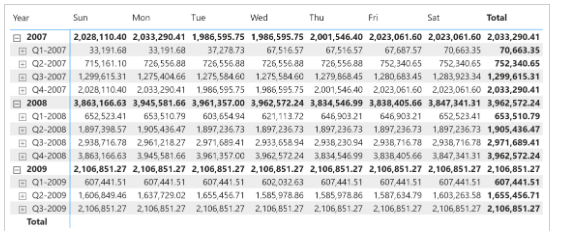
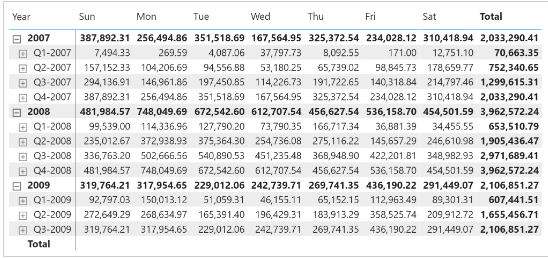
**Filtering other date attributes**

Once you mark the Date table as a date table, DAX automatically removes any filter from the *Date* table every time [**CALCULATE**](https://dax.guide/calculate/?aff=dax-patterns) filters the date column of the *Date* table. This behavior is by design. Its goal is to simplify the writing of time intelligence calculations. Indeed, if DAX did not remove the filters, it would be necessary to manually add a [**REMOVEFILTERS**](https://dax.guide/removefilters/?aff=dax-patterns) over the *Date* table every time a DAX time intelligence function is used, resulting in a negative development experience.

The automatic removal of the filters from the *Date* table might introduce issues for some particular reports. For example, if a report computes the year-to-date of sales by slicing the amount by day of the week, the result obtained by only using the time intelligence function [**DATESYTD**](https://dax.guide/datesytd/?aff=dax-patterns) is wrong. Figure 19 shows that the result of *Sales YTD* for each day of the week is slightly smaller or equal to the row total, which is showing the value for all the days of the week.



The reason for the inaccurate value is that [**DATESYTD**](https://dax.guide/datesytd/?aff=dax-patterns) applies a filter on the Date[Date] column. Because Date is marked as a date table, DAX automatically applies a [***REMOVEFILTERS***](https://dax.guide/removefilters/?aff=dax-patterns)( ‘Date’ ) modifier to the same [**CALCULATE**](https://dax.guide/calculate/?aff=dax-patterns) where [**DATESYTD**](https://dax.guide/datesytd/?aff=dax-patterns) is used in a filter argument – thus removing the filter on the day of the week. Therefore, the number shown is the year-to-date regardless of any filter on the weekday. The day-of-week filter only affects the last day of the period specified on the rows of the report – year or quarter. The correct result, shown in Figure 20, requires a different approach.



There are two options to obtain the correct value: either reiterate the filter over the day of the week in the [**CALCULATE**](https://dax.guide/calculate/?aff=dax-patterns) statement, or update the data model.

Restoring the filter over the day of the week requires adding [***VALUES***](https://dax.guide/values/?aff=dax-patterns) ( Date[Day of Week] ) only if the columns are filtered, like in the following code:

Sales YTD (day of week) :=

IF (

    [ShowValueForDates],

    IF (

        ISFILTERED ( 'Date'[Day of Week] ),

        CALCULATE (

            [Sales Amount],

            DATESYTD ( 'Date'[Date] ),

            VALUES ( 'Date'[Day of Week] )

        ),

        CALCULATE (

            [Sales Amount],

            DATESYTD ( 'Date'[Date] )

        )

    )

)

This first solution works well, but it comes with a significant shortcoming: there are two different versions of the calculation depending on whether the Date[Day of Week] column is filtered or not. On large models, this might have a noticeable impact on performance.

There is another solution to this scenario that requires updating the data model. Instead of using the Date table to select the day of the week, we can store the day of the week in a separate table that filters Sales without being related to Date. This way, the automatic filter removal over Date does not affect the existing filter over the day of the week. For example, the Day of Week table can be created as a calculated table:

Day of Week =

SELECTCOLUMNS (

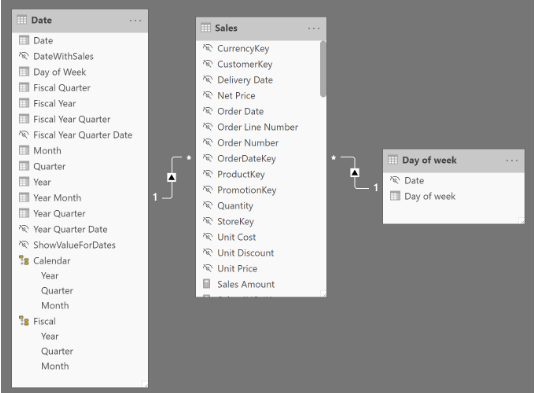
    'Date',

    "Date", 'Date'[Date],

    "Day of Week", 'Date'[Day of Week]

)

The Day of week table must have a relationship between Sales[Order Date] and ‘Day of week'[Date], meaning the model must look like the one in Figure 21.



Please note that we created the new Day of Week table using all the dates in Date to create the relationship with the existing Sales[Order Date] column. It is possible to obtain the same behavior by creating a table with only seven values (Sunday through Saturday), but that choice requires an additional column in the Sales table – thus consuming more memory for the data model.

Slicing by Day of Week in the newly created table is compatible with any time intelligence calculation and respects any filter on the Day of Week table; this is because the two filters (Date and Day of the week) belong to two different tables.

The additional table could consolidate any set of attributes required by specific business rules. We built an example with the day of the week, but you can use any other set of attributes (like working days, holidays, seasons), provided that such attributes depend on Order Date.