Nieuwe lege query aanmaken,  
advanced editor openen en deze tekst inplakken.

let

StartDate = #date(2021, 1, 1),

EndDate = Date.AddDays(Date.EndOfYear(DateTime.Date(DateTime.FixedLocalNow())),365) /\*was "#date(2017, 1, 1)" Updated on 201802027: hard Coded End of Year caused some formulas to break, switching to dynamic date\*/,

//Used for 'Offset' Column calculations, you may Hard code CurrentDate for testing e.g. #date(2017,9,1)

CurrentDate = DateTime.Date(DateTime.FixedLocalNow()),

// Specify the last month in your Fiscal Year, e.g. if June is the last month of your Fiscal Year, specify 6

FiscalYearEndMonth = 1,

#"==SET PARAMETERS ABOVE==" = 1,

#"==Build Date Column==" = #"==SET PARAMETERS ABOVE==",

ListDates = List.Dates(StartDate, Number.From(EndDate - StartDate)+1, #duration(1,0,0,0)),

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

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

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

#"==Add Calendar Columns==" = #"Changed Type to Date",

#"Added Calendar MonthNum" = Table.AddColumn(#"==Add Calendar Columns==", "MonthNum", each Date.Month([Date]), Int64.Type),

#"Added Month Name" = Table.AddColumn(#"Added Calendar MonthNum", "Month", each Text.Start(Date.MonthName([Date]),3), type text),

#"Added Month Name Long" = Table.AddColumn(#"Added Month Name", "MonthLong", each Date.MonthName([Date]), type text),

#"Added Calendar Quarter" = Table.AddColumn(#"Added Month Name Long", "Quarter", each "Q" & Text.From(Date.QuarterOfYear([Date]))),

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

#"==Add Fiscal Calendar Columns==" = #"Added Calendar Year",

#"Added FiscalMonthNum" = Table.AddColumn(#"==Add Fiscal Calendar Columns==", "FiscalMonthNum", each if [MonthNum] > FiscalYearEndMonth

then [MonthNum] - FiscalYearEndMonth

else [MonthNum] + (12 - FiscalYearEndMonth), type number),

#"Added FiscalMonth Name" = Table.AddColumn(#"Added FiscalMonthNum", "FiscalMonth", each [Month]),

#"Added FiscalMonth Name Long" = Table.AddColumn(#"Added FiscalMonth Name", "FiscalMonthLong", each [MonthLong]),

#"Added FiscalQuarter" = Table.AddColumn(#"Added FiscalMonth Name Long", "FiscalQuarter", each "FQ" & Text.From(Number.RoundUp([FiscalMonthNum] / 3,0))),

#"Added FiscalYear" = Table.AddColumn(#"Added FiscalQuarter", "FiscalYear", each "FY" &

Text.End(

Text.From(

if [MonthNum] > FiscalYearEndMonth

then [Year] + 1

else [Year]

)

, 2

)),

#"==Add Calendar Date Offset Columns==" = #"Added FiscalYear",

// Can be used to for example to show the past 3 months(CurMonthOffset = 0, -1, -2)

#"Added CurMonthOffset" = Table.AddColumn(#"==Add Calendar Date Offset Columns==", "CurMonthOffset", each ( Date.Year([Date]) - Date.Year(CurrentDate) ) \* 12

+ Date.Month([Date]) - Date.Month(CurrentDate), Int64.Type),

// Can be used to for example to show the past 3 quarters (CurQuarterOffset = 0, -1, -2)

#"Added CurQuarterOffset" = Table.AddColumn(#"Added CurMonthOffset", "CurQuarterOffset", each /\*Year Difference\*/

( Date.Year([Date]) - Date.Year(CurrentDate) )\*4

/\*Quarter Difference\*/

+ Number.RoundUp(Date.Month([Date]) / 3)

- Number.RoundUp(Date.Month(CurrentDate) / 3),

Int64.Type),

// Can be used to for example to show the past 3 years (CurYearOffset = 0, -1, -2)

#"Added CurYearOffset" = Table.AddColumn(#"Added CurQuarterOffset", "CurYearOffset", each Date.Year([Date]) - Date.Year(CurrentDate), Int64.Type),

// Can be used to for example filter out all future dates

#"Added FutureDate Flag" = Table.AddColumn(#"Added CurYearOffset", "FutureDate", each if [Date] > CurrentDate then "Future" else "Past" ),

// FiscalYearOffset is the only Offset that is different.

// FiscalQuarterOffset = is same as CurQuarterOffset

// FiscalMonthOffset = is same as CurMonthOffset

#"==Add FiscalYearOffset==" = #"Added FutureDate Flag",

#"Filtered Rows to CurrentDate" = Table.SelectRows(#"==Add FiscalYearOffset==", each ([Date] = CurrentDate)),

CurrentFiscalYear = #"Filtered Rows to CurrentDate"{0}[FiscalYear],

#"Continue...Orig Table" = #"==Add FiscalYearOffset==",

#"Added CurFiscalYearOffset" = Table.AddColumn(#"Continue...Orig Table", "CurFiscalYearOffset", each Number.From(Text.Range([FiscalYear],2,2)) -

Number.From(Text.Range(CurrentFiscalYear,2,2))

/\*Extract the numerical portion, e.g. FY18 = 18\*/),

#"==Add General Columns==" = #"Added CurFiscalYearOffset",

// Used as 'Sort by Column' for MonthYear columns

#"Added MonthYearNum" = Table.AddColumn(#"==Add General Columns==", "MonthYearNum", each [Year]\*100 + [MonthNum] /\*e.g. Sep-2016 would become 201609\*/, Int64.Type),

#"Added MonthYear" = Table.AddColumn(#"Added MonthYearNum", "MonthYear", each [Month] & "-" & Text.End(Text.From([Year]),2)),

#"Added MonthYearLong" = Table.AddColumn(#"Added MonthYear", "MonthYearLong", each [Month] & "-" & Text.From([Year])),

#"Added WeekdayNum" = Table.AddColumn(#"Added MonthYearLong", "WeekdayNum", each Date.DayOfWeek([Date]), Int64.Type),

#"Added Weekday Name" = Table.AddColumn(#"Added WeekdayNum", "Weekday", each Text.Start(Date.DayOfWeekName([Date]),3), type text),

#"Added WeekdayWeekend" = Table.AddColumn(#"Added Weekday Name", "WeekdayWeekend", each if [WeekdayNum] = 0 or [WeekdayNum] = 6

then "Weekend"

else "Weekday"),

#"==Improve Ultimate Table" = #"Added WeekdayWeekend",

#"----Add WeekSequenceNum----" = #"==Improve Ultimate Table",

#"Filtered Rows Sundays Only (Start of Week)" = Table.SelectRows(#"----Add WeekSequenceNum----", each ([WeekdayNum] = 0)),

#"Added Index WeekSequenceNum" = Table.AddIndexColumn(#"Filtered Rows Sundays Only (Start of Week)", "WeekSequenceNum", 2, 1),

#"Merged Queries Ultimate Table to WeekSequenceNum" = Table.NestedJoin(#"==Improve Ultimate Table",{"Date"},#"Added Index WeekSequenceNum",{"Date"},"Added Index WeekNum",JoinKind.LeftOuter),

#"Expanded Added Index WeekNum" = Table.ExpandTableColumn(#"Merged Queries Ultimate Table to WeekSequenceNum", "Added Index WeekNum", {"WeekSequenceNum"}, {"WeekSequenceNum"}),

// somehow it ends up being unsorted after Expand Column, should not matter for the end table, but makes it harder to debug and check everything is correct. Thus sorting it.

#"ReSorted Rows by Date" = Table.Sort(#"Expanded Added Index WeekNum",{{"Date", Order.Ascending}}),

#"Filled Down WeekSequenceNum" = Table.FillDown(#"ReSorted Rows by Date",{"WeekSequenceNum"}),

#"Replaced Value WeekSequenceNum null with 1" = Table.ReplaceValue(#"Filled Down WeekSequenceNum",null,1,Replacer.ReplaceValue,{"WeekSequenceNum"}),

#"Inserted Start of Week (WeekDate)" = Table.AddColumn(#"Replaced Value WeekSequenceNum null with 1", "WeekDate", each Date.StartOfWeek([Date]), type date),

#"Inserted Week of Year" = Table.AddColumn(#"Inserted Start of Week (WeekDate)", "Week of Year", each Date.WeekOfYear([Date]), Int64.Type),

#"----WeekSequenceNum Complete----" = #"Inserted Week of Year",

Current\_WeekSequenceNum = #"----WeekSequenceNum Complete----"{[Date = CurrentDate]}?[WeekSequenceNum],

#"Added Custom CurWeekOffset" = Table.AddColumn(#"----WeekSequenceNum Complete----", "CurWeekOffset", each [WeekSequenceNum] - Current\_WeekSequenceNum, Int64.Type),

// Adding a DayofYear 1 to 365

// And YTD, QTD, MTD Columns (can help with showing YTD Numbers across multiple years)

#"==Updates 2019-Feb DayofYear and YTD QTD MTD Columns" = #"Added Custom CurWeekOffset",

#"Inserted Day of Year" = Table.AddColumn(#"==Updates 2019-Feb DayofYear and YTD QTD MTD Columns", "Day of Year", each Date.DayOfYear([Date]), Int64.Type),

#"Added Flag\_YTD" = Table.AddColumn(#"Inserted Day of Year", "Flag\_YTD", each if Date.DayOfYear([Date]) <= Date.DayOfYear(CurrentDate)

then "YTD"

else null),

#"Added Flag\_MTD" = Table.AddColumn(#"Added Flag\_YTD", "Flag\_MTD", each if Date.Day([Date]) <= Date.Day(CurrentDate)

then "MTD"

else null),

#"Added Flag\_QTD" = Table.AddColumn(#"Added Flag\_MTD", "Flag\_QTD", each //Compare Month Number in Quarter (1,2,3) for [Date] and CurrentDate

if Number.Mod(Date.Month([Date])-1, 3) + 1

<= Number.Mod(Date.Month(CurrentDate)-1, 3) + 1

then "QTD"

else null),

#"==Update 2019-Mar CurrentDatOffset" = #"Added Flag\_QTD",

#"Added CurrentDayOffset" = Table.AddColumn(#"==Update 2019-Mar CurrentDatOffset", "CurrentDayOffset", each [Date] - CurrentDate),

#"Changed Type1" = Table.TransformColumnTypes(#"Added CurrentDayOffset",{{"CurrentDayOffset", Int64.Type}})

in

#"Changed Type1"