```
let CreateDateTable = (StartDate as date, EndDate as date, optional Culture as nullable text) as table =>

let

DayCount = Duration.Days(Duration.From(EndDate - StartDate)) +1,

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]), type number),

InsertQuarter = Table.AddColumn(InsertYear, "Quarter Num", each Date.QuarterOfYear([Date]), type number),

InsertCalendarQtr = Table.AddColumn(InsertQuarter, "Quarter Year", each "Q" & Number.ToText([Quarter Num]) & " " & Number.ToText([Year]),type text),

InsertCalendarQtrOrder = Table.AddColumn(InsertCalendarQtr, "Quarter Year Order", each [Year] * 10 + [Quarter Num], type number),

InsertMonth = Table.AddColumn(InsertCalendarQtrOrder, "Month Num", each Date.Month([Date]), type number),

InsertMonthName = Table.AddColumn(InsertMonth, "Month Name", each Date.ToText([Date], "MMMM", Culture), type text),

InsertMonthNameShort = Table.AddColumn(InsertMonthName, "Month Name Short", each Date.ToText([Date], "MMM", Culture), type text),

InsertCalendarMonth = Table.AddColumn(InsertMonthNameShort, "Month Year", each (try(Text.Range([Month Name],0,3)) otherwise [Month Name]) & " " & Number.ToText([Year]), type text),

InsertCalendarMonthOrder = Table.AddColumn(InsertCalendarMonth, "Month Year Order", each [Year] * 100 + [Month Num], type number),

InsertWeek = Table.AddColumn(InsertCalendarMonthOrder, "Week Num", each Date.WeekOfYear([Date]), type number),

InsertCalendarWk = Table.AddColumn(InsertWeek, "Week Year", each "W" & Number.ToText([Week Num]) & " " & Number.ToText([Year]), type text),

InsertCalendarWkOrder = Table.AddColumn(InsertCalendarWk, "Week Year Order", each [Year] * 100 + [Week Num], type number),

InsertWeekEnding = Table.AddColumn(InsertCalendarWkOrder, "Week Ending", each Date.EndOfWeek([Date]), type date),
InsertDay = Table.AddColumn(InsertWeekEnding, "Month Day Num", each Date.Day([Date]), type number),

InsertDayInt = Table.AddColumn(InsertDay, "Date Int", each [Year] * 10000 + [Month Num] * 100 + [Month Day Num], type number),

InsertDayWeek = Table.AddColumn(InsertDayInt, "Day Num Week", each Date.DayOfWeek([Date]) + 1, type number),

InsertDayName = Table.AddColumn(InsertDayWeek, "Day Name", each Date.ToText([Date], "dddd", Culture), type text),

InsertWeekend = Table.AddColumn(InsertDayName, "Weekend", each if [Day Num Week] = 1 then "Y" else if [Day Num Week] = 7 then "Y" else "N", type text),

InsertDayNameShort = Table.AddColumn(InsertWeekend, "Day Name Short", each Date.ToText([Date], "ddd", Culture), type text),

InsertIndex = Table.AddIndexColumn(InsertDayNameShort, "Index", 1, 1),

InsertDayOfYear = Table.AddColumn(InsertIndex, "Day of Year", each Date.DayOfYear([Date]), type number),

InsertCurrentDay = Table.AddColumn(InsertDayOfYear, "Current Day?", each Date.IsInCurrentDay([Date]), type logical),

InsertCurrentWeek = Table.AddColumn(InsertCurrentDay, "Current Week?", each Date.IsInCurrentWeek([Date]), type logical),

InsertCurrentMonth = Table.AddColumn(InsertCurrentWeek, "Current Month?", each Date.IsInCurrentMonth([Date]), type logical),

InsertCurrentQuarter = Table.AddColumn(InsertCurrentMonth, "Current Quarter?", each Date.IsInCurrentQuarter([Date]), type logical),

InsertCurrentYear = Table.AddColumn(InsertCurrentQuarter, "Current Year?", each Date.IsInCurrentYear([Date]), type logical),

InsertCompletedDay = Table.AddColumn(InsertCurrentYear, "Completed Days", each if DateTime.Date(DateTime.LocalNow()) > [Date] then "Y" else "N", type text),

InsertCompletedWeek = Table.AddColumn(InsertCompletedDay, "Completed Weeks", each if (Date.Year(DateTime.Date(DateTime.LocalNow())) > Date.Year([Date])) then "Y" else if (Date.Year(DateTime.Date(DateTime.LocalNow())) < Date.Year([Date])) then "N" else if (Date.WeekOfYear(DateTime.Date(DateTime.LocalNow())) > Date.WeekOfYear([Date])) then "Y" else "N", type text),

InsertCompletedMonth = Table.AddColumn(InsertCompletedWeek, "Completed Months", each if (Date.Year(DateTime.Date(DateTime.LocalNow())) > Date.Year([Date])) then "Y" else if (Date.Year(DateTime.Date(DateTime.LocalNow())) < Date.Year([Date])) then "N" else if (Date.Month(DateTime.Date(DateTime.LocalNow())) > Date.Month([Date])) then "Y" else "N", type text),
InsertCompletedQuarter = Table.AddColumn(InsertCompletedMonth, "Completed Quarters", each if
(Date.Year(DateTime.Date(DateTime.LocalNow())) > Date.Year([Date])) then "Y" else if
(Date.Year(DateTime.Date(DateTime.LocalNow())) < Date.Year([Date])) then "N" else if
(Date.QuarterOfYear(DateTime.Date(DateTime.LocalNow())) > Date.QuarterOfYear([Date])) then "Y"
else "N", type text),

InsertCompletedYear = Table.AddColumn(InsertCompletedQuarter, "Completed Years", each if
(Date.Year(DateTime.Date(DateTime.LocalNow())) > Date.Year([Date])) then "Y" else "N", type text)

in

InsertCompletedYear

in

CreateDateTable
```