|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | let CreateDateTable = (StartDate as date, EndDate as date, optional Culture as nullable text) as table =>  let  DayCount = Duration.Days(Duration.From(EndDate - StartDate)),  Source = List.Dates(StartDate,DayCount,#duration(1,0,0,0)),  TableFromList = Table.FromList(Source, Splitter.SplitByNothing()),  ChangedType = Table.TransformColumnTypes(TableFromList,{{"Column1", type date}}),  RenamedColumns = Table.RenameColumns(ChangedType,{{"Column1", "Date"}}),  InsertYear = Table.AddColumn(RenamedColumns, "Year", each Date.Year([Date])),  InsertQuarter = Table.AddColumn(InsertYear, "QuarterOfYear", each Date.QuarterOfYear([Date])),  InsertMonth = Table.AddColumn(InsertQuarter, "MonthOfYear", each Date.Month([Date])),  InsertDay = Table.AddColumn(InsertMonth, "DayOfMonth", each Date.Day([Date])),  InsertDayInt = Table.AddColumn(InsertDay, "DateInt", each [Year] \* 10000 + [MonthOfYear] \* 100 + [DayOfMonth]),  InsertMonthName = Table.AddColumn(InsertDayInt, "MonthName", each Date.ToText([Date], "MMMM", Culture), type text),  InsertCalendarMonth = Table.AddColumn(InsertMonthName, "MonthInCalendar", each (try(Text.Range([MonthName],0,3)) otherwise [MonthName]) & " " & Number.ToText([Year])),  InsertCalendarQtr = Table.AddColumn(InsertCalendarMonth, "QuarterInCalendar", each "Q" & Number.ToText([QuarterOfYear]) & " " & Number.ToText([Year])),  InsertDayWeek = Table.AddColumn(InsertCalendarQtr, "DayInWeek", each Date.DayOfWeek([Date])),  InsertDayName = Table.AddColumn(InsertDayWeek, "DayOfWeekName", each Date.ToText([Date], "dddd", Culture), type text),  InsertWeekEnding = Table.AddColumn(InsertDayName, "WeekEnding", each Date.EndOfWeek([Date]), type date)  in  InsertWeekEnding  in  CreateDateTable |