*November 2024*

*From a Database to Power BI Overview*

Step 1: Access the database in a relational database management system (RDBMS) like SQLite or Azure Data Studio.

Step 2: Explore the tables to get a feel for content and key relationships, notably keys.

Step 3: Determine what will be presented in Power BI. This will help to decipher the view.

Step 4: Create a view using SQL including ‘Last Refresh Date’ and save it to the database. Permission may be needed from the database administrator, as not everyone has access to a database.

Note: From Step 5 onward, I will describe my experience with the RDBMS SQLite.

Step 5: Export the view as a Text/CSV file to the hard drive.

Step 6: Open a blank report in Power BI.

Step 7: Get data (choose the Text/CSV option) from the location on the hard drive.

Step 8: Within Power Query Editor, rename this staging query with the word ‘Source’ in the name and disable its load.

Step 9: Reference this staging query any number of times to create unique fact and dimension tables. Note: there can be more than one fact or dimension table.

Step 10: Select ‘New Source’ (Home) 🡪 ‘Blank Query’ 🡪 ‘Advanced Editor’ (View). Delete the pre-existing M code and write fresh code to capture the essence of the date dimension table. See the end of this document for the M code.

Step 11: Reference the staging query to create the ‘Last Refresh Date’ query.

Step 12: Close & apply the fact, dimension and last refresh date queries to Power BI.

*Within Power Query Editor*

Step 9a: Rename the referenced dimension queries with the prefix Dim (i.e., Dim\_Xxx) and the referenced fact query(ies) with the prefix Fact (i.e., Fact\_Yyy).

Step 9b: In each referenced query, ‘Choose Columns’ to select only those columns to be included in that table.

Step 9c: ‘Remove Duplicates’ in dimension tables where applicable.

Step 9d: Modify data types.

Step 9e: Sort rows where applicable.

Step 9f: Perform any other data transformations in Power Query where applicable.

Step 11a: After the ‘Remove Duplicates’ applied step, replace the section of the formula within the curly braces with 0 (a zero) to assign a floating date reference point.

Step 11b: In the query pane, right click and ‘Drill Down’ on the single date.

Step 11c: Rename the final applied step to ‘Drill Down to Date’.

*Within Power BI*

Step 13: Within the model view, ensure that the sensed primary and foreign key relationships are accurate. Note however that the ‘Last Refresh Date’ table does not have a relationship with any other table. It can float freely.

Step 14: Within the model view, hide the foreign key of the many side of the relationships.

Step 15: Within the model view, hide the source column of any explicit (DAX) measure.

Step 16: Within the report view, select ‘Date’ in the date dimension table, and under ‘Table tools’ ‘Mark as date table’ choosing the column ‘Date’.

Step 17: Within the report view, select the month name column in the date dimension table, and under ‘Column tools’ ‘Sort’ by the month number column.

Step 18: Within the report view, select the day name column in the date dimension table, and under ‘Column tools’ ‘Sort’ by the day of the week number column.

Step 19: If a map visual is planned, then a column should be designated as a location. Within the report view, choose that column, and under ‘Column tools’ select the correct ‘Data category’ from the drop- down menu.

From Step 10 - Date dimension table M code.

Modify the start and end dates to satisfy a specific dataset.

Copy and paste into the ‘Advanced Editor’

let

StartDate = #date(2019,01,01),

EndDate = #date(2021,12,31),

Source = {Number.From(StartDate)..Number.From(EndDate)},

#"Converted to Table" = Table.FromList(Source, Splitter.SplitByNothing(), null, null, ExtraValues.Error),

#"Renamed Columns" = Table.RenameColumns(#"Converted to Table",{{"Column1", "Date"}}),

#"Changed Type" = Table.TransformColumnTypes(#"Renamed Columns",{{"Date", type date}}),

#"Inserted Year" = Table.AddColumn(#"Changed Type", "Year", each Date.Year([Date]), Int64.Type),

#"Inserted Quarter" = Table.AddColumn(#"Inserted Year", "Quarter", each Date.QuarterOfYear([Date]), Int64.Type),

#"Added QuarterID" = Table.AddColumn(#"Inserted Quarter", "QuarterID", each Text.From([Year]) & "Q" & Text.From([Quarter]), type text),

#"Inserted Month" = Table.AddColumn(#"Added QuarterID", "MonthNum", each Date.Month([Date]), Int64.Type),

#"Added MonthID" = Table.AddColumn(#"Inserted Month", "MonthID", each Text.From([Year]) & Text.PadStart(Text.From([MonthNum]), 2, "0"), type text),

#"Inserted MonthName" = Table.AddColumn(#"Added MonthID", "MonthName", each Text.Start(Date.MonthName([Date]), 3), type text),

#"Inserted DayOfMonth" = Table.AddColumn(#"Inserted MonthName", "DayOfMonth", each Date.Day([Date]), Int64.Type),

#"Inserted DayOfWeek" = Table.AddColumn(#"Inserted DayOfMonth", "DayofWeek", each Date.DayOfWeek([Date]), Int64.Type),

#"Inserted DayName" = Table.AddColumn(#"Inserted DayOfWeek", "DayName", each Date.DayOfWeekName([Date]), type text)

in

#"Inserted DayName