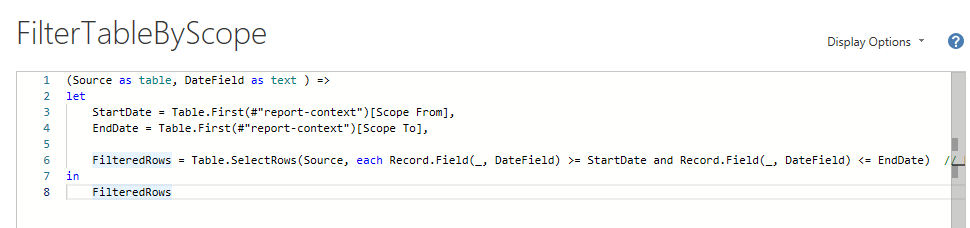
List of Steps To Generate Report

1. Import all data tables (sources)
2. Create two Parameters:
   1. Area
      1. Type = Text
      2. Suggested Values = List of available Values [OR, EUFS]
      3. Select Current Value
   2. Type
      1. Type = Text
      2. Suggested Values = List of available Values [Annual, Mid-Year]
      3. Select Current Value
3. Filter Prioritization Exercise
   1. Get only active records
   2. Filter By Area (filter based on Parameter)
   3. Filter By Type (filter based on Parameter)
   4. Keep First Row
4. In PowerQuery, create a new folder called **Functions**
   1. Add a new query (call it **GetDates**) and add the following query <https://devinknightsql.com/2015/06/16/creating-a-date-dimension-with-power-query/>
   2. Add a new query (call it **FilterTableByScope**) and add the following query. This will help filter all the source tables based on the scope dates provided in the Prioritization Exercise.



1. In PowerQuery, create a new folder called **Source Data**
   1. Create a DateDim table using **GetDate** function  
      = GetDates(Table.First(#"report-context")[Scope From], Table.First(#"report-context")[Scope To])
   2. Move all imported tables into this folder
2. In PowerQuery, create a new folder called **Filtered Data**
   1. For each relevant table in the source data (i.e. table that needs to be filtered by date), create a filtered table using the **FilterTableByScope** function  
      = FilterTableByScope(penalties, "PenaltyDate")
3. Close and Apply
4. In Modelling View, link the tables and establish relationships. In our sample report, CompanyID was the common denominator.
5. To compute the number of **change requests** submitted **by the account** within the scope can be done by creating a measure on the filtered table. Create a new measure   
   CountMeasure = COUNTROWS(*filteredTableName*)

NOTE: Measures work based on row-context and will work only **when** the relationship is established

1. To add a weight based on the number of **change requests**, add another measure on the **same** filtered table,   
   WeightMeasure = IF(*CountMeasure* > 0, 10, 1)
2. To sum up all the weights, add a new measure on the **account** table, and sum all the **weight measure**

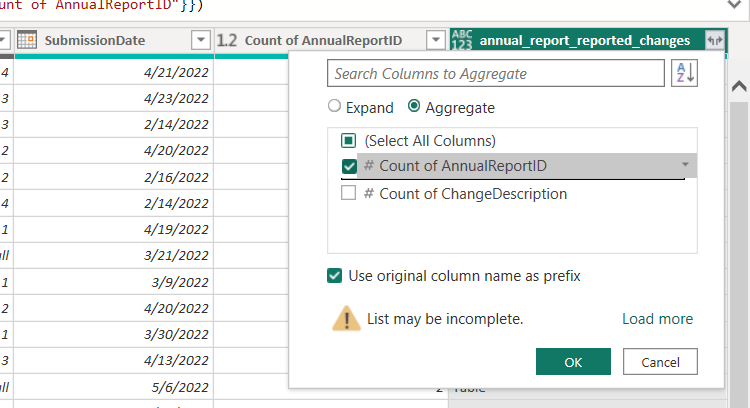
## Assigning Weight Based on Latest Tier Change

1. [HOW TO] – in order to get the latest tier/rank change by account [PowerQuery]
   1. [Optional] Filter tier/rank history table to get a list of all tier/rank changes within the scope defined reporting context (i.e Prioritization Exercise)
   2. Duplicate the filtered table
   3. In the duplicate, do GROUP BY
      1. Company ID
      2. New Column = Latest Change Date
      3. Operation = MAX
      4. Column = Created On/Rank Change Date
   4. Merge Query as New between the filtered table and GROUP BY table. INNER JOIN ON Company ID and [Latest Change Date, Created On/Rank Change Date]
2. [PowerQuery] - Add a conditional column assigning the correct weight based on the change type (e.g. Low to High, set weight as 3, otherwise, set weight as 1)

## Conditional Weights Based On Count/Measure

1. Create a measure using CALCULATE(  
    COUNTROW (*TableName*),  
    USERELATIONSHIP(*col1, col2*)  
   )
2. Create a **conditional column in DAX**
3. Add the conditional column as MAX() to the Report View

## Comparing Number of Tickets Submitted v/s Reported In Annual Report

1. In PowerQuery do a merge of query between **filtered annual report** and tickets submitted in annual report (source table)
2. Expand column as **aggregate**
3. Close and Apply.  
   In Report Builder, add a column in **filtered annual report** that compares the aggregated column created above and the measure storing the number of tickets  
     
   ARMisinfoTicketsWeight = IF(filtered\_annual\_reports[NumARTickets] <> [NumTickets], 5, 0)

## Summing Across Columns (AR Weights)

1. Go to **Power Query**
2. Duplicate **filtered annual report**
3. Remove all unnecessary columns
4. Select Annual Report ID, and **Unpivot Other Columns**
5. Save query as **Unpivoted Filtered Annual Reports**
6. Merge **filtered annual report** and **Unpivoted Filtered Annual Reports (LEFT JOIN)**
7. Expand columns as an **Aggregate** (SUM of values)
8. Rename column.
9. Close and Apply. Include the new column in report.

## Automatic Inclusion of Late Submissions

1. Go to **Power Query**
2. Merge **filtered annual report** with **annual report cycles**
3. Expand column to add **Annual Report Cycle (End Date)**
4. Create a **conditional column** called **Late Submission**.
5. If **Submission Date** is blank or after Annual Report Cycle (End Date), then set column text as **Include**, otherwise, set it to **Default**.   
   