box, and you can position it anywhere in your report. Click again in the layout area where you want the Text Field to appear. When you release the mouse button, RPTwin displays the Text Properties dialog. Type your new text in the Text box and click OK. The new text field appears on the design canvas.

## Editing a Text Field

To change the contents of an existing Text Field, double-click in the Text Field. RPTwin opens the Text Properties dialog with the existing text highlighted in the Text window. You can change all or part of the existing text, or just type your new text. Click OK for the new text to appear in place of the old text.

See [Changing Field Properties](#) for more information.

## **Working With Fields**

You can perform many of the following operations on several fields at once. See [Selecting Multiple Fields at Once](#) for more information.

### **Related Topics:**

[Deleting a Field](#)

[Moving a Field](#)

[Copying a Field](#)

[Resizing a Field](#)

[Changing Font Settings](#)

[Changing Text Properties](#)

[Selecting Multiple Fields at Once](#)

## **Deleting a Field**

To delete a field from your report, click once anywhere in the field to select it (the field is selected if four square grabber handles appear at the corners) and press the DELETE key. Or, select the field and choose Cut from the Edit menu (or press CTRL-X). You may later Paste the field wherever you like – even in another [band](#) or report layout – after you Cut it, using Paste from the Edit menu (or press CTRL-V). The field is pasted in the same relative position in the new band as in the band from which it was Cut.

## **Moving a Field**

To move a field in your report layout, position the cursor anywhere in the field, click and hold the mouse button down, drag the field to the desired location, then release the mouse button. If the Snap to [Grid](#) option is turned on, you can move the field to any intersection of grid lines. If this option is turned off, you can move the field freely about the layout area.

To temporarily override the Snap to Grid [function](#), press and hold the CTRL key while you drag the field to the desired location.

## Copying a Field

To copy a field, click once anywhere in the field to select it (the field is selected if four square grabber handles appear at the corners). Then choose Copy from the Edit menu (or press CTRL-C). You may later Paste the field wherever you like using Paste from the Edit menu (or press CTRL-V). The field is pasted in the same relative position in the new [band](#) as it was in the band from which it was copied. If you Paste into the same band that you copied it from, the new Pasted copy appears on top of the original. Just move the new field off of the original one to reveal both copies.

## **Resizing a Field**

To resize a field, click anywhere in the field to select it (the selected field shows four square grabber handles at the corners). Click on one of the grabber handles, hold the mouse button down, and drag the corner to a new position. In this way, you can make the field taller, wider, or smaller.

When resizing a field, you may have to enlarge the field to see all of its contents, either because the contents are too long to fit in the field box, or because the font size is too large to fit. If the contents don't look right in the field, select the Word Wrap option from the Data or Text Properties dialog. See [for more information](#).

## **Changing Font Settings**

To change the Font and type size of text, use the Font and Size buttons, respectively, on the Toolbar.

When you click on one of these buttons, RPTwin opens the related drop down list showing all of the available font types or sizes. Click the desired font or size selection to reformat the field contents.

If you resize or change your font, and the field contents do not fit in the field box, resize the field.

## Changing Text Properties

Field contents can be **bolded**, *italicized*, underlined, double underlined, and justified using the appropriate buttons in the Toolbar. To format a field, click anywhere in the field to select it, then use any of the desired formatting buttons in any combination.

When a field is selected, you can tell which format options apply to that field by looking at the state of the relevant buttons in the Toolbar. If a button looks pushed in, that option is enabled for the selected field.

To turn off a format option, click on the appropriate Toolbar button to reset it.

See [for more information](#).

## Selecting Multiple Fields at Once

You can change [field properties](#) on more than one field at a time. For example, you can bold, italicize, underline, justify, change the font and font size, delete, copy, paste or move multiple fields all at once. There are two ways to select multiple fields:

1. Select the first field by clicking on it. Then, press and hold the SHIFT key and click on the next field you want to select. Notice that both fields are now active with the "grabber" handles. You can continue to select additional fields by Shift-clicking on them in this way.
2. Or, hold the mouse button down and drag a rectangle around all of the desired fields. When you release the mouse button, any field partially or completely encompassed by the rectangle is selected.

When you have selected all the desired fields, applying a formatting operation, such as underlining. All selected fields are changed at the same time.

## **OLE Objects**

OLE (Object Linking and Embedding) is a powerful RPTwin extension that allows embedding and linking of objects created by other Windows applications. Objects such as Microsoft Word documents, Lotus Spreadsheets, bitmap image graphics among many others can be created, linked and embedded into any RPTwin report. It makes no difference which application is being used. If the application is Windows based and resides in your environment, you can choose or define the specific application, create the object and place it on the report.

### **Related Topics:**

[Inserting an OLE Object](#)

[To insert an OLE object in a report](#)

## Inserting an OLE Object

Inserting an OLE object involves selecting the OLE tool from the toolbox or "OLE Object" from the Insert menu, clicking on the report area for the object to appear, and choosing an object type from the Insert Object dialog.

With Create New as the default and using Microsoft Paint as an example, Microsoft Paint will become the object application server and run within the RPTwin workspace where you can create a new bitmap image from scratch. If you select Create from File, you can open an existing bitmap using the Browse button to locate the file.

The Link check box establishes a link between the Object and RPTwin so that changes made to the object external of RPTwin will be automatically updated in RPTwin. If Link is not used, the object in RPTwin is independent of the server application and can only be accessed within RPTwin.

The optional Display as Icon check box displays a file icon in place of the object itself. This is useful as a placeholder for the object during the design process and when printed.

During the creation (or editing) of an object, the application server and RPTwin **function** simultaneously where only certain applicable functions are allowed. For example, you cannot open another object file while creating an object. During the creation process, you will see a bordered area on the RPTwin design canvas for the new object. (See for more information)

When you finish creating or editing the object, click outside of the application workspace. The object (e.g., spreadsheet, chart, etc.) then appears on the RPTwin design canvas, and subsequently in the report when previewed or printed.

**Note:** Server applications have varying behavior under OLE. In most cases, application menus are integrated with RPTwin menus that allow varying degrees of functionality. For example, some Windows applications allow you to use the Close option from the File menu to return back to RPTwin.

## To insert an OLE object in a report

---

1. Click the ToolBox option from the View menu to display the ToolBox.
2. Click the  button on the toolbox. Move the cursor to the desired location on the report and click the mouse button. The Insert Object dialog displays.
  - If creating a new object, verify that the Create New radio button is selected and choose an object type from the Object Type list box (e.g. Microsoft Excel Chart). The associated application will open within the RPTwin window (Microsoft Excel). Create the object to be inserted.
  - If you want to embed a previously created object, click on the Create From File radio button and enter the path or use the Browse button. Select the Link check box if you want to link the object to the application for external editing purposes.
3. Click outside of the OLE application area. The application closes and the object remains embedded in the RPTwin report.

**Note:** OLE objects can also be inserted from the Insert menu and the floating menu made active with the right mouse button.

See your Microsoft Windows documentation for more information regarding OLE.

# Working With OLE Objects

Once inserted on the RPTwin design canvas, OLE objects can be selected like fields. You can also select more than one OLE object at once, but only for the purpose of moving objects. When selected, objects can be moved, copied, deleted, converted, resized, and opened for edits in the server application. Multiple OLE objects cannot be opened at the same time. (See for more information)

## Related Topics:

[Converting an OLE Object](#)

[Deleting an OLE Object](#)

[Resetting an OLE Object Size](#)

[Editing an OLE Object](#)

[Moving an OLE Object](#)

[Copying an OLE Object](#)

## Converting an OLE Object

Some OLE objects can be converted within the application they were created. For example, a Microsoft Word Document can be converted to a Microsoft Word Picture. This may be important for certain drawing features you want included in the RPTwin report. You cannot convert OLE objects across applications, such as from Microsoft Excel to Microsoft Word.

To convert an OLE object, click once anywhere in the field to select it (the field is selected if four square grabber handles appear at the corners). Then choose the "Object" menu item at the bottom of the Edit menu (or from the floating right mouse button menu), followed by "Convert". The Convert dialog then displays.

**Note:** The Object menu item is preceded in name by the type of object that you just selected. For example, a selected bitmap image would appear in the menu as "Bitmap Image Object". The same is true if you use the floating menu accessed with the right mouse button.

Choose the new object type to convert to and click OK. In many other object server applications, the "Activate as" radio button is available as an option to activate future objects as other object types. This of course applies only to objects created by their applications (e.g., Paint, Word, Excel, etc.)

As with inserting an object, the "Display As Icon" option is available when converting an object. See for more information.

## Deleting an OLE Object

To delete an OLE object from your report, click once anywhere on the object to select it (the field is selected if four square grabber handles appear at the corners). Then press the DELETE key or choose "Cut" from the Edit menu (or press CTRL-X). The Cut [function](#) saves the object in the clipboard allowing you to later Paste the object wherever you like – even in another [band](#) or report layout. The Paste (or press CTRL-V) function is also in the Edit menu.

## Resetting an OLE Object Size

To reset an OLE object to its original size, click once anywhere on the object to select it (the field is selected if four square grabber handles appear at the corners). Then choose Reset Object Size from either the Edit menu or the floating menu made active by the right mouse button. The object's size is then restored to its original dimensions.

**Note:** Resetting an OLE object is not defined by RPTwin, but by the server application. Some applications do not reset objects to their original size and dimensions.

## **Editing an OLE Object**

---

To edit an OLE object, simply double-click on the object itself. The object server application automatically starts with that object in the RPTwin workspace. You can also click on the object and choose the "Object" menu item from the Edit menu. The Edit menu Object item varies depending on the selected object type (Document Object, Chart Object, Bitmap Image Object, etc.). Then select "Edit" to edit the object using its application. When finished with your changes, click outside of the application window to close it. The updated object remains on the RPTwin design area.

You can also use the "Open" menu item from "Object" in the Edit menu to edit the object. The only difference is that the object server application opens externally of RPTwin. When finished with changes, simply close the application window. The updated object remains on the RPTwin design area.

## **Moving an OLE Object**

To move an OLE object in your report layout, click on the object and drag it to the desired location. If the Snap to [Grid](#) option is turned on, the object will automatically snap to the closest intersection of grid lines. If this option is turned off, you can move the object freely about the layout area.

To temporarily override the Snap to Grid [function](#), press and hold the CTRL key while you drag the field to the desired location.

## **Copying an OLE Object**

To copy an OLE object, click once anywhere on the object to select it (the object is selected if four square grabber handles appear at the corners). Then choose Copy from the Edit menu (or press CTRL-C). You may later Paste the object wherever you like using Paste from the Edit menu (or press CTRL-V). The object is pasted in the same relative position in the new [band](#) as it was in the band from which it was copied. If you Paste into the same band that you copied from, the pasted object copy appears on top of the original. Simply drag the new object off of the original to its destination.

## Working With Headers and Footers

Headers and footers allow you to organize and identify the different sections of your report.

You can include any kind of report element in your headers and footers. Some common uses for the different headers and footers are:

<a href="#">Report Header</a>	Report title or Report Cover Page(s). Appears once at the beginning of the report.
<a href="#">Page Header</a>	Column names, report title (repeated), etc. Appears at the top of each page.
<a href="#">Group Header</a>	Name of the current group. Appears at the top of the group.
<a href="#">Group Footer</a>	Group totals or other summary information. Appears at the bottom of the group.
<a href="#">Page Footer</a>	Pagination, date, etc. Appears at the bottom of each page.
<a href="#">Report Footer</a>	Grand totals or other summary information. Appears once at the end of the report.

### Related Topics:

[Adding Headers and Footers](#)

[Removing Headers and Footers](#)

[Adding Text and/or Data to Header and Footer Bands](#)

[Modifying Properties of Headers and Footers](#)

## **Adding Headers and Footers**

You can turn on different headers and footers by selecting [Report Header](#), [Report Footer](#), [Page Header](#), or [Page Footer](#) from the Layout menu. A check mark appears next to each header or footer that is currently turned on. Group Headers and Footers are created in a special way. See [for more information](#).

## Removing Headers and Footers

You can turn off headers or footers by again selecting the [Report Header](#), [Report Footer](#), [Page Header](#), or [Page Footer](#) option in the Layout menu. RPTwin removes the check mark in the menu next to the element you are removing. If you remove a Header/Footer, all the information in the [band](#) is also permanently deleted. Group Headers and Footers are removed in a special way. See [for more information](#).

## **Adding Text and/or Data to Header and Footer Bands**

The process of adding text, data or other elements to the header and footer bands is the same as adding these elements to any other part of the report.

## **Modifying Properties of Headers and Footers**

The Properties of headers and footers include such items as height, location, printing options, borders, and patterns. You can modify all of these properties using the [Band Properties](#) dialog for the particular header or footer. See [Band Properties](#) for more information.

## Page Layout Options and Preferences

RPTwin provides four dialogs to let you control the general layout of your reports, and specify your preferences for how certain report features should work.

- Print Setup**      Accessed from the File menu, this dialog lets you set printer options that apply to all new reports you create in the current work session. These options may be set differently from those in Page Layout.
- Page Layout**      Accessed from the Layout menu, this dialog lets you set page and printer options that apply to the report you are currently working on (only). These options may be set differently from those in Print Setup.
- Current Layout**      Accessed from the Options menu, this dialog lets you set layout options that apply to the report you are currently working on (only). These options may be set differently from those in Preferences.
- Preferences**      Accessed from the Options menu, this dialog lets you set default layout options that apply to all new reports you create (unless you override an option using the Current Layout dialog). These options may be set differently from those in Current Layout.

### Related Topics:

[Print Setup Dialog](#)

[Page Layout Dialog](#)

[Current Layout Dialog](#)

[Preferences Dialog](#)

[Changing the Overall Report Width](#)

## **Print Setup Dialog**

Accessed from the File menu, this is a standard Windows Print Setup dialog where you set the Printer Type, Page Orientation (Portrait or Landscape), and Paper Size.

Changes made to your Print Setup (for instance, switching to Landscape orientation) affects all new reports you create in the current work session. When you close RPTwin, the settings revert to your Windows defaults. If you want to make these changes permanent, use the Printers option in the Control Panel program that comes with Windows.

## **Page Layout Dialog**

Accessed from the Layout menu, this dialog lets you override the current default Printer Type, Page Orientation, and Paper Size (as shown in Print Setup) for your current report. Page layout settings are stored with each report layout so that each report retains its individual settings. You can also use this dialog to set your report margins and the number of columns for the report.

### **Related Topics:**

[Setting Margins](#)

[Setting Multiple Columns](#)

### ***Setting Margins***

The margin settings in the Preferences dialog determine the margins of all new reports. You may enter different values for your current report in the Page Layout dialog. Margin values refer to the Units of Measure— inches or centimeters— as set in the Current Layout dialog.

**Note:** Changing the left or right margins controls how much data can be printed on a given page. It does not determine how wide your report is (since a report can take up many pages in width). You can get a sense of how wide your report is by looking at the Design window. A dotted vertical line appears in the report layout indicating where the right margin falls.

## ***Setting Multiple Columns***

---

A report can have one or more columns. In a multi-column report, the data in the [Detail Band](#) runs down the first column, then back up to the top and down the second column, then down the third column – and only then does it go on to the next page. To change the number of columns in the Page Layout dialog, type a value in the Number of Columns box to specify the default for new reports.

**Note:** Changing the number of columns only affects a Vertical report layout.

## Current Layout Dialog

---

Accessed from the Options menu, the Current Layout dialog lets you control the report layout options for the current report such as [data field](#) formats, margins settings, measurement units, and [grid](#) properties.

### Related Topics:

[Snap Objects To Grid](#)

[Show Grid](#)

[Subdivisions](#)

[Measurement Units](#)

[Show Text Borders](#)

[Add Names to New Data Fields](#)

[Show Data Instead of Formulas](#)

[Enable Case Sensitive Sort Options](#)

[Leading Zero on Days and Months](#)

[1000 Separator](#)

[Decimal Separator](#)

[Negative Format](#)

[Currency Symbol](#)

[Symbol Position](#)

[Curr. Neg. Format](#)

### ***Snap Objects To Grid***

The [Grid](#) is an optional array of dots in the [Design window](#) that helps you align the elements of your report.

If Snap To Grid is turned on, an element placed or moved on the screen automatically snaps to this grid, aligning its edges with the lines of the grid. When the Snap to Grid option is off, you can move your elements freely about the screen. To temporarily override the Snap to Grid [function](#), hold down the CTRL key when you drag an object to another location. When you release the CTRL key, the Snap to Grid function returns to its previous state.

You can also turn Snap to Grid on and off by clicking the Snap to Grid button in the Toolbar, or by selecting Snap to Grid from the Layout menu in RPTwin.

**Show Grid**

The [Grid](#) may be shown or hidden while working on your report. Snap to Grid works even if the grid is hidden. The grid dots only appear on the design area, never in a previewed or printed report.

### ***Subdivisions***

Subdivisions per X or Y Unit determines the size of the [grid](#). You can change the X (horizontal) and Y (vertical) grid spacing independently of one another. To change the subdivisions, type any number between 1 and 64 in the appropriate box. The default value is 32 (which means the dots are positioned 1/32 of an inch apart). If you use a value above 32 in either X or Y, the snap divisions will operate correctly, but they cannot be shown in the [Design window](#).

If you change the grid density, the location of items previously placed on the screen is not changed.

### ***Measurement Units***

The units in the ruler and in all dialogs can be displayed in either inches (the default) or centimeters.

### **Show Text Borders**

To make a report easier to read, you may want to turn off the boxes around [text fields](#) in the [Design window](#). Click the Show Text Borders box to check or uncheck this option.

***Add Names to New Data Fields***

If checked, this option adds the name of the [DataSet](#) column to the report layout when you drag in a new [Data Field](#) from the [DataSet Columns List](#).

**Show Data Instead of Formulas**

If checked, this option displays actual data from your [DataSet](#) in the report layout, instead of formulas. This is convenient if you want to see how your report layout looks with real data as you are designing it. See [for more information](#).

### ***Enable Case Sensitive Sort Options***

If checked, the Case Sensitive checkbox option in some other dialogs (e.g. the [Sorting](#) and [Grouping](#) dialog in the Layout menu) is enabled so that you can choose it. See [Case Sensitive Sorting or Grouping](#) for more information.

***Leading Zero on Days and Months***

If checked, dates with any leading zeros will be displayed on any previewed or printed report. For example, 1/1/96 would become 01/01/96.

### **1000 Separator**

Choose a character to act as the thousands separator for printed and previewed report numbers. The comma (,) is most common (e.g., 3,129,293).

### ***Decimal Separator***

Choose a character to act as the decimal separator for printed or previewed real numbers. The period (.) is most common (e.g., 3.14).

### ***Negative Format***

Choose a format from the list box to act as the negative format for printed or previewed negative numbers. Depending on the report application, the "-" sign is placed before or after the number in varying formats (e.g., -1, - 1, 1- or 1 -).

### ***Currency Symbol***

Choose a character to act as the symbol for currency in any printed or previewed report. The dollar sign (\$) is most common.

### ***Symbol Position***

Choose a format from the list box to act as the currency symbol position for printed or previewed currency. Depending on the report application, the currency symbol (usually a "\$") is placed before or after the currency amount in varying formats (e.g., \$1, \$ 1, 1\$ or 1 \$).

***Curr. Neg. Format***

Choose a format from the list box to act as the negative format for printed or previewed currency amounts. Depending on the report application, the "-" sign and "\$" symbol are placed before or after the currency amount in varying formats (e.g., - \$ 10, \$ - 10, \$ 10 -, etc.).

## Preferences Dialog

The Preferences dialog lets you control the default settings for report layout options such as [data field](#) formats, margins settings, measurement units, and [grid](#) properties.

### Related Topics:

[Default Data Formats](#)

[Fit All Columns on One Page](#)

[Setting Margins](#)

### ***Default Data Formats***

The Default Data Format options in the Preferences dialog let you set default formatting options for different types of report data. You can, of course, override these default formats at any time for any given field. See for more information.

The different data types are listed in the dialog. To change a default format, click on the arrow in the appropriate pull-down list box, and use the mouse to select the desired format.

The Show Text Borders, Add Names to New Data Fields, Snap Objects To [Grid](#), Show Grid, Measurement Units, Number Formats, and Enable Case Sensitive Sort options in the Preferences dialog are used as described in the [Current Layout dialog](#) section, except they apply to all new reports you create, rather than only to your current report.

### ***Fit All Columns on One Page***

By default, RPTwin lays out a multi-column report using as many pages as necessary to fit all the columns side-by-side. If you select the Fit All Columns on One Page option, RPTwin automatically forces all the columns in a multi-column report to fit on one page.

To fit all columns in a Quick [Columnar report](#), check the Fit all columns on one page option in the Preferences dialog.

To fit all columns in a Guided [Group/Totals report](#), check the Fit all columns on one page option that appears under the columns list in the first dialog that RPTwin opens when you choose that Guided Report.

### ***Setting Margins***

The margin settings in the Preferences dialog affect all new reports that you create. Margin values refer to the Units of Measure – inches or centimeters.

**Note:** Changing the left or right margins controls how much data can be printed on a given page. It does not determine how wide your report is (since a report can take up many pages in width). You can get a sense of how wide your report is by looking at the Design window. A dotted vertical line appears in the report layout indicating where the right margin falls.

## Changing the Overall Report Width

---

The overall width of your report layout is independent of the page size and/or margins. For example, you may have a wide report that prints across several physical pages.

In the [Design window](#), the overall width is represented by the white area, set off from a shaded background. When you scroll to the right in the Design window, you eventually see the edge of the report layout. You can change the overall width in the following way:

Move the cursor over the right edge of the report layout (the cursor changes); then click and drag the edge left or right to the desired size.

If the overall width of your report is larger than the area available inside the margins on the specified paper size, a dotted vertical line appears in the report layout showing where the report breaks across paper pages when it is printed. If you don't want the report to print across pages, find a way to make the overall width smaller (e.g., try Fit All Columns on One Page as described earlier), change the Print Setup to a wider paper size, and/or change the orientation to Landscape so the report prints wide across the paper.

If you are dragging the report edge to the left to make it narrower, you cannot drag it past any field in the report layout. Move or resize these items to the left if you need a smaller overall width.

## **Aligning Fields**

The Align feature, accessed from the Layout menu, helps you line up fields in your report. To use Align, first select the fields you want to align (you must select multiple fields). Then choose Align from the Layout menu, and pick the alignment option you want from the sub menu choices: Top, Bottom, Left, Right, and Baseline.

Top alignment means that all selected fields line up along their top edge. Bottom, Left, and Right alignment work similarly. Baseline alignment means that you will align the fields along their text baselines – that is, an imaginary line drawn right at the bottom of a letter that has no descender (like A or a).

## Sorting and Grouping

RPTwin lets you sort and group the data in a report in any way you need to present it clearly.

[Sorting](#) lets you determine the order in which you want your rows of data arranged. The order can be ascending or descending for the selected data, and you can sort on multiple levels.

[Grouping](#) sorts the report data into several Groups (categories). For example, in a sales report, you might want to group together all sales made by a given salesperson, and get a total for each salesperson as well. To do so, you would group on salesperson and ask for a total of all the sales in that group. You can have any number of groups and subgroups, each of which can have its own header and footer.

To Sort or Group your data, select Sorting and Grouping from the Layout menu. RPTwin opens the Sorting/Grouping dialog.

**Note:** Grouping only makes sense when data is sorted. You cannot choose to Group without Sorting <sup>¾</sup> but you can choose to Sort without Grouping (which essentially means you do not want Group Headers or Footers).

### Related Topics:

---

[Picking a Group or Sort Order](#)

[Rearranging Sort or Group Orders](#)

[Removing Sort or Group Orders](#)

[Sorting or Grouping on a Calculated Field](#)

[Case Sensitive Sorting or Grouping](#)

## Picking a Group or Sort Order

The [DataSet Columns list](#) on the left of the [Sorting/Grouping](#) dialog shows all the column names in your RPTwin DataSet file. Double-click on an item in the DataSet Columns list to automatically place the item into the Sort/Group On list on the right. Or, highlight the item and click the Add button.

The final order of the items in the Sort/Group On list is important. If you have more than one item in the Sort/Group On list, RPTwin sorts first on the first entry, and within that, RPTwin sorts on the second entry, and so forth. You can rearrange the items in your Sort/Group On list.

Use the radio buttons and check boxes beneath the Sort/Group On list to specify whether you want to Sort only, or to Group and Sort. If you Group and Sort, you can further specify whether you want Group Headers or Footers or both. Click on the elements you want. Having both Group Headers and Footers is the default. If you don't want a header or footer, click in the appropriate box to uncheck the item.

You can also choose whether each item should be sorted in Ascending or Descending order.

Click OK when everything is set as you want.

**Note:** If a number appears to the left of a column name in the Sorting/Grouping dialog, it indicates the order in which the data is currently sorted, and the arrows indicate how the data is sorted (Ascending or Descending). You can change the sort or grouping order for your report, but the current sort order may be useful if you want to save time. A long report will run much more quickly if you do not require a new sort or grouping order.

If you choose to Group as well as Sort, the group bands are displayed in the Design windowHASH\_2f12264c with groups and subgroups hierarchically laid out. Nothing appears in these bands until you place the fields and formulas to perform the functions you desire.

**Note:** The Guided Report called "Groups/Totals" automatically fills in your group headers and footers for you <sup>¾</sup> it makes the assumption that you want a total for every numeric field in your report. Try using that Guided Report, and examine the report layouts it creates, to get some useful ideas about laying out your own group headers and footers.

## **Rearranging Sort or Group Orders**

---

If at any time you want to change the Sort or Group order of your report, open the [Sorting](#) and [Grouping](#) dialog. You can change the order of the items in the Sort/Group On list by clicking and dragging the items up or down to new positions in the Sort/Group on list.

## Removing Sort or Group Orders

To remove a group or a sort order from your report, open the [Sorting](#) and [Grouping](#) dialog. Highlight the item to be removed in the Sort/Group On list and click the Remove button. You can only remove items from the Group On list, not the [DataSet Columns list](#).

Click OK to confirm the removal of the group. When the [Design window](#) is displayed, the bands for that group are automatically removed from the report layout.

## Sorting or Grouping on a Calculated Field

Sometimes you may want to group on some data that is not in your [DataSet](#) Column list. For example, ORDER\_DATE may be a column, but you want to Group on Quarters, not individual dates. To sort on Quarters, you need to create a [formula](#) to group on.

To add a formula, click on the "Sort/Group on Calculated Value" button in the [Sorting](#) and [Grouping](#) dialog. RPTwin opens the Formula Editor. Use this dialog as described in the [Editing Formulas](#) section earlier in this section. For example, you would enter the following formula:

### Quarter (ORDER\_DATE)

to group the data by Quarter. When you have created your formula, click OK twice to accept the formula and return to the Sorting and Grouping dialog.

The same technique can be used to sort on a calculated value such as the Sales Total for each quarter.

## Case Sensitive Sorting Sorting or Grouping

By default, [sorting](#) and [grouping](#) are done in a case-insensitive manner. For example, the following names are considered the same for sorting purposes:

**VanAllen**

**vanallen**

**vanAllen**

If you want to perform a case-sensitive sort or grouping, check the Case Sensitive check box in the lower right region of the Sorting/Grouping dialog.

A case sensitive sort would look more like this:

**vanallen**

**vanAllen**

**VanAllen**

Case sensitive sorting prioritizes lower case over upper case.

## Changing Field Properties

RPTwin lets you change the properties of the Data and [Text fields](#) in the [Field Properties](#) dialog.

To open the appropriate Properties dialog, double-click on the field you want to change. Alternatively, click once on a field to select it, then use the Layout menu to choose the appropriate Properties item.

The [Data Field](#) Properties dialog shown below has many more options than the Text Field Properties dialog, which is described later in this section.

### Related Topics:

[Properties Common to Both Data Fields and Text Fields](#)

[Data Field Properties](#)

[Text Field Properties](#)

## Properties Common to Both Data Fields and Text Fields

Although the [Data Field](#) and Text [Field Properties](#) dialogs are different in appearance, they share many of the same features. Properties that appear in both dialogs are explained below.

**Note:** Double clicking an OLE object allows you to edit that object in the OLE server application. See [Editing an OLE Object](#) for more information.

### Related Topics:

---

[Changing a Field's Height or Width](#)

[Changing a Field's Position](#)

[Word Wrap in Fields](#)

[Allowing a Field to Move Automatically](#)

[Working With Repeating Values](#)

[Adding Borders to Fields](#)

[Adding Patterns to Fields](#)

[Adding Field Names](#)

### ***Changing a Field's Height or Width***

These properties control the height and width of a [Data Field](#) or Text Field in a printed report as well as in the [Design window](#). The more common way of setting the height or width of a field is to resize it directly with the mouse, but if you need to set the size to a particular value, you can enter those height and width values in the Height and Width text boxes in the appropriate Properties dialog.

### ***Changing a Field's Position***

The Position property controls the position of a [Data Field](#) or Text Field, relative to the left and top margins of the report [band](#) in which the field is located. It is usually easier just to move the field to the position you want by dragging it with the mouse in the [Design window](#). However, if you need to set a precise distance for certain forms, you can enter the desired values here.

### ***Word Wrap in Fields***

Word wrap allows long data to continue on to the next line if it does not fit within the width of the field you have laid out. If Word Wrap is turned off, data that is too long to fit in the field is truncated.

If you use Word Wrap, you can also specify how you want all subsequent data and text to be moved. When Word Wrap is turned on, you may also want to use the "Can Move Down" option for subsequent fields in the report. If you check the Can Move Down check box and the field contents wrap to a new line and therefore expand the field downwards, the remaining fields in the report are automatically moved down. See [Can Move Down](#) and [Word Wrap](#) for more information.

**Note:** Word Wrap cannot always be used with Data Fields. See [Word Wrap for Data Fields](#), discussed later in this section for more information.

### ***Allowing a Field to Move Automatically***

You can control whether a field will be stationary in your report, or will move to adjust to the needs of the report.

There are two options:

- ◆ Can Move Down
- ◆ Can Slide Left

Notice that when you apply either of these options, the position of the selected field does not change in the **Band** layout area in the [Design window](#), but the output is repositioned appropriately in the Preview and printed copy.

- If Can Move Down is turned on, the field automatically moves down when the data in the field above it expands (when the Word Wrap option applies, a text field expands downward to fit all of its text).
- If Can Move Down is turned off, the field stays in a fixed position and cannot move down, even if the field above it expands downward (due to Word Wrap). If this happens, the expanded field may write on top of the field below it.
- If you select Can Slide Left, the field automatically moves to the left until it encounters either another field or the left margin of the report. In other words, if there is empty space between the selected field and the left margin or any other field on its left, that space is removed, and the selected field is placed to the immediate right of the margin or the previous field.

This is useful to squeeze out extra space between fields. For example, suppose there are two Data Fields: CITY and STATE. If CITY is set to Adjust Width to Data (discussed later), it may shrink to the left. This could leave a gap between CITY and STATE. Applying Can Slide Left on STATE will close the gap.

**Note:** In this example, you will probably want to insert an extra text field in your layout, between CITY and STATE, that contains a comma and a space, so that the State name does not push all the way over and touch the City name. Another technique would be to use ConcatenationConcatenation>main (described earlier), to put everything into one data field, like this: CITY & ", " & STATE

## **Working With Repeating Values**

---

A repeating value is a [Data Field](#) or Text Field in your [Detail Band](#) that contains the same value two or more times in a row. For example, if you create a report that lists the attributes for each entity in a data model, the report output might look as follows:

```
CUSTOMER  customer-address
CUSTOMER  customer-name
CUSTOMER  customer-number
CUSTOMER  customer-status-code
EMPLOYEE  employee-address
EMPLOYEE  employee-name
EMPLOYEE  employee-number
```

CUSTOMER and EMPLOYEE are repeating values.

By default, all Data Field data is always printed, repeating or not. To remove the duplicate information generated by repeating values, use the suppress options in the Properties dialog:

When you choose the Suppress option, a repeating value is displayed only once in a listing. For example, using the Suppress option in the report above would generate the following output:

```
CUSTOMER      customer-address
               customer-name
               customer-number
               customer-status-code
EMPLOYEE      employee-address
               employee-name
               employee-number
```

If you check the Redisplay after Group check box, a repeating value is displayed only once per group. If the repeating value continues into a new group, it is repeated again once at the top of the new group.

If you check the Redisplay after Page check box, a repeating value is displayed only once per page. If the repeating value continues onto a new page, it is repeated again once at the top of the new page.

**Note:** Another way to achieve a similar effect of suppressing repeating values is to put the (repeating) Data Field into the [Group Header](#) instead of in the Detail [band](#). This is how the [Guided Reports](#) method does it. Try creating some Guided Group/Totals reports, and examine the layout produced to see the techniques used.

## ***Adding Borders to Fields***

---

Using borders allows you to highlight or emphasize the information in certain fields. It can also add visual interest to your report presentation.

You can add a border on any or all sides of any field in your report. Click on the desired Borders check box (left, right, top, bottom, or all) to draw a corresponding border. As you add borders in the Properties dialog, the picture to the right (representing the field) shows the borders you have selected.

To delete a border, click on the Borders check box again to uncheck the selection. The border is immediately removed.

You can also add borders to entire Bands, using the [Band Properties](#) dialog, described later.

**Hint:** To display a sum line below a column of numbers and above the total, add a top border to the Sum field in your [Group Footer](#) or [Report Footer](#).

**Note:** You cannot use borders on an OLE object. Borders for OLE objects must be created with the object application.

### ***Adding Patterns to Fields***

Use the Patterns option in the Properties dialog to choose different types of gray shading to highlight different fields in your report.

To add a pattern to a field, click on the arrow next to the Pattern box in the Properties dialog. RPTwin opens a drop down list showing the available patterns, including None. Scroll the patterns until the one you want is shown. Click on the pattern to select it. The selected pattern is displayed in the pattern window.

To remove a pattern, follow the procedure above and replace the old pattern with a new pattern, a clear pattern, or None.

### ***Adding Field Names***

Any Data or Text field may optionally be given a Name by typing it in the Name entry box in the Properties dialogs.

## **Data Field Properties**

The following properties, which apply to Data Fields, can be changed in the Data [Field Properties](#) dialog.

### **Related Topics:**

[Changing the Format of Data Fields](#)

[Blank If Zero](#)

[Word Wrap for Data Fields](#)

[Adjusting Width Automatically for Data Fields](#)

[Squeezing Up If No Data](#)

[Pushing Down Other Fields](#)

### **Changing the Format of Data Fields**

A [Data Field](#) can contain any of following types of data: text, numbers, times, dates or datetimes (i.e., a date and a time together). Each type of data has its own set of possible formats. The Format property determines how to display the data in the current Data Field. For example, a date can be formatted and displayed in many ways:

<b>Appearance</b>	<b>Format Used</b>
Apr 15, 1995	Mon dd, yyyy
4/15/95	mm/dd/yy
April 15, 1995	Month dd, yyyy
4/15/1995	mm/dd/yyyy

and so on...

A number can also have many formats. For example, the value .432 can be displayed in a number of different formats:

<b>Appearance</b>	<b>Format Used</b>
.432	9.999
\$0.43	\$9.99
43.20%	9.99%

and so on...

When you create a new Data Field, a format is assigned to the field by default (Page number, Text, Date, etc.) You can change these default formats using the Preferences dialog, which is accessed from the Options menu.

To change the format for a Data Field, click on the arrow next to the Format drop-down list in the [Data Field Properties](#) dialog. Scroll to the format you want and release the mouse.

The **General** format for each data type displays the data as closely as possible to the format it receives from the database.

***Blank If Zero***

Sometimes you may want to suppress the display of data fields that evaluate to zero. This can make certain types of reports look less cluttered, by eliminating zero values. To suppress the display of zero values, click the Blank if Zero check box, so a checkmark appears in the box. To display zero values, click again to remove the checkmark.

### ***Word Wrap for Data Fields***

Word Wrap cannot always be used with Data Fields. If the Width property is set to Adjust Width To Data, the Word Wrap option is disabled. If Word Wrap is enabled and clicked, the list box in the Height options becomes active (The "If height increases due to word wrap, then move down" list box). See [Pushing Down Other Fields](#) for more information.

### ***Adjusting Width Automatically for Data Fields***

The pull-down list to the right of the Width box in the Data [Field Properties](#) dialog provides options for controlling the width of your field depending on its contents. The options are:

- ◆ Fixed Width
- ◆ Adjust Width to Data
- ◆ Expand Right to Margin or Next Item

Choose Fixed Width to maintain the width you have given the field, regardless of the size of the data. This option maintains a constant relationship between the field and any adjacent items in the report. When this option is selected, you can also choose Word Wrap.

Choose Adjust Width to Data to stretch the field to the right to accommodate your data. The field will vary in width so other items in the report may also shift their positions. If you use this option, text justification features (center, left, right, full) are disabled. All data appears Left justified.

Choose Expand Right to Margin or Next Item to expand the space available in the Data Field to the right until it encounters the right margin or another field. Other items will stay in position while the Data Field expands as far as it can to accommodate the data. When this option is selected, you can also choose Word Wrap.

### ***Squeezing Up If No Data***

If a [Data Field](#) contains no data, checking the "Can be squeezed up if no data" check box shrinks the field so that it occupies no vertical space in the report. If you leave this box unchecked, especially in [Vertical report](#) layouts, the report may contain a blank line where the empty field is located.

For example, suppose you have a list of mailing addresses in the following lay out:

CUST\_NAME

ADDR1

ADDR2

CUST\_CITY

Now, suppose that for some customers there is only information in the ADDR1 field (ADDR2 is empty or NULL). If you don't use the Squeeze up option, your printout will contain a blank line between ADDR1 and CUST\_CITY for those particular customers.

In this case, use the Squeeze up option with ADDR2. Note that the field will only squeeze up if doing so will not cause any remaining fields to overlap.

### ***Pushing Down Other Fields***

When Word Wrap is turned on for a field, the "If height increases..." options are enabled in the Data Properties dialog. The options are:

- ◆ All Items Below
- ◆ Items Immediately Below
- ◆ Items Immediately Below and Below Right
- ◆ None of the Items Below

Choose All Items Below to push every item in the report downwards if Word Wrap in the current field expands the field downwards.

Choose Items Immediately Below and Items Immediately Below and Below Right to push down all fields immediately below or immediately below and to the right of the current field.

Choose None of the Items Below to leave other report items in place. If you choose this option, some fields can be overwritten. To prevent this, you might want to leave some blank space in the [band](#) below the Word Wrapped field to absorb the change in height.

**Note:** A field will only be pushed down if the [Can Move Down](#) Property is turned on for that field as well.

## **Text Field Properties**

---

To change Text [Field Properties](#), double-click on the Text Field in your report. RPTwin opens the Text Field Properties dialog.

Changing Text Field properties is done in the same manner as changing [Data Field](#) properties. See [Changing Field Properties](#) for more information about the properties in this dialog.

## Changing Band Properties

The [Design window](#) contains a number of horizontal Bands—for the [Report Header](#) and Footer, Group Headers and Footers, [Page Header](#) and Footer, and the [Detail Band](#). You can change the properties associated with each of the report bands in a report layout.

To open the [Band](#) Detail dialog, double-click on any empty area in a band, or in the band's title bar. You can also select any field in the band and then click on the [Band Properties](#) item in the Layout menu.

All band dialogs have the following general format.

### Related Topics:

[Changing the Band Height](#)

[Fixed Height Option](#)

[Using Borders in Bands](#)

[Using Patterns in Bands](#)

[Print Options in Bands](#)

## Changing the Band Height

The Height property controls the height of the [Band](#) layout area on the printed report as well as in the [Design window](#). Changing band height is a perfect way to create extra space in your report. You can type in a value in this box, although it is usually easier to get what you want by clicking and dragging the bottom edge of the Band layout area to the desired size. When you move the mouse cursor to the very bottom of a band, you will notice that the cursor changes.

You can then click and hold the mouse button down to drag the band area up or down like a window shade. Band height is measured in inches or centimeters, as specified in the Current Layout and Preferences dialogs.

### **Fixed Height Option**

If Fixed Height is checked, the [band](#) will print with the same height regardless of the quantity of report data.

Unchecking this box permits the band to expand downwards so that all the data is displayed. The need to stretch the band is evident if you are using Word Wrap on any of the Text and/or Data Fields in the band. If the data in these fields overflows due to word wrap, you may want the band to be flexible to accommodate the text.

## Using Borders in Bands

Using borders allows you to highlight or emphasize the information in the [band](#). It can also add visual interest to your report presentation.

You can add a border to any or all sides of a report band. Click on the desired Borders check box to draw a corresponding border line. As a preview, the graphical representation in the dialog box shows the border(s) you have selected.

To delete the border, click on the Borders check box again to uncheck the selection. The border is immediately removed.

## Using Patterns in Bands

You can use a pattern to emphasize information in the [band](#), or you can use a pattern to de-emphasize, or even obliterate information.

To add a pattern to your band, click on the arrow button to the right of the Pattern box in the Band dialog. RPTwin opens a drop-down box showing the available patterns, including None. Scroll the patterns until the one you want is shown. Click on the pattern to select it. The selected pattern is displayed in the pattern window.

To remove a pattern, follow the procedure above and replace the old pattern with a new pattern, a clear pattern, or None.

## Print Options in Bands

Each [band](#) has specific options available to control printing, which appear in the On Printing section of each Band dialog. Different options appear for different Band types. Examples of how and where you might use these options are provided below.

### Start New Page Before Band is Printed

Starts a new page, then prints the band data, starting at the top of the page. For example, use this option to start each group on a new page.

**Bands used in:** [Group Header](#), Detail, [Group Footer](#), and [Report Footer](#).

### Start New Page After Band is Printed

Prints the data in the band, then starts a new page. For example, when used in the [Report Header](#), this allows you to create a cover page for your report with just the title and/or other report information on it.

**Bands used in:** Report Header, Detail, and Group Footer.

### Reset Page Number Before Band is Printed

Resets the page number to 1 before the band data is printed. For example, if each group is set to start on a new page, it might be convenient to provide each group with its own page numbering.

**Bands used in:** Group Header, Group Footer, and Report Footer.

### Reset Page Number After Band is Printed

Resets the page number to 1 after the band data is printed. For example, you can have several pages for the Report Header. You might want to start your page numbering for the body of the report after the report title page(s) are printed.

**Bands used in:** Report Header, Group Footer, and [Page Footer](#).

### Don't Allow Page Break Inside Band

Keeps the band information together. For example, you're using a Report Footer with several important details in it that you don't want to get separated. Or, you want everything in your [Detail Band](#) to stay together for each instance (row) of related data.

**Bands used in:** Group Header, Detail, Group Footer, and Report Footer.

### Replace [Page Header](#) with this Band

Instead of printing the Page Header at the top of a new page, this prints the Group Header. For example, you want to mark the top of each page with information about the current group.

**Bands used in:** Group Header only

### Replace Page Footer with this Band

Instead of printing the Page Footer at the bottom of a new page, this prints the Group or Report Footer. For example, the group or report footer may contain important group or report totals that you want seen instead of the page footer.

**Bands used:** Group Footer and Report Footer.

## Changing OLE Object Properties

An OLE object in a report can be changed in two ways. First, you can alter it in the application that it is linked to (See [Working With OLE Objects](#)); second, you can alter its properties with respect to the rest of the report. Object properties include its size, position and the option to hide the object during printing.

To change the current properties of an OLE object, click once over the object to select it (the field is selected if four square grabber handles appear at the corners). Then choose OLE Object Properties from the Layout menu or the floating menu (using the right mouse button). The OLE Object Properties dialog box displays:

The Position and Size values are in the default units being used in the report. You can enter the exact object position and size, though it is easier to use your mouse and drag the object or the object corners accordingly. The position and size values are automatically recorded as you move and size the object with the mouse.

Regarding the object position, there are two checkbox options:

- ◆ Can Move Down
- ◆ Can Slide Left

If Can Move Down is selected, the OLE object will appear in its report layout position, unless there is a field above it that has expanded to take up more space. In this case, it will move down below that field instead of the position defined in the From Top field.

If Can Slide Left is selected, The OLE object will appear on the left side of the report, unless there is a field to the left of the OLE object. In this case, it will move to the left up from that field. This happens regardless of the value in the From Left field. The OLE object will also slide left if placed beyond the report border to the right.

Selecting the Hide When Printing checkbox will hide the OLE object from previewing and printing. The object will still be visible on the design canvas.

# Previewing a Report

To preview a report, click on the Preview button  in the Design window's Toolbar or select Print Preview on the File menu. When you preview a report, the active report appears in the [Preview window](#), shown below, just as it would be printed, complete with headers, footers, groups, and fields.

**Note:** If you have specified a special Sort or Grouping order, the sorting occurs before previewing can take place. For a very large DataSet, this could take some time. A message appears showing how to cancel the process if it becomes too lengthy.

## Related Topics:

[Preview Toolbar](#)

[Paging Through a Report](#)

[Displaying One or Two Report Pages](#)

[Magnifying a Report](#)

[Closing the Preview Window](#)

## Preview Toolbar

The following tools appear on the Toolbar in the [Preview window](#):



Each of the buttons on the Preview Toolbar is explained below.



**Print** the report.



**Close** the Preview window and return to the [Design window](#).



**Go to First Page** of the report.



**Go to Previous Page** of the report.



**Go to Next Page** of the report.



**Go to Last Page** of the report.



**Single Page Display** of the report.



**Double Page Display** of the report.



**Zoom In** (make larger).



**Zoom Out** (make smaller).



## Paging Through a Report

If there are several pages in a report, you can cycle through the pages using the page control buttons (which operate like VCR buttons).

The bottom left hand corner of the [Preview window](#) shows the current page number. When you reach the end of the report, the Next Page button is automatically dimmed.

You can page backwards through the report, one page or two pages at a time, depending on the number of pages selected. Or, you can jump to the first page using the Go to First Page button or the last page using the Go to Last Page button.

**Note:** RPTwin composes pages in the background while you are examining a given page. If you try to jump to the end of a long report before it has completed this work, there will be a delay, since the appearance of the last page depends upon the composition of all the previous pages. The status bar at the bottom of the Preview window indicates the progress of the page composition process.



### **Displaying One or Two Report Pages**

You can display one or two pages on your screen at the same time using the page display buttons, however, you can't change this selection if the display is magnified. When you resume normal magnification, these options again become available.



## Magnifying a Report

There are two ways to change the magnification in the [Preview window](#).

1. Click the mouse on the report to magnify the preview. If you continue clicking the mouse, the zoom factor returns to the smallest size display.
2. Click on the plus and minus magnifying buttons on the Toolbar at the top of the Preview window to change the zoom factor.

**Note:** The mouse cursor turns into a magnifying glass whenever it passes over the previewed report. When previewing a report at the smallest size, some of the details may not appear just right. If this happens, Zoom Out, or make a test print to be sure.



### **Closing the Preview Window**

When you have reviewed your Previewed report, you can print it directly from the Preview window's Toolbar by clicking the Print button. Or, close the [Preview window](#) by clicking the Close button (to the right of the Print button).



## Printing a Report

RPTwin uses standard Windows printing conventions. You can print a report using the Print button in the [Preview window](#), the Print item on the File Menu, or the Print button on the Toolbar. The Print dialog lets you choose to print the entire report or selected pages, to receive one or more copies of the report, etc.

**Note:** If you have designed a complicated report for a [DataSet](#), and want to use it with another DataSet - rather than starting from scratch with a completely new report, see [Changing the DataSet](#) on how to clone a report. Similarly, if you want to share a report file with others, be sure that the DataSet filename is accessible to everyone using the report. See [Using the No Path Option](#) for more information on how to share reports.

## Changing the DataSet

Each report layout stores the name of one (and only one) [DataSet](#) that it automatically uses as the source of its data. In general, this is the DataSet that you used when you first created the report. However, you can change the DataSet that a given report will use. To do so:

Select Current DataSet from the Options menu.

The Current DataSet dialog has two sections. The DataSet Currently In Use By This Report section, shows the name of the DataSet file that the report is using right now. The DataSet Linked To This Report section, shows the DataSet name that is stored permanently with the report file. The DataSet will usually be the same in both boxes. Why are two DataSet names listed?

- ◆ You may want to use the current report layout to make a report on some data from a different DataSet, without changing the default DataSet that this report file will use in the future. In this case, you will want to keep the linked DataSet unchanged, and just change the current DataSet. In this way, you can use the same report with another DataSet instead of designing a new one (especially if the DataSets are similar in structure). See [Changing the Current DataSet](#) for more information.
- ◆ If you want to permanently use a report layout with a different DataSet (even with different column names), you can create a clone report. Open the report and change its current DataSet. Then, to make the clone report, click the Link button (to make the new DataSet permanent). Finally, use the Save As item on the File menu to save the report under a new name. Make whatever changes you like to the report layout; it is a separate report file with a different DataSet from the original one.

### Related Topics:

[Changing the Current DataSet](#)

[Making the Change Permanent](#)

[Using the No Path Option](#)

## Changing the Current DataSet

To change the current [DataSet](#), click the Select button and choose a new DataSet file for your report. The DataSet you choose is copied into the current DataSet box.

If you change the DataSet, the new DataSet must resemble the original one (it must contain all the columns that are named in the report, and their data types must be the same as the originals) or else the report will have trouble calculating its formulas and knowing where to place each piece of data.

If you choose a DataSet that does not match your original one, RPTwin warns you. If you proceed with the new DataSet, you will need to change the formulas in your Data Fields so that they refer to appropriate columns in your new DataSet. Otherwise, when you print or preview the report, the words Bad [Formula](#) appear wherever formulas no longer make sense with the new DataSet.

### **Making the Change Permanent**

To make your new current [DataSet](#) the permanently linked DataSet for this report, click the Link button. The current DataSet is copied into the linked DataSet box.

## Using the No Path Option

When the full pathname is listed for a [DataSet](#), RPTwin looks for that DataSet only in that exact location on the disk. This is fine if the DataSet is on your own disk – but what if you want to share a report file with someone else? Since the DataSet name is saved inside the report file, there may be different drive letters or directory names on the computer the report is run on (your co-worker might work on drive D: rather than drive C:, or he/she may have installed RPTwin in a directory with a different name).

### No Path

The No Path button in the Current DataSet dialog lets you strip off the full pathname part of the DataSet name in the linked DataSet box, leaving just the filename itself. For example, if the linked DataSet box shows this pathname:

C:\RPTWIN\ENTITY.LWD

after you choose No Path, it would show just the filename:

ENTITY.LWD

When a DataSet is stored without its full pathname (using the No Path button in this dialog), RPTwin looks for the DataSet in the following way: First, it will look in the default DATASETS directory. If your DataSet is not there, it will look in the directory that contains the report file (.LWR) you are using. If it cannot find it there, it will look in the current directory. If you plan to share report files, RPTwin will be able to find the associated DataSets as long as you make sure that each DataSet is saved either in the RPTwin default directory or in the same directory as the report with which it is linked.

**G Remember:** The DataSet (\*.LWD) and the report layout (\*.LWR) are separate files. The only link between them is a reference in the report file to the name of the DataSet file.

---

**Note:** If you open a report layout or print the report, RPTwin does not automatically update the contents of the DataSet used as the data for the report. The actual contents of the DataSet are changed only when ERwin or BPwin generates new report output and the file is saved again in the DataSet file.

## Creating Guided Reports

---

Whenever you start a New report, you have the option to create a Guided Report. An RPTwin Guided Report walks you through the process of report creation in a manner similar to using a Microsoft Wizard. There are two types of [Guided Reports](#):

- Groups/Totals**      For columnar reports with automatic [grouping](#) and [sorting](#), complete with headers, footers, and totals of all numeric columns.
- Vertical**              For Vertical reports where you control which data should appear, in what order, and with optional sorting.

Each Guided Report presents a series of simple dialogs (similar to the figure shown below), in which you can choose the options you want in your report.

The Guided Report dialogs share many common features. For example, at the bottom of each dialog, there are buttons that let you go back to the previous step (Back), or forward to the next step (Next), to skip to the first or last dialog in the series, or to cancel the report creation process.

A set of step-by-step instructions for creating each type of Guided Report follows below.

### Related Topics:

[To create a Group/Totals report](#)

[To create a Vertical Guided Report](#)

## To create a Group/Totals report

1. Choose the New option on the File menu and select the report data file (.LWD) that you want to use for the report. In the New Report dialog, click the Group/Totals button in the [Guided Reports](#) group box on the right.
2. When RPTwin displays the first Guided Report dialog, choose the [DataSet](#) columns that you want to include in your report. Double-click on a column name in the left-hand side of the dialog, to move it into the right-hand (included) list. To remove a column from the right-hand list, click on it in the right-hand list to highlight it and then click Remove.
3. The order of the column names is important, as it determines the order of the fields in your report. You can rearrange the order by grabbing a column and moving it up or down in the right-hand list.
4. Optionally, click the Fit all columns on one page check box to fit all fields onto one page, even if this means truncating some of the data. This option is useful when most of your data will fit on one page, but a few long data items would otherwise force the report to use more than one page across to fit everything.
5. Click the Next button to go on.
6. When RPTwin displays the second dialog, choose which columns you want to use to group your report. Again, the order of your choices is important, as it determines the order of your groupings. You may group on more than one column (which will produce automatic subgroups). See [Grouping](#) for more information.
7. Click the Next button to go on.
8. When RPTwin displays the final dialog in the series (which are similarly named for all Guided Reports), enter a title to appear in the [Report Header](#) (or you can keep the default title). Then, click on the appropriate button to perform the next action.
  - To preview the report on screen, click the Print Preview button.
  - To print the report, click the Print... button.
  - To work on the report in the [Design window](#), click the Design button.

## To create a Vertical Guided Report

1. Choose the New option on the File menu and select the report data file (.LWD) that you want to use for the report. In the New Report dialog, click the Vertical button in the [Guided Reports](#) group box on the right.
2. When RPTwin displays the first Guided Report dialog, choose the [DataSet](#) columns that you want to include in your report. Double-click on a column name in the left-hand side of the dialog, to move it into the right-hand (included) list. To remove a column from the right-hand list, click on it in the right-hand list to highlight it and then click Remove.
3. The order of the column names is important, as it determines the order of the fields in your report. You can rearrange the order by grabbing a column and moving it up or down in the right-hand list.
4. When RPTwin displays the second dialog, choose which column(s) you want to use to Sort. Double click on a column name in the left-hand side of the dialog, and it will appear in the right-hand (sort) list. To remove a column from the sort list, click on it to highlight it and then click Remove. Again, the order of your choices is important, as it determines the sort order. You can rearrange the order by grabbing a column and moving it up or down in the right-hand list. You may sort on more than one column, which will produce automatic sub-[sorting](#). Sorting may be in Ascending or Descending order; just click the appropriate button.
5. When RPTwin displays the final dialog in the series (which are similarly named for all Guided Reports), enter a title to appear in the [Report Header](#) (or you can keep the default title). Then, click on the appropriate button to perform the next action.
  - To preview the report on screen, click the Print Preview button.
  - To print the report, click the Print... button.
  - To work on the report in the [Design window](#), click the Design button.

## Using Sum, Average, and Other Aggregate Functions

RPTwin provides certain functions (called **aggregate functions**) that you can use to perform calculations on the values in a report. Some aggregate functions (e.g., Sum, Avg, Min, Max, and Count) are **context sensitive** and perform calculations based on where you place them in the report. For example, an aggregate [function](#) may produce one total when you place it in the [Group Footer](#) (it generates a group total) and a different total when you place it in the [Page Footer](#) (it generates a page total).

Other aggregate functions (e.g., ReportSum and GroupSum) perform group-level or report level calculations regardless of where they are placed in the report. For example, ReportSum, a [report-level function](#), produces the same total whether it is placed in the Group Footer or in the Page Footer (it generates the total for all the records processed in the report).

See [Overview](#) for more information.

### Related Topics:

[Context Sensitive Aggregates](#)

[Look-Ahead \(Two-Pass\) Feature](#)

[Report-Level and Group-Level Functions](#)

## Context Sensitive Aggregates

Aggregate functions work in different ways, they are context sensitive depending on where they are placed in your report. For example, if you place a:

- ◆ Sum in a [Detail Band](#), it produces a running sum for the current group.
- ◆ Sum in a [Group Footer](#), it produces a total for that Group.
- ◆ Sum in a [Report Footer](#), it produces a total for the entire report.

## Look-Ahead (Two-Pass) Feature

Part of being context-sensitive means that these aggregate functions must know what to do when they appear in Headers. What would it mean to calculate the sum in a [Report Header](#)? Nothing, since there is no data in a report before the Report Header. The same kind of problem exists with Group Headers – what would it mean to calculate the Sum in a group, in the [Group Header](#)? There is no data for the group before its Group Header. Therefore, aggregate functions must act differently when they are placed in Headers.

If you put a Sum [function](#), for instance, into your Report Header, RPTwin automatically senses that it must look ahead through the data to calculate the Sum for the entire report. Similarly, if you put a Sum function into your Group Header, RPTwin automatically knows that it must scan ahead through the data until the group changes and calculate the sum for the entire Group. In RPTwin, all aggregate functions act in this way.

Because it supports Look-Ahead (often called Two-Pass feature) calculations, RPTwin lets you reference report totals, group totals, and other aggregate functions at any point in your report layout. For example, you can place a:

- ◆ Report Grand Total at the very top of your report.
- ◆ Group Total in the Group Header for any group.

**Note:** Two-pass functionality applies only to the aggregate functions, which include: Sum, Avg, Min, Max, and Count.

## Report-Level and Group-Level Functions

---

RPTwin provides some special aggregate functions which allow you even greater control over how aggregates get evaluated. These include:

ReportSum	GroupSum
ReportMax	GroupMax
ReportMin	GroupMin
ReportAvg	GroupAvg
ReportCount	GroupCount

These special aggregates are not context-sensitive in the same way as the standard aggregate functions. Report-level functions always produce the desired value for the entire report, no matter where you place them in the layout. Group-level functions always produce the value for the entire current group, no matter where you place them in the layout (except that if you place a [Group-level function](#) in a [Report Header](#) or [Report Footer](#), it is automatically promoted to a [Report-level function](#) and acts accordingly).

You can use these special aggregates to generate percent of total fields in detail lines or in group headers/footers. For example, if want to make a report that looks (in part) like this:

SALESMAN: J. Smith					
DATE	AMT	CUM	%Salesman	%Total	
1/2/94	2000	2000	20%	10%	
1/4/94	3000	5000	30%	15%	
1/6/94	5000	10000	50%	25%	
SALESMAN: P. Brown					
DATE	AMT	CUM	%Salesman	%Total	
1/3/94	1000	1000	10%	5%	
1/5/94	6000	7000	60%	30%	
1/7/94	3000	10000	30%	15%	

create three formulas as shown below:

<b>Field</b>	<b>Formula</b>
CUM	Sum(ORDER_AMOUNT)
%Salesman	ORDER_AMOUNT/GroupSum(ORDER_AMOUNT)
%Total	ORDERAMOUNT/ReportSum(ORDER_AMOUNT)

**Note:** RPTwin uses an on-demand look-ahead system for resolving formulas as it encounters them. It is not a simple two-pass system. But the result is essentially the same, and therefore the common term "two-pass report" is not applicable. This is an on-demand feature, so it will not slow down reports that do not need it.

## Filtering: Using Subsets of Data

---

Generally speaking, you use [ERwin](#) to define the entities that you want to include in the report output file. However, if you need to report on just a subset of the data in your [DataSet](#), you can use the Filter dialog in RPTwin.

To open the Filter dialog, choose Filter on the Options menu. RPTwin displays the Filter dialog.

**{Entity Name} = "CUSTOMER" or {Entity Name} = "OVERDUE-NOTICE"**

Click OK to close the [Formula](#) Editor then click Include to display only these entities in the report.

When you click OK and run the report, only the rows of data that meet your filter criteria will be included in the output. You can group, sort, or do anything else on this filtered DataSet that you can do on a full DataSet. Although the report output only shows the data that meets the specified filter criteria, all the ERwin data remains in your DataSet. This lets you use the same DataSet to produce any number of different report snapshots.

If you want to show all the ERwin data model information except the data that meets the filter criteria, click the Exclude button in the Filter dialog instead of the Include button.

**Note:** The formula you enter in the Filter dialog must always evaluate to TRUE or FALSE. For example, Entity Name = CUSTOMER would be a valid Filter (because it is either TRUE or FALSE), but CUSTNUM \* 2 would not! Your Filters can be as complex as you like.

## Understanding Formulas

---

The way you tell RPTwin where to put a piece of information is by placing a [Data Field](#) on the report layout. The way you tell the field what information to display is by giving it a [formula](#). The formula can be as simple as a single [DataSet](#) Column name, such as CUST\_NAME or ORD\_DATE, or it can be more complicated, such as DayOfWeek (ORD\_DATE).

This section describes the components from which you create formulas, and gives examples of how you might use them to create some useful formulas.

### **Related Topics:**

[DataSet Column Names](#)

[Arithmetic Formulas](#)

[Text Formulas](#)

[Formula Types](#)

[Operators](#)

[Functions](#)

## DataSet Column Names

The simplest formulas consist of a single [DataSet](#) Column name. For example, if you have a column named CUSTOMER, then CUSTOMER is a valid [formula](#). Every time you drag a column name from the [DataSet Columns List](#) onto the report to create a field, the DataSet Column name automatically placed in the Formula Editor.

**Note:** A DataSet Column name must begin with a letter, and contain nothing but letters, numbers, or the underline character ( ). If your Column name contains any spaces, punctuation characters, + or - signs, or anything other than letters, numbers, and underlines, it cannot be a converted into a valid RPTwin formula.

If a column name contains an invalid character, you can easily ensure that it can be converted into a valid formula by enclosing the name in curly braces ({}).

---

For example, the following table shows three invalid column names on the left, the reason why the name is invalid in the center, and a valid version of each name on the right:

Invalid Name	Reason	Valid Name
CUST NAME	Contains an embedded space	{CUST NAME}
1234X	Starts with a number	{1234X}
QTY*10	Includes a multiplication symbol (*)	{QTY*10}

In most cases, RPTwin will detect that a Column name contains illegal characters, and automatically add the curly braces for you. If for some reason RPTwin does not detect the problem, or if you need to manually type Column names, you can add the curly braces yourself in the Formula Editor.

Be sure that you put the curly braces immediately around the problem Column name. For example, if the Column name is:

### Count of Rows in Groups^

where the ^ character represents an extra space at the end of the Column name, the curly braces should be placed to include the trailing space like this:

#### {Count of Rows in Groups }

The two examples below show how **not** to add braces:

{ Count of Rows in Groups }     (space at start)

{Count of Rows in Groups }     (2 spaces at end)

In order to produce a valid formula, all the characters inside the curly braces must be actual characters in the Column name.

## Arithmetic Formulas

You can also build a [formula](#) that does normal arithmetic. For example, perhaps you have a [DataSet](#) of product orders that includes a column named PRICE that gives the price of the product, and a column named QTY that tells how many were ordered. You can calculate the extended price (i.e., the price of QTY items at PRICE amount each) with the formula:

$QTY * PRICE$

Valid arithmetic operators are:

- + addition
- subtraction
- \* multiplication
- / division

If necessary, you may build more complex arithmetic formulas from simpler ones. For example, if you want to add a one-dollar processing fee to the above extended price, you could use the formula:

$QTY * PRICE + 1.00$

You could also write this formula as:

$1.00 + QTY * PRICE$

and it would work just the same way, because RPTwin performs multiplications (and divisions) before it performs additions (or subtractions). If you want an addition performed first, use parentheses, as in:

$(1.00 + PRICE) * QTY$

RPTwin always calculates values inside parentheses before performing other calculations.

**Note:** You can enter the one-dollar amount in the above example as 1.00, 1.0, 1, or 01, but you cannot write \$1 because RPTwin doesn't understand dollar signs in numbers. Extra spaces between column names and operators (such as \*) are acceptable.

## Text Formulas

---

The '&' operator concatenates two text values; if you have two [DataSet](#) Columns LAST\_NAME and FIRST\_NAME, you can put them together with the [formula](#):

FIRST\_NAME & LAST\_NAME

So if FIRST\_NAME has the value Scarlett and LAST\_NAME has the value O'Hara, the result of the formula would be ScarlettO'Hara. If you want to insert a space between the first and last name, you would instead use the formula:

FIRST\_NAME & " "& LAST\_NAME

where there is a space between the quotation marks. This combines two [text fields](#) together with one space in between. Now, your result would be: Scarlett O'Hara (The quotes don't get printed).

## Formula Types

If you try to create a [formula](#) like:

```
CUST_NAME + 1.00
```

RPTwin generates an error indicating that it expected a Number but found Text instead.

Number and Text are two of the data types understood by the formula editor. Other types include Date, Time and Datetime (which includes both date and time). To use an operator (like + or \*) or a [function](#) properly, you have to know what data types RPTwin expects for its inputs and the type of the result that it will calculate.

For example, the operators +, -, \* and / all expect Numbers for the inputs and return a Number as the result. Since the data type of CUST\_NAME is Text, you can't use it with +. But, you can use PRICE or QTY, because they are Numbers.

The '&' operator, on the other hand, expects Text inputs, and generates a Text result.

Sometimes a value of one type is a lot like a value of another type. In such cases, RPTwin may let you use a value of the first type even though an operator or function is expecting the other type. For example, RPTwin knows how to convert Numbers into Text, so the following statement is a valid formula:

```
4 & 2
```

even though '&' expects Text inputs and 4 and 2 are Numbers. In this case, RPTwin converts the 4 from the Number 4 to the Text value 4, and similarly the number 2 is treated like a text value so the result will be 42. Because RPTwin can convert any type (Number, Date, DateTime, or Time) to Text, any function expecting Text will also accept values of other types as well.

However, if an operator or function expects a Number, you can't use Text or Date or Time. (What would Scarlett O'Hara + 1 mean?) You can use Datetime when an operator or function is expecting a Time, because Datetimes include a Time value. You can also use a value of Time when an operator or function is expecting a Datetime; in this case the Datetime will have the given Time on the day Jan 1, 0001.

**Note:** If you have a Text value that you know is really a date or a number, and you need to pass it to an operator or function that expects a date or a number, you can call a type-conversion function that tells RPTwin how to read your Text value as a value of one of the other (expected) types (e.g., ToText ToText>mainToText).

## Operators

---

RPTwin supports three types of operators:

- ◆ Arithmetic operators (+, -, \*, and /)
- ◆ Text operator (&)
- ◆ Comparison operators (<=, <, =, >, =>), which return True/False values and are used only in the test of an [If formula](#)

When you use a comparison operator, the comparison performed depends on the types of the inputs. For example, if X and Y are numbers, as in SALARY < 25000, the comparison is done by comparing numbers. If X and Y are text, as in LAST\_NAME < Morrison, then they are compared alphabetically (e.g., aardvark < zebra is true). If they are both dates or are both times, they are compared as such. If the inputs are different types, they are converted to text and compared alphabetically.

Sometimes data from a [DataSet](#) comes in as Null, which means that no data was entered in the column. Null indicates that there is no meaningful value for that field. If you try to operate on a null value the result is a null value and nothing is printed in the report for that field. The two operators that are designed to test for the null value are the True/False operators **is null** and **is not null**, which return True if their argument is null or not null, respectively.

## Functions

Functions can perform a great deal of specialized work in your reports, computing values based on the data in your [DataSet](#). Like operators, functions compute a result value (with a specific type) when given inputs (with specific types).

For example, the [formula](#):

Left (Hello There!, 5)

specifies the [function](#) named Left, with inputs Hello There! and 5. The result is a text value consisting of the first 5 characters of Hello There! (i.e., Hello).

For a complete list of RPTwin functions, see [Overview](#).

## Changing Report Defaults

To speed up the report creation process, RPTwin uses some default settings that control how it lays out your reports. The default settings are stored in the RPTWIN.INI file, which is created in your Windows directory when you first open RPTwin. If you are comfortable editing Windows .INI files, you can change the default settings to suit your needs.

RPTWIN.INI settings that can be changed include the default:

- ◆ Report Title used for [Quick Reports](#) and suggested for [Guided Reports](#).
- ◆ Amount of space between rows in the Quick Columnar and Guided Group/Totals reports.
- ◆ Font, type size and type attributes used for different sections of the reports.

# Overview



The RPTwin [Formula](#) Editor, which you can open by clicking the Add Formula tool on the ToolBox, lets you enter a formula to manipulate data in any field in a report. A formula may include one or more of the functions listed in the Formula Editor.

This appendix alphabetically lists the functions RPTwin supports and provides information to help you use the functions in RPTwin formulas. The sample [function](#) shown below illustrates the format used to define each function.

See [Data Fields](#) and [Changing a Data Field/Editing Formulas](#) for more information.

## Sample Function

Each function in this appendix is documented as shown below.

<b>Function Name</b>	<b>Left</b>								
<b>Function Purpose</b>	<b>Purpose:</b> Returns the specified number of leftmost characters of a text string.								
<b>Syntax</b>	<b>Format:</b> Left( text, number )								
<b>Arguments</b>	<b>Arguments:</b> (text) Text or a formula that produces text, (number) Integer number.								
<b>Return Value</b>	<b>Returns:</b> Text.								
<b>Additional Information</b>	<b>Notes:</b> If the text string contains fewer than the specified number of characters, then all of the text string is returned.								
<b>Usage</b>	<b>Example</b>								
	<table><thead><tr><th><b>Expression</b></th><th><b>Returns</b></th></tr></thead><tbody><tr><td>Left("Hello There!", 5)</td><td>Hello</td></tr><tr><td>Left("Hello There!", 50)</td><td>Hello There!</td></tr><tr><td>Left("Hello There!", 0)</td><td>[no output]</td></tr></tbody></table>	<b>Expression</b>	<b>Returns</b>	Left("Hello There!", 5)	Hello	Left("Hello There!", 50)	Hello There!	Left("Hello There!", 0)	[no output]
<b>Expression</b>	<b>Returns</b>								
Left("Hello There!", 5)	Hello								
Left("Hello There!", 50)	Hello There!								
Left("Hello There!", 0)	[no output]								
<b>Related Functions</b>	<b>Related Functions:</b> Right <a href="#">Right</a>								

## Abs

**Purpose:** Returns the absolute value of a number.

**Format:** Abs( number )

**Arguments:** (number) Number or a [formula](#) that produces a number.

**Returns:** Number

**Notes:** The absolute value of a positive number N is N; the absolute value of a negative number (-N) is N. The absolute value of zero is 0.

### Example

<b>Expression</b>	<b>Returns</b>
Abs(-10)	10
Abs(10)	10

# Age

**Purpose:** Returns the number of whole years since a given date.

**Format:** Age(date)

**Arguments:** (date) Date, datetime, or [formula](#) that produces one of these two types.

**Returns:** Number

## Example

If you give this [function](#) your birth date, it will return your age in whole years.

Expression	Returns
Age(MakeDate(4,1,53))	42 (if today is 4/1/95)

**Related Functions:** DaysBetween[DaysBetween](#), YearsBetween[YearsBetween](#)

# Avg

**Purpose:** Group [function](#) that returns the average value for the rows in the current group.

**Format:** Avg( number)

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number

**Notes:** This function (like the other standard aggregate functions – Sum, Avg, Min, Max, Count) is context sensitive. If this function is used in a [data field](#) in a [Group Footer](#), it will return the average for rows in the current group. If in a [Report Footer](#), it will return the average for the entire report. If in the [Detail Band](#), it will return a running average for the current group. See [for more information](#).

## Example

If your group has two records, and a PRICE column that contains 10.00 for the first row and 20.00 for the second row, the value computed by:

Avg(PRICE)

in the group footer would be 15.00 (that is, the average of 10 and 20). In the detail [band](#), Avg(PRICE) would compute to values of 10.00 and 15.00 respectively (a running average, that is the average of the given formula for all the rows in the group up to and including the current one).

**Related Functions:** Avg [Avg](#), Count [Cos](#), GroupAvg [GroupAvg](#), GroupCount [GroupCount](#), GroupMax [GroupMax](#), GroupMin [GroupMin](#), GroupSum [GroupSum](#), Min [Min](#), ReportAvg [ReportAvg](#), ReportCount [ReportCount](#), ReportCumAvg [ReportCumAvg](#), ReportCumMax [ReportCumMax](#), ReportCumMin [ReportCumMin](#), ReportCumSum [ReportCumSum](#), ReportMax [ReportMax](#), ReportMin [ReportMin](#), ReportSum [ReportSum](#), Sum [Sum](#)

# Cos

**Purpose:** Returns the cosine of a number of radians.

**Format:** Cos(number)

**Arguments:** (number) Number or [formula](#) that produces a number.

**Returns:** Number

**Notes:** The Cosine [function](#) serves as the standard mathematical cosine function.

## Example

Function	Result
Cos(0)	1
Cos(3.1415926536)	-1

**Related Functions:** Sin[Sin](#), Tan[Tan](#)

# Count

**Purpose:** Group [function](#) that returns the number of records in the current group.

**Format:** Count()

**Returns:** Number

**Notes:** This function (like the other standard aggregate functions – Sum, Avg, Min, Max, Count) is context sensitive. If this function is used in a [data field](#) in a [Group Footer](#), it will return the number of rows in the current group. If in a [Report Footer](#), it will return the number of rows in the entire report. If in the [Detail Band](#), it will return a running count for the current group. See [for more information](#).

## Example

If your group has two records, the value computed by:

Count()

in the group footer would be 2. In the detail [band](#), Count () would compute values of 1 and 2 respectively (a running count, or a count of all the records in the group up to and including the current one).

**Related Functions:** Avg[Avg](#), Count[Cos](#), GroupAvg[GroupAvg](#), GroupCount[GroupCount](#), GroupMax[GroupMax](#), GroupMin[GroupMin](#), GroupSum[GroupSum](#), Max[Max](#), Min[Min](#), ReportAvg[ReportAvg](#), ReportCount[ReportCount](#), ReportCount[ReportCount](#), ReportCumMax[ReportCumMax](#), ReportCumMin[ReportCumMin](#), ReportCumSum[ReportCumSum](#), ReportMax[ReportMax](#), ReportMin[ReportMin](#), ReportSum[ReportSum](#), Sum[Sum](#)

# Date

**Purpose:** Returns a value of type date that represents the date when the report is run.

**Format:** Date()

**Returns:** Date

## Example

If you run the report on April 15, 1995:

Expression	Returns
Date()	4/15/95 (exact output depends on field's selected format)
ToText(Date(), "Day, Month dd, yyyy")	Saturday, April 15, 1995

**Related Functions:** DateTime [Date](#)Time, MakeDate [MakeDate](#), MakeTime [MakeTime](#), ToText [ToText](#)

# DateTime

**Purpose:** Returns a value of type datetime that represents the date and time when the report was run.

**Format:** DateTime()

**Returns:** Datetime

## Example

If you run the report on April 15, 1995, at 2:35 p.m.:

Expression	Returns
DateTime()	4/15/95 2:35 p.m. (exact output depends on the default Datetime format)
ToText(DateTime(), "Day, Month dd, yyyy hh12:MM am")	Saturday, April 15, 1995 2:35 p.m.

**Related Functions:** Date[Date](#), MakeDate[MakeDate](#), MakeTime[MakeTime](#), ToText[ToText](#)

# DayName

**Purpose:** Returns a text value containing the name of the day of the week of a date.

**Format:** DayName( date )

**Arguments:** (date) A date or datetime, or a [formula](#) that produces a date or datetime.

**Returns:** Text

## Example

On Saturday, April 15, 1995:

Expression	Returns
DayName(Date())	Saturday

**Related Functions:** DayNameAbr[DayNameAbr](#), DayOfMonth[DayOfMonth](#), DayOfWeek[DayOfWeek](#), DayOfYear[DayOfYear](#), ToText[ToText](#)

# DayNameAbr

**Purpose:** Returns a text value containing the abbreviated name of the day of the week of a date.

**Returns:** DayNameAbr( date)

**Arguments:** (date) A date or datetime, or a [formula](#) that produces a date or datetime.

**Returns:** Text

## Example

On Saturday, April 15, 1995:

Expression	Returns
DayNameAbr(Date())	Thu

**Related Functions:** DayName[DayName](#), DayOfMonth[DayOfMonth](#), DayOfWeek[DayOfWeek](#), DayOfYear[DayOfYear](#), ToText[ToText](#)

# DayOfMonth

**Purpose:** Returns the day of the month of a date.

**Format:** DayOfMonth( date )

**Arguments:** (date) A date or datetime, or a [formula](#) that produces a date or datetime.

**Returns:** Number

## Example

On April 15, 1995:

Expression	Returns
DayOfMonth(Date())	15

**Related Functions:** DayName[DayName](#), DayNameAbr[DayNameAbr](#), DayOfWeek[DayOfWeek](#), DayOfYear[DayOfYear](#), ToText[ToText](#)

# DayOfWeek

**Purpose:** Returns a number representing the day of the week for a date, in the range 1-7, where 1 = Sunday and 7 = Saturday.

**Format:** DayOfWeek( date )

**Argument:** (date) A date or datetime, or a [formula](#) that produces a date or datetime.

**Returns:** Integer number in the range 1-7.

## Example

On Saturday, April 15, 1995:

Expression	Returns
DayOfWeek(Date())	7 (because 7=Saturday)

**Related Functions:** DayName[DayName](#), DayNameAbr[DayNameAbr](#), DayOfMonth[DayOfMonth](#), DayOfYear[DayOfYear](#), ToText[ToText](#)

# DayOfYear

**Purpose:** Returns a number equal to how many days a date is into the year, where Jan 1 is day 1.

**Format:** DayOfYear(date )

**Argument:** (date) A date or datetime, or a [formula](#) that produces a date or datetime.

**Returns:** Integer number in the range 1-365 if the date is not in a leap year, and 1-366 if it is.

## Example

On April 15, 1995:

Expression	Returns
DayOfYear(Date())	105 (because 31+28+31+15=105)

**Related Functions:** DayName[DayName](#), DayNameAbr[DayNameAbr](#), DayOfMonth[DayOfMonth](#), DayOfWeek[DayOfWeek](#), ToText[ToText](#)

# DaysBetween

**Purpose:** Returns the number of whole days between two dates.

**Format:** DaysBetween( date )

**Arguments:** (date) A date or datetime, or a [formula](#) that produces a date or datetime.

**Returns:** Integer number.

## Example

Assume that today is 1/1/95:

Expression	Returns
DaysBetween(MakeDate(1,1,94), Date())	365

**Related Functions:** Age [Age](#), YearsBetween [YearsBetween](#)

# GroupAvg

**Purpose:** Group [function](#) that returns the average value for the rows in the current group.

**Format:** GroupAvg( number)

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number.

**Notes:** This function (unlike the standard aggregate functions – Sum, Avg, Min, Max, Count) is not context sensitive within the scope of a group. No matter whether it is placed in a [Group Header](#), [Detail Band](#), or [Group Footer](#), it will return the average for all rows in the current group. However, if this function is used outside its normal scope, it does vary in its meaning. Used in a [Report Header](#) or [Report Footer](#), it will be "promoted" to act like a ReportAvg function, and return the average for the entire report. See [for more information](#).

## Example

If your group has two records, and a PRICE column that contains 10.00 for the first row and 20.00 for the second row, the value computed by:

GroupAvg(PRICE)

in the Group Footer is 15.00 (that is, the average of 10 and 20). In the Group Header, or the Detail [Band](#), GroupAvg(PRICE) would still return 15.00.

**Related Functions:** Avg [Avg](#), Count [Cos](#), GroupCount [GroupCount](#), GroupMax [GroupMax](#), GroupMin [GroupMin](#), GroupSum [GroupSum](#), Max [Max](#), Min [Min](#), ReportAvg [ReportAvg](#), ReportCount [ReportCount](#), ReportCount [ReportCount](#), ReportCumMax [ReportCumMax](#), ReportCumMin [ReportCumMin](#), ReportCumSum [ReportCumSum](#), ReportMax [ReportMax](#), ReportMin [ReportMin](#), ReportSum [ReportSum](#), Sum [Sum](#)

# GroupCount

**Purpose:** Group [function](#) that returns the count of the rows in the current group.

**Format:** GroupCount( number)

**Arguments:** (number) Number or [formula](#) that produces a number.

**Returns:** Number.

**Notes:** See the explanation for GroupAvg [GroupAvg](#)

## Example

If your report has two groups with two records in the first group and three records in the second group, the value computed by:

GroupCount

in the [Group Footer](#) or [Detail Band](#) is 2 (for the first group) and 3 (for the second group). In the [Report Header](#) or [Report Footer](#), the GroupCount total is 5 (the total of all rows in all the groups in the report).

**Related Functions:** Avg [Avg](#), Count [Cos](#), GroupAvg [GroupAvg](#), GroupMax [GroupMax](#), GroupMin [GroupMin](#), GroupSum [GroupSum](#), Max [Max](#), Min [Min](#), ReportAvg [ReportAvg](#), ReportCount [ReportCount](#), ReportCount [ReportCount](#), ReportCumMax [ReportCumMax](#), ReportCumMin [ReportCumMin](#), ReportCumSum [ReportCumSum](#), ReportMax [ReportMax](#), ReportMin [ReportMin](#), ReportSum [ReportSum](#), [Sum](#)

# GroupMax

**Purpose:** Group [function](#) that returns the maximum value for the rows in the current group.

**Format:** GroupMax( number)

**Arguments:** (number) Number or [formula](#) that produces a number.

**Returns:** Number.

**Notes:** See the explanation for GroupAvg[GroupAvg](#).

## Example

If your group has two records, and a PRICE column that contains 10.00 for the first row and 20.00 for the second row, the value computed by:

GroupMax(PRICE)

is 20.00 (the maximum value for the rows in the current group).

**Related Functions:** Avg[Avg](#), Count[Cos](#), GroupAvg[GroupAvg](#), GroupCount[GroupCount](#), GroupMin[GroupMin](#), GroupSum[GroupSum](#), Max[Max](#), Min[Min](#), ReportAvg[ReportAvg](#), ReportCount[ReportCount](#), ReportCount[ReportCount](#), ReportCumMax[ReportCumMax](#), ReportCumMin[ReportCumMin](#), ReportCumSum[ReportCumSum](#), ReportMax[ReportMax](#), ReportMin[ReportMin](#), ReportSum[ReportSum](#), Sum[Sin](#)

# GroupMin

**Purpose:** Group [function](#) that returns the minimum value for the rows in the current group.

**Format:** GroupMin( number)

**Arguments:** (number) Number or [formula](#) that produces a number.

**Returns:** Number.

**Notes:** See the explanation for [GroupAvg](#).

## Example

If your group has two records and a PRICE column that contains the values 10.00 for the first row and 20.00 for the second row, the value computed by:

GroupMin(PRICE)

is 10.00 ( the minimum value for the rows in the current group).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# GroupSum

**Purpose:** Group [function](#) that returns the sum of all the rows in the current group.

**Format:** GroupSum( number)

**Arguments:** (number) Number or [formula](#) that produces a number.

**Returns:** Number.

**Notes:** See the explanation for [GroupAvg](#).

## Example

If your group has two records and a PRICE column that contains the values 10.00 for the first row and 20.00 for the second row, the value computed by:

GroupSum(PRICE)

is 30.00 (the sum of the value of the rows in the current group).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [SumSum](#)

# Hour

**Purpose:** Returns a number from 0-23 corresponding to the hour of a time.

**Format:** Hour( time)

**Arguments:** (time) A time or datetime, or a [formula](#) that produces a time or datetime.

**Returns:** Integer number in the range 0-23 where 0 corresponds to 12 AM and 23 corresponds to 11 PM.

## Example

At 2:35 pm:

Expression	Returns
Hour(Time())	14 (because 2:35 pm is 14:35 in 24-hour format)

To generate the hour of the current time in 12-hour format, you can use the following formula:

```
If Hour(Time()) = 0 Then "12 AM"  
Else If Hour(Time()) <= 12 Then  
    Hour(Time()) & " AM"  
Else  
    Hour(Time ()) - 12 & " PM"
```

**Related Functions:** Minute[Minute](#), Second[Second](#), ToText[ToText](#)

# If...Then...Else

**Purpose:** Returns one value if the statement is true, or a second value (if any) if the comparison is false.

**Format:** If test Then returnValue1 [ Else returnValue2 ]

**Arguments:** (test) Any comparison test that evaluates to true or false. The comparison test is specified by one or more comparisons, in the form:

testValue1 comparisonOperator testValue2

The comparisonOperators can be any of the following:

Operator	Description
=	(equals)
<>	(is not equal to)
>	(greater than)
<	(less than)
>=	(greater than or equal to)
<=	(less than or equal to)
is in	(is included in a list of values)
contains	(is contained in a value)

If more than one comparison is specified, each one of which may evaluate to true or false, then a "logical operator"(e.g., "and" or "or") is placed between each comparison. These determine whether the entire test evaluates to true or false in the form:

comparison1 logicalOperator1 comparison2

The logicalOperators can be any of the following:

Operator	Description
and	(the test is true if BOTH of the comparisons are true)
or	(the test is true if EITHER of the comparisons is true)

One more special operator, "not" can be placed before any comparison. This reverses its result, making it true if it was false and vice versa. This takes the form:

not comparison

The "is null" or "is not null" test has the form:

testValue is null | is not null

where "is null" means that testValue is blank and "is not null" means that testValue is not blank.

Expression	Returns
If testValue Then returnValue1 Else returnValue2	returnValue1(if expression is true) or returnValue2 (if expression is false and the Else value is defined) or Nothing (if expression is false and Else value is not defined)

returnValue1 and returnValue2 can equal any value or a [formula](#) that produces a value.

## Returns:

If the comparison evaluates to true, then returnValue1 is returned by the [function](#).

If the comparison evaluates to false, then the value returned depends upon whether the optional Else

part of the function is used. If the Else clause is not included in the formula, then nothing is returned. If an Else clause is included in the formula, then returnValue2 is returned.

**Notes:**

This function allows your report to display different data depending on a condition that is evaluated at runtime.

The test formula must use one of the Comparison Operators (=, <, etc.) or the "is null"/"is not null" conditions to make its test. The values returned by the formula may be of any type. If returnValue1 and returnValue2 are included in the formula and are different data types, the expression returns the same data type as returnValue2.

If...Then...Else functions can be nested inside each other to provide several levels of comparison.

**Example**

The examples below assume numeric fields COMMISSION, PRICE, YEARSWITHCOMPANY, QUOTA, SALES and SALARY, and a text field TAXABLE that has values "Yes" or "No".

<b>Expression</b>	<b>Returns</b>
If PRICE > 100.00 Then PRICE	PRICE or nothing (if PRICE is less than 100)
If PRICE > 100.00 Then 100 Else PRICE	PRICE or 100.00 (whichever is less)
If salary > 100,000 Then "Executive" Else "Supervisor"	Executive or Supervisor (if salary is less than 100,000)
If SALARY > 200,000 or YEARSWITHCOMPANY > 20 Then "Senior Staff" Else "Junior Staff"	Senior Staff or Junior Staff (if salary is less than 200,000 or if employed less than 20 years)
If TAXABLE = Yes Then PRICE * 1.06 Else PRICE	PRICE plus 6% or PRICE (if not taxable)
If SALES > QUOTA Then If YEARSWITHCOMPANY <= 5 Then SALARY * 0.1 Else SALARY * 0.2	Nothing (if SALES is less than QUOTA) or SALARY plus 10% or SALARY plus 20% (if employed 5 years or more)

## InitCap

**Purpose:** Returns text with all characters in lowercase except for those that begin words.

**Format:** InitCap(text)

**Arguments:** (text) Text or a [formula](#) that produces text.

**Returns:** Text.

**Example**

<b>Expression</b>	<b>Returns</b>
InitCap ("dog kennels")	Dog Kennels

If ITEM = "dOg KEnneLs", then:

<b>Expression</b>	<b>Returns</b>
InitCap(ITEM)	Dog Kennels

**Related Functions:** [LCase](#), [UCase](#)

# LCase

**Purpose:** Returns text with all characters in lowercase.

**Format:** LCase(text)

**Arguments:** (text) Text or a [formula](#) that produces text.

**Returns:** Text.

## Example

Expression	Returns
LCase("Dog Kennels")	dog kennels

If ITEM = "dOg KEnneLs", then:

Expression	Returns
LCase(ITEM)	dog kennels

**Related Functions:** [InitCap](#), [UCase](#)

# Left

**Purpose:** Returns the specified number of leftmost characters of a text string.

**Format:** Left( text, number )

**Arguments:** (text) Text or a [formula](#) that produces text, (number) Integer number.

**Returns:** Text.

**Notes:** If the text string contains fewer than the specified number of characters, then all of the text string is returned.

## Example

Expression	Returns
Left("Hello There!", 5)	Hello
Left("Hello There!", 50)	Hello There!
Left("Hello There!", 0)	[no output]

**Related Functions:** [Mid](#), [Right](#)

# LTrim

**Purpose:** Returns the characters of a text string, but with initial spaces removed.

**Format:** LTrim( text)

**Arguments:** (text) Text or a [formula](#) that produces text.

**Returns:** Text.

**Notes:** This [function](#) is useful if your data comes from the database with spaces at the beginning but you don't want those spaces to show up on your report.

## Example

Expression	Returns
LTrim(" Joe Banks")	"Joe Banks"

If ITEM = " Dog Kennels", then:

Expression	Returns
LTrim(ITEM)	"Dog Kennels"

**Related Functions:** [RTrim](#), [Trim](#)

# MakeDate

**Purpose:** Returns a date value from three numbers.

**Format:** MakeDate(MM, DD, YY)

**Arguments:** (MM) Number from 1-12 that represents the month of the year, (DD) Number from 1-31 that represents the day of the month, (YYYY) Number from 0-9999 that represents the last two digits of the year.

**Returns:** Date

**Notes:** If you give only two digits for the year, MakeDate will assume that it is in the twentieth century (1900-1999).

This [function](#) lets you display a number as a date and include the return value in a text string (e.g., "Please return your video rental(s) by Saturday, April 15.").

## Example

Expression	Returns
MakeDate(4, 15, 1995)	4/15/95 (output depends on field format)

To return a text string based on the return value of MakeDate, use the following [formula](#):

```
If Date() = MakeDate(1, 1, 2001) Then
    "Welcome to the Third Millennium, everyone!"
Else
    "Welcome, everyone."
If BIRTHDAY < MakeDate(1, 1, 20) Then
    "Remembers the Gold Standard."
Else
    "Doesn't know what real money is."
```

**Related Functions:** [MakeTime](#), [ToDate](#)

# MakeMoney

**Purpose:** Returns a money value (which will display with a dollar sign and commas) from a number.

**Format:** MakeMoney( number )

**Arguments:** (number) Number or a [formula](#) that returns a number

**Returns:** Money

**Notes:** This [function](#) lets you display a number as money and include the return value in a text string (e.g., "The total amount is \$25.00").

## Example

Assuming PRICE is a [DataSet](#) column of number type with a value of 2539.6:

Expression	Return
MakeMoney(PRICE)	\$2,539.60
"The price is " & PRICE & "."	The price is 2539.6
"The price is " & MakeMoney(PRICE) & "."	The price is \$2,539.60

# MakeTime

**Purpose:** Returns a time value from three numbers.

**Format:** MakeTime(HH, MI, SS)

**Arguments:** (HH) Number from 0-23 that represents the hour of the day, where 12 AM = 0 and 11 PM = 23, (MI) Number from 0-59 that represents the minute, (SS) Number from 0-59 that represents the second.

**Returns:** Time

**Notes:** The MakeTime [function](#) lets you specify a time value in an expression since you cannot directly type a specific time in a [formula](#).

## Example

MakeTime(17, 0, 0) is 5:00 pm, so:

```
If Time () >= MakeTime(17, 0, 0) Then
    "You're working late, I see..."
Else
    "Get back to work."
```

**Related Functions:** [MakeDate](#), [ToDate](#)

# Max

**Purpose:** Returns the maximum value for the records in the current group.

**Format:** Max( number )

**Arguments:** (number) Number or a [formula](#) that returns a number.

**Returns:** Number

**Notes:** This [function](#) (like the other standard aggregate functions – Sum, Avg, Min, Max, Count) is context sensitive. If this function is used in a [data field](#) in a [Group Footer](#), it will return the maximum value for rows in the current group. If in a [Report Footer](#), it will return the maximum for the entire report. If in the [Detail Band](#), it will return a running maximum for the current group. See for more information.

## Example

If your group has two records, and a PRICE column that contains 10.00 for the first record and 20.00 for the second record, the value computed by:

Max(PRICE)

in the group footer would be 20.00 (that is, the maximum of 10 and 20). In the detail [band](#), Max (PRICE) would compute values of 10.00 and 20.00 respectively (a running maximum, or the maximum of the given formula for all the records in the group up to and including the current one).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# Mid

**Purpose:** Returns the specified number of characters from the middle of a text string and uses a second specified value to determine the starting point. For example, to extract "win" from "RPTwin Reports", specify 4 as the starting point and 3 as the return value.

**Format:** Mid (text, number1, number2)

**Arguments:** (text) Text or a [formula](#) that produces text, (number1) Integer number that identifies the starting point for the [function](#), (number2) Integer number that identifies the number of characters to extract.

**Returns:** Text

**Notes:** If the text string contains fewer than the specified number of characters, then all of the text string is returned. Intervening spaces and punctuation symbols are included. Trailing spaces are ignored.

## Example

Expression	Returns
Mid("Hello There!", 7, 5)	There
Mid("Hello There!", 7, 50 )	Hello There!
Mid("Hello There!", 7, 0)	[no output]

**Related Functions:** [Left](#), [Right](#)

# Min

**Purpose:** Returns the minimum value for the records in the current group.

**Format:** Min( number )

**Arguments:** (number) Number or a [formula](#) that returns a number.

**Returns:** Number

**Notes:** This [function](#) (like the other standard aggregate functions – Sum, Avg, Min, Max, Count) is context sensitive. If this function is used in a [data field](#) in a [Group Footer](#), it will return the minimum value for rows in the current group. If in a [Report Footer](#), it will return the minimum for the entire report. If in the [Detail Band](#), it will return a running minimum for the current group. See [for more information](#).

## Example

If your group has two records, and a PRICE column that contains 10.00 for the first record and 20.00 for the second record, the value computed by:

Min(PRICE)

in the group footer would be 10.00 (that is, the minimum of 10 and 20). In the detail [band](#), Min (PRICE) would compute values of 10.00 and 10.00 respectively (a running minimum, or the minimum of the given formula for all the records in the group up to and including the current one).

**Related Functions:** [Avg](#), [Count](#), [GroupCount](#), [GroupAvg](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [SumSum](#)

# Minute

**Purpose:** Returns a number from 0-59 corresponding to the minutes of a time.

**Format:** Minute( time )

**Arguments:** (time) Time or datetime value, or [formula](#) that returns a time or datetime value.

**Returns:** Number

## Example

At 2:35 pm:

Expression	Returns
Minute(Time ())	35

**Related Functions:** [Hour](#), [Second](#), [ToText](#)

# Mod

**Purpose:** Returns the remainder when you divide a number by a second number.

**Format:** Mod( number1, number2 )

**Arguments:** (number1) Number or a [formula](#) that returns a number, (Number2) Number or a formula that returns a number.

**Returns:** Number

## Example

Expression	Returns
Mod(10, 0)	0
Mod(10, 2)	0
Mod(11, 2)	1
Mod(PRICE, 1) * 100	cents of PRICE, if PRICE is in dollars

# Month

**Purpose:** Returns a number from 1-12 corresponding to the month of a date.

**Format:** Month( date)

**Arguments:** (date) Date or datetime value or [formula](#) that returns a date or datetime value.

**Returns:** Number

## Example

On April 15, 1995:

Expression	Returns
Month(Date ())	4

**Related Functions:** [MonthName](#), [MonthNameAbr](#), [ToText](#)

# MonthName

**Purpose:** Returns a text value containing the long name (January, February, March, etc.) of the month of a date.

**Format:** MonthName( date)

**Arguments:** (date) Date or datetime value or [formula](#) that returns a date or datetime value.

**Returns:** Text

## Example

On April 15, 1995:

Expression	Returns
MonthName(Date())	April

**Related Functions:** [Month](#), [MonthNameAbr](#), [ToText](#)

# MonthNameAbr

**Purpose:** Returns a text value containing the abbreviated name (Jan, Feb, Mar, etc.) of the month of a date.

**Format:** MonthNameAbr( date )

**Arguments:** (date) Date or datetime value or [formula](#) that returns a date or datetime value.

**Returns:** Text

## Example

On April 15, 1995:

Expression	Returns
MonthNameAbr(Date())	Apr

**Related Functions:** [Month](#), [MonthName](#), [ToText](#)

# PageNum

**Purpose:** Returns the number of the current page of the report.

**Format:** PageNum()

**Returns:** Number

## Example

On the third page of a report:

Expression	Returns
"Page" & PageNum ()	Page 3

# Quarter

**Purpose:** Returns a number from 1-4 that corresponds to the quarter of a date:

Months	Quarter
January-March	1
April-June	2
July-September	3
October-December	4

**Format:** Quarter( date )

**Arguments:** (date) Date or datetime value or [formula](#) that returns a date or datetime value.

**Returns:** Number

## Example

On April 15, 1995:

Expression	Returns
Quarter(Date())	2

**Related Functions:** [Month](#), [ToText](#), [Week](#)

# RecNum

**Purpose:** Returns the record number of the current [DataSet](#) record.

**Format:** RecNum()

**Returns:** Number

**Notes:** Since the numbers go from 1 to the number of records in your DataSet, each record has a unique number. This number provides an unambiguous way to refer to a specific record, though RPTwin does not use this number internally for any purpose.

## Example

Expression	Returns
RecNum()	1 (for the first record in the DataSet)

# Replace

**Purpose:** Returns text string, but with some of its characters replaced with new characters.

**Format:** Replace( mainText, oldText, newText )

**Arguments:** (mainText) Text or a [formula](#) that returns text, (oldText) A string of text to be replaced. Case is not ignored, so the characters to be replaced must be specified exactly. (newText) Text to replace the oldText. This can be an empty string: "".

**Returns:** Text

## Example

Expression	Replace
Replace("Momma", "m", "p")	Moppa
Replace("Momma", "M", "P")	Pomma
Replace("momma", "m", "p")	poppa
Replace("Dog Kennels", "Ke", "Tu")	Dog Tunnels
Replace("RPTwin", "RPT", "")	win

# ReportAvg

**Purpose:** [Function](#) that returns the average value for all the rows in the entire report.

**Format:** ReportAvg( number )

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number.

**Notes:** This function (unlike the standard aggregate functions – Sum, Avg, Min, Max, Count) is not context sensitive at all. No matter where it is placed in the report, it will return the average for all rows in the entire report. See for more information.

## Example

If your report has two records, and a PRICE column that contains 10.00 for the first row and 20.00 for the second row, the value computed by:

ReportAvg(PRICE)

anywhere in the report would be 15.00 (that is, the average of 10 and 20).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# ReportCount

**Purpose:** [Function](#) that returns the count of all the rows in the entire report.

**Format:** ReportCount( number)

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number.

**Notes:** See the explanation for ReportAvg.

## Example

If your report has two records, the value computed by:

ReportCount(PRICE)

anywhere in the report would be 2.

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# ReportCumAvg

**Purpose:** [Function](#) that returns the average value for all the rows processed so far (a running total).

**Format:** ReportCumAvg( number)

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number.

**Notes:** This function (like the other standard aggregate functions – Sum, Avg, Min, Max, count) is context sensitive. If this function is used in a [data field](#) in a [Group Footer](#), it returns the Average for all the rows processed so far. If this function is placed in the [Detail Band](#), it returns the average for the current group. See [for more information](#).

## Example

For example, if your report has processed two records with a PRICE column that contains the values 10.00 and 20.00, the value computed by:

ReportCumAvg

is 15.00 (the average of the rows processed so far).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# ReportCumMax

**Purpose:** [Function](#) that returns the highest value for all the rows processed so far.

**Format:** ReportCumMax( number)

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number.

**Notes:** See the explanation for ReportCumAvg.

## Example

For example, if your report has processed two records with a PRICE column that contains the values 10.00 and 20.00, the value computed by:

ReportCumMax

is 20.00 (the highest value of all the rows processed so far).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# ReportCumMin

**Purpose:** [Function](#) that returns the lowest value for all the rows processed so far.

**Format:** ReportCumMin( number)

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number.

**Notes:** See the explanation for ReportCumAvg.

## Example

For example, if your report has processed two records with a PRICE column that contains the values 10.00 and 20.00, the value computed by:

ReportCumMin

is 10.00 (the lowest value of all the rows processed so far).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# ReportCumSum

**Purpose:** [Function](#) that returns the total accumulated value for all the rows processed so far (a running total).

**Format:** ReportCumSum( number)

**Arguments:** (number) Number or [formula](#) that produces a number – often just the name of a numeric column from your [DataSet](#), but any formula returning a number may be used.

**Returns:** Number.

**Notes:** See the explanation for ReportCumAvg.

## Example

For example, if your report has two records that have been processed and a PRICE column that contains the values 10.00 and 20.00, the value computed by:

ReportCumSum

is 30.00 (the total value of all the rows processed so far).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportMax](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# ReportMax

**Purpose:** [Function](#) that returns the maximum value for all the rows in the entire report.

**Format:** ReportMax( number)

**Arguments:** (number) Number or [formula](#) that produces a number

**Returns:** Number.

**Notes:** See the explanation for ReportAvg.

## Example

For example, if your report has two records and a PRICE column that contain the values 10.00 and 20.00, the value computed by:

ReportMax

for the entire report is 20.00 (that is , the highest value in the report).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMin](#), [ReportSum](#), [Sum](#)

# ReportMin

**Purpose:** [Function](#) that returns the minimum value for all the rows in the entire report.

**Format:** ReportMin( number)

**Arguments:** (number) Number or [formula](#) that produces a number

**Returns:** Number.

**Notes:** See the explanation for ReportAvg.

## Example

For example, if your report has two records and a PRICE column that contain the values 10.00 and 20.00, the value computed by:

ReportMin

for the entire report is 10.00 (that is, the lowest value in the report).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportSum](#), [Sum](#)

# ReportSum

**Purpose:** [Function](#) that returns the sum of all the rows in the entire report.

**Format:** ReportSum( number)

**Arguments:** (number) Number or [formula](#) that produces a number

**Returns:** Number.

**Notes:** See the explanation for ReportAvg.

## Example

For example, if your report has two records and a PRICE column that contain the values 10.00 and 20.00, the value computed by:

ReportSum

for the entire report is 30.00 (that is, the total value of all the rows in the report).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [Sum](#)

# Right

**Purpose:** Returns the specified number of the rightmost characters of a text string.

**Format:** Right( mainText, number )

**Arguments:** (mainText) Text or a [formula](#) that returns text, (Number)  
Number of the rightmost characters of mainText to return.

**Returns:** Text

**Notes:** If the text string contains fewer than the specified number of characters, then all of the text string is returned.

## Example

Expression	Returns
Right("Hello There!", 5)	here!
Right("Hello There!", 50)	Hello There!
Right("Hello There!", 0)	[no output]

**Related Functions:** [Left](#), [Mid](#),

# Round

**Purpose:** Rounds a number up or down. Returns the closest number to the specified number using the specified precision.

**Format:** Round( numberToRound, precisionNumber )

**Arguments:** (numberToRound) Number or a [formula](#) that returns a number, (precisionNumber) Precision to be used in rounding.

**Returns:** Number

**Notes:** Use this [function](#) to discard fractional amounts (Round(number, 1)), or to show numbers in round thousands (Round(number, 1000)), or round to the thousandths (Round(number, .001)).

## Example

Expression	Returns
Round(98.6, 1)	99 (Round to whole number)
Round(98.3, 1)	98
Round(5678, 100)	5700 (Round to hundreds)
Round(499.9, 1000)	0 (Round to thousands)
Round(500.0, 1000)	1000
Round(12345, 500)	12500 (Round to five-hundreds)
Round(34567, 500)	34500
Round(3.141593, .0001)	3.1416 (Round to ten-thousandths)

**Related Functions:** [Trunc](#)

# RTrim

**Purpose:** Returns a text string with final (trailing) spaces removed. This is useful if your data comes from the database with spaces at the end but you don't want those spaces to show up on your report.

**Format:** RTrim( text )

**Arguments:** (text) Text or a [formula](#) that returns text.

**Returns:** Text

## Example

Expression	Returns
RTrim("Joe Banks ")	Joe Banks (with no trailing spaces)

**Related Functions:** [LTrim](#), [RTrim](#)

# Second

**Purpose:** Returns a number from 0-59 corresponding to the seconds of a time.

**Format:** Second( time )

**Arguments:** (time) Time or datetime value, or [formula](#) that returns a time or datetime value.

**Returns:** Number

## Example

At 2:35:55 pm:

Expression	Returns
Seconds(Time())	55

**Related Functions:** [Hour](#), [Minute](#), [ToText](#)

# Sign

**Purpose:** Returns the sign of a number.

**Format:** Sign( number )

**Arguments:** (number) Number or a [formula](#) that returns a number.

**Returns:** Number, as described below.

<b>If the Number is</b>	<b>Then</b>
Greater than zero	1
Equal to zero	0
Less than zero	-1

## **Example**

```
If Sign(Sum(SALES)) = -1 Then
    "Sales were Negative!"
Else
    Sum (SALES)
```

# Sin

**Purpose:** Returns the sine of a number of radians.

**Format:** Sin(number)

**Arguments:** (number) Number or [formula](#) that produces a number.

**Returns:** Number

**Notes:** The Sine [function](#) serves as the standard mathematical sine function.

## Example

Function	Result
Sin(0)	0
Sin(3.1415926536 / 2)	1

**Related Functions:** Cos[Cos](#), Tan[Tan](#)

# Sum

**Purpose:** Returns the sum of the values for the records in the current group.

**Format:** Sum( number )

**Arguments:** (number) Number or a [formula](#) that returns a number.

**Returns:** Number

**Notes:** This [function](#) (like the other standard aggregate functions – Sum, Avg, Min, Max, Count) is context sensitive. If this function is used in a [data field](#) in a [Group Footer](#), it will return the total for rows in the current group. If in a [Report Footer](#), it will return the total for the entire report. If in the [Detail Band](#), it will return a running total for the current group. See for more information.

## Example

If your group has two records, and a PRICE column that contains 10.00 for the first record and 20.00 for the second record, the value computed by:

Sum(PRICE)

in the group footer would be 30.00 (that is, the sum of 10 and 20). In the detail [band](#), Sum (PRICE) would compute values of 10.00 and 30.00 respectively (a running sum, or the sum of the given formula for all the records in the group up to and including the current one).

**Related Functions:** [Avg](#), [Count](#), [GroupAvg](#), [GroupCount](#), [GroupMax](#), [GroupMin](#), [GroupSum](#), [Max](#), [Min](#), [ReportAvg](#), [ReportCount](#), [ReportCount](#), [ReportCumMax](#), [ReportCumMin](#), [ReportCumSum](#), [ReportMax](#), [ReportMin](#), [ReportSum](#)

# Tan

**Purpose:** Returns the tangent of a number of radians.

**Format:** Tan(number)

**Arguments:** (number) Number or [formula](#) that produces a number.

**Returns:** Number

**Notes:** The Tangent [function](#) serves as the standard mathematical tangent function.

## Example

Function	Result
Tan(0)	0
Tan(3.1415926536 / 4)	1

**Related Functions:** Sin[Sin](#), Cos[Cos](#)

# Time

**Purpose:** Returns a value of type time that represents the time when the report was run.

**Format:** Time()

**Returns:** Number

## Example

If you run the report at 2:35 pm:

Expression	Returns
Time()	2:35:00 pm (exact output depends on the field's selected format)
Hour(Time())	14 (because Hour gives 24-hour range)

# ToDate

**Purpose:** Returns a date and/or time derived from a text string with a specified format. The format uses codes to specify how the resulting date or datetime will be displayed.

**Format:** ToDate( text, format )

**Arguments:** (text) A text string whose values will be interpreted as a date according to the given format, (format) A text string that represents the format the date should be displayed in. The returned value will be the same as the format, with the following character sequences replaced with the indicated values:

If Argument Format is	Then Return Format is
A.M., P.M.	A.M. or P.M.
AM, PM	AM or PM
D	Day of week, 1-7
DAY	Name of day of week, Sunday-Saturday
DD	Day of month, 1-31
DDD	Day of year, 1-366
DY	Abbreviated day of week, Sun-Sat
HH	Hour, 1-12
HH12	Hour, 1-12
HH24	Hour, 0-23
MI	Minutes, 0-59
MM	Month of year, 1-12
MON	Abbreviated month name, Jan-Dec
MONTH	Month name, January-December
Q	Quarter, 1-4
SS	Seconds, 0-59
WW	Week of year, 1-54 (1st week may be partial)
YY	Last two digits of year, 00-99
YYYY	Four-digit year

**Returns:** Date

**Notes:** Characters in the format other than codes will be ignored (so format text "mm/dd/yy" is

equivalent to "mmddy" or "mm dd yy"). If the text contains time information, only the [function](#) sets the date to January 1, 0001, with the corresponding time.

### Example

The following examples assume that the RPTwin default data format settings apply. If you change the default settings, the returns may be different.

<b>Expression</b>	<b>Returns</b>
ToDate("4/15/95", "mm/dd/yy")	April 15, 1995
ToDate("4/15/95", "mm/dd/yyyy")	April 15, 1995
ToDate("4-3-95", "mmddy")	April 3, 1995
ToDate("4-3-95", "ddmmy")	March 4, 1995
ToDate("1/1/95 2:35 pm", "mm/dd/yy hh12:mi am")	January 1, 1995 at 2:35 pm.

**Related Functions:** [ToNumber](#), [ToText](#)

# ToNumber

**Purpose:** Returns a number extracted from a text string.

**Format:** ToNumber( text )

**Arguments:** (text) Text or a [formula](#) that returns text.

**Returns:** Number

**Notes:** Pays attention only to digits, signs and the decimal point, discarding dollar signs, commas, spaces, etc. Use this [function](#) before performing arithmetic on text characters that represent numbers.

## Example

Expression	Returns
"42"	42 (text value)
1 + "42"	Error in formula: Cannot convert Text to Number
ToNumber ("285.47")	285.47 (number)
1 + ToNumber ("194,997")	194998 (number)

**Related Functions:** [ToText](#)

# ToText

**Purpose:** Returns a text value that represents a date and/or time displayed according to a given format.

**Format:** ToText( date, format )

**Arguments:** (date) A date or a [formula](#) that produces a date, a time or a datetime (date and time values combined.), (format) A text string that uses codes to represent the format the date should be displayed in. The returned value will be the same as the format, with the codes replaced as indicated in the following table:

<b>If Argument Format is</b>	<b>Then Return Format is</b>
A.M., P.M.	A.M. or P.M.
AM, PM	AM or PM.
D	Day of week, 1-7
DAY	Name of day of week, Sunday-Saturday
DD	Day of month, 1-31
DDD	Day of year, 1-366
DY	Abbreviated day of week, Sun-Sat
HH	Hour, 1-12
HH12	Hour, 1-12
HH24	Hour, 0-23
MI	Minutes, 0-59
MM	Month of year, 1-12
MON	Abbreviated month name, Jan-Dec
MONTH	Month name, January-December
Q	Quarter, 1-4
SS	Seconds, 0-59
WW	Week of year, 1-54 (1st week may be partial)
YY	Last two digits of year, 00-99
YYYY	Four-digit year

The codes may be written in lower, mixed or upper case, with the results as follows: If the code represents a number (such as MM for the month number) uppercase returns a leading zero if the number has only one digit and lowercase does not return a leading zero. For example, if today's date is April 5, 1995, then:

<b>Expression</b>	<b>Returns</b>
ToText(Date(), "MM/DD/YY")	04/05/95 (with leading zeros in month and day)
ToText(Date(), "mm/dd/yy")	4/5/95 (without leading zeros in month and day)

If you have a long column of dates it is easier to read if you choose the first option. If you just want to return a month or day number by itself, choose the second option to avoid returning a leading zero.

If the code represents text, such as MONTH for the name of the month, the capitalization of the code affects the capitalization of the resulting text. For example, if the current month is April, then:

<b>Expression</b>	<b>Returns</b>
ToText(Date (), "MONTH")	APRIL
ToText(Date (), "Month")	April
ToText(Date (), "month")	april

These are the only three options for capitalization of the output text; that is, "MoNtH" will produce "April", not "ApRiL".

**Returns:** Text

**Example**

On April 15, 1995 at 2:35 pm:

**Expression**

ToText(Date ( ),  
"Day, Month dd, yyyy hh12:MI a.m.")

ToText(Date ( ), "ddMonyy")

ToText(Date ( ), "MM/DD/YY HH24:MI:SS")

**Returns**

Saturday, April 15, 1995 2:35

15Apr95

04/15/95 14:35:00

**Related Functions:** [ToDate](#), [ToNumber](#)

# Trim

**Purpose:** Returns a text string with leading and trailing spaces removed.

**Format:** Trim( text )

**Arguments:** (text) Text or a [formula](#) that returns text.

**Returns:** Text

**Notes:** This is useful if your data comes from the database with spaces at the beginning or end but you don't want those spaces to show up in your report.

## Example

Expression	Returns
Trim(" Joe Banks ")	Joe Banks (with no leading or trailing spaces)

**Related Functions:** [LTrim](#), [RTrim](#)

# Trunc

**Purpose:** Returns the largest number not greater than N (the specified number) that is a multiple of P (the specified precision).

**Format:** Trunc( number, precision )

**Arguments:** (number) Number or a [formula](#) that returns a number. (Precision) Number or a formula that returns a number.

**Returns:** Number

**Notes:** Use to discard fractional amounts (Trunc(N, 1)), or to show numbers in thousands (Trunc(N, 1000)), or thousandths (Trunc(N, .001)).

This [function](#) differs from the Round function only in that it always rounds down; that is, Trunc always throws away fractional parts while Round rounds them up or down.

## Example

Expression	Returns
Trunc(98.6, 1)	98 (Truncate to whole number)
Trunc(98.3, 1)	98
Trunc(5678, 100)	5600(Truncate to hundreds)
Trunc(499.9, 1000)	0 (Truncate to thousands)
Trunc(500.0, 1000)	0
Trunc(12345, 500)	12000(Truncate to five-hundreds)
Trunc(34567, 500)	34500
Trunc(3.141593, .0001)	3.1415 (Truncate to ten-thousandths)

**Related Functions:** [Round](#)

# UCase

**Purpose:** Returns a text string with all characters converted to uppercase.

**Format:** Ucase( text )

**Arguments:** (text) Text or a [formula](#) that returns text.

**Returns:** Text all in uppercase

## Example

If ITEM is "dOg KEnneLs":

Expression	Returns
UCase(ITEM)	DOG KENNELS
UCase("Dog Kennels")	DOG KENNELS

**Related Functions:** [InitCap](#), [LCase](#)

# Week

**Purpose:** Returns the number from 1-54 of the week of the year of a date.

**Format:** Week( date )

**Arguments:** (date) Date or datetime, or a [formula](#) that returns a date or datetime.

**Returns:** Number

**Notes:** Weeks are defined to begin on Sunday and end on Saturday, so while January 1st is always in the first week of the year, January 2nd can fall in the second week of the year if it falls on a Sunday. In 1995, for example, January 8th (being a Sunday) was the first day of the second week of the year.

## Example

Expression	Returns
Week(MakeDate(1, 1, 95))	1
Week(MakeDate(1, 2, 95))	1
Week(MakeDate(1, 8, 95))	2
Week (MakeDate(1, 9, 95))	2
Week (MakeDate(1, 15, 95))	3

**Related Functions:** [Month](#), [Quarter](#), [ToText](#), [Year](#)

# Year

**Purpose:** Returns a number corresponding to the year of a date.

**Format:** Year( date )

**Arguments:** (date) Date or datetime, or a [formula](#) that returns a date or datetime.

**Returns:** Number

## Example

Assume that today is 4/15/54:

Expression	Returns
Year(MakeDate(1, 1, 95))	1995
Year(Date())	1954

**Related Functions:** [Month](#), [Quarter](#), [ToText](#), [Week](#)

# YearsBetween

**Purpose:** Returns the number of whole years between two dates.

**Format:** YearsBetween( date )

**Arguments:** (date) A date or datetime, or a [formula](#) that produces a date or datetime.

**Returns:** Integer number.

## Example

Assume that today is 4/15/95:

Expression	Returns
DaysBetween(MakeDate(1,1,90), Date())	5

**Related Functions:** [Age](#), [DaysBetween](#)

# Glossary of Terms

[Aggregate function](#)

[Band](#)

[Band properties](#)

[Blank report](#)

[BPwin](#)

[Columnar report](#)

[Data field](#)

[DataSet](#)

[DataSet Columns list](#)

[Design window](#)

[Detail band](#)

[ERwin](#)

[Field properties](#)

[Filtering](#)

[Formula](#)

[Function](#)

[Grid](#)

[Group footer](#)

[Group header](#)

[Group-level function](#)

[Group/Totals report](#)

[Grouping](#)

[Guided reports](#)

[Page footer](#)

[Page header](#)

[Preview window](#)

[Quick reports](#)

[Report footer](#)

[Report header](#)

[Report-level function](#)

[RPTwin report data file](#)

[RPTwin report definition file](#)

[Sorting](#)

[Text fields](#)

[Two-pass reporting](#)

Vertical report

## Aggregate function

A [function](#) that summarizes the data values in the report. A context-sensitive aggregate function (Avg, Count, Max, Min, and Sum) performs calculations based on its location in the report. For example, if you place Count in the [Page Footer](#), it counts and totals the records on the current page. If you place Count in the [Report Footer](#), it counts and totals the records for the entire report.

Non-context sensitive aggregate functions include both Report-level (ReportAvg, ReportCount, ReportMax, ReportMin, and ReportSum) and Group-level (GroupAvg, GroupCount, GroupMax, GroupMin, and GroupSum) functions. A [report-level function](#) performs calculations for the entire report regardless of where it is located in the report. For example, if you place ReportCount in either the [Page Header](#) or Page Footer, it counts and totals the records for the entire report. A [group-level function](#) works the same way except it performs calculations for each group in the report.

## Band

A horizontal section in the [Design window](#) where you place report objects (e.g., report title, page numbers, etc.) or fields. Every report must include a [Detail Band](#) and may optionally include other bands (e.g., [Report Header](#) and Footer, Group Headers and Footers, [Page Header](#) and Footer). To add or remove an optional band, select the appropriate toggle option in the View menu.

## **Band properties**

RPTwin includes a [Band](#) Detail Editor that lets you define the characteristics of a band. You can control the appearance of a band by changing its properties including: format (e.g., whether or not it has a border or pattern), size (e.g., height and width), and printing options (e.g., start new page after/before band is printed).

## Blank report

A Quick Report layout option that lets you build a report from a blank [Design window](#). To add data fields, use the drag-and-drop method to copy an item from the [DataSet](#) Column list to the report. To add report objects (e.g., page breaks, [text fields](#), etc.), use the point and click method to select a tool in the Toolbox and insert the object into the report.

## **BPwin**

Logic Works' business process modeling tool in which you can generate a customized or predefined report data file and send the information to RPTwin for additional formatting.

## **Columnar report**

A multi-column Quick Report layout option in which RPTwin automatically arranges the report data in tabular columns and inserts the column name at the top of each column.

## Data field

A field that displays information about the [ERwin](#) data model or [BPwin](#) business process model from which the report was generated. For example, if the report provides information about the entities in a data model, Entity Name might be a data field in the report. RPTwin displays the actual value in the data field (e.g., CUSTOMER, MOVIE, etc.) when you print or preview the report.

## **DataSet**

A report data file (.LWD) that includes information from the [ERwin](#) data model or [BPwin](#) business process model. RPTwin uses the DataSet sent from ERwin or BPwin to layout the report. RPTwin automatically links the report definition file (.LWR) to the DataSet that was used to start the report in RPTwin. If the DataSet changes after the .LWR is saved (e.g., you modify the model and generate a new .LWD file), RPTwin asks you to link the .LWR file to the new .LWD file and warns you if there are differences between the previous and current DataSet.

## **DataSet Columns list**

A moveable list box that includes all of the data fields in the current report. To add data to the report, select and drag the [data field](#) from the [DataSet](#) Columns list and drop it where you want it to appear in the [Design window](#). If you delete a data field from the Design window, it is still included in the DataSet Columns list box.

## Design window

The RPTwin work area in which you design or enhance a report. The Design window includes a ruler, [grid](#), toolbar, toolbox, and one or more bands where you can place objects or fields. When you start a new report or open an existing report, the report objects automatically appear in the Design window. The Preview button on the Toolbar lets you switch to the [Preview window](#) to see a WYSIWIG version of the report before you print it.

## Detail band

An area in the body of the report that displays the detail information from the [ERwin](#) data model or [BPwin](#) business process model from which the report is generated. For example, if the report provides a list of attributes for each entity in the data model, the Detail [band](#) might include the actual entity and attribute names.

## **ERwin**

Logic Works' data modeling tool in which you can generate a customized or predefined report data file and send the information to RPTwin for additional formatting.

## Field properties

RPTwin includes a [Data Field](#) Properties Editor and a Text Field Properties Editor that let you define the characteristics of a field. You can control the appearance of a field by changing many of its properties, including its format (e.g., bold, italic), size (e.g., height and width), and position (e.g., distance from the margin or top of the report). You can also modify the properties that control how RPTwin displays the data in a field. For example, the Word Wrap option determines if RPTwin prints text on the next line or if it truncates (cuts off) the text at the field boundary line.

## Filtering

RPTwin includes a Filter Editor that lets you select a subset of data for a report. For example, if the report data file includes all the entities in the data model, you can filter the data to include only a subset of entities in the report. RPTwin uses the [formula](#) that you enter to determine how to filter the report data.

## Formula

RPTwin includes a Formula Editor that lets you add a formula to a field. RPTwin uses the formula to determine how to format and/or calculate a field's output. A formula can simply be the name of a [data field](#) or it can include other functions. For example, RPTwin uses the formula **U Case ({Attribute Name})** to convert the attribute names in the report from lower case to upper case characters (e.g., customer-name displays as CUSTOMER-NAME).

## Function

A predefined statement that you can insert in a [formula](#) to perform calculations and manipulate the output of report data. For example, the GroupSum function totals the values in a group and the UCase function displays text in upper case characters. RPTwin provides a list of supported functions in its Formula Editor.

## Grid

An optional array of dots that can be shown or hidden in the [Design window](#) to help you align the objects in the report.

## **Group footer**

An area in the body of the report that summarizes information about a group. For example, if the report is grouped by entity, the group footer might include the total number of attributes for each entity.

## **Group header**

An area in the body of the report that introduces information about a group. For example, if the information in the report is grouped by entity, the group header might include Entity Name as the group title.

## Group-level function

A type of aggregate [function](#) that calculates the values for each group regardless of where you place it in the [Design window](#). For example, if you want to display the group total before the group detail information, you can place a [formula](#) with the GroupSum function in the [Group Header band](#). RPTwin scans ahead to total the records in each group and then places the sum in the Group Header area. As an exception, if you place a group-level function in the [Report Header](#) or [Report Footer](#), RPTwin promotes it to a [report-level function](#).

## **Group/Totals report**

A multi-column Guided Report layout option that lets you specify how you want RPTwin to group and/or sort the information in the report. After you respond to a series of prompts, RPTwin automatically arranges, groups, sorts and totals the report data as you specified.

## Grouping

RPTwin includes a [Sorting](#) and Grouping Editor that lets you group (categorize) report information based on the [data field](#)(s) that you select. For example, if a report includes two data fields, Entity Name and Attribute Name, and is grouped by Entity Name, RPTwin displays the first entity name followed by its attributes. Then, it displays the next entity name followed by its attributes, and so on.

## Guided reports

An option on the New Report dialog that takes you through a step-by-step process to create a [columnar report](#) (with group totals) or a [vertical report](#). After you respond to a series of prompts, RPTwin uses the information to automatically lay out the report.

## **Page footer**

An area at the bottom of every report page that displays information about the contents of the current page (e.g., report title).

## **Page header**

An area at the top of every report page that displays information about the contents of the current page (e.g., page number).

## Preview window

The RPTwin window in which you can preview a report on screen before you print it. The Preview window has its own toolbar that lets you display the report in one or two page increments, page forward or backward, zoom in or zoom out, switch to the [Design window](#), and print the report.

## Quick reports

An option in the New Report dialog that lets you quickly create a columnar, vertical, or [blank report](#) layout. If you choose a columnar or vertical Quick Report, RPTwin automatically arranges the report data fields in the specified format. If you choose the blank report option, RPTwin displays a blank [Design window](#) and lets you arrange the data fields as you like.

## **Report footer**

The area at the bottom of the report that displays information summarizing the entire report. For example, if the report identifies all of the entities in the data model, the Report Footer area might include the total number of entities in the data model.

## **Report header**

The area at the beginning of the report that displays information about the entire report (e.g., the report's title).

## Report-level function

A type of aggregate [function](#) that calculates the values for the entire report regardless of where you place it in the [Design window](#). For example, if you want the report totals to appear on the first page of the report, you can insert a [formula](#) with a report-level function such as ReportSum in the [Report Header band](#). RPTwin scans ahead to total the records in the whole report and then places the sum in the Report Header area.

## **RPTwin report data file**

A file that is saved in [ERwin](#) or [BPwin](#) as an .LWD file and includes the report data (e.g., contents, [filtering](#), and formatting options) from the current model. RPTwin uses the information in this file to automatically layout the report.

## **RPTwin report definition file**

A file that is saved in RPTwin as an .LWR file and includes report layout information (e.g., fonts, position of report objects, [grouping](#) and [sorting](#) options, etc.) as well as the pathname of the report data file to which it is linked.

## Sorting

By default, RPTwin sorts report information in ascending order (e.g., A-Z). But, RPTwin includes a Sorting and [Grouping](#) Editor that lets you change the sorting order to descending (e.g., Z-A).

## **Text fields**

You can add a text field using the Text tool anywhere in the report to provide additional information about the report (e.g., report title, [page header](#), labels, etc.).

## Two-pass reporting

A method that RPTwin uses with certain Aggregate functions (Sum, Avg, Min, Max, and Count) to calculate group-level and report-level values regardless of where the [function](#) is located in the report. For example, if you want to display the report totals on the first page, you can place the [report-level function](#), ReportSum, in the [Report Header](#). In the first pass, RPTwin scans ahead and calculates the totals for all records processed in the report. Then, it returns to the header and inserts the report total.

## Vertical report

A Quick or Guided Report layout option in which RPTwin arranges the data fields vertically (one-after-the-other) in a single column. RPTwin inserts a label before each field (e.g., **ENTITY NAME:** CUSTOMER; where ENTITY NAME is the field label and CUSTOMER is the value in the [data field](#)).

