# Calculated column vs Measure

## Option 1

When you insert Columns or Measures using the **Modeling** tab, they are assigned to whichever *table is currently selected*, or the *first table in the field list* by default.

* Measures can be reassigned to new “Home” tables (under the “*Properties*” options in the **Modeling** tab), but the Option 2 allows you to be more deliberate about placing them

## Option 2

* Right-click within the table (in the Data view) or the Field List (in either the Data or Report view)

## Option 3

* New quick measure
  + Quick measures are pre-built formula templates that allow you to drag and drop fields, rather than write DAX from scratch
* Note: Assigning measures to specific tables doesn’t have any impact on functionality – it’s just a way to keep them organized.

# Implicit vs Explicit Measures

## Implicit Measures

Only accessible within the specific visualization in which it was created, and cannot be referenced elsewhere

## Explicit Measures

Can be used anywhere in the report and referenced within other DAX calculations to create “measure trees”

Each measure value in a report is ***like an island***, and calculates according to its own filter context (even *Totals* and *Grand Totals*)

Measure Name = function name ( table name [ column name ] )

Calculated columns do not always use functions, but measures always use functions

* In a Calculated Column, **=Transactions[quantity]** returns the value from the quantity column in each row (since it evaluates one row at a time)
* In a Measure, **=Transactions[quantity]** will return an error since PowerBI doesn’t know how to translate that as a single value (you need some sort of aggregation)

For Column references, use the fully qualified name (i.e. Table[Column])

For measure references, use the measure name (i.e. [Measure])

*Referenced*

*Table Name*

**Total Quantity**: =SUM(*Transactions*[***quantity***])

**Measure Name** Function Name ***Referenced Column name***

# Related()

Returns related values in each row of a table based on relationships with other tables

=RELATED(ColumnName)

e.g. ColumnName

Product\_Lookup(ProductName)

Territory\_Lookup(Country)

RELATED works almost exactly like a VLOOKUP function – it uses the relationship between tables (defined by primary key and foreign keys) to pull values from one table into a new column of another

Since this function requires row context, it can only be used as a Calculated Column or as part of an iterator function that cycles through all the rows in a table (FILTER, SUMX, MAXX, etc.)

Avoid using RELATED to create redundant calculated columns unless you absolutely need them, since those extra columns increase file size. Instead, use RELATED within a measure like FILTER or SUMX.

COUNT() – Counts the number of cells in a column that contains numbers

COUNTA() – Counts the number of non-empty cells in a column (numerical and non-numerical)

DISTINCTCOUNT() – Counts the number of distinct or unique values in a column

COUNTROWS() – Counts the number of rows in the specified table or a table defined by an expression

# CALCULATE()

=CALCULATE(Expression,[Filter1],[Filter2],..)

Expression should contain name of an existing measure or a DAX formula for a valid measure

e.g.

[Total Orders]

SUM(Returns\_Data[ReturnQuantity])

[Filter1][Filter2]…

List of simple Boolean (True/False) filter expressions

(note: these require simple, fixed values; you cannot create filters based on measures or reference measures)

e.g.

Territory\_Lookup[Country] = “USA”

Calendar[Year] > 1988

CALCULATE works just like SUMIF or COUNTIF in excel, except it can evaluate measures based on ANY sort of calculation (not just a sum, count, etc.); it may help to think of it like **CALCULATEIF**

CALCULATE ***modifies*** and ***overrules*** any competing filter contexts

Step1

Filter context is detected and applied

Step 1.5 – Filters modified by CALCULATE

Step 2 – Filters flow “downstream” to all related tables

Step 3 – Measure formula evaluates against the filtered table

# ALL()

Returns all rows in a table or all values in a column, ignoring any filters that have been applied

Does not return a single number, returns a table or a subset of a table. Almost never used by itself, but is used as a component of other functions, most commonly CALCULATE.

=ALL(Table or ColumnName, [ColumnName1],[ColumnName2],…)

The table or column that you want to clear filters on

e.g.

Transactions

Products[ProductCategory]

[ColumnName1],[ColumnName2],…

List of columns that you want to clear filters on (optional)

**Notes**: If your first parameter is a table, you can’t specify additional columns. All columns must include the table name, and come from the same table.

e.g.

Customer\_Lookup[CustomerCity], Customer\_Lookup[CustomerCountry)

Products[ProductName]

Instead of adding filter context, ALL **removes it**. This is often used when you need unfiltered values that won’t react to changes in filter context (e.g. **% of Total** where the denominator needs to remain fixed)