



Pentaho Advanced Reporting Guide

Understanding the Pentaho Reporting File Definition

Version 1.1 Milestone 6
May 24, 2006

Copyright © 2005 Pentaho Corporation. Redistribution permitted. All trademarks are the property of their respective owners.

For the latest information, please visit our web site at www.pentaho.org

Table of Contents

Introduction	4
Report Layout	5
Report Sections	5
Base XML	5
Position Objects	6
Report Settings.....	7
Report Header and Footer	8
<reportheader></reportheader>	9
<reportfooter></reportfooter>	10
Page Header and Footer	12
Page	12
<pageheader>	13
<pagefooter>	15
Groups.....	16
<groupheader>.....	18
<groupfooter>	20
Items.....	21
<items></items>	21
<items>	22
Types of Report Objects.....	23
Report Object Attributes	24
anchor	24
band	24
date-field	26
drawable-field	28
drawable-url-field	29
drawableref.....	29
image-field, image-ref, imageurl	29
label	30
line	32
message-field	32
number-field	35
rectangle.....	37
resource-field	38
resource-label	40
shape-field	42
string-field.....	43
Functions and Expressions	44
Summary (Grand and Group) based Calculations	44
CountDistinctFunction	45
GroupCountFunction	45
ItemAvgFunction	46
ItemColumnQuotientExpression	46
ItemCountFunction	46
ItemHideFunction	46
ItemMaxFunction.....	47
ItemMinFunction.....	47
ItemPercentageFunction	47
ItemSumFunction	47
TotalCalculationFunction	48
TotalGroupCountFunction.....	48
TotalGroupSumFunction.....	48
TotalGroupSumQuotientFunction	49
TotalGroupSumQuotientPercentFunction	49
TotalItemCountFunction.....	50

Item Calculations	50
ColumnAverageExpression	51
ConvertToNumberExpression.....	51
BSHExpression	52
PercentageExpression	52
Advanced Calculations	53
CreateGroupAnchorsFunction	53
CreateHyperLinksFunction	53
ElementColorFunction	54
ElementVisibilitySwitchFunction	54
HideElementByNameFunction	55
Hide Page Head and Footer if report is not a pageable export	55
PageFunction	55
PageOfPagesFunction	56
PageTotalFunction	56
TextFormatExpression.....	56
Property Reference	56
Pentaho Extended Functions for Charting	57
Collect Chart Data.....	57
Set Chart Type and attributes.....	58
Display Chart.....	60
Configuration	60
CSV default separator	60
CSV write column name into first row	60
Excel output set paper size.....	60
Excel output set paper orientation	60
PDF set author	62
PDF set title	62
PDF encoding	62
PDF embed fonts.....	62
PDF version.....	62
PDF Security, Allow Copy	62
PDF Security, Allow Printing	62
PDF Security, Allow Modifying Contents	63
PDF Security, Allow Modifying Annotations	63
PDF Security, Allow Fill-in of Document Forms.....	63
PDF Security, Allow access for screenreaders	63
PDF Security, Allows reassembly of document	63
PDF Security, Allow low quality printing.....	63
PDF Security, Encrypt	63
PDF Security, Password.....	64
PDF Security, Owner	64

Introduction

When looking at any production, managerial, financial or operational report, each displays detail data, organizes it by certain categories and specifies which pieces of data gets presented on the first page, last page or on every page. In addition, these report types can contain:

- Analytical computations that are simple like performing subtotaling or averaging by group or complex like computing each detail line time's percentage of their group's total or highlighting line items or groups for exception reporting,
- Objects like lines or bands to emphasize totaling or other metrics
- Images for corporate logos or watermarks

This guide will explain:

- The report xml definition for a Pentaho Reporting document based on JFreeReport 0.8.7.
- Report layout and object placement
- Formatting
- The use of functions
- Configuration and layout

Report Layout

The report xml definition for Pentaho Reports contains the following major sections; Report Header, Report Footer, Page Header, Page Footer, Groups including each group along with their corresponding header and footer, and Details/Items.

Report Sections

Elements	Purpose
include	Allows a report to reference another xml document for other report sections. For instance, multiple reports can reference a single report with only a page header defined for easier report maintenance.
parser-config	Sets variables with corresponding values for a report to use at runtime. Thus, common values can be set throughout a report like common text, background color or fonts. Note: The parser-config only defines replacement values for the parser itself. There is no parser-config anymore once the report has been parsed.
configuration	Allows parameters to be set for special output handling or debugging purposes.
watermark	Printed before any other object is printed on a new page. This band can consume the complete space of the page and will never trigger a pagebreak. The watermark-band is intended to fill the page background.
reportheader	Determines content that prints on the first page of a report.
reportfooter	Determines content that prints on the last page of a report.
pageheader	Determines content that prints at the top of each page of a report.
pagefooter	Determines content that prints at the bottom of each page of a report.
groups	Determines how the report is grouped and the content of each group's header and footer.
items	Determines the fields and content will be the line items or details.
functions	Computes results based on predefined or customized mathematical expressions or complex scripts.

Base XML

Below is the base definition for a report.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE report PUBLIC "-//JFreeReport//DTD report definition//EN//simple/version 0.8.5"
"http://jfreereport.sourceforge.net/report-085.dtd">
<report>
  <include>
  </include>

  <parser-config>
  </parser-config>

  <configuration>
  </configuration>

  <watermark>
  </watermark>

  <reportheader>
  </reportheader>

  <pageheader>
  </pageheader>

  <groups>
  </groups>

  <items>
  </items>

  <pagefooter>
```

```

</pagefooter>

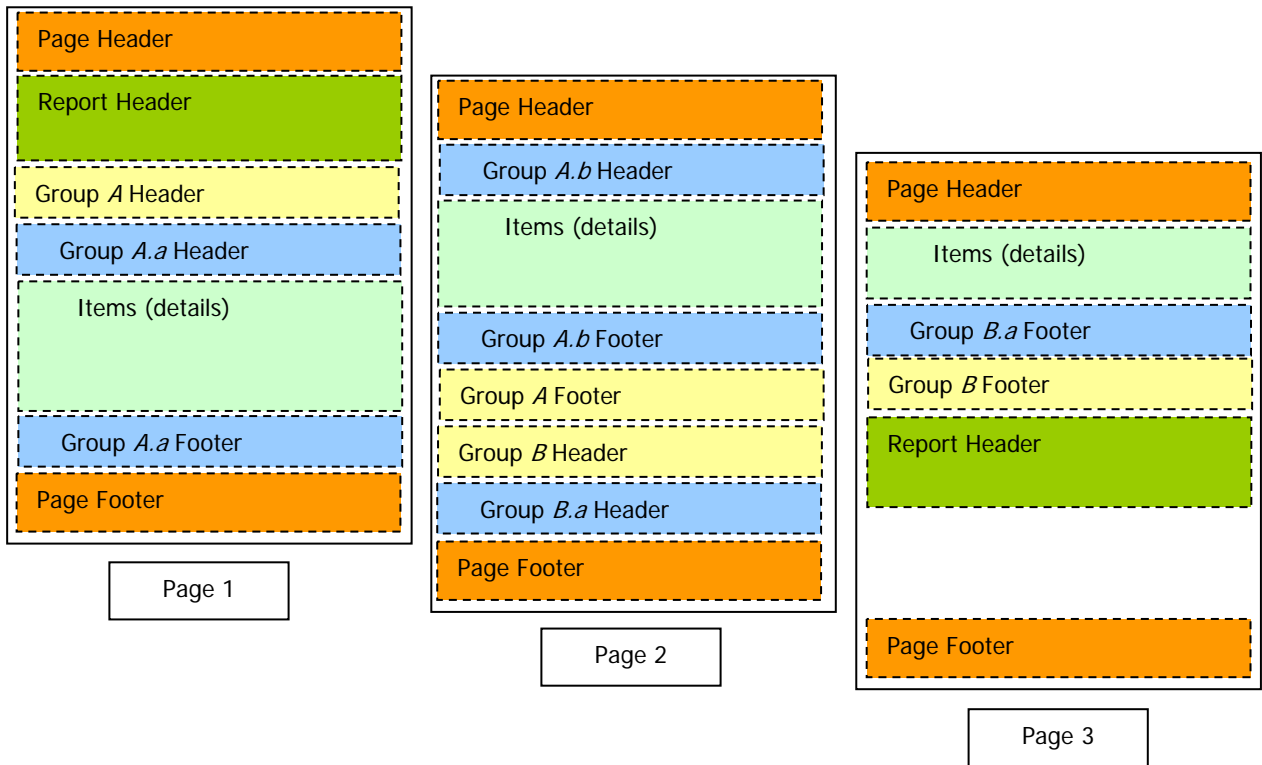
<reportfooter>
</reportfooter>

<functions>
</functions>

<watermark>
</watermark>
</report>

```

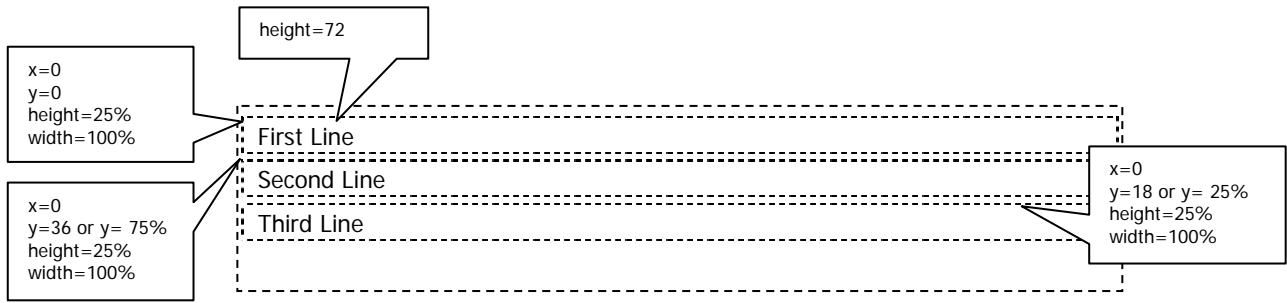
Example of Report Output



Position Objects

The position of an object in a section is declared by defining the x and y coordinates within a particular section. The top left starting point is x=0 and y=0 and is also the default position. When using an object, the mandatory attributes that need to be specified are the x (horizontal position), y (vertical position), height and width. These attributes are defined in 1/72 inch unit increments. These can be defined in units or percent. When using percent to define y and height, the container's height must be specified. The container's height is computed automatically. If a height is defined, it serves as a minimum-height. Therefore a band can increase its height beyond that size. Relative x and width values are resolved against the width of the parent (which corresponds to 100%). Relative y and height values are resolved against the used height of an container. For that computation, the reporting engine first computes the effective height of all absolutely positioned elements and uses that height as 100%. Then it computes the complete layout for all elements, with all relative sizes resolved against that height.

X & Y Coordinate Example



As with every element container, you may define default font settings for sub elements without their own font definition.

Report Settings

In a report definition, report attributes set margins, page size and orientation.

Example

```
<report rightmargin="72" topmargin="72" bottommargin="72" leftmargin="72" name="Quadrant For Region" orientation="portrait" pageformat="LETTER" >
```

<report>

Attributes	Description
pageFormats	<p>Defines the page height and width based on predefined page types.</p> <p>Values: PAPER10X11 PAPER10X13 PAPER10X14 PAPER12X11 PAPER15X11 PAPER7X9 PAPER8X10 PAPER9X11 PAPER9X12 A0 A1 A2 A3 A3_TRANSVERSE A3_EXTRA A3_EXTRATRANSVERSE A3_ROTATED A4 A4_TRANSVERSE A4_EXTRA A4_PLUS A4_ROTATED A4_SMALL A5 A5_TRANSVERSE A5_EXTRA A5_ROTATED A6 A6_ROTATED A7 A8 A9 A10 ANSIC ANSID ANSIE ARCHA ARCHB ARCHC ARCHD ARCHE B0 B1 B2 B3 B4 B4_ROTATED B5 B5_TRANSVERSE B5_ROTATED B6 B6_ROTATED B7 B8 B9 B10 C4 C5 C6 COMM10 DL DOUBLEPOSTCARD DOUBLEPOSTCARD_ROTATED ENV9 ENV10 ENV11 ENV12 ENV14 ENVC0 ENVC1 ENVC2 ENVC3 ENVC4 ENVC5 ENVC6 ENVC65 ENVC7 ENVCHOU3 ENVCHOU3_ROTATED ENVCHOU4 ENVCHOU4_ROTATED ENVDL ENVINVITE ENVISOB4 ENVISOB5 ENVISOB6 ENVITALIAN ENVKAKU2 ENVKAKU2_ROTATED ENVKAKU3 ENVKAKU3_ROTATED ENVMONARCH ENVPERSONAL ENVPRC1 ENVPRC1_ROTATED ENVPRC2 ENVPRC2_ROTATED ENVPRC3 ENVPRC3_ROTATED ENVPRC4 ENVPRC4_ROTATED ENVPRC5 ENVPRC5_ROTATED ENVPRC6 ENVPRC6_ROTATED ENVPRC7 ENVPRC7_ROTATED ENVPRC8 ENVPRC8_ROTATED ENVPRC9 ENVPRC9_ROTATED ENVPRC10 ENVPRC10_ROTATED ENVYOU4 ENVYOU4_ROTATED EXECUTIVE FANFOLDUS FANFOLDGERMAN FANFOLDGERMANLEGAL FOLIO ISOB0 ISOB1 ISOB2 ISOB3 ISOB4 ISOB5 ISOB5_EXTRA ISOB6 ISOB7 ISOB8 ISOB9 ISOB10 LEDGER LEGAL LEGAL_EXTRA LETTER LETTER_TRANSVERSE LETTER_EXTRA LETTER_EXTRATRANSVERSE LETTER_PLUS LETTER_ROTATED LETTER_SMALL MONARCH NOTE POSTCARD POSTCARD_ROTATED </p>

	PRC16K PRC16K_ROTATED PRC32K PRC32K_ROTATED PRC32K_BIG PRC32K_BIGROTATED QUARTO STATEMENT SUPERA SUPERB TABLOID TABLOIDEXTRA
orientation	Determines the page orientation. Values: portrait landscape
height	Determines the height of each page if pageFormat is not specified. Value is a whole number and each increment is equivalent to 1/72 of an inch.
width	Determines the width of each page if pageFormat is not specified. Value is a whole number and each increment is equivalent to 1/72 of an inch.
name	Report Name (optional)
leftmargin	Determines the left margin of each page. Value is a whole number and each increment is equivalent to 1/72 of an inch.
rightmargin	Determines the right margin of each page. Value is a whole number and each increment is equivalent to 1/72 of an inch.
topmargin	Determines the top margin of each page. Value is a whole number and each increment is equivalent to 1/72 of an inch.
bottommargin	Determines the bottom margin of each page. Value is a whole number and each increment is equivalent to 1/72 of an inch.
pagespan	Determines the number of pages the report width will extend.

Report Header and Footer

This section defines content that will print once at the beginning and end of the report. All objects contained within this section will inherit this attribute unless specified directly by the objects.

The report header can contain any band-element. The height of the report header is ignored, if the header and footer is printed on the same page.

Report Section	XML Syntax	Description
Report Header	<reportheader> </reportheader>	Prints on the first page.
Report Footer	<reportheader> </reportheader>	Prints on the last page.

Example

First Line Second Line Third Line

--


```
<reportheader height="72" alignment="left" font="Arial" fontsize="12">
<label height="25%" width="100%" alignment="left" x="0" y="0">First Line</label>
<label height="25%" width="100%" alignment="left" x="0" y="25%">Second Line</label>
<label height="25%" width="100%" alignment="left" x=0 y="50%">Third Line</label>
<\reportheader>
```

or

```
<reportheader alignment="left" font="Arial" fontsize="12">
<label height="18" width="100%" alignment="left" x="0" y="0">First Line</label>
<label height="18" width="100%" alignment="left" x="0" y="18">Second Line</label>
<label height="18" width="100%" alignment="left" x=0 y="36">Third Line</label>
<\reportheader>
```

<reportheader></reportheader>

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
fixedposition	Defines the absolute position on the page where the band should be printed. If the current cursor position is before that y-position, then empty space is added. If the current position is after that y-position, the band is printed on the next page (on that y-position).
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF export. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is used as a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean

	Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikethr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number and each increment is equivalent to 1/72 of an inch. Objects don't inherit this setting. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
pagebreak-after-print	If set to true, the page breaks after the report header is printed. Type: Boolean Default value: false
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the report header will printed. Type: Boolean Default Value: true

<reportfooter></reportfooter>

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left

	Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
fixedposition	Defines the absolute position on the page where the band should be printed. If the current cursor position is before that y-position, then empty space is added. If the current position is after that y-position, the band is printed on the next page (on that y-position). Type: Integer
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Value: false
font-encoding	Specifies the encoding to be used for the PDF export. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikethr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number and each increment is equivalent to 1/72 of an inch. Objects don't inherit this setting. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section.

	Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
pagebreak-before-print	If set to true, the page breaks before the report footer is printed. Type: Boolean Default Value: false
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: True
vertical-alignment	Sets the vertical position of the text. Type: String Default Value: top Values: top middle bottom
visible	Sets whether the report header will printed. Type: Boolean Default Value: true

Page Header and Footer

This section defines content that will print every page of the report. The page header can contain any band-element aka section. The page header/footer should not contain dynamic elements. The page header cannot create page breaks. If the combined content in the page header and page footer is larger than the space for a single page or no other content can fit on the page after the page header has printed and space for the page footer has been reserved, the report processing will fail.

As with every section, the font settings for this section are inherited to the sub elements without an own font definition.

Page

Report Section	XML Syntax	Description
Page Header	<pageheader> </pageheader>	Prints before any content is printed on the page.
Page Footer	<pagefooter> </pagefooter>	Prints after the last content for the page is printed. The page footer is always positioned at the bottom of a page, regardless how much space of the page is filled.

Example

First Line
 Second Line
 Third Line

```
<pageheader height="72" alignment="left" font="Arial" fontsize="12">
<label height="25%" width="100%" alignment="left" x="0" y="0">First Line</label>
<label height="25%" width="100%" alignment="left" x="0" y="25%">Second Line</label>
<label height="25%" width="100%" alignment="left" x=0 y="50%">Third Line</label>
<\pageheader>
```

or

```
<pageheader alignment="left" font="Arial" fontsize="12">
<label height="18" width="100%" alignment="left" x="0" y="0">First Line</label>
<label height="18" width="100%" alignment="left" x="0" y="18">Second Line</label>
<label height="18" width="100%" alignment="left" x=0 y="36">Third Line</label>
<\pageheader>
```

<pageheader>

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String

	Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikethr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number and each increment is equivalent to 1/72 of an inch. Objects don't inherit this setting. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
onfirstpage	If set to false, content is not printed on first page. Type: Boolean Default Value: true
onlastpage	If set to false, content is not printed on last page. Type: Boolean Default Value: true
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the page header will printed. Type: Boolean Default Value: true

<pagefooter>

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstriker	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number and each increment is equivalent to 1/72 of an inch. Objects don't inherit this setting. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section.

	Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
onfirstpage	If set to false, content is not printed on first page. Type: Boolean Default Value: true
onlastpage	If set to false, content is not printed on last page. Default is true. Type: Boolean Default Value: true
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the report header will printed. Type: Boolean Default Value: true

Groups

This section defines content that will print for each group defined on the report. The tag encapsulates all groups. This tag helps to keep parsing simple. If no groups are defined, a default group is created and contains all elements of the report data row.

A fields list may only contain strings defining the names of the items which form a group. This is not limited to items from the data model, you may also enter expressions here.

If you define a subgroup, then you'll have to include all fields of the parent group and at least one new field. The field is not required to exist. If the field does not exist during the report processing, the value 'null' is used as replacement for the field's content.

The group ordering is defined by the contents of the <fields> tag of the groups. A less specific group is printed before any more specific group. A group is considered more specific than an other group, if it contains more fields in its field list.

Report Section	XML Syntax	Description
Groups	<groups> </groups>	There can be one or more groups definitions in the "groups" element. If no groups are defined, a default group is created

		to contain all data elements of the current report.
Group	<group></group>	Defines a group inside a groups element. (A group has a unique name and a list of fields.
Group Header	<groupheader></groupheader>	Prints as the header for a group.
Group Footer	<groupfooter></groupfooter>	Prints as the footer for a group

Example

The text in italics is sample data

```

1ST GROUP BY
BEGIN <first group by A>
  2ND GROUP BY
  BEGIN <second group by A.a>
    <details>
  2ND GROUP BY
  END <second group by A.a>
  2ND GROUP BY
  BEGIN <second group by A.b>
    <details>
  2ND GROUP BY
  END <second group by A.b>
1ST GROUP BY
END <first group by A>
1ST GROUP BY
BEGIN <first group by B>
  2ND GROUP BY
  BEGIN <second group by B.a>
    <details>
  2ND GROUP BY
  END <second group by B.a>
  2ND GROUP BY
  BEGIN <second group by B.b>
    <details>
  2ND GROUP BY
  END <second group by B.b>
1ST GROUP BY
END <first group by B>

```

```

<groups>
  <group name="FIRST GROUP ">
    <fields>
      <field>FIRST_GROUP_COLUMN_FIELD</field>
    </fields>
    <groupheader height="36" font="Arial" fontsize="10">
      <label height="18" alignment="left" width="100%" x="0" y="0">1ST GROUP BY</label>
      <message-field height="18" alignment="left" width="100%" x="0" y="18">
        BEGIN $(FIRST_GROUP_COLUMN_FIELD)
      </message-field>
    </groupheader>
    <groupfooter height="36" font="Arial" fontsize="10">
      <label height="18" alignment="left" width="100%" x="0" y="0">1ST GROUP BY </label>
      <message-field height="18" alignment="left" width="100%" x="0" y="18">
        END $(FIRST_GROUP_COLUMN_FIELD)
      </message-field>
    </groupfooter>
  </group>
  <group name="SECOND GROUP ">
    <fields>
      <field>FIRST_GROUP_COLUMN_FIELD</field>
      <field>SECOND_GROUP_COLUMN_FIELD</field>
    </fields>
    <groupheader height="36" font="Arial" fontsize="10">

```

```

<label height="18" alignment="left" width="90%" x="10%" y="0">2ND GROUP BY</label>
<message-field height="18" alignment="left" width="90%" x="10%" y="18">
  BEGIN $(SECOND_GROUP_COLUMN_FIELD)
</message-field>
</groupheader>
<groupfooter height="36" font="Arial" fontsize="10">
<label height="18" alignment="left" width="90%" x="0" y="0">2ND GROUP BY</label>
<message-field height="18" alignment="left" width="90%" x="10%" y="18">
  END $(SECOND_GROUP_COLUMN_FIELD)
</message-field>
</groupfooter>
</group>
</group>
</groups>

```

<groupheader>

Attributes	Description
alignment	<p>Sets the horizontal justification of the text based on a defined width.</p> <p>Type: String Default Value: left Values: left center right</p>
color	<p>Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below.</p> <p>Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow</p>
excel-wrap-text	<p>When outputting to Excel, sets the text to word wrap.</p> <p>Type: Boolean Default Value: true</p>
fixedposition	<p>Defines the absolute position on the page where the band should be printed. If the current cursor position is before that y-position, then empty space is added. If the current position is after that y-position, the band is printed on the next page (on that y-position).</p>
font-embedded	<p>Specifies the font to embed into the PDF file.</p> <p>Type: Boolean Default Value: false</p>
font-encoding	<p>Specifies the encoding to be used for the PDF.</p> <p>Type: String</p>
fontname	<p>Sets the font name.</p> <p>Type: String</p>
fontsize	<p>Sets the font size.</p> <p>Type: Integer</p>
fontstyle	<p>Sets font style. This is a shortcut for defining fsbold and fsitalic.</p> <p>Type: String Values: plain bold italic bolditalic</p>
fsbold	<p>Sets font to italics.</p> <p>Type: Boolean Default Value: false</p>
fsitalic	<p>Sets font to italics.</p> <p>Type: Boolean Default Value: false</p>

fsstrikethr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number and each increment is equivalent to 1/72 of an inch. Objects don't inherit this setting. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
pagebreak-before-print	If set to true, page break will occur before the group header is printed. Type: Boolean Default Value: false
pagebreak-after-print	If set to true, page break will occur after the group header is printed. Type: Boolean Default Value: false
repeat	Repeats the group header at the top of each new page if it spans over to another page. Repeated group headers are printed directly below the page header. Type: Boolean Default Value: false
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the report header will printed. Type: Boolean Default Value: true

<groupfooter>

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
fixed-position	Defines the absolute position on the page where the band should be printed. If the current cursor position is before that y-position, then empty space is added. If the current position is after that y-position, the band is printed on the next page (on that y-position).
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikethr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number and each increment is equivalent to 1/72 of an inch. Objects don't inherit this setting.

	Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
pagebreak-after-print	If set to true, page break will occur before the group footer is printed. Type: Boolean Default Value: false
pagebreak-before-print	If set to true, page break will occur after the group footer is printed. Default is false. Type: Boolean Default Value: false
repeat	This will repeat the group footer if it spans over pages.
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the report header will printed. Type: Boolean Default Value: true

Items

This section defines content that will be printed for each data row.

`<items></items>`

Report Section	XML Syntax	Description
Details/Items	<code><items></code> <code></items></code>	Prints data rows. If no details/items band is defined, only printing is disabled. All calculations are performed regardless of the appearance of the items. Accepts the same attributes as all other bands. In the default case, the number of itembands printed corresponds to the number of rows in the tablemodel. If the tablemodel is empty, no itembands will be generated (but all

		other header and footer will be printed).
--	--	---

<items>

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care. Type: Boolean Default Value: false
fixed-position	Defines the absolute position on the page where the band should be printed. If the current cursor position is before that y-position, then empty space is added. If the current position is after that y-position, the band is printed on the next page (on that y-position).
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikethr	Sets font to strikethrough. Type: Boolean

	Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
pagebreak-after-print	Page break after printing details section. Type: Boolean Default Value: false
pagebreak-before-print	Page break before printing details section. Type: Boolean Default Value: false
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed. Type: Boolean Default Value: true

Types of Report Objects

Report Objects	Description
anchor-field	Marks a specific point in an document as either the source or destination of a hypertext link. This allows you to create links from one hypertext document to another, as well as to different sections within the same document.
band	Defines a sub-band within an other band i.e. group header, report footer, etc. This can be used to group objects and to trigger the visibility or change the style of the whole group using a single operation. Bands can have their own layout manager

	attached, which is also the only way to stack dynamic elements below each other in a safe way.
date-field	Returns the date from a specified field based on the current record.
drawable-field	Returns a drawable object from a specified field on the current record.
drawable-url-field	Returns a drawable object's URL from a specified field on the current record.
drawableref	Returns a drawable object's loaded from the specified URL. If the URL is a relative path, the URL will be resolved using the report definition's location. The core library allows loading WMF (Windows Meta Format) files using this field. Support for SVG images requires JfreeReport-Ext.
image-field	Returns a image object from a specified field on the current record.
imageref	Returns a image reference from a specified field on the current record.
imageurl-field	Returns an image loaded from an URL found in the specified field on the current record.
label	Writes static, immutable text.
line	Draws a line.
message-field	Writes multiple data types (text, string field, date field, and numeric field) into one object.
number-field	Returns the numeric value from a specified field on the current record.
rectangle	Draw a rectangle.
resource-field	Used for localizing reports.
resource-label	Used for localizing reports.
resource-message	Used for localizing reports.
shape-field	Returns a shape value from a specified field on the current record.
string-field	Returns a string value from a specified field on the current record.

Report Object Attributes

anchor

Attributes	Description
!fieldname	Specify fieldname. The field must either contain URLs or Strings.
name	Used to reference the element later. Functions pick up elements by their name. Type: String
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

band

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean

	Default Value: false
font-embedded	Specifies the font to embed into the PDF file. Type: String
font-encoding	Specifies the encoding to be used for the PDF. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. Type: String
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikerthr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true

vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed. Type: Boolean Default Value: true
width	Set the horizontal width of the object.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

date-field

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.
excel-format	When outputting to Excel, use another date/number format. The format is excel-specific and defined in the Excel-Help. If it is invalid, the resulting excel file might be unreadable. . If no format is specified here, the Java format string is used (as defined by the 'format' attribute). Type: String
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. The field must contain java.util.Date objects.
format	Defines the formatting of the date field based on the Date Formatting Table. mmm dd, yyyy = Dec. 25, 2005 mm/dd/yy = 12/25/05 mm/dd/yy h:m = 12/25/05 8:00 am
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF. Type: String
fontname	Sets the font name. Type: String

fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikethr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer The height is required. If the height is smaller than the font size, no content is printed.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
nullstring	If the value read from the given field is null or is no instance of 'java.util.Date', replace with a predefined value.
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed. Type: Boolean Default Value: true

width	Set the horizontal width of the object. This attribute is required. If no width is given, nothing gets printed.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

Date Formatting Table

Symbol	Meaning	Type	Example
G	Era	Text	"GG" -> "AD"
Y	Year	Number	"yy" -> "03" "yyyy" -> "2003"
M	Month	Text or Number	"M" -> "7" "M" -> "12" "MM" -> "07" "MMM" -> "Jul" "MMMM" -> "December"
D	Day in month	Number	"d" -> "3" "dd" -> "03"
H	Hour (1-12, AM/PM)	Number	"h" -> "3" "hh" -> "03"
H	Hour (0-23)	Number	"H" -> "15" "HH" -> "15"
K	Hour (1-24)	Number	"k" -> "3" "kk" -> "03"
K	Hour (0-11 AM/PM)	Number	"K" -> "15" "KK" -> "15"
M	Minute	Number	"m" -> "7" "m" -> "15" "mm" -> "15"
S	Second	Number	"s" -> "15" "ss" -> "15"
S	Millisecond (0-999)	Number	"SSS" -> "007"
E	Day in week	Text	"EEE" -> "Tue" "EEEE" -> "Tuesday"
D	Day in year (1-365 or 1-364)	Number	"D" -> "65" "DDD" -> "065"
F	Day of week in month (1-5)	Number	"F" -> "1"
W	Week in year (1-53)	Number	"w" -> "7"
W	Week in month (1-5)	Number	"W" -> "3"
A	AM/PM	Text	"a" -> "AM" "aa" -> "AM"
Z	Time zone	Text	"z" -> "EST" "zzz" -> "EST" "zzzz" -> "Eastern Standard Time"
'	Escape for text	Delimiter	"'hour' h" -> "hour 9"
"	Single quote	Literal	"ss"SSS" -> "45'876"

drawable-field

Attributes	Description
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. The field must contain org.jfree.ui.Drawable objects.
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer The height is required.
href	Sets a hyperlink for the section.

	Type: String
name	Used to reference the element later. Functions pick up elements by their name. Type: String
width	Set the horizontal width of the object. The width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

drawable-url-field

Attributes	Description
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. The field must contain URLs or Strings.
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer Default: The height is required.
href	Sets a hyperlink for the section. Type: String
name	Used to reference the element later. Functions pick up elements by their name. Type: String
width	Set the horizontal width of the object. The width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

drawableref

Attributes	Description
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. The field must contain org.jfree.ui.Drawable objects.
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer Default: The height is required.
href	Sets a hyperlink for the section. Type: String
name	Used to reference the element later. Functions pick up elements by their name. Type: String
width	Set the horizontal width of the object. The width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

image-field, image-ref, imageurl

Attributes	Description
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. The field must contain java.awt.Image objects or org.jfree.report.ImageContainer objects.
height	Determines the minimum height. Value is a whole number where each increment is

	equivalent to 1/72 of an inch or percentage. Type: Integer The height is required.
href	Sets a hyperlink for the section. Type: String
keepAspectRatio	Defines, whether the image's original aspect ratio should be preserved during scaling. This has no effect if no scaling is used. Type: Boolean Default Value: false
name	Used to reference the element later. Functions pick up elements by their name. Type: String
scale	Shrinks or expands the image to completely fill the element's defined bounds. If scaling is disabled and the unscaled image does not fit into the element's boundaries, the image is clipped at the element bounds. Type: Boolean Default Value: false
visible	Sets whether the object will be printed. Type: Boolean Default Value: true
width	Set the horizontal width of the object. The width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
Y	Defines the vertical starting position of an object defined by a value or percentage.

label

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care. For labels, using the dynamic width should seldom be necessary and can almost always be avoided by using carefully selected height values.
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF.

	Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikrthr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer The height is required.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed.

	Type: Boolean Default Value: true
width	Set the horizontal width of the object. The width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

line

The reporting engine will construct the bounds for the line shape using the x1, y1 coordinates and the given width or height. If no width or height is given, the missing values will be computed from the difference between x2 and x1 and y2 to y1.

The values x1, y1 and x2, y2 define the start and end-point of the line within the element. This is used to specify the direction of the line.

Line shapes will always be scaled, without preserving the aspect ratio.

Attributes	Description
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer Default: If no height is specified, the height will be derived from the difference between y1 and y2.
name	Used to reference the element later. Functions pick up elements by their name. Type: String
stroke-style	Defines how the lines of the shape get drawn. Type: String Default Value: solid Predefined Values: solid dashed dotted dot-dash dot-dot-dash
weight	The weight defines the width of the pen used to draw the line. Type: Integer Default: 1
width	Set the horizontal width of the object. Default: If no width is specified, the width will be derived from the difference between x1 and x2.
x1	Defines the horizontal starting position of the line defined by a value or percentage.
x2	Defines the horizontal ending position of the line defined by a value or percentage.
y1	Defines the vertical starting position of the line defined by a value or percentage.
y2	Defines the vertical ending position of the line defined by a value or percentage.

message-field

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right

color	<p>Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below.</p> <p>Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow</p>
dynamic	<p>A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.</p>
excel-wrap-text	<p>When outputting to Excel, sets the text to word wrap.</p> <p>Type: Boolean Default Value: true</p>
font-embedded	<p>Specifies the font to embed into the PDF file.</p> <p>Type: Boolean Default Value: false</p>
font-encoding	<p>Specifies the encoding to be used for the PDF.</p> <p>Type: String</p>
fontname	<p>Sets the font name.</p> <p>Type: String</p>
fontsize	<p>Sets the font size.</p> <p>Type: Integer</p>
fontstyle	<p>Sets font style. This is a shortcut for defining fsbold and fsitalic.</p> <p>Type: String Values: plain bold italic bolditalic</p>
fsbold	<p>Sets font to italics.</p> <p>Type: Boolean Default Value: false</p>
fsitalic	<p>Sets font to italics.</p> <p>Type: Boolean Default Value: false</p>
fsstrikethr	<p>Sets font to strikethrough.</p> <p>Type: Boolean Default Value: false</p>
fsunderline	<p>Sets font to underline.</p> <p>Type: Boolean Default Value: false</p>
height	<p>Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage.</p> <p>Type: Integer The height is required.</p>
href	<p>Sets a hyperlink for the section.</p> <p>Type: String</p>
line-height	<p>The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable.</p>

	Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
nullstring	If a null value, replace with a predefined value.
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed. Type: Boolean Default Value: true
width	Set the horizontal width of the object. Width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

Number Formatting

Symbol	Location	Localized?	Meaning
0	Number	Yes	Digit
#	Number	Yes	Digit, zero shows as absent
.	Number	Yes	Decimal separator or monetary decimal separator
-	Number	Yes	Minus sign
,	Number	Yes	Grouping separator
E	Number	Yes	Separates mantissa and exponent in scientific notation. <i>Need not be quoted in prefix or suffix.</i>
;	Sub-pattern boundary	Yes	Separates positive and negative sub-patterns
%	Prefix or suffix	Yes	Multiply by 100 and show as percentage
\u2030	Prefix or suffix	Yes	Multiply by 1000 and show as per mille
¤ (\u00A4)	Prefix or suffix	No	Currency sign, replaced by currency symbol. If doubled, replaced by international currency symbol. If present in a pattern, the monetary decimal separator is used instead of the decimal separator.
'	Prefix or suffix	No	Used to quote special characters in a prefix or suffix, for example, "'###'" formats 123 to "#123". To create a single quote itself, use two in a row: "# o' 'clock".

Date Formatting

Symbol	Meaning	Type	Example
G	Era	Text	"GG" -> "AD"
y	Year	Number	"yy" -> "03" "yyyy" -> "2003"
M	Month	Text or Number	"M" -> "7" "M" -> "12" "MM" -> "07"

			"MMM" -> "Jul" "MMMM" -> "December"
d	Day in month	Number	"d" -> "3" "dd" -> "03"
h	Hour (1-12, AM/PM)	Number	"h" -> "3" "hh" -> "03"
H	Hour (0-23)	Number	"H" -> "15" "HH" -> "15"
k	Hour (1-24)	Number	"k" -> "3" "kk" -> "03"
K	Hour (0-11 AM/PM)	Number	"K" -> "15" "KK" -> "15"
m	Minute	Number	"m" -> "7" "m" -> "15" "mm" -> "15"
s	Second	Number	"s" -> "15" "ss" -> "15"
S	Millisecond (0-999)	Number	"SSS" -> "007"
E	Day in week	Text	"EEE" -> "Tue" "EEEE" -> "Tuesday"
D	Day in year (1-365 or 1-364)	Number	"D" -> "65" "DDD" -> "065"
F	Day of week in month (1-5)	Number	"F" -> "1"
W	Week in year (1-53)	Number	"w" -> "7"
W	Week in month (1-5)	Number	"W" -> "3"
A	AM/PM	Text	"a" -> "AM" "aa" -> "AM"
Z	Time zone	Text	"z" -> "EST" "zzz" -> "EST" "zzzz" -> "Eastern Standard Time"
'	Escape for text	Delimiter	""hour' h" -> "hour 9"
"	Single quote	Literal	""ss"SSS" -> "45'876"

number-field

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.
excel-format	When outputting to Excel, use another date/number format. The format is excel-specific and defined in the Excel-Help. If it is invalid, the resulting excel file might be unreadable.
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the

	function/expression/report property from where to read the data. The field must contain instances of <code>java.lang.Number</code> .
format	Defines the formatting of the date field based on the Date Formatting Table. 123456789.1234 => \$ ###,###,###.00 = \$ 123,456,789.12 123456789.1234 => ###,###,### = 123,456,789 123456 => 00##### = 00123456
font-embedded	Specifies the font to embed into the PDF file. Type: String
font-encoding	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikerthr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer The height is required.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
nullstring	If a null value or a non-number value is encountered, replace with a predefined

	value.
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed. Type: Boolean Default Value: true
width	Set the horizontal width of the object.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

Number Formatting

Symbol	Location	Localized?	Meaning
0	Number	Yes	Digit
#	Number	Yes	Digit, zero shows as absent
.	Number	Yes	Decimal separator or monetary decimal separator
-	Number	Yes	Minus sign
,	Number	Yes	Grouping separator
E	Number	Yes	Separates mantissa and exponent in scientific notation. <i>Need not be quoted in prefix or suffix.</i>
;	Sub-pattern boundary	Yes	Separates positive and negative sub-patterns
%	Prefix or suffix	Yes	Multiply by 100 and show as percentage
\u2030	Prefix or suffix	Yes	Multiply by 1000 and show as per mille
¤ (\u00A4)	Prefix or suffix	No	Currency sign, replaced by currency symbol. If doubled, replaced by international currency symbol. If present in a pattern, the monetary decimal separator is used instead of the decimal separator.
'	Prefix or suffix	No	Used to quote special characters in a prefix or suffix, for example, "'#'" formats 123 to "'#123'". To create a single quote itself, use two in a row: "' o' 'clock'".

rectangle

Attributes	Description
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
draw	Sets the outline of the object to be visible. The outline is drawn using the color and stroke defined for the element.

	Type: Boolean Default: false
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care. Type: Boolean Default: false
fill	Fills the rectangle with the color defined by the color attribute. Type: Boolean Default: true
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
name	Used to reference the element later. Functions pick up elements by their name. Type: String
reserved-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
stroke-style	Defines how the lines of the shape get drawn. Type: String Default Value: solid Predefined Values: solid dashed dotted dot-dash dot-dot-dash
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
weight	Defines the width of the pen used to draw the outline of the rectangle. Type: Boolean Default Value: true
width	Set the horizontal width of the object. Width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

resource-field

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or

	<p>from list of the predefined colors below.</p> <p>Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow</p>
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.
excel-wrap-text	<p>When outputting to Excel, sets the text to word wrap.</p> <p>Type: Boolean Default Value: true</p>
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. The field should contain a string. That string must be a recognized key value of the used ResourceBundle, or nothing will be printed.
font-embedded	<p>Specifies the font to embed into the PDF file.</p> <p>Type: Boolean Default Value: false</p>
font-encoding	<p>Specifies the encoding to be used for the PDF.</p> <p>Type: String</p>
fontname	<p>Sets the font name.</p> <p>Type: String</p>
fontsize	<p>Sets the font size.</p> <p>Type: Integer</p>
fontstyle	<p>Sets font style. This is a shortcut for defining fsbold and fsitalic.</p> <p>Type: String Values: plain bold italic bolditalic</p>
fsbold	<p>Sets font to italics.</p> <p>Type: Boolean Default Value: false</p>
fsitalic	<p>Sets font to italics.</p> <p>Type: Boolean Default Value: false</p>
fsstrikethr	<p>Sets font to strikethrough.</p> <p>Type: Boolean Default Value: false</p>
fsunderline	<p>Sets font to underline.</p> <p>Type: Boolean Default Value: false</p>
height	<p>Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage.</p> <p>Type: Integer Default: Height is required.</p>
href	<p>Sets a hyperlink for the section.</p> <p>Type: String</p>
line-height	The height of a single text line in text elements. If the line-height is set and is greater

	<p>than the font size, extra padding will be added between the lines making text more readable.</p> <p>Type: Integer</p>
name	<p>Used to reference the element later. Functions pick up elements by their name.</p> <p>Type: String</p>
nullstring	<p>If a null value is encountered or the value read from the field could not be resolved using the defined resource bundle, replace with a predefined value.</p>
reserve-literal	<p>Determines the text printed when the text does not fit completely into an element.</p> <p>Type: String Default value: "..."</p>
trim-text-content	<p>Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout.</p> <p>Type: Boolean Default Value: true</p>
vertical-alignment	<p>Sets the vertical position of the text.</p> <p>Default Value: top Values: top middle bottom</p>
visible	<p>Sets whether the object will be printed.</p> <p>Type: Boolean Default Value: true</p>
width	<p>Set the horizontal width of the object.</p>
x	<p>Defines the horizontal starting position an object defined by a value or percentage.</p>
y	<p>Defines the vertical starting position of an object defined by a value or percentage.</p>

resource-label

Attributes	Description
alignment	<p>Sets the horizontal justification of the text based on a defined width.</p> <p>Type: String Default Value: left Values: left center right</p>
color	<p>Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below.</p> <p>Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow</p>
dynamic	<p>A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.</p>
excel-wrap-text	<p>When outputting to Excel, sets the text to word wrap.</p> <p>Type: Boolean Default Value: true</p>
font-embedded	<p>Specifies the font to embed into the PDF file.</p> <p>Type: Boolean Default Value: false</p>
font-encoding	<p>Specifies the encoding to be used for the PDF.</p> <p>Type: String</p>
fontname	<p>Sets the font name.</p>

	Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikerthr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Type: Integer Default: Height is required.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed. Type: Boolean Default Value: true

width	Set the horizontal width of the object. Width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

shape-field

Attributes	Description
color	<p>Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below.</p> <p>Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow</p>
draw	<p>Defines, whether the outline of the shape found in the defined field should be drawn.</p> <p>Type: Boolean Default Value: false</p>
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. The field must contain java.awt.Shape objects.
fill	<p>Defines, whether the shape should be filled.</p> <p>Type: Boolean Default Value: false</p>
height	<p>Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage.</p> <p>Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.</p>
href	<p>Sets a hyperlink for the section.</p> <p>Type: String</p>
name	<p>Used to reference the element later. Functions pick up elements by their name.</p> <p>Type: String</p>
stroke-style	<p>Defines how the lines of the shape get drawn.</p> <p>Type: String Default Value: solid Predefined Values: solid dashed dotted dot-dash dot-dot-dash</p>
visible	<p>Sets whether the object will be printed.</p> <p>Default Value: top Values: top middle bottom</p>
weight	<p>Sets whether the object will be printed.</p> <p>Type: Boolean Default Value: true</p>
width	Set the horizontal width of the object.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

string-field

Attributes	Description
alignment	Sets the horizontal justification of the text based on a defined width. Type: String Default Value: left Values: left center right
color	Sets the font color by the using RGB-triplet (#ffffff for white; #ff0000 for red, etc) or from list of the predefined colors below. Type: String Default Value: black Predefined Values: black blue cyan darkGray gray green lightGray magenta orange pink red white yellow
dynamic	A dynamic element increases the height of the element to fit the content of that element. Computing that dynamic height is expensive, so use it with care.
excel-wrap-text	When outputting to Excel, sets the text to word wrap. Type: Boolean Default Value: true
fieldname	The fieldname defines the name of the column of the tablemodel or the name of the function/expression/report property from where to read the data. Any value found in the column will be converted into a String using String.valueOf (...).
font-embedded	Specifies the font to embed into the PDF file. Type: Boolean Default Value: false
font-encoding	Specifies the encoding to be used for the PDF. Type: String
fontname	Sets the font name. Type: String
fontsize	Sets the font size. Type: Integer
fontstyle	Sets font style. This is a shortcut for defining fsbold and fsitalic. Type: String Values: plain bold italic bolditalic
fsbold	Sets font to italics. Type: Boolean Default Value: false
fsitalic	Sets font to italics. Type: Boolean Default Value: false
fsstrikethr	Sets font to strikethrough. Type: Boolean Default Value: false
fsunderline	Sets font to underline. Type: Boolean Default Value: false
height	Determines the minimum height. Value is a whole number where each increment is equivalent to 1/72 of an inch or percentage. Height is required.

	Type: Integer Default: If no height is specified, then height will be determine by all the objects contained.
href	Sets a hyperlink for the section. Type: String
line-height	The height of a single text line in text elements. If the line-height is set and is greater than the font size, extra padding will be added between the lines making text more readable. Type: Integer
name	Used to reference the element later. Functions pick up elements by their name. Type: String
nullstring	If a null value, replace with a predefined value.
reserve-literal	Determines the text printed when the text does not fit completely into an element. Type: String Default value: "..."
trim-text-content	Defines, whether leading and trailing whitespaces of the generated lines get removed. Important for a clean layout. Type: Boolean Default Value: true
vertical-alignment	Sets the vertical position of the text. Default Value: top Values: top middle bottom
visible	Sets whether the object will be printed. Type: Boolean Default Value: true
width	Set the horizontal width of the object. Width is required.
x	Defines the horizontal starting position an object defined by a value or percentage.
y	Defines the vertical starting position of an object defined by a value or percentage.

Functions and Expressions

Functions have access to the data row and can access other functions or expressions or the data source. Functions are stateful meaning they maintain their state during the report generation. For stateless user-defined computations, consider using an expression instead of functions, as expression are cheaper to compute and maintain when using huge reports. Function parameters are given by property elements.

The `delevel` attribute can be used to prioritize the functions. Functions with an higher dependency level are executed before any function with lower dependency levels. Dependency levels lower than 0 are not allowed.

An expression is a stateless user-defined function. It can access the data row and the report properties (property-ref) to perform its task. Using the data row, an expression has access to the data source and other functions and expressions. Expression parameters are given by property elements.

Summary (Grand and Group) based Calculations

Function Class	Description
CountDistinctFunction	Counts the distinct occurrences of an certain value of an column. This functionality is similar to the SQL <code>distinct()</code>

	function.
GroupCountFunction	A report function that counts groups in a report. If a null-group name is given, all groups are counted.
ItemAvgFunction	Calculates the average of fields (columns) from the current row.
ItemCountFunction	Counts items in a report
ItemHideFunction	Hides equal values in a group where only the first changed value is printed.
ItemMaxFunction	Calculates the maximum value of one field (column).
ItemMinFunction	Calculates the minimum value of one field (column).
ItemPercentageFunction	Calculates the percentage value of a numeric field. The total sum is taken and divided by the number of items counted.
ItemSumFunction	A report function that calculates the sum of one field (column).
TotalCalculationFunction	Stores the result of a calculation for a group or the complete report. The field value, that was read when the group finished, is stored and returned when the group gets active again in a higher processing level. This function can be used to calculate total values for a group, TotalMax for the group, which returns the maximum value encountered for that field in the current group. The computed values are available to all bands of the group.
TotalGroupCountFunction	Counts the total of groups in a report. If a null group name is given, all groups are counted.
TotalGroupSumFunction	Calculates the sum of one field (column) . This function produces a global total.
TotalGroupSumQuotientFunction	A report function that calculates the quotient of two summed fields (columns) . This function produces a global total.
TotalGroupSumQuotientPercentFunction	A report function that calculates the quotient percent of two summed fields (columns). This function produces a global total.
TotalItemCountFunction	A report function that counts the total number of items contained in groups in a report. If a null group name is given, all groups are counted.

CountDistinctFunction

Counts the distinct occurrences of an certain value of an column. This functionality is similar to the SQL distinct() function.

Example

```
<function class="org.jfree.report.function.CountDistinctFunction" name="Cnt_Dstnct_USA">
  <properties>
    <property name="field">Country</property>
    <property name="group">Product</property>
    <property name="value">USA</property>
  </properties>
```

GroupCountFunction

A report function that counts groups in a report. If a null-group name is given, all groups are counted.

The group to be counted is defined using the property "group". An optional container group can be defined using the property "parent-group". When the group start event of that group is encountered, the counter will be reset to '0'. If the group property is not set, all group starts get counted.

Example

```
<function class="org.jfree.report.function.GroupCountFunction" deplevel="" name="Orders Count">
  <properties>
```

```
<property name="group">ORDER NUMBER GROUP</property>
<property name="parentGroup">CUSTOMER GROUP</property>
</properties>
</function>
```

ItemAvgFunction

A report function that calculates the average of one field (column) from the TableModel. This function produces a running total, no global total. The function can be used in two ways:

- to calculate an average value for the entire report
- to calculate an average value within a particular group

This function expects its input values to be either java.lang.Number instances or Strings that can be parsed to java.lang.Number instances using a java.text.DecimalFormat.

The function understands two parameters:

- field parameter is required and denotes the name of an ItemBand-field which gets summed up.
- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

```
<function class="org.jfree.report.function.ItemAvgFunction" name="Actual_Avg">
  <properties>
    <property name="field">ACTUAL</property>
    <property name="group">RegionGroup</property>
  </properties>
</function>
```

ItemColumnQuotientExpression

A report function that calculates the quotient of two fields (columns) from the current row.

This function expects its input values to be java.lang.Number instances.

The function understands two parameters.

- dividend parameter is required and denotes the name of an Item Section field which is used as dividend.
- divisor parameter is required and denotes the name of an Item Section field which is used as divisor.

```
<expression name="saved" class="org.jfree.report.function.ItemColumnQuotientExpression">
  <properties>
    <property name="dividend">article.price</property>
    <property name="divisor">article.reducedPrice</property>
  </properties>
</expression>
```

ItemCountFunction

A report function that counts items in a report. If the "group" property is set, the item count is reset to zero whenever the group changes.

```
<function class="org.jfree.report.function.ItemCountFunction" name="Product_Count">
  <properties>
    <property name="field">Product</property>
    <property name="group">RegionGroup</property>
  </properties>
</function>
```

ItemHideFunction

The ItemHideFunction hides equal values in a group. Only the first changed value is printed. This function uses the property tag to define the name of the object in the Item Section that should be made visible or invisible by this function. The property tag, field, defines the field in the datasource or the expression which should be used to determine the visibility.

```
<function class="org.jfree.report.function.ItemHideFunction" name="HideCategory">
  <properties>
```

```
<property name="field">Category</property>
</properties>
</function>
```

ItemMaxFunction

A report function that calculates the maximum value of one field (column) from the TableModel.

The function can be used in two ways:

- to calculate a maximum value for the entire report
- to calculate a maximum value within a particular group

This function expects its input values to be either java.lang.Number instances or Strings that can be parsed to java.lang.Number instances using java.text.DecimalFormat.

The function understands two parameters:

- field parameter is required and denotes the name of an ItemBand-field which gets summed up.
- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

```
<function class="org.jfree.report.function.ItemMaxFunction" name="Actual_Sales">
  <properties>
    <property name="field">ACTUAL</property>
    <property name="group">RegionGroup</property>
  </properties>
</function>
```

ItemMinFunction

A report function that calculates the minimum value of one field (column) from the TableModel. The function can be used in two ways:

- to calculate a minimum value for the entire report
- to calculate a minimum value within a particular group

This function expects its input values to be either java.lang.Number instances or Strings that can be parsed to java.lang.Number instances using java.text.DecimalFormat.

The function understands two parameters:

- field parameter is required and denotes the name of an ItemBand-field which gets summed up.
- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

```
<function class="org.jfree.report.function.ItemMinFunction" name="Actual_Sales">
  <properties>
    <property name="field">ACTUAL</property>
    <property name="group">RegionGroup</property>
  </properties>
</function>
```

ItemPercentageFunction

Calculates the percentage value of a numeric field. The total sum is taken and divided by the number of items counted.

```
<function class="org.jfree.report.function.ItemPercentageFunction" name="Item_Percentage">
  <properties>
    <property name="field">ACTUAL</property>
  </properties>
</function>
```

ItemSumFunction

A report function that calculates the sum of one field (column) from the TableModel. This function produces a running total, no global total. For a global sum, use the TotalGroupSumFunction function. The function can be used in two ways:

- to calculate a sum for the entire report

- to calculate a sum within a particular group

This function expects its input values to be either java.lang.Number instances or Strings that can be parsed to java.lang.Number instances using java.text.DecimalFormat.

The function understands two parameters:

- field parameter is required and denotes the name of an ItemBand-field which gets summed up.
- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

```
<expression class="org.jfree.report.function.ItemSumFunction" name="RegionActual">
  <properties>
    <property name="field">ACTUAL</property>
    <property name="group">RegionGroup</property>
  </properties>
```

TotalCalculationFunction

A report function that stores the result of a calculation for a group or the complete report. The field value, that was read when the group finished, is stored and returned when the group gets active again in a higher processing level.

This function can be used to calculate total values for a group, e.g. a TotalMax for the group, which returns the maximum value encountered for that field in the current group. The computed values are available to all bands of the group.

The function understands two parameters:

- field parameter is required and denotes the name of an ItemBand-field which gets summed up.
- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

TotalGroupCountFunction

A report function that counts the total of groups in a report. If a null group name is given, all groups are counted.

A group can be defined using the property "group". If the group property is not set, all group starts get counted.

```
<function class="org.jfree.report.function.TotalGroupCountFunction" deplevel="" name="Orders Count">
  <properties>
    <property name="group">ORDER NUMBER GROUP</property>
  </properties>
</function>
```

TotalGroupSumFunction

A report function that calculates the sum of one field (column) from the TableModel.

This function produces a global total. The total sum of the group is known when the group processing starts and the report is not performing a prepare-run. The sum is calculated in the prepare run and recalled in the printing run.

The function can be used in two ways:

- to calculate a sum for the entire report
- to calculate a sum within a particular group

This function expects its input values to be either java.lang.Number instances or Strings that can be parsed to java.lang.Number instances using java.text.DecimalFormat.

The function understands two parameters:

- field parameter is required and denotes the name of an ItemBand-field which gets summed up.

- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

```
<function class="org.jfree.report.function.TotalGroupSumFunction" deplevel="" name="Orders Count">
  <properties>
    <property name="group">ORDER NUMBER GROUP</property>
    <property name="field">CUSTOMER</property>
  </properties>
</function>
```

TotalGroupSumQuotientFunction

A report function that calculates the quotient of two summed fields (columns) from the TableModel. This function produces a global total. The total sum of the group is known when the group processing starts and the report is not performing a prepare-run. The sum is calculated in the prepare run and recalled in the printing run.

The function can be used in two ways:

- to calculate a quotient for the entire report
- to calculate a quotient within a particular group

This function expects its input values to be either java.lang.Number instances or Strings that can be parsed to java.lang.Number instances using java.text.DecimalFormat.

The function understands three parameters.

- dividend parameter is required and denotes the name of an Item Section field which is used as dividend.
- divisor parameter is required and denotes the name of an Item Section field which is used as divisor.
- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

```
<function class="org.jfree.report.function.TotakGroupSumQuotientFunction" deplevel=""
name="Year_Over_Ratio">
  <properties>
    <property name="group">REGION GROUP</property>
    <property name="dividend">Y2000</property>
    <property name="divisor">Y2001</property>
  </properties>
</function>
```

TotalGroupSumQuotientPercentFunction

A report function that calculates the quotient of two summed fields (columns) from the TableModel. This function produces a global total. The total sum of the group is known when the group processing starts and the report is not performing a prepare-run. The sum is calculated in the prepare run and recalled in the printing run.

The function can be used in two ways:

- to calculate a quotient for the entire report
- to calculate a quotient within a particular group

This function expects its input values to be either java.lang.Number instances or Strings that can be parsed to java.lang.Number instances using java.text.DecimalFormat.

The function understands three parameters.

- dividend parameter is required and denotes the name of an Item Section field which is used as dividend.
- divisor parameter is required and denotes the name of an Item Section field which is used as divisor.
- group parameter denotes the name of a group. When this group is started, the counter gets reset to null.

```
<function class="org.jfree.report.function.TotakGroupSumQuotientPercentFunction" deplevel=""
name="Year_Over_Percentage">
  <properties>
    <property name="group">REGION GROUP</property>
    <property name="dividend">Y2000</property>
    <property name="divisor">Y2001</property>
  </properties>
</function>
```

TotalItemCountFunction

A report function that counts the total number of items contained in groups in a report. If a 'null' group name is given, all groups are counted.

A group can be defined using the property 'group'. If it is not set, all group starts get counted.

```
<function class="org.jfree.report.function.TotalItemCountFunction" deplevel="" name="Total Items
Counted">
  <properties>
    <property name="group">REGION GROUP</property>
  </properties>
</function>
```

Item Calculations

Function Class	Description
ColumnAverageExpression	An expression that takes values from one or more fields and returns the average of them.
ConvertToNumberExpression	Converts string to a numeric type.
BSHExpression	Used to calculate values within a single row of a report. Expressions can use a dataRow to access other fields, expressions or functions within the current row in the report.
ItemColumnQuotientExpression	Calculates the quotient of two fields (columns) from the current row.
PercentageExpression	Computes the percentage or percent difference between two columns.
TimeDiffAndFormatFunction	Computes the Time between two date fields
ColumnAverageExpression	An expression that takes values from one or more fields and returns the average of them.
ColumnDifferenceExpression	An expression that takes values from two or more fields and returns the difference of them.
ColumnDivisionExpression	An expression that takes values from two or more fields and divides all values.
ColumnMaximumExpression	An expression that takes values from two or more fields and returns the the greatest value of them

ColumnMinimumExpression	An expression that takes values from two or more fields and returns the the smallest value of them
ColumnMultiplyExpression	An expression that takes values from two or more fields and and multiplies all values
ColumnSumExpression	An expression that takes values from two or more fields and returns the the sum of all values
ConvertToDateExpression	Converts string to a date type
DateExpression	Constructs a date using the specified parameters.
VariableDateExpression	Constructs a date using the specified parameters and using values read from various fields.
DateSpanExpression	Computes the difference between two dates.
CapitalizeStringExpression	Capitalizes the first letter of all words found in the string.
ToLowerCaseStringExpression	Converts the string to all lower case letters
ToUpperCaseStringExpression	Converts the string to all upper case letters
TokenizeStringExpression	Replaces all occurrences of a certain substring with an other string.
SubStringExpression	Returns a partial string.
MapStringExpression	Replaces various predefined values with other constant values.
MapIndirectExpression	Replaces various predefined values with values read from other fields.
ResourceBundleLookupExpression	Returns a translated value from a resource bundle.
ResourceMessageExpression	Formats a string using a MessageFormat string read from an ResourceBundle. This is the expression-counterpart of the resource-message field.

ColumnAverageExpression

An expression that takes values from one or more fields and returns the average of them.

Example

```
<expression name="Average_Column" class="org.jfree.report.function.AverageExpression">
  <properties>
    <property name="field[0]">Jan_Column</property>
    <property name="field[1]">Feb_Column</property>
    <property name="field[2]">Mar_Column</property>
  </properties>
</expression>
```

ConvertToNumberExpression

Converts string to a numeric type.

Example

```
<expression name="C_Txt_t_Nmbr" class="org.jfree.report.function.ConvertToNumberExpression">
  <properties>
    <property name="field[0]">Invoice</property>
  </properties>
</expression>
```

BSHExpression

An expression is a lightweight function that does not maintain a state. Expressions are used to calculate values within a single row of a report. Expressions can use a `dataRow` to access other fields, expressions or functions within the current row in the report.

Examples

```
<expression class="org.jfree.report.modules.misc.beanshell.BSHExpression" name="totalPrice">
  <properties>
    <property name="expression">Object getValue()
      {
        if (dataRow == null)
        {
          return null;
        }
        Number OrderCount = (Number) dataRow.get("&quot;QUANTITYORDERED&quot;");
        Number SalePrice = (Number) dataRow.get("&quot;PRICEEACH&quot;");
        if (OrderCount == null || SalePrice == null) { return null;}
        float totalPrice = OrderCount.floatValue() * SalePrice.floatValue();
        return new Float (totalPrice);
      }
    </property>
  </properties>
</expression>

<expression class="org.jfree.report.modules.misc.beanshell.BSHExpression" name="isVarianceNegative">
  <properties>
    <property name="expression">Object getValue()
      {
        Object value = dataRow.get("&quot;VARIANCE&quot;");
        if (value instanceof Number == false)
        {
          return Boolean.FALSE;
        }
        Number number = (Number) value;
        if (number.doubleValue() &lt; 0)
        {
          return Boolean.TRUE;
        }
        return Boolean.FALSE;
      }
    </property>
  </properties>
</expression>
```

PercentageExpression

Computes the percentage for a column in relation to a base column.

The function understands two parameters.

- dividend parameter is required and denotes the name of an Item Section field which is used as dividend.
- divisor parameter is required and denotes the name of an Item Section field which is used as divisor.

If either the dividend or the divisor are not numeric, the expression will return null.

The formula used is as follows, Percent := dividend / divisor.

If the flag 'useDifference' is set, the difference between base and subject is used instead. Percent := (divisor - dividend) / divisor

```
<expression name="saved" class="org.jfree.report.function.PercentageExpression">
  <properties>
    <property name="dividend">article.price</property>
    <property name="divisor">article.reducedPrice</property>
    <property name="useDifference">true</property>
  </properties>
</expression>
```

Advanced Calculations

Function Class	Description
CreateGroupAnchorsFunction	Adds an anchor to a group to all elements (objects) with the name specified in 'element'. The link target is read from a specified field. The column referenced by this field should contain URLs or Strings.
CreateHyperLinksFunction	Adds hyperlinks to all elements with the name specified in 'element'. The link target is read from a specified field. The column referenced by this field should contain URLs or Strings.
ElementColorFunction	Alternates between true and false for each item within a group. The function value determines when to use an object's color. If the function evaluates to true, the named object is painted with color assigned to colorTrue, else the object is painted with color assigned to colorFalse. Typically, used when applying negative formatting.
ElementVisibilitySwitchFunction	Alternates between true and false for each item within a group. The functions value affects a defined objects visibility. If the function evaluates to true, the named object is visible, else the element is invisible. Typically, used for row banding.
HideElementByNameFunction	Hides all objects with a given name, as long as the defined field does not contain the element name.
HidePageBandForTableExport	Hides the page header and footer if the current export type is not a pageable export (like printing, PDF or plain text).
PageFunction	Counts pages.
PageOfPagesFunction	Combines the PageFunction and PageTotalFunction.
PageTotalFunction	Prints the total number of pages of an report.
TextFormatExpression	Uses a java.text.MessageFormat to concatenate and format one or more values evaluated from an expression, function or report data source.
property-ref	Used to pass values from action sequeneces into a report.

CreateGroupAnchorsFunction

Adds an anchor to a group to all elements with the name specified in 'element'. The link target is read from a specified field. The column referenced by this field should contain URLs or Strings.

Example

```
<function class="org.jfree.report.function.CreateGroupAnchorsFunction" name="Grp_Anchr">
  <properties>
    <property name="field">Country</property>
    <property name="group">Product</property>
  </properties>
```

CreateHyperLinksFunction

Adds hyperlinks to all elements (objects) with the name specified in 'element'. The link target is read from a specified field. The column referenced by this field should contain URLs or Strings.

Example

```
<expression class="org.jfree.report.function.TextFormatExpression" name="URLCreateExpression">
  <properties>
    <property name="pattern">http://localhost:8080/pentaho/ViewAction?&amp;solution=test
&amp;path=reporting&amp;action=open_sales.xaction&amp;customer={0}</property>
    <property name="field[0]">CUSTOMERNAME</property>
  </properties>
</expression>

<function class="org.jfree.report.function.CreateHyperLinksFunction" name="AddHRefs">
  <properties>
    <property name="field">URLCreateExpression</property>
```

```

    <property name="element">URL-Field</property>
  </properties>
</function>

<items fontname="Arial" fontsize="9" fontstyle="plain" height="12" visible="true" vertical-
alignment="middle">
<rectangle name="background" x="15%" y="0" width="70%" height="100%" color="#DFDFDF" weight="0"/>
<string-field height="12" width="55%" fieldname="CUSTOMERNAME" x="15%" alignment="left" name="URL-
Field"></string-field>
<number-field height="12" width="15%" fieldname="QUANTITYORDERED" x="70%" alignment="right"
format="###,###.00"></number-field>
</items>

```

ElementColorFunction

A function that alternates between true and false for each item within a group. The function value determines when to use an object's color. If the function evaluates to true, the named object is painted with the color=True, else the object is painted with color=False.

Use a property tag to name an object contained in the Item section whose color should be affected by this function.

```

<expression class="org.jfree.report.modules.misc.beanshell.BSHEExpression" name="isVarNeg">
<properties>
  <property name="expression">Object getValue()
  {
    Object value = dataRow.get("&quot;VARIANCE&quot;");
    if (value instanceof Number == false)
    {
      return Boolean.FALSE;
    }
    Number number = (Number) value;
    if (number.doubleValue() &lt; 0)
    {
      return Boolean.TRUE;
    }
    return Boolean.FALSE;
  }
</property>
</properties>
</expression>

<function class="org.jfree.report.function.ElementColorFunction" name="changeColor">
  <properties>
    <property name="element">Variance Field</property>
    <property name="field">isVarNeg</property>
    <property name="colorTrue">red</property>
    <property name="colorFalse">black</property>
  </properties>
</function>

<items fontname="Arial" fontsize="10" vertical-alignment="middle">
  <string-field fieldname="POSITIONTITLE" height="15" vertical-alignment="middle" width="40%" x="15%"
y="0"/>
  <number-field alignment="right" fieldname="ACTUAL" format="$#,##0" height="15" width="15%" x="55%"
y="0"/>
  <number-field alignment="right" fieldname="BUDGET" format="$#,##0" height="15" width="15%" x="70%"
y="0"/>
  <number-field alignment="right" fieldname="VARIANCE" format="$#,##0" height="15" name="Variance
Field" width="15%" x="85%" y="0"/>
  <line color="gray" height="0" weight=".25" width="85%" x1="15%" x2="100%" y1="16" y2="16"/>
</items>

```

ElementVisibilitySwitchFunction

A function that alternates between true and false for each item within a group. The functions value affects a defined objects visibility. If the function evaluates to true, the named object is visible, else the element is invisible.

Objects do not define their own background color attribute. To create a background, a rectangle shape element must be created behind the element.

The ElementVisibilitySwitchFunction is used to trigger the visibility of an named object. If the object is in the background, it will alternate between an object being visible and not being visible.

The ElementVisibilitySwitchFunction defines two parameters:

element

The name of the element aka object in the item band that should be modified. The element must be named using the "name" attribute, only the first occurrence of that named element will be modified.

initial-state

The initial state of the function. (true or false) defaults to false. This is the reverts of the object's visibility (set to false to start with an visible object, set to true to hide the object in the first item row).

```
<function name="bkgrndTrggr" class="org.jfree.report.function.ElementVisibilitySwitchFunction">
  <properties>
    <property name="element">background</property>
    <property name="numberOfElements">1</property>
  </properties>

  <items fontname="Arial" fontsize="9" fontstyle="plain" height="12" layout-cachable="false"
  visible="true" vertical-alignment="middle">
    <rectangle name="background" x="15%" y="0" width="70%" height="100%" color="#DFDFDF" weight="0"/>
    <string-field height="12" width="55%" fieldname="CUSTOMERNAME" x="15%" alignment="left" name="URL-
    Field"></string-field>
    <number-field height="12" width="15%" fieldname="QUANTITYORDERED" x="70%" alignment="right"
    format="###,###.00"></number-field>
  </items>
```

HideElementByNameFunction

This function hides all objects with a given name, as long as the defined field does not contain the element name.

```
<function class="org.jfree.report.function.HideElementByNameFunction" name="LeaderVisibility">
  <properties>
    <property name="element">leader</property>
    <property name="field">recordID</property>
  </properties>
</function>

<band name="leader" x="0" y="0">
  ...
</band>
```

Hide Page Head and Footer if report is not a pageable export

Hide the page header and footer if the current export type is not a pageable export (like printing, PDF or plain text).

This function accepts two parameters:

hidePageBands - defaults to true; hides the page header and footer for non-pageable exports.

disableRepeatingHeader - defaults to false; all group headers, which have the repeat flag set to true get that flag reset to false. On a page break, no repeated header will be printed. As this may be confusing to the user and not always appropriate, this property defaults to false and must be enabled explicitly.

Example

```
<function class="org.jfree.report.function.HidePageBandForTableExportFunction"
name="hidePageHeaderandPageFooter">
  <properties/>
    <property name="hidePageBands">true</property>
    <property name="disableReportingHeader">true</property>
  </properties>
</function>
```

PageFunction

A report function that counts pages. This method is only useable when a report processor is used, which generated PageEvents. The PageableReportProcessor is one of them.

As with all page dependent functions: The function will only be active, when the page events get fired, this usually only happens during the last pagination run and the printing. The function level will be negative when this happens.

```
<function class="org.jfree.report.function.PagesFunction" name="Page"/>
```

PageOfPagesFunction

A report function that combines the PageFunction and PageTotalFunction. For example, it outputs the page as 3/14 which is page 3 of 14 total pages.

Restrictions for both classes apply to this one also.

```
<function class="org.jfree.report.function.PageOfPagesFunction" name="PageofPages"/>
```

PageTotalFunction

Prints the total number of pages of an report. If a group is specified, this function expects the group to have the manual page break enabled.

This function will only work as expected in group mode if the named group has page break set to true.

```
<function class="org.jfree.report.function.PagesTotalFunction" name="PageTotal"/>
```

TextFormatExpression

A TextFormatExpression uses a java.text.MessageFormat to concatenate and format one or more values evaluated from an expression, function or report data source.

The TextFormatExpression uses the pattern property to define the global format-pattern used when evaluating the expression. The data row fields used to fill the expressions placeholders are defined in a list of properties where the property-names are numbers.

The property counting starts at "0".

The Syntax of the 'pattern' property is explained in java.text.MessageFormat.

Example:

```
<expression name="expr" class="org.jfree.report.function.TextFormatExpression">
  <properties>
    <property name="pattern">Invoice for your order from {0, date, EEE, MMM d, yyyy}</property>
    <property name="fields[0]">printdate</property>
  </properties>
</expression>
```

```
<expression class="org.jfree.report.function.TextFormatExpression" name="leader.name">
  <properties>
    <property name="pattern">{0} {1}</property>
    <property name="field[0]">leader.firstName</property>
    <property name="field[1]">leader.lastName</property>
  </properties>
</expression>
```

Property Reference

Gets the current date.

Example

```
<property-ref report.date/>
```

Accepts either "text" or "serialized-base64" encoding.

For example, these are variables that could be passed from a Pentaho action.

Example


```
<property-ref name="productname"/> <!--variable passed from Pentaho *.xaction-->
<property-ref name="username"/> <!--variable passed from Pentaho *.xaction-->
```

These property references need to be defined before it can be used in a report.

Pentaho Extended Functions for Charting

Collect Chart Data

Function Class	Description
PieSetCollectorFunction	Data collector for Pie Charts
CategorySetCollectorFunction	Data collector for Bar, Line, and Area

Line, Bar, and Area Chart Data Collector

Pentaho 1.0

```
<expression class="org.pentaho.jfree.reportcharts.CategorySetCollectorFunction" name="chartDataSource">
  <properties>
    <property name="categoryColumn">CUSTOMERNAME</property>
    <property name="seriesName[0]">Customer Sales</property>
    <property name="seriesName[1]">Customer Margin</property>
    <!-- <property name="seriesName[n]">Series Name</property> -->
    <property name="valueColumn[0]">Total Sales</property>
    <property name="valueColumn[1]">Margin</property>
    <!-- <property name="valueColumn[n]">Column Name</property> -->
    <property name="summaryOnly">false</property>
    <!-- <property name="group">Customer_Type</property> --><!-- Only use if summaryOnly=true -->
  </properties>
</expression>
```

Pentaho 1.1 or higher

```
<expression class="org.pentaho.plugin.jfreereport.reportcharts.CategorySetCollectorFunction"
name="chartDataSource">
  <properties>
    <property name="categoryColumn">CUSTOMERNAME</property>
    <property name="seriesName[0]">Customer Sales</property>
    <property name="seriesName[1]">Customer Margin</property>
    <!-- <property name="seriesName[n]">Series Name</property> -->
    <property name="valueColumn[0]">Total Sales</property>
    <property name="valueColumn[1]">Margin</property>
    <!-- <property name="valueColumn[n]">Column Name</property> -->
    <property name="summaryOnly">false</property>
    <!-- <property name="group">Customer_Type</property> --><!-- Only use if summaryOnly=true -->
  </properties>
</expression>
```

Pie Chart Data Collector

Pentaho 1.0

```
<expression class="org.pentaho.jfree.reportcharts.PieSetCollectorFunction" name="ChartDataSource">
  <properties>
    <property name="group">RegionGroup</property><!-- Only use if summaryOnly=true -->
    <property name="seriesColumn">REGION</property>
    <property name="valueColumn">RegionBudget</property>
    <property name="summaryOnly">true</property>
  </properties>
</expression>
```

Pentaho 1.1 or higher

```
<expression class="org.pentaho.plugin.jfreereport.reportcharts.PieSetCollectorFunction"
name="ChartDataSource">
  <properties>
    <property name="group">RegionGroup</property><!-- Only use if summaryOnly=true -->
    <property name="seriesColumn">REGION</property>
    <property name="valueColumn">RegionBudget</property>
    <property name="summaryOnly">true</property>
  </properties>
</expression>
```

Set Chart Type and attributes

Function Class	Description
PieChartExpression	Sets Pie Chart as the chart type and sets formatting attributes.
BarChartExpression	Sets Bar Chart as the chart type and sets formatting attributes.
LineChartExpression	Sets Line Chart as the chart type and sets formatting attributes.
AreaChartExpression	Sets Area Chart as the chart type and sets formatting attributes.
MultiPieChartExpression	Sets Multi Pie Chart as the chart type and sets formatting attributes.

Pie Chart (Pentaho 1.0)

```
<expression class="org.pentaho.jfree.reportcharts.PieChartExpression" name="PieChart">
  <properties>
    <property name="dataSource">chartDataSource</property><!--Set Data Collector -->
    <property name="title">RegionDepartmentChartTitle</property>
    <property name="titleFont">Arial-BOLD-14</property><!--fontfamilyname-style-pointsize-->
    <property name="labelFont">Arial--8</property>
    <property name="legendFont">Arial--6</property>
    <property name="backgroundColor">#DDDDDD</property>
    <property name="showBorder">true</property>
    <property name="legendLocation">BOTTOM</property>
    <property name="showLegend">true</property>
    <property name="chartWidth">500</property>
    <property name="chartHeight">400</property>
    <property name="useDrawable">false</property>
    <property name="ignoreNulls">true</property>
    <property name="ignoreZeros">true</property>
    <property name="pieLabelFormat">{2}</property>
    <property name="pieLegendLabelFormat">{0}</property>
    <property name="explodeSegment">maxValue</property>
    <property name="explodePct">.20</property>
  </properties>
</expression>
```

Bar Chart (Pentaho 1.0)

```
<expression class="org.pentaho.jfree.reportcharts.BarChartExpression" name="BarChart">
  <properties>
    <property name="dataSource">chartDataSource</property><!--Set Data Collector -->
    <property name="title">Sales By Customer</property>
    <property name="titleFont">Arial-BOLD-9</property><!--fontfamilyname-style-pointsize -->
    <property name="labelFont">Arial--9</property>
    <property name="legendFont">Arial--6</property>
    <property name="backgroundColor">white</property>
    <property name="showBorder">false</property>
    <property name="horizontal">true</property>
    <property name="drawBarOutline">true</property>
    <property name="seriesColor[0]">#2954A9</property>
    <property name="legendLocation">BOTTOM</property>
    <property name="showLegend">false</property>
    <property name="maxBarWidth">0.10</property>
    <property name="maxCategoryLabelWidthRatio">7.0</property>
    <property name="chartHeight">600</property>
    <property name="chartWidth">480</property>
    <property name="useDrawable">true</property>
  </properties>
</expression>
```

Pie Chart

```
<expression class="org.pentaho.plugin.jfreereport.reportcharts.PieChartExpression" name="PieChart">
  <properties>
    <property name="dataSource">chartDataSource</property><!--Set Data Collector -->
    <property name="title">RegionDepartmentChartTitle</property>
    <property name="titleFont">Arial-BOLD-14</property><!--fontfamilyname-style-pointsize-->
    <property name="labelFont">Arial--8</property>
    <property name="legendFont">Arial--6</property>
    <property name="backgroundColor">#DDDDDD</property>
    <property name="showBorder">true</property>
    <property name="legendLocation">BOTTOM</property>
    <property name="showLegend">true</property>
    <property name="chartWidth">500</property>
    <property name="chartHeight">400</property>
    <property name="useDrawable">false</property>
    <property name="ignoreNulls">true</property>
    <property name="ignoreZeros">true</property>
  </properties>
</expression>
```

```

<property name="pieLabelFormat">{2}</property>
<property name="pieLegendLabelFormat">{0}</property>
<property name="explodeSegment">maxValue</property>
<property name="explodePct">.20</property>
</properties>
</expression>

```

Bar Chart

```

<expression class="org.pentaho.plugin.jfreereport.reportcharts.BarChartExpression" name="BarChart">
<properties>
  <property name="dataSource">chartDataSource</property><!--Set Data Collector -->
  <property name="title">Sales By Customer</property>
  <property name="titleFont">Arial-BOLD-9</property><!--fontfamilyname-style-pointsize -->
  <property name="labelFont">Arial--9</property>
  <property name="seriesColor[0]">#018FE1</property><!--HTML Color Code -->
  <property name="seriesColor[1]">#CB02D2</property><!--HTML Color Code -->
  <!-- <property name="seriesColor[n]">#CCCCCC</property><!--HTML Color Code --> -->
  <property name="legendFont">Arial--6</property>
  <property name="backgroundColor">white</property>
  <property name="showBorder">>false</property>
  <property name="horizontal">>true</property>
  <property name="drawBarOutline">>true</property>
  <property name="legendLocation">BOTTOM</property>
  <property name="showLegend">>false</property>
  <property name="maxBarWidth">0.10</property>
  <property name="maxCategoryLabelWidthRatio">7.0</property>
  <property name="chartHeight">600</property>
  <property name="chartWidth">480</property>
  <property name="useDrawable">>true</property>
  <property name="stacked">>false</property>
</properties>
</expression>

```

Line Chart

```

<expression class="org.pentaho.plugin.jfreereport.reportcharts.LineChartExpression" name="LineChart">
<properties>
  <property name="dataSource">chartDataSource</property><!--Set Data Collector -->
  <property name="title">Actual vs. Budget</property>
  <property name="titleFont">Arial-BOLD-14</property><!--fontfamilyname-style-pointsize -->
  <property name="labelFont">Arial--8</property>
  <property name="legendFont">Arial--6</property>
  <property name="backgroundColor">#DDDDDD</property>
  <property name="showBorder">>true</property>
  <property name="horizontal">>false</property>
  <property name="seriesColor[0]">#018FE1</property><!--HTML Color Code -->
  <property name="seriesColor[1]">#CB02D2</property><!--HTML Color Code -->
  <!-- <property name="seriesColor[n]">#CCCCCC</property><!--HTML Color Code --> -->
  <property name="legendLocation">BOTTOM</property>
  <property name="showLegend">>true</property>
  <property name="chartHeight">300</property>
  <property name="chartWidth">540</property>
  <property name="useDrawable">>false</property>
  <property name="labelRotation">7.0</property>
</properties>
</expression>

```

Area Chart

```

<expression class="org.pentaho.plugin.jfreereport.reportcharts.AreaChartExpression" name="AreaChart">
<properties>
  <property name="dataSource">chartDataSource</property><!--Set Data Collector -->
  <property name="title">Actual vs. Budget</property>
  <property name="titleFont">Arial-BOLD-14</property><!--fontfamilyname-style-pointsize -->
  <property name="labelFont">Arial--8</property><!--fontfamilyname-style-pointsize -->
  <property name="legendFont">Arial--6</property><!--fontfamilyname-style-pointsize -->
  <property name="backgroundColor">#DDDDDD</property>
  <property name="showBorder">>true</property>
  <property name="horizontal">>false</property>
  <property name="dataSource">chartDataSource</property>
  <property name="seriesColor[0]">#018FE1</property><!--HTML Color Code -->
  <property name="seriesColor[1]">#CB02D2</property><!--HTML Color Code -->
  <!-- <property name="seriesColor[n]">#CCCCCC</property><!--HTML Color Code --> -->
  <property name="legendLocation">BOTTOM</property>
  <property name="showLegend">>true</property>
  <property name="chartHeight">300</property>
  <property name="chartWidth">540</property>
  <property name="useDrawable">>false</property>

```

```
<property name="labelRotation">7.0</property>
</properties>
</expression>
```

Multi Pie Chart

```
<expression class="org.pentaho.plugin.jfreereport.reportcharts.MultiPieChartExpression"
name="MulitPieChart">
  <properties>
    <property name="dataSource">chartDataSource</property><!--Set Data Collector -->
    <property name="title">Actual vs. Budget</property>
    <property name="titleFont">Arial-BOLD-14</property><!--fontfamilyname-style-pointsize -->
    <property name="labelFont">Arial--8</property><!--fontfamilyname-style-pointsize -->
    <property name="legendFont">Arial--6</property><!--fontfamilyname-style-pointsize -->
    <property name="backgroundColor">#DDDDDD</property>
    <property name="showBorder">true</property>
    <property name="seriesColor[0]">#018FE1</property><!--HTML Color Code -->
    <property name="seriesColor[1]">#CB02D2</property><!--HTML Color Code -->
    <!-- <property name="seriesColor[n]">#CCCCCC</property><!--HTML Color Code --> -->
    <property name="legendLocation">BOTTOM</property>
    <property name="showLegend">true</property>
    <property name="chartHeight">300</property>
    <property name="chartWidth">540</property>
    <property name="useDrawable">>false</property>
    <property name="multipieByRow">>false</property>
    <property name="multipieLabelFormat">{2}</property>
  </properties>
</expression>
```

Display Chart

Use the imageurl-field to display the chart in the report.

```
<imageurl-field dynamic="true" x="5%" y="35" fieldname="BarChart"/>
<imageurl-field dynamic="true" x="5%" y="35" fieldname="LineChart"/>
<imageurl-field dynamic="true" x="5%" y="35" fieldname="PieChart"/>
```

Configuration

CSV default separator

Defines the default separator, that should be used for the output.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.csv.Separator">,</property>
</configuration>
```

CSV write column name into first row

Defines whether to write the column names of the data row into the first row.

```
<configuration>
  <property name="org.jfree.report.modules.output.csv.WriteDatarowNames">>true</property>
</configuration>
```

Excel output set paper size

Defines the Paper that should be used in the excel print setup. If no paper is defined, a suitable paper will be selected automatically.

```
<configuration>
  <property name=" org.jfree.report.modules.output.table.xls.Paper">LETTER</property>
</configuration>
```

Excel output set paper orientation

Defines the Paper that should be used in the excel print setup. If no paper is defined, an suitable paper will be selected automatically. Valid values are "Landscape" or "Portrait"

```
<configuration>
  <property name="org.jfree.report.modules.output.table.xls.PaperOrientation">LETTER</property>
</configuration>
```

HTML encoding

Defines the encoding for the HTML.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.Encoding">UTF-8</property>
</configuration>
```

HTML body fragment only

Defines whether to define a body fragment only. This removes the header and body tags and makes it easy to include the generated content in own pages.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.BodyFragment">>false</property>
</configuration>
```

HTML, XHTML 1.0 or HTML 4.0 format

Defines whether to generate XHTML1.0 or HTML4.0 content.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.GenerateXHTML">>false</property>
</configuration>
```

HTML default title

Defines the default title for generated HTML documents.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.Title">My First Report</property>
</configuration>
```

HTML set author

Defines a author for generated HTML documents.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.Author">Shakespeare</property>
</configuration>
```

HTML set empty cells to use CSS

Defines, whether empty cells will be controlled using CSS. If this is set to true, the Table gets the style property "empty-cells: show" assigned and the otherwise required no-break-space entities will no longer be generated.

The Internet Explorer is not able to use correct CSS2 yet. Thus, the default is set to "false". This is a local configuration setting and can be enabled in the report.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.EmptyCellsUseCSS">>false</property>
</configuration>
```

HTML image size in Points or Pixels

Defines, whether image sizes should be given in device independent Point unit (1/72 inch) or in pixels (which is device dependent and makes it hard to predict the final size in the HTML-Page).

The default is pixels, false.

```
<configuration>
  <property
name="org.jfree.report.modules.output.table.html.UseDeviceIndependentImageSize">>false</property>
</configuration>
```

HTML table carrier a border definition

Defines, whether table rows should carry a border definition (in addition to the cell definitions). This is a workaround for a layout bug in the Mozilla browser family.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.TableRowBorderDefinition">false</property>
</configuration>
```

HTML Table, use proportional column widths

Enables the use of proportional column widths instead of static ones. The resulting table will have a width of 100% and the columns will have the proportional equivalent of their static widths as width.

```
<configuration>
  <property name="org.jfree.report.modules.output.table.html.ProportionalColumnWidths">false</property>
</configuration>
```

PDF set author

The author string of the document.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.Author">Larry Augustin</property>
</configuration>
```

PDF set title

The title of the document.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.Title">Sales Report</property>
</configuration>
```

PDF encoding

Defines PDF encoding for report.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.Encoding">Cp1252</property>
</configuration>
```

PDF embed fonts

Defines whether to embed fonts in the generated PDF. This will result in larger PDF files, but makes sure that all characters are displayed properly.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.EmbedFonts">Cp1252</property>
</configuration>
```

PDF version

The PDF Version that should be created. Valid values are "1.2", "1.3", "1.4" or "1.5"

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.Version">1.4</property>
</configuration>
```

PDF Security, Allow Copy

Defines, whether the PDF Security setting allows copying of the document's contents by default. PDF Security flags are only used with security enabled and a password set.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowCopy">false</property>
</configuration>
```

PDF Security, Allow Printing

Defines, whether the PDF Security setting allows printing of the document's contents by default

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowPrinting">false</property>
</configuration>
```

PDF Security, Allow Modifying Contents

Defines, whether the PDF Security setting allows modifying of the document by default.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowModifyContents">false</property>
</configuration>
```

PDF Security, Allow Modifying Annotations

Defines, whether the PDF Security setting allows document annotations by default.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowModifyAnnotations">false</property>
</configuration>
```

PDF Security, Allow Fill-in of Document Forms

Defines, whether the PDF Security setting allows the fill-in of document forms (input fields etc).

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowFillIn">false</property>
</configuration>
```

PDF Security, Allow access for screenreaders

Defines, whether the PDF Security setting allows access for screen readers by default. Although this could make it easier to grab the contents your PDF document, enabling this property will allow blind people to access your content. It is usually a good idea to always enable this property.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowScreenReaders">false</property>
</configuration>
```

PDF Security, Allows reassembly of document

Defines, whether the PDF Security setting allows reassembly of the document by default.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowAssembly">false</property>
</configuration>
```

PDF Security, Allow low quality printing

Defines, whether the PDF Security setting allows low quality printing of the document's contents by default.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.AllowDegradedPrinting">false</property>
</configuration>
```

PDF Security, Encrypt

Defines whether the PDF file should be encrypted by default. Set this to "none" "40bit" or "128bit". Access restrictions have no effect if the document is not encrypted, and not all security settings are active on 40-Bit encryption.

No encryption

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.Encryption">none</property>
</configuration>
```

40 bit encryption

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.Encryption">40bit</property>
</configuration>
```

128 bit encryption

```
<configuration>
```

```
<property name="org.jfree.report.modules.output.pageable.pdf.Encryption">128bit</property>
</configuration>
```

Warning: Specifying passwords in the property file can introduce a security risk. Do not use these settings on the client side or in unprotected (world-readable) server environments. In these cases use other means of defining the password.

PDF Security, Password

The default user password for the document. Users are able to access the document within the defined security restrictions.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.UserPassword">password</property>
</configuration>
```

PDF Security, Owner

The default owner password for the document. The owner has all permissions on the document and will be able to change the security settings.

```
<configuration>
  <property name="org.jfree.report.modules.output.pageable.pdf.OwnerPassword">password</property>
</configuration>
```