myInsight for Documentum

Fusion Interface User Guide

AMEXIO

Contents

1. Version History

Date	Changes	Version number
2-June-2016	Updates for myInsight v5.1. Updated new variables.	1.0
21-April-2017	Update for myInsight v6.0.	1.1
18-October-2018	Update for myInsight 7.0.	1.2
26-March-2021	Update for myInsight 7.4.	1.3
7-April-2023	Update for myInsight 8.0. Reordeded variables chapter.	1.4
22-September-2023	Update for myInsight 8.1.	1.5
17-February-2025	Update for myInsight 9.0.	1.6

2. Product Description

With myInsight for Documentum, end-users can request reports from predefined report definitions. They can see the reports displayed on their computer screen or receive reports automatically in their e-mail or at a specified location inside or outside the Documentum repository. The preferred format can be chosen by the end-user, without the end-user needing any knowledge about DQL, HTML or XSL.

myInsight for Documentum categorizes functionalities according to the user's role in the report generator. There are 3 predefined roles:

- Users in the *Report User* role, which supplies predefined reports from the system, require no knowledge of either DQL or style sheets.
- Users in the *Report Administrator* role can schedule reports so that they are generated automatically at a predefined time and location.
- Users in the *Report Builder* role can define new report specifications by configuring the DQL statements, and they can identify and compose the desired style sheets.

myInsight for Documentum can be accessed by anyone who has been given one of these default roles.

myInsight for Documentum can produce output in any format that can be generated using XSL style sheets. For example: reports can be presented in PDF, HTML, text file, Microsoft Excel spread sheet or Microsoft Word format. myInsight for Documentum can also e-mail the output file automatically. In this case the recipient does not need to be a Documentum user and can even be someone from outside the organisation.

myInsight for Documentum is integrated into Documentum Webtop, Documentum Administrator, Documentum D2 Classic, Documentum Smart View and Documentum xCP, in line with the corporate philosophy of Documentum. This enables end users to work in an environment that they are already familiar with. Due to its full integration within the Documentum environment, no additional components need to be installed on the end user's local machine.

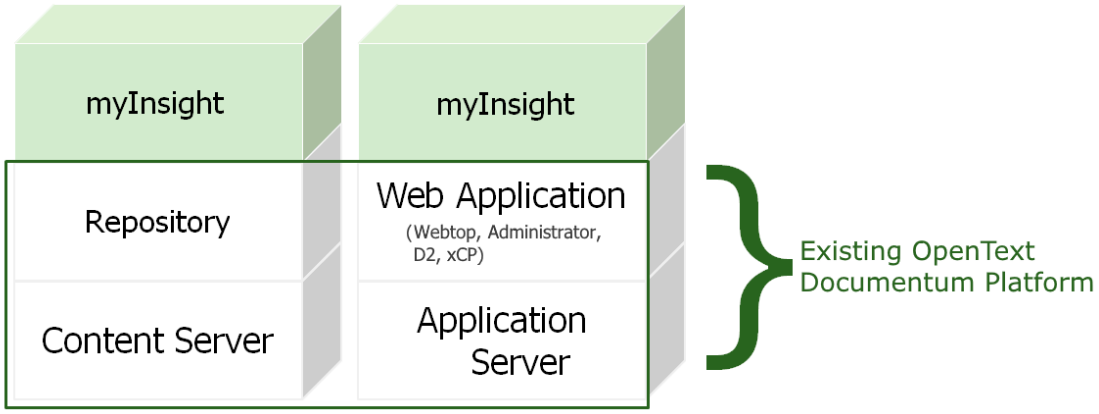


Figure 1: myInsight integration into the OpenText Documentum platform

As shown in the previous figure, myInsight for Documentum components are located on both Repository level as Web Application Level. Both use standard OpenText | Documentum subcomponents for its functionality.

As of version 7.0, myInsight for Documentum can be installed separate from the Content Server.

In December 2015 euroscript has been renamed to Amplexor. The product euroscript Documentum Report Generator has been renamed to myInsight for Documentum. In 2020, Amplexor joined the Acolad Group under the name Acolad Digital. In October 2023, Acolad Digital became AmeXio.

For the reader's convenience, the abbreviation 'myInsight' will be used in this document, instead of the full product title 'myInsight for Documentum'.

2.1. Document Description

One of the most challenging aspects of creating a stunning data visualization is presenting the data in the right way.

Before you can even start with customizing the report to your preferences you have to think about the data that you want to retrieve, how the stylesheet/presentation expects the data to be formatted and how to create/structure the query. When you can finally check the result, it is often not what you had envisioned. This means that you can go back to the drawing board, modify the query, check the new results, rinse and repeat.

This is a very time consuming and needlessly complex process that forces you to think of the way that you have to format the data without seeing a visual representation of that data and the result.

The Fusion Interface was created to alleviate this problem by giving you the ability to see the visual result of your data manipulation while you are formatting the data. This means that you only need to have a general sense of the data that you need to query, since you can fine-tune your results afterwards.

For example: I need the ID, name, creation date and content type of all documents that I have created. You can then select the chart that you would like to show and start manipulating the data by counting, grouping and filtering. Once you have created the perfect visualization, you can copy the settings that you used, save them and edit the query if necessary. The report will now use your settings and display your visualization.

The Fusion Interface bridges the gap between IT and business by allowing business users to make data visualizations on actual data, without the need for extensive technical (query) knowledge.

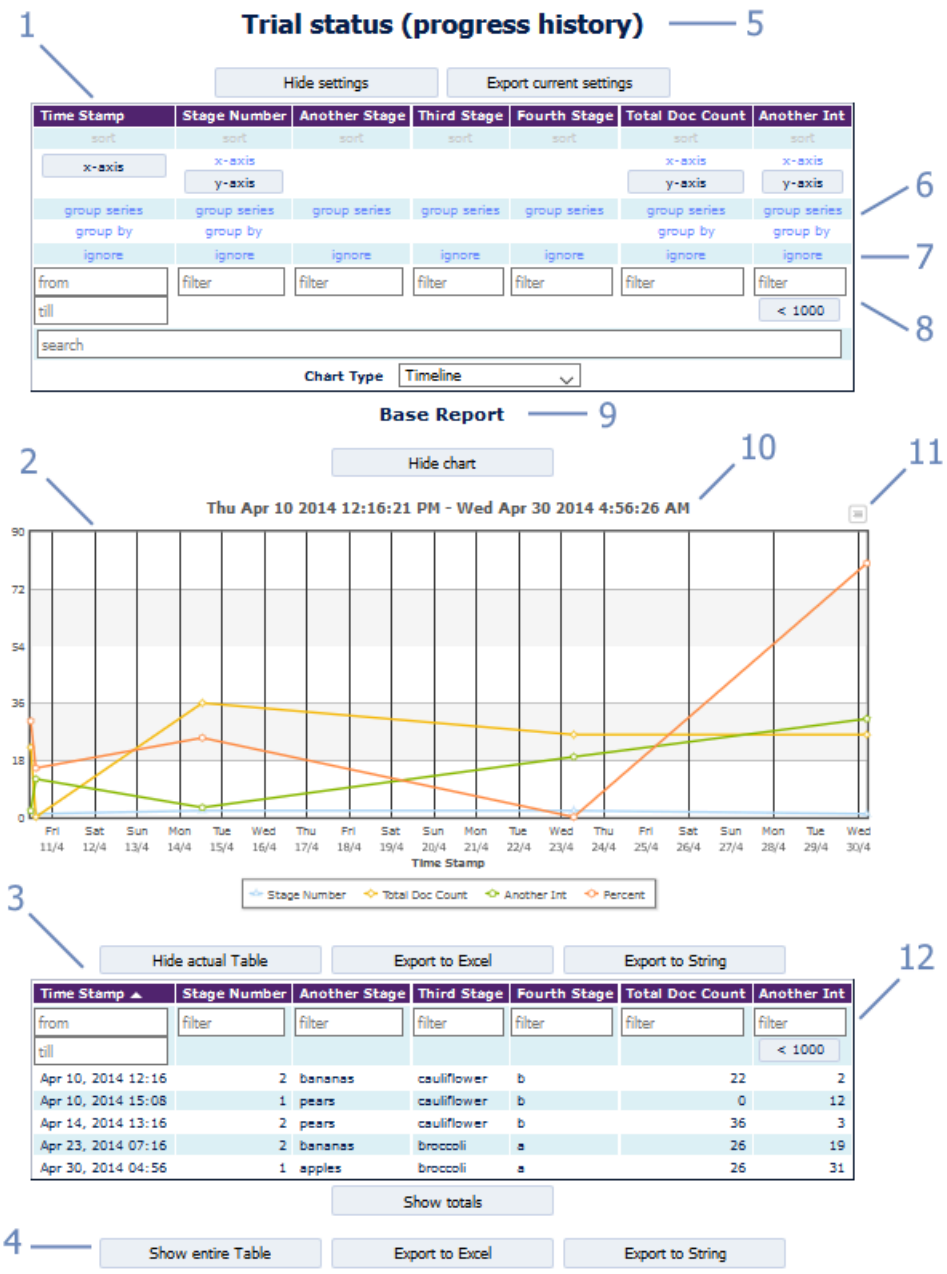


Figure 2: A complete overview of Fusion Interface in myInsight

1. Settings table. These settings are used to configure the data. Settings may or may not be available depending on the chosen chart type.
2. The actual chart. It's a visual representation based on the selected data and settings.
3. The actual table. This table displays the data that is used in the chart.
4. Show entire Table will display the original data. It is collapsed on default.
5. Title. This defaults to the report name, but can be customized.
6. Grouping. Data in the chart can be grouped by a value (or a series for the Scatterplot chart).
7. Ignore. This button removes the column from the chart and the actual table.
8. Filters. Several filters are available to filter the data used for the chart.
9. 'Breadcrumb' functionality. When you change settings, you can revert back to an older configuration by clicking on the breadcrumb.
10. Chart title.
11. Export button. This button allows you to export the chart to a certain formats (e.g. PDF, PNG).
12. Filter on the actual table. This works similar to the filter in the settings table, but it can also filter after grouping.

3. Settings

Formatting the visual representation of the data can be done in the settings table. In this table, the user can define report specific settings, and export them for later use.

The table below explains the rows of the settings table (there is one column in the settings table for each column in the data table).

Name	Description
sort	Sort the data on that column ascending or descending.
x-axis	The column that is used for the x-axis. In most cases, any type of data can be used as the x-axis. Note that the x-axis is not always plotted on the bottom of the graph.
y-axis	The column that is used for the y-axis. Can be an integer, percentage or double. For 3D graphs, all non-ignored suitable columns are plotted. Only one is plotted for 2D charts.
group-by	Groups all identical records for the selected column together in a single row. Now every record on the x-axis represents one group. In the settings table it replaces the normal y-axis with groupers.

sub-group	Only appears when data is grouped. Divides each group into multiple subgroups. The number of times a chart can be sub-grouped varies.
	• *General Charts: 1 time, with following exceptions:*
	  • Column (2D): 2 times.
	  • Pie: Many times.
	• *2D charts: 0 times*
	• *3D charts 1 time.*

groupers	Only appears when data is grouped. • *Count*: counts the number of records per group. This number is the same for every column. • *Distinct*: count the number of unique values per group. If a group contains [“a”, “a”, “b”, “b”, “c”]. It will count 3 different values. • *Sum*: sums all the values of a column per group. • *Minimum*: gives the lowest value of that column per group. • *Maximum*: gives the highest value of that column per group. • *Average*: gives the average value of that column per group.

Name	Description
group series	Only used for scatterplots. For every group, the points on the y-axis in that group are counted as one dataset.
ignore	Do not show the data for that column (filters still apply).
filter	Looks for the occurrences of the data typed in the filter. It will keep all rows that have the search term occur in the selected column. The following special characters can be used to refine searches: • **'<'** or **'>'**, followed by a number. This will count any numbers less or greater than that number. • **'>='** or **'<='**, followed by a number. This will count any numbers less (<) or greater (>) than and including that number. • **'='**counts only exact matches are counted. • **'!='** counts the rows where no exact match is found. • **[*min*,*max*]** To search in a range of numbers. This includes the maximum and minimum. **(*min*,*max*)** only includes numbers between min and max, but not the minimum and maximum themselves. **[*min*,*max*)** and **(*min*,*max*]** are also legal. Multiple filters can be added to one column. In which case only rows are counted that match both terms. Filters can be removed by clicking on them. If combined with grouping, filters are applied before the rows are grouped.
search	Similar to filter, but any column may contain the search term.
chart type	Select the type of chart you want.
breadcrumbs	Home is always visible and resets the settings back to how the chart was first presented. If the chart is changed by drilling down in the chart itself, a new breadcrumb will be added. Clicking on an instance of the breacrumb will take you back to that level.

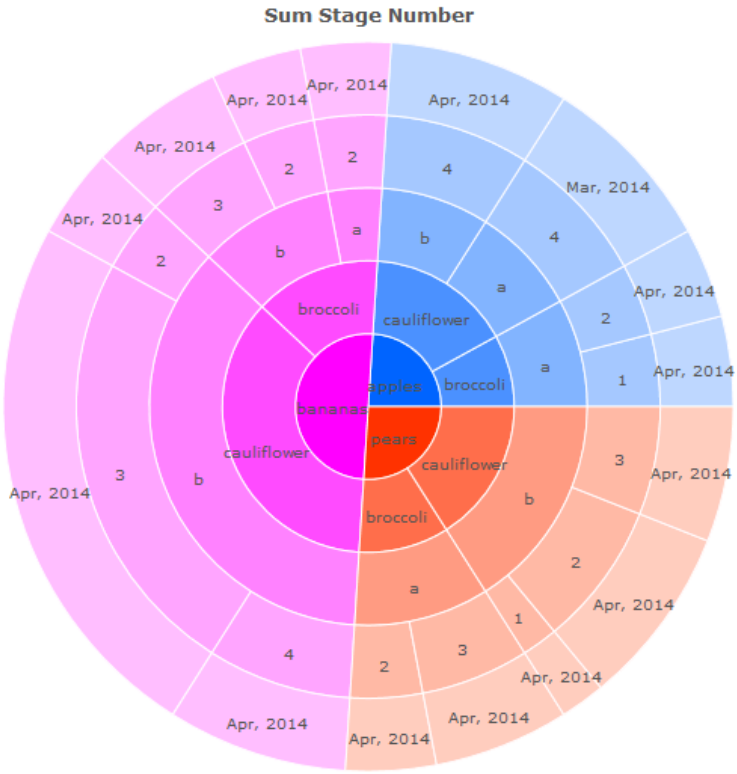
3.1. Dimensions

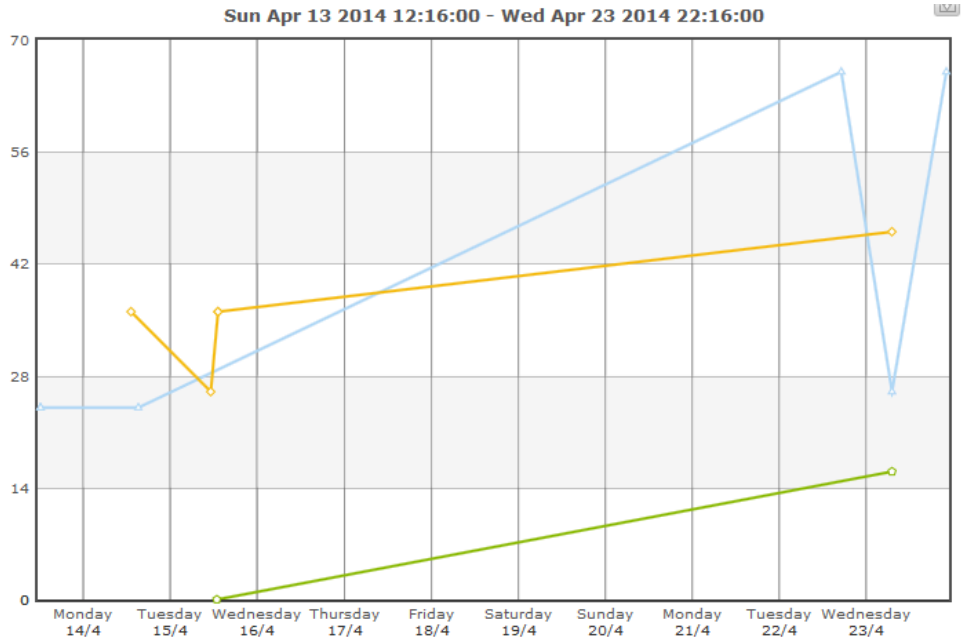
The way the data is presented depends on the amount of dimensions. The amount of dimensions depends on the chart type.

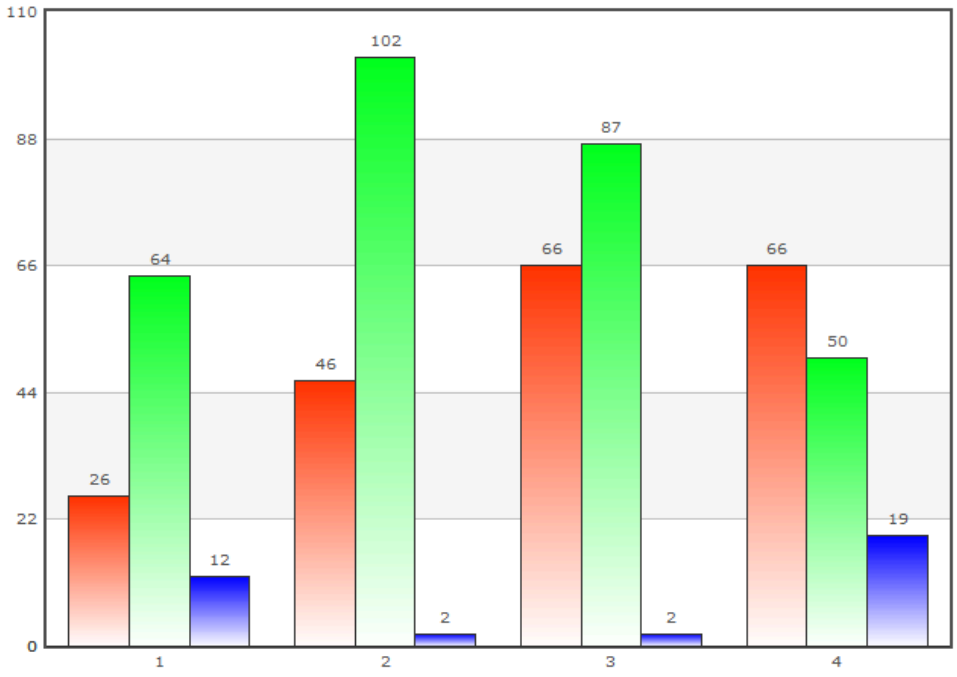
- 2D: This has an X-axis and a Y-axis. Ungrouped with 1 Y-axis or grouped with 1 'grouper'.
- 3D: Multiple series. Examples of this would be a bar chart grouped in clusters, ungrouped with multiple Y-axis, or grouped with multiple groupers and grouped with 1 sub-group.
- 4D: Groups are divided into multiple subgroups, which in turn are divided into multiple subgroups.

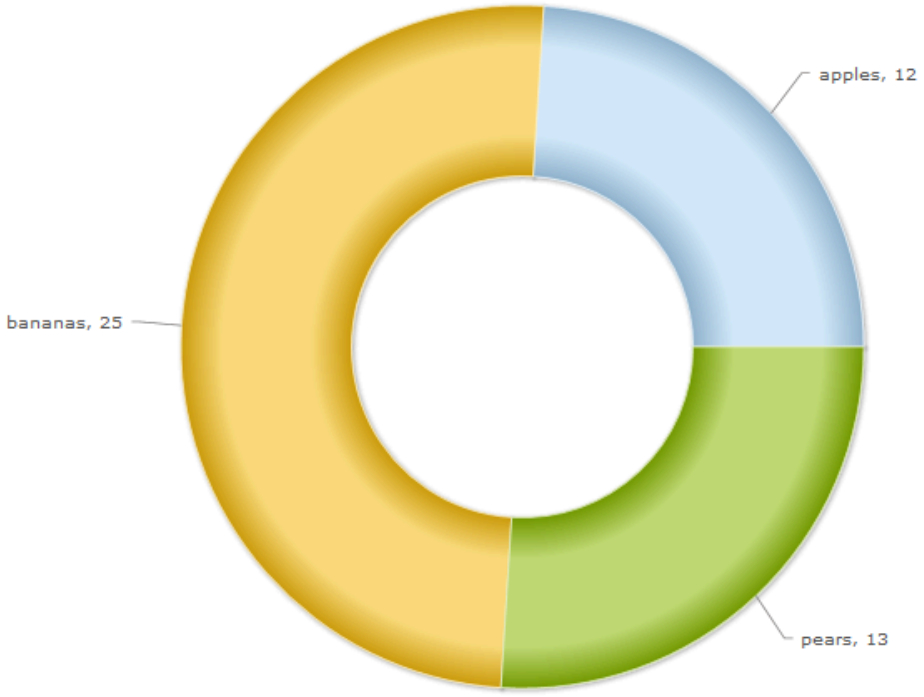
4. Chart types

Charts in FusionInterface are grouped into five groups. General charts, Scatterplot, 3D, 2D and 'other' charts. A group consists of one or more types that can use the same data.

Group	Description
General charts	Charts that change appearance depending on the data they show. • Column: Standard vertical columns. • 3D gives a depth effect. This is purely cosmetic. 3D will take longer to load so it is not recommended to use with a lot of data. • 2D can be sub-grouped an additional time. • Bar: Standard Bar chart. • Line: Standard Line chart • Smoothed: Similar to the Line chart but the line is smoothed. • Area: Line chart where the area beneath the line is filled.  Figure 3: Example of a multipie • Pie: When not sub-grouped, a normal Pie chart. • When sub-grouped, changes into multipie. This divides the chart into an inner circle of groups and an outer cirle of subgroups. • These subgroups can be divided into smaller subgroups. • Clicking on a group will zoom in on the group.

Group	Description
Scatterplot	This chart type contains one chart, the Timeline. It is the only type of graph where the table rows are not plotted on the graph as regular intervals. This graph type is automatically chosen if there is a time column available. The x-axis can also be an integer, float or percentage.  It also is the only chart that has the group series option. This is similar to the group by function but shows every group as a separate line instead. In practice it means that you can have multiple series next to each other that have different points on the x-axis.

Group	Description
3D Charts	Charts that have an x-axis with one or more series plotted against the y-axis. Since this type has three dimensions, groups can be divided into subgroups.  • *Inverse*: the data axis is reversed. For example, an inverse column chart has the big bars for 0 and small bars for low numbers. • *Logarithmic*: the data axis is displayed logarithmic. This means that the absolute distance between 1 and 10 is the same as between 10 and 100. Good for displaying records that have a lot of range in their data. • *Marimekko*: similar to the Stacked Column, but the total amount per row is represented by width instead of by height. • *Radar*: the amount is shown by the distance to the center of the graph. • *Stacked Charts*: data sets are stacked on top of each other. The total height is the sum of all columns for that row. • *Zoom Line*: looks similar to the Line chart but has extra functionality. You can drag your mouse over a portion of the graph to zoom in on that part. In the top right corner of the chart next to the export button are additional options. For example, a button to compare two sections of the chart.

Group	Description
2D Charts	Charts with just one series on the y-axis.  • *Doughnut*: Every value is represented with a portion of the donut. • *Pareto*: Chart that shows the amount per record from highest to lowest. Also shows the cumulative total percentage.
Other charts	• *Bubble*: Groups on X-axis, subgroups on Y-axis. The Y-axis decides the size of the points. • *Heat map*: Works a bit like the bubble, higher numbers are shown with different colors. The colors can be changed with the *customColors* variables. • *Box & Whisker*: Shows average and deviation for each series on the X-axis.

4.1. List of available charts

Some charts use multiple FusionCharts depending on the supplied data. These are grouped under "General Charts" in the selection.

- Display name: Name used in the Chart Type selection of the settings table.
- Chart name: Javascript alias used for FusionCharts.
- Chart type: the dimensions used for the charts. Scatter and Misc have special rules.

- Default: Whether this chart is displayed on default in the chart selection. They can still be added with variables such as ***addChartType*** and *onStartChartType*.
- Link: Link to the FusionChart website, this has the properties for each chart. They can be set with the ***setChartProperties*** variable.

Display name	Chart name	Chart type	Default	Link
Area	msarea	2D 3D	Yes	**Area 2D** **Multi-series Area 2D**
Area - Smoothed	splinearea mssplinearea	2D 3D	Yes	**Single-series Spline Area 2D** **Multi-series Spline Area**
Bar - 2D	bar2d msbar2d	2D 3D	Yes	**Bar 2D** **Multi-series Bar 2D**
Bar - 3D	bar3d msbar3d	2D 3D	Yes	**Bar 3D** **Multi-series Bar 3D**
Box & Whisker	boxandwhisker2d	Misc	No	**Box and Whisker 2D**
Bubble	bubble	Misc	Yes	**Bubble Chart**
Column (2D)	column2d mscolumn2d msstackedcolumn2d	2D 3D 4D	Yes	**Column 2D** **Multi-series Column 2D** **Multi-series Stacked Column 2D**
Column (3D)	column3d mscolumn3d	2D 3D	Yes	**Column 3D** **Multi-series Column 3D**
Doughnut	doughnut2d	2D	Yes	**Doughnut 2D**
Doughnut - 3D	doughnut3d	2D	No	**Doughnut 3D**
Error Line	errorline	3D	No	**Error Line 2D**
Error Scatter	errorscatter	Scatter	No	**Error Scatter Chart**

Display name	Chart name	Chart type	Default	Link
Funnel	funnel	2D	No	**Funnel Chart**
Heat map	heatmap	Misc	No	**Heat Map Chart**
Inverse Area	inversemsarea	3D	Yes	**Inverse Y-axis Area Chart**
Inverse Column	inversemscolumn2d	3D	Yes	**Inverse Y-axis Column 2D Chart**
Inverse Line	inversemsline	3D	Yes	**Inverse Y-axis Line 2D Chart**
Line	line msline	2D 3D	Yes	**Line 2D** **Multi-series Line 2D**
Line - Smoothed	spline msspline	2D 3D	Yes	**Singe-series Spline 2D** **Multi-series Spline**
Logarithmic Column	logmscolumn2d	3D	Yes	**Logarithmic Column 2D Chart**
Logarithmic Line	LogMSLine	3D	No	**Logarithmic Line 2D Chart**
Marimekko	marimekko	3D	Yes	**Marimekko**
Pareto	pareto2d	2D	Yes	**Pareto 2D**
Pie	pie2d multilevelpie	2D 4D+	Yes	**Pie 2D** **Multi-level Pie Chart**
Pie - 3D	pie3d	2D	No	**Pie 3D**
Pyramid	pyramid	2D	No	**Pyramid Chart**
Radar	radar	3D	No	**Radar Chart**

Display name	Chart name	Chart type	Default	Link
Scroll Area	scrollarea2d	3D	No	**Scroll Area**
Scroll Column	scrollcolumn2d	3D	No	**Scroll Column**
Scroll Line	scrollline2d	3D	No	**Scroll Line**
Stacked Area	stackedarea2d	3D	Yes	**Stacked Area 2D**
Stacked Bar	stackedbar3d	3D	Yes	**Stacked Bar 3D**
Stacked Column – 2D	stackedcolumn2d	3D	No	**Stacked Column 2D**
Stacked Column – 3D	stackedcolumn3d	3D	Yes	**Stacked Column 3D**
Spark Line	sparkline	2D	No	**Spark line**
Zoom Line	zoomline	3D	Yes	**Zoom Line**

5. Variables

5.1. Variable Values

The table below (table 5) contains an overview of all variables that can be used to customize a FusionInterface report. Variables that have a string as input have the following rules:

- If multiple values can be set (see below), they are split using a comma ','.
- If a value consist of multiple parts, they are split using a colon ':'.
- To use a comma in a string use [$comma$].
- To use a colon in a string use [$colon$].

If another variable is referenced here, it is written within dollar signs.

[] (black brackets)	Means that multiple values can be set. For example: [string], would have "string1, string2, string3" as a possible input.
: (colon)	Means that a variable consists of multiple parts. A variable with multiple values can also have multiple parts. For example "column:value,column:value".
[:] (grey brackets)	Means that a variable part is optional and not need filled in. Mind however that string**[:**string**]** **[:**string**]** needs to be filled in as follows: "string1::string2" (when variable 2 is ignored).
<>	Means that either one of these values should be used.
[] (purple brackets)	Means an int/double number within this range should be used.
true (orange)	Represents a Boolean.
column (MediumBlue)	is used for a name/label of a column.
string (DeepSkyBlue)	Represents a String.

0 (purple)	is used for an Integer or Double.
#FF0000 (green)	Represents a Hex Color.
(deprecated) (grey)	is used to include deprecated table settings. Please do not use these and if your current code does use deprecated settings, update the code to use the new settings.
date (red)	Represents a Date

5.2. Showing/hiding parts of the page

Name	Syntax	Description	Default	Example
showTitle	<true/false>	Shows the report title.	true	
displayVariables	<true/false>	Variables chosen by the user are displayed in the report below the title.	false	
noTabs	<true/false>	true: hide tab-buttons and display all tabs in one sequence page. false: use tabs	false	
showSettings	<true/false/ collapse/ noButton/ filterOnly>	true: show with button to show/hide. false: hide this entire part. collapse: hidden, but with button to show/hide and export. noButton: show, but hide show/hide and export buttons. columnNames: show names as well as values in the breadcrumbs.	true	
showBreadcrumbs	<true/false/ columnNames>		true	
showChart	<true/false/ collapse/ noButton>		true	
showDetailsTable	<true/false/ collapse/ noButton>		true	
showEntireTable	<true/false/ collapse/ noButton>		collapse	

Name	Syntax	Description	Default	Example
tableOnly	<true/false>	true: hide all other parts (settings table, etc.). If another variable from above is set, it will override this setting.	false	
chartOnly	<true/false>		false	

5.3. Grouping and sorting data

These variables can be set to automatically group and sort data. The easiest way to do this is to first do the grouping and sorting in the settings table in an active report. Then export the settings to clipboard using 'export current settings' button, and import them using the 'Import' button in the myInsight report properties.

Name	Syntax	Description	Default	Example
sortColumn (deprecated)	column:boolean	Use sortColumns instead	0: true	r_object_id:false
sortColumns	[column:boolean]	true: ascending false: descending	0: true	r_object_id:false ,clicks:true
groupColumn	column	The column to group on.	none **or** date column for history	
subGroupColumns	[column]	Divides the groups into subgroups based on the chosen columns. Columns after the first column split the group further and add one dimension of data.	none	
subGroupColumn (deprecated)	column	Use subGroupColumns instead.	none	

Name	Syntax	Description	Default	Example
subSubGroupcolumn (deprecated)	column	Use subGroupColumns instead.	none	
customGroupers	[column:<count/sum/ average/minimum/ maximum/distinct>]	Adds a grouper to a column.	none	document:count, reviews:sum, clicks:sum, clicks:minimum, clicks:maximum
seriesGroupColumn	column	The grouped series column, only applies to the scatterplot.	none	
keyColumn	column	The column used for the x-axis, overwritten by group column.	first date/ string column	
yColumn	column	y-Column for 2-dimensional charts.	none	
zColumn	column	z-Column for special chart types.	none	
ignoreColumns	[column]	Columns to ignore.	none	
filters	[column :comparator column]	Adds a filter to a column. • comparator: <, >, =, LIKE, etc. • value.	none	all:<4, stage_ number:<:4, total_doc_count :>:15

5.4. Chart specific variables

Name	Syntax	Description		Default	Example
drilldownColumns	[column [:action]]	The columns to automatically group on when the table or chart is clicked on (if no subgroup is set). **Action** can be set to the following: regroup: Default action. hideChart: Ungroup, and set the X-column to this. Also hide the chart and show the table. Should only be set last after the regroup columns. If a column does not exist, this action also happens.		none	
customColors	[color]	Set the colors to custom values. The colors will repeat if needed.	color: hexadecimal color (#-sign is optional) comparator: <, >, =, LIKE	none	#FF0000, #FF9933, #FF1234, #0000FF
conditionalColors	[<column/all> :<int/double/date> :color [:column]]	Apply colors to the chart based on criteria such as minimum number. It can be combined with customColors. See **Conditional colors** on page 36 for more info.		none	DF0101:<,100, FF9933:> :stage_number, ED15D4:>=0 ::num_docs

Name	Syntax	Description	Default	Example
onStartChartType	string	Name of any available chart.	**Column2D** or **Timeline** depending on whether there is a date field or not	
removeDefaultChartTypes	boolean	Remove all the chart types that can usually be chosen. Should be used in combination with **addChartTypes**.	false	
addChartTypes	[string [:<2D/3D/4D/Scatter>] [:int]]	• JavaScript chart name from (e.g. *canvasBgColor*) • **2D/3D/4D/Scatter**, depending on the FusionChart that is used. Only needed if the type is not one of the charts that are currently available (See **List of available charts** on page 15). • Label for chart selector. • Maximum numbers of records to show in chart.	none	column3d :2D:2D-column :300

Name	Syntax	Description	Default	Example
setChartProperties	[string :<string/int/ boolean/color>]	Set various native properties of the chart. • value • name *See* **List of available charts** *on page 15 for the links that to all possible values.*	none	cnvsBgClr :CC66FF, cnvsBrdrClr :33CC3, caption :Some Chart, showValues :true, captionFontSize :14
setTrendlineProperties	[string :<string/int /boolean /color]		none	startvalue:200, color:33CC3, displayvalue :target for 2015
customLegends	[string: [color]]	Replace the legend with a custom version. Automatically set with $customColors$.	none	Less than five :FF9900, five or more :FF0000
chartHeightPercentage	[0-100 int]	Chart height in proportion to the screen height.	80, or 50 in widget	
hideZeroesInChart	boolean	If true, hide numbers in the chart that are 0.	false	

Name	Syntax	Description	Default	Example
maxRecordsModifier	[0-1 double]	Affects the number of records shown in the **chart**. If there are many records, some are normally left out of the chart. The value for maxRecodsModifier affects when this happens. Percentage: 0 no maximum < 1 increases performance, decreases number of records in chart 1 unmodified > 1 increases maximum number of records in chart.	1	

5.5. Table specific variables

Name	Syntax	Description	Default	Example
d2Actions	string:string[:column][:column:column]	• event type: The D2 action to be executed when clicking on the link. For example D2_ACTION_CONTENT_VIEW. • The column containing the event link. • The column containing the ID's needed for the D2 action. This column will also be set to ignore. • Any other parameters can be added by the format :name:value, the field for that column is used as value.	none	D2_ACTION_CONTENT_VIEW:doc_count:r_object_id

Name	Syntax	Description	Default	Example
xcpActions	string:string[:column]	• Action name. This can be xCPSelectObject or xCPSendEvent1-5. Must match the name selected in xCP builder. • Column of the table that can be clicked on to perform the action. • Optional column to be used for adding the ID.	none	xCPSelectObject: object_name: r_object_id
smartviewActions	string:string[:column] [:column:column]	• event type: The Smarview action to be executed when clicking on the link. • The column containing the event link. • The column containing the ID's needed for the action. This column will also be set to ignore. • Any other parameters can be added by the format :name:value, the field for that column is used as value.	none	CUSTOM _APP _PARAM: doc_count: r_object_id
externalLinks	column:string [:column][:string]	Constructs an external link. • The column that can be clicked on. • First part of the URL, can be empty. • Optional. Variable part of the URL, this can point to a value from another column. • Optional. Last part of the URL.	none	name:http [$colon$]// domain[$colon $]8080/ da/component/ drl?objectId= :object_id: &versionLabel= CURRENT

Name	Syntax	Description	Default	Example
replaceWindowURL	column:string [:column][:string]	Similar to external links but in this case the current browser is being redirected to the specified URL in stead of a new window.	none	
recordsPerTablePage	int	int value, preferred values: {5, 10, 20, 50, 100, 200, 500, 1000}	100	
addPercentageColumns (deprecated)	[column]	Use tableCellRendering instead.	none	
tableCellStyling	<column/any_number /any_string> : <left /center/right>	• Aligns columns to the left, center or right. • Can be a specific column, or any number/string type column.	none	any_string:left, some_column: right
tableCellRendering	[column:<percentage /button/custom>]	• percentage: shows a percentage bar for numbers 0-100 • button: styles the cell like a button • custom: cell content can be overridden by a stylesheet calling FI using template addCellRenderingScript	none	any_string :percentage

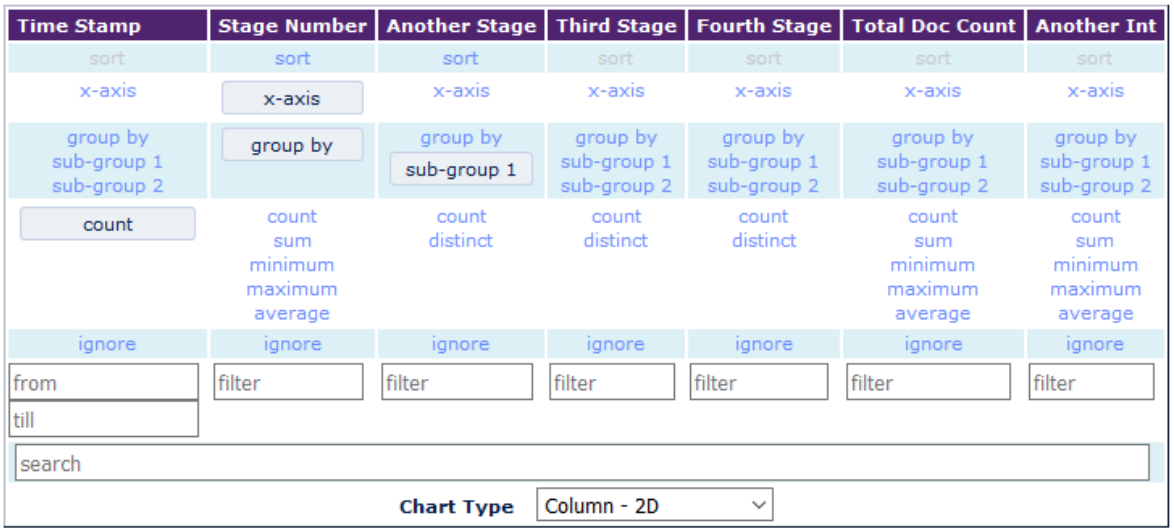
5.6. Other variables

Name	Syntax	Description	Default	Example
disableClickEvents	boolean	true: most click events in chart and table are disabled. Does not include links to external sites. false: clicks are enabled where possible. This affects clicks for groupers, filters, details and d2Actions.	false	
timeGrouper	year, month, day, hour or minute	When grouping on a time column, it will group the time per selected period.	day	
dropdownFilters	[column]	Add filters separately from the settings table. They will show the distinct values for the selected column. If there are more than 15 possible values, a range is shown instead.	none	
hideColumns	[column]	Fully hides the data in this column. This includes the settings table. The data can still be used for things like D2 event and grouping.	none	
translation	<de/en/es/fr/nl/ar>	Sets a reference to a native language support file. Example: Dutch to refer to **dutch.js**.	browser's default	

Name	Syntax	Description	Default	Example
refreshInterval	[double]:string	The report will reload automatically after a set time • double refresh interval in minutes, defaults to 5. • object id of the report.	none	10 :09000751800 e333b
varCombine	[string:string]	• Set a variable with a combination of 1 of more variables. See **Combine Variables** on page 40 for more information.	none	reportTitle :Docs grouped by [$timeGrouper $]

6. Tips and Tricks

6.1. Chart title



Number of 'Time Stamp' / 'Stage Number'(group), 'Another Stage'(sub-group)

The default chart title gets created by the following logic:

- To the right of the '/' sign are the group columns. Grouping is done around each unique field, for example each day. Groups can be divided into subgroups. An example of this would be to show data for each day per employee.
- To the left of the '/' sign are the groupers which gather the data for each group.

The default chart title can be overridden using **setChartProperties**. Add 'caption:Some Caption' to set the chart title to 'Some Caption'. To completely hide the chart title, set it to 'caption:'. To add a comma ',' or colon ':' in the chart title, use the text '[$comma$]' or '[$colon$]'.

6.2. Only show basic table

To only show the basic table in your output, copy the following lines and paste them into the variable import field:

$showTitle$ false
$showBreadcrumbs$ false
$tableOnly$ true
$varCombine$ $showDetailsTable$:nobuttons[$comma$] totals[$colon$]false

6.3. Drilldown columns

The drilldownColumns variable is designed to provide a more intuitive way of grouping and filtering for end users. This section will demonstrate how to use it. Note that settings table is shown here for clarity, but it can be hidden for end users.

The data in the following image shows users who like apples or pears for fruits, and broccoli or cauliflower for vegetable.

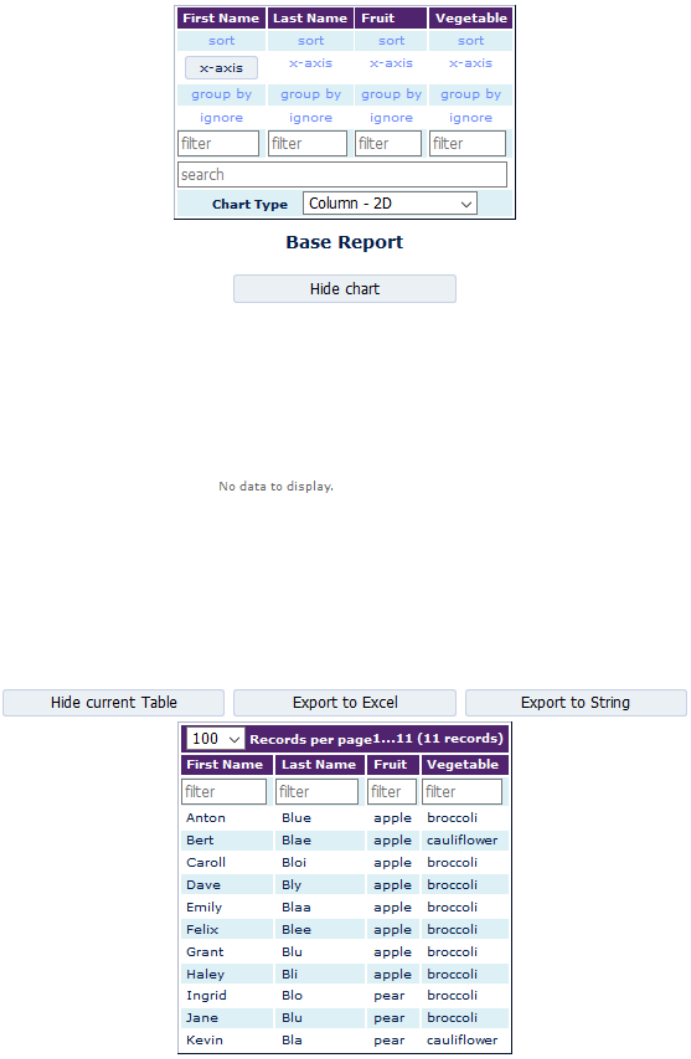


Figure 4: Drilldown colums - data overview

Suppose you would want an overview of who likes which fruit. Group the data on the fruit columns:

```
groupColumn = fruit
```

The result would be:

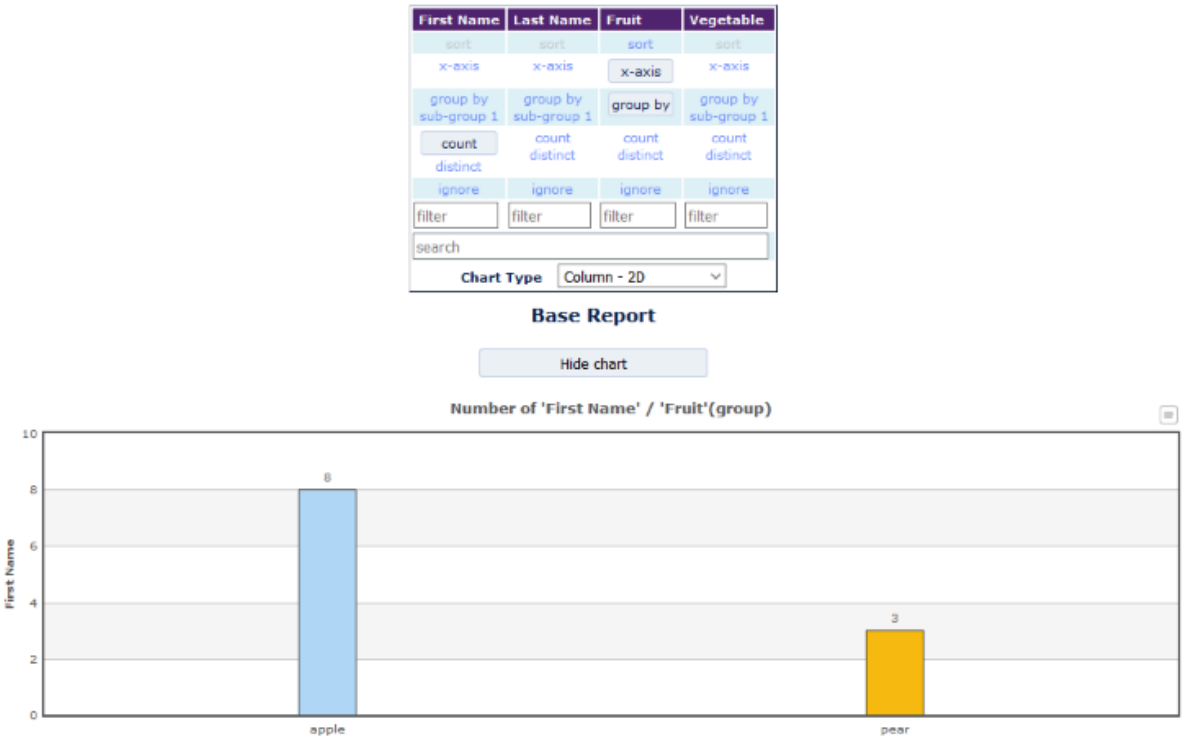


Figure 5: Drilldown columns - data grouped by fruit

Now when the user clicks on the apple column, a filter will be added on the fruit column to apples. Note that a breadcrumb is added to match the filter:

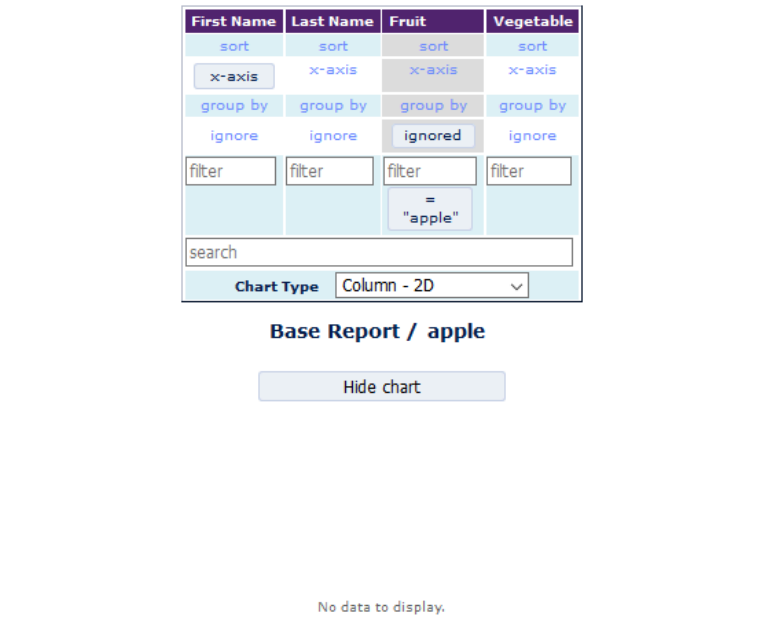


Figure 6: Drilldown columns - data is filtered by apple

The chart in is gone in **Figure 6** on page 34. A user could now group on vegetable to get an overview of vegetable preferences for the people who like apple. But that can be automated with the drilldownColumns variable:

```
drilldownColumns = vegetable
```

Now when the chart is clicked again, a second filter will be added (see breadcrumb):

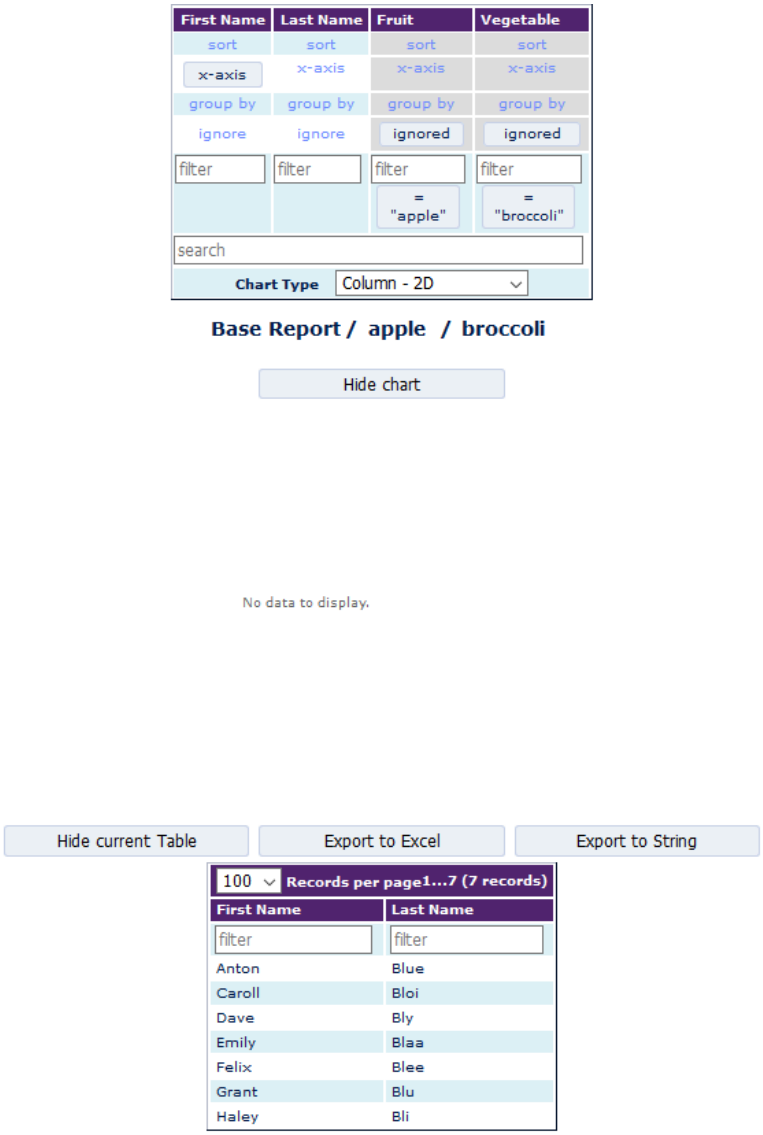


Figure 7: Drilldown columns - automatically filter with drillDownColumns

The table now shows an overview of the remaining users that like apples and broccoli. The chart is no longer useful, so it can be collapsed. This is done as follows:

```
drilldownColumns = vegetable, null:hideChart
```

Null is used to target no new column for regrouping, and hideChart is used to collapse the chart. hideChart automatically opens the table if it is collapsed.

6.4. Conditional colors

The conditional colors variable works in the following way: You create a filter for a specific range, and then you assign a color to this range. Multiple conditional colors can be added (split by comma). The color that applies first will be added.

The following arguments can be applied, split by a colon sign (:).

Name	Example
Filter Column Name	some_column
Filter value	integer>4
Color	ff0000
Display Column	integer_column
Extra filter Column	all
Extra filter value	=abc

To add a conditional color:

1. Open/run the report.

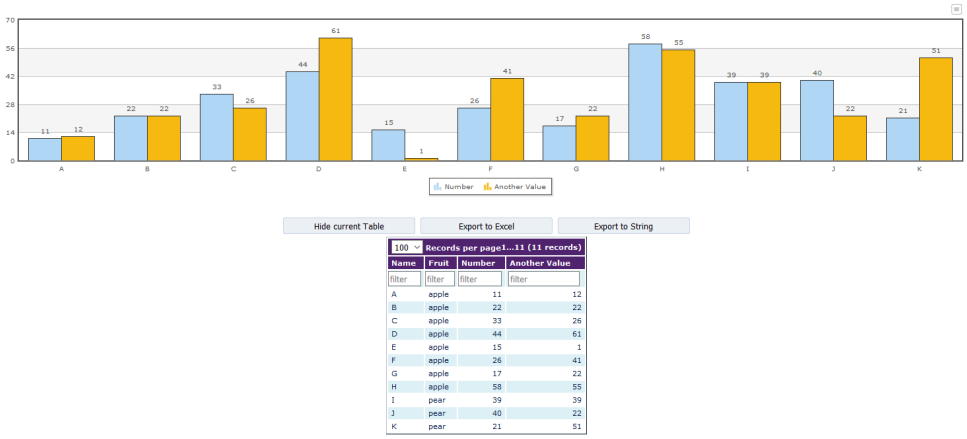
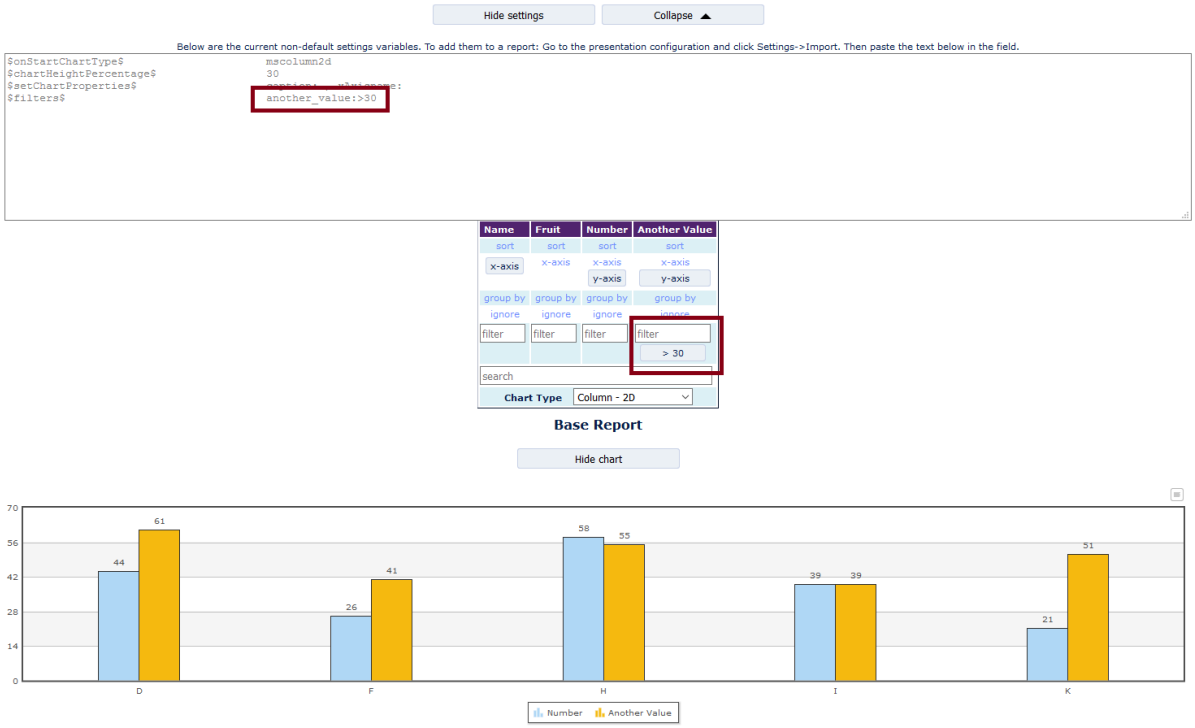


Figure 8: Report (example)

2. Add one or more filters via the settings or current table:
3. Click the "export current settings" button on top of your report.
4. Copy the line describing the filter(s), for instance to highlight where the field 'another_value' is greater than 10:

```
$filters$ another_value:>10
```



Note

Instead of a single column, the value "all" can also be entered. In this case all columns will be targeted.

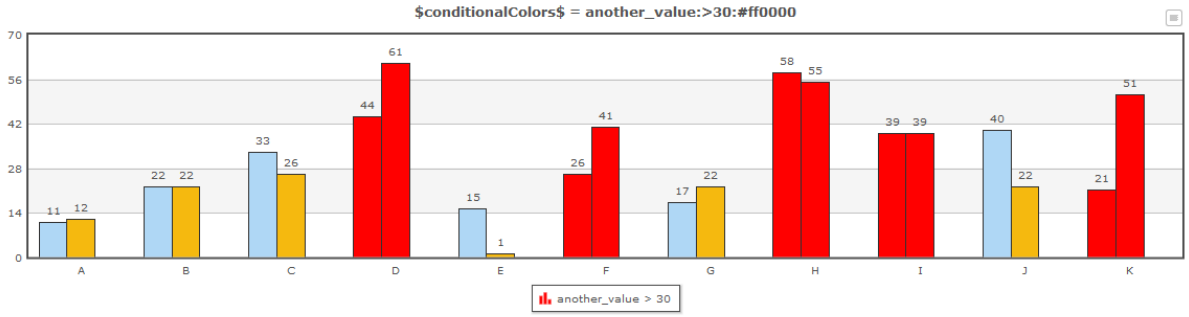
5. Add the filter string to a conditional variable, in the above example:

```
$conditonalColors$ another_value:>10
```

6. Add a colon (:) to the variable followed by a color hex code, the '#' sign is optional. The result would then be:

```
$conditionalColors$ another_value:>10:#ff0000
```

7. Run the report again and the color should be applied as you can see in this example:



Hide current Table | Export to Excel | Export to String

100 Records per page1...11 (11 records)

Name	Fruit	Number	Another Value
filter	filter	filter	filter
A	apple	11	12
B	apple	22	22
C	apple	33	26
D	apple	44	61
E	apple	15	1
F	apple	26	41
G	apple	17	22
H	apple	58	55
I	pear	39	39
J	pear	40	22
K	pear	21	51

Figure 9: Report with conditional colors (example)

8. The previous example has multiple fields ('Number' and 'Another Value'). It is possible to apply the conditional color to just one field by adding the column name in the conditional color variable. To only apply the color to the field 'Number' the filter would be:

```
$conditionalColors$ another_value:>10:ff0000:number
```

The resulting report looks like the following:

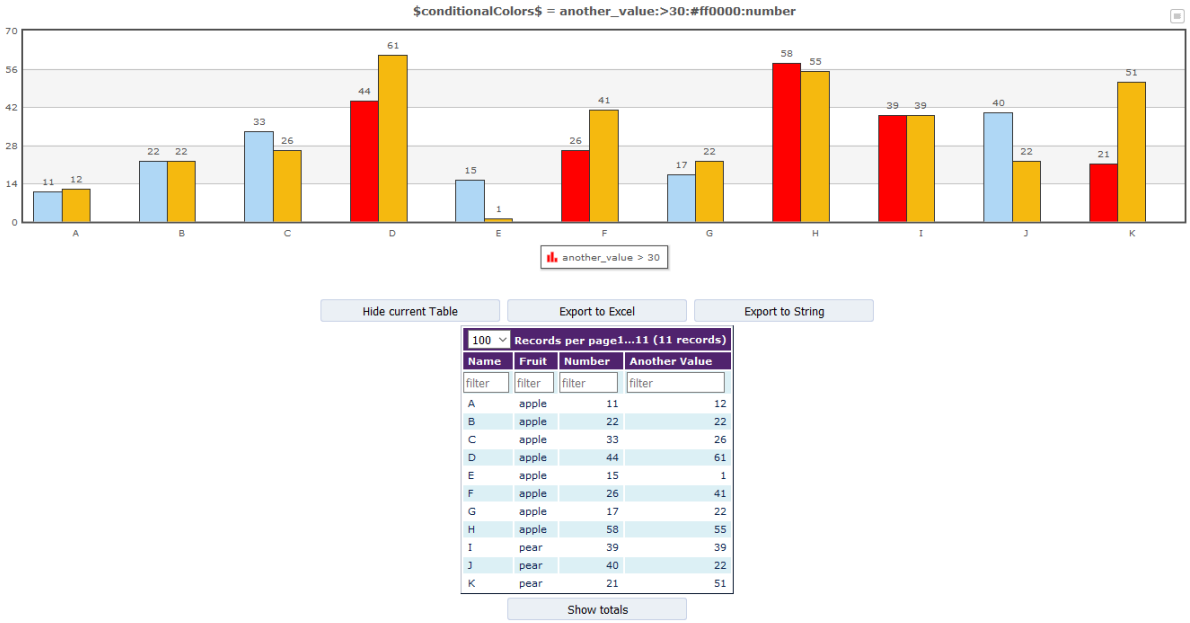


Figure 10: Report with conditional colors, single field highlighted (example)

> **Note** This step can work with data that is grouped once. The column that is used for the grouper is chosen.

9. Additional filters can be applied. For example, to only make fields red where 'Number' is greater than 10 and 'fruit' is 'apple', the result would be:

```
another_value:>30:#ff0000::fruit:=apple
```

and it would look like this:

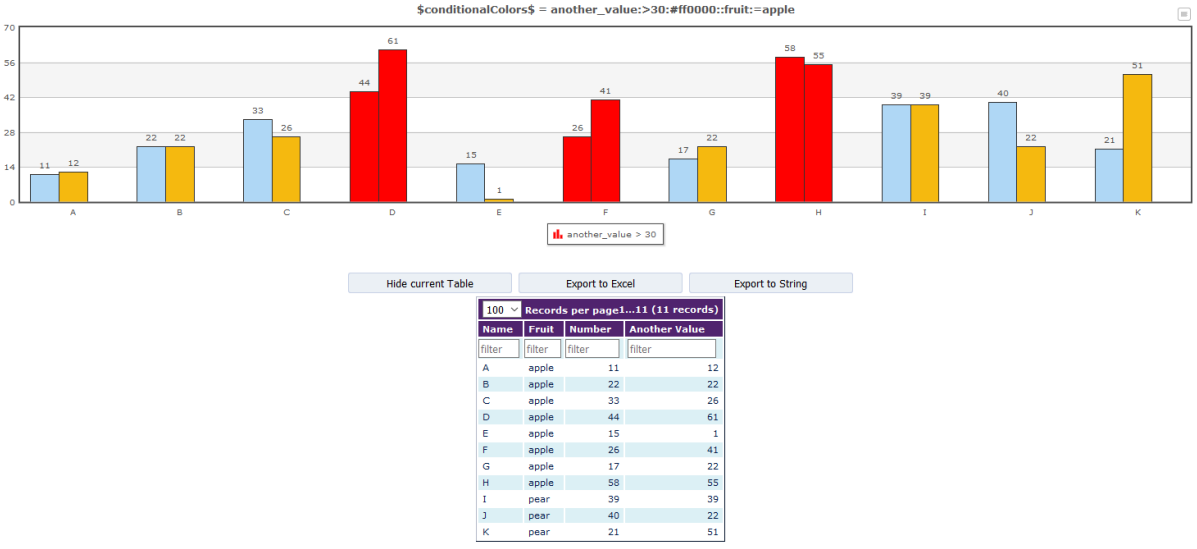


Figure 11: Report with conditional colors, only apple highlighted (example)

> **Note** This step only works for data that is not grouped with FusionInterface.

The layout of the variable looks as follows:

[
<column/*all*>
:comparator
[**:<***int*/double/date/*string*>]
:color
[**:column**]
]

> **Note** Note that there is a fifth column parameter. This parameter can be used if the color should be displayed on another column than the column we execute the filter on.

6.5. Combine Variables

It can happen that you need to use one variable multiple times in the same report. This can be a problem if the user needs to set this variable by hand. The varCombine variable allows you to freely re-use and combine variables.

This is added in the following way:

```
$VariableName$: VariableValue
```

It is possible to use a value of another variable. To do this add it within

```
[$ $]
```

Multiple variables can be set by splitting the values with comma. Comma's and colons can be used in a variable by using [$comma$] and [$colon$] respectively.

Example 1, include group column in report title:

```
$reportTitle$: Docs grouped by[$colon$] [$groupColumn$]
```

This will include the $groupColumn$ variable (if set) in the title. Only valid Fusion variables can be set. Variables from the report definition can be used as long as they are the same name as FI one and not overwritten.

Example 2, hide the totals and buttons in the details table:

```
$varCombine$ $showDetailsTable$:nobuttons[$comma$] totals[$colon
$]false
```

These are advanced settings that can normally not be set in the variable select screen. In this case the value of the $showDetailsTable$ variable will become:

```
nobuttons, totals:false
```

If the $showDetailsTable$ variable is already set, this will be ignored.

6.6. Other

- The number of records (rows * numerical columns) that the chart will show is usually limited to 300-1000 depending on chart type. When this number is exceeded, the stepsize in the increased. This means that if there are 3000 records in the table, only every 3rd row is shown. A warning will show at the bottom of the settings page.
- The number of rows that FusionInterface can handle is much more. Up to about 50.000. Neither of the previous actions affects the calculations used for grouping, summing etc.
- Performance of FusionInterface has been tested on recent versions of Firefox, Chrome, Internet Explorer and Edge. Because results differ between browsers, the default maximum in the chart are halved in IE and Edge to have similar loading times.

The number of records shown can be changed with the variable **$maxRecordModifier$**.

List of Tables

List of Figures

Index