Qlikview Set Analysis Guide

==============================

Set Analysis provides a method of defining groups (sets) of information that are independent of the current selections and/or can be created based on other sets (such as your current selections). Set analysis is a powerful tool for comparison analysis. For example, you can create a chart that displays this year's data versus last year's data, product 1 compared to products 2 & 3, or even the selected data versus the non-selected data.

Syntax

All set analysis expressions need to include an aggregation function such as sum, count, max, min, etc…

Set analysis syntax is broken down into three types: Identifiers, operators, and modifiers.

Identifiers are used to denote the record sets. Identifiers can loosely be compared to a FROM statement in SQL, but are being used at an expression level. Generally, you can think of it as what pond you will be fishing in for the rest of the equation. Identifiers must begin and end with curly brackets {}.

Sum({*set identifier*} Sales)

1 represents all the records in the document (fishing in the whole pond, ignoring any current selections)

$ represents only the records selected (fishing in only the section of the pond you've roped off based on your other selections)

1-$ represents the inverse of the records selected (fishing in everything but the roped off sections of the pond)

Example of sum of all sales (ignoring current selections)
Sum({1}Sales)

Other identifiers can be used outside of the most common ones listed above. Bookmarks, alternate states, and backwards/forward selections are also possibilities.

$1 is the equivalent of the back button

$_1 is the equivalent of the forward button

If you had a bookmark with ID BM02, your expression would look as the following

Sum ({BM02} Sales)

Operators are used to "operate" on one or more identifiers. Operators can be thought of as a set operators in SQL, but they are used on the identifier at the expression level.
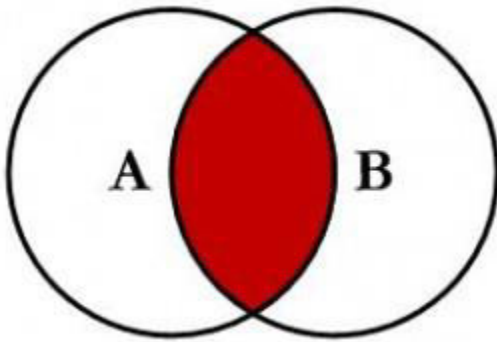+ returns the set of records that belong to any of the two sets (similar to a UNION in SQL)
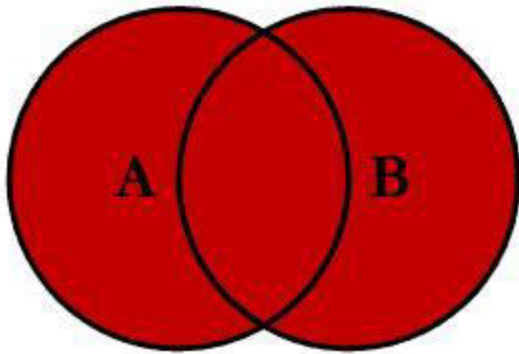

– returns the set of records that belong to the first identifier while excluding any overlapping data in the second (similar to a MINUS or EXCEPT in SQL)
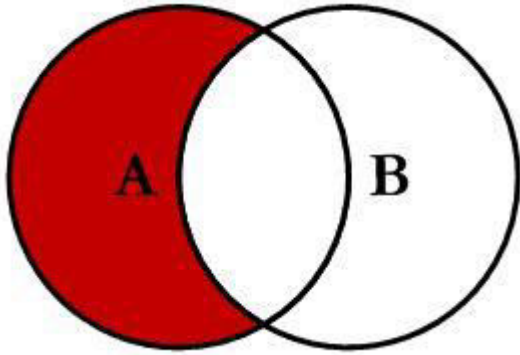
* returns the set of records that only belong to the overlapping data from the first and second sets (similar to an INTERSECT in SQL)

/ returns the set of records that belong to either but not both of the sets (similar to an XOR or Symmetric Difference)
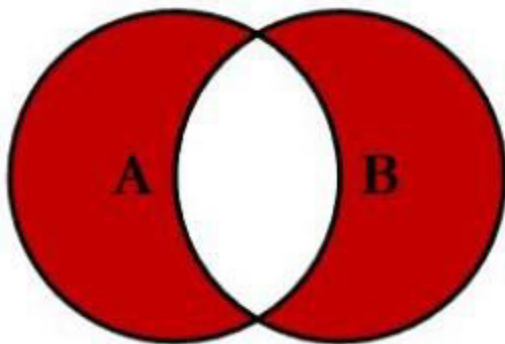
Examples of operators:

Sum({1-$} Sales) is sum of all sales excluding the current selections

Sum({$*BM01}Sales) is sum of sales only were current selections and the bookmark named BM01 overlap

Modifiers are used to make additional and specific alterations to the expression and the set of records being used. Modifiers are similar to a WHERE statement in SQL. It is used to modify the set of records used in the expression based on alterations to field-specific comparisons. Modifiers use one or more field names, each followed by selection(s) that can be made in the field. They begin and end with angle brackets <> but are still contained within the curly brackets of the entire set analysis portion of the expression.
Sum({*set identifier <set modifiers>* } Sales)

Example of sum of all sales (ignoring current selections) for the year of 2007
Sum ({1 <Year = {2007}> } Sales) *–note values need to be in curly brackets*

Example of sum of all sales (ignoring current selections) for the year of 2007 or 2008 and the country is USA
Sum ({1 <Year = {2007, 2008}, Country = {'USA'}> } Sales) *–note character values need single quotes*

Example of sum of sales (affected by current selections) only returning the set of records where OrderDate = DeliveryDate and the region is Ohio or Pennsylvania
Sum({ $ < OrderDate = DeliveryDate, Region = {'OH', 'PA'} > } Sales)

Searches can be used in set analysis, but they must be defined by double quotes.

Example of sum of sales (affected by current selections, using the inverse of selections) for the year of 20-something
Sum ({1-$ <Year = {"20*"}> } Sales

To omit selections made against fields, you can leave the modifier empty

Example of sum of sales (affected by current selections) except for any values selected in the field of Year. Selection made against Year are ignored in this expression.

Sum ({$ <Year = > } Sales)