

Introduction to DAX for Power BI

Using Data Analysis Expressions to solve common business problems in Power BI

Overview

In this two-day course, students will be introduced to using Data Analysis Expressions (DAX), which is the expression language that is used to create custom calculations in the Power BI Data model. The course covers some of the theoretical underpinnings of the data model and DAX, but the emphasis is on using DAX to solve common business problems. Students will learn how to write their own calculated columns, measures, and tables, how to visualize the way Power BI computes DAX calculations, and how to troubleshoot custom code. This instructor-led course consists of a combination of lecture, demonstrations, and hands-on labs.

Software Used During the Class

The standard lab setup for this class consists of a Window 10 PC with Power BI Desktop and DAX Studio installed.

Audience

This course is intended for business users who have been using Power BI to build analytic solutions and are ready to take advantage of the power and flexibility that DAX provides. Learning DAX is a very common “next step” for experienced Power BI users.

Prerequisites

Although experience with Power BI Desktop is not strictly required for this class, students with at least some familiarity will be better equipped to get the most out of the lab work.

DAX functions that are covered in this class include:

Date and time functions: CALENDAR(), CALENDARAUTO(), DATE(), NOW(), TODAY(), WEEKDAY(), WEEKNUM(), YEAR(), MONTH(), DAY()

Filter functions: ALL(), ALLEXCEPT(), CALCULATE(), CALCULATETABLE(), FILTER(), LOOKUPVALUE(), REMOVEFILTERS()

Information functions: CONTAINS(), HASONEFILTER(), HASONEVALUE(), ISINSCOPE(), ISBLANK(), ISEMPTY(), ISERROR(), USERNAME()

Logical functions: AND(), COALESCE(), IF(), IFERROR(), NOT(), OR(), SWITCH(), TRUE(), FALSE()

Math and Statistical Functions: DIVIDE(), SUM(), SUMX(), AVERAGE(), AVERAGEX(), COUNT(), COUNTROWS(), COUNTX(), DISTINCTCOUNT(), MAX(), MAXX(), MEDIAN(), MEDIANX(), MIN(), MINX(), RANKX()

Table manipulation functions: ADDCOLUMNS(), SELECTCOLUMNS(), CROSSJOIN(), DATATABLE(), NATURALINNERJOIN(), SUMMARIZE(), VALUES()

Text functions: CONCATENATEX(), FIND(), LEFT(), LEN(), MID(), REPLACE(), SEARCH(), TRIM(), VALUE()

Time intelligence functions: DATEADD(), DATESBETWEEN(), DATESMTD(), DATESQTD(), DATESYTD(), PARALLELPERIOD(), SAMEPERIODLASTYEAR(), TOTALMTD(), TOTALQTD(), TOTALYTD()

Course Outline

Getting started with DAX

What is DAX?

Basics of DAX syntax

Creating a calculated column

Creating a calculated measure

Creating a calculated table

Differences in evaluation context

Rules of evaluation

Working with variables

Commenting your code

Testing your code

Performing basic calculations

Implicit measures

Adding quick measures

Doing basic math

Using logic in your calculations

Aggregating and summarizing data

Working with DAX data types

Working with context in the data model

Context defined

Data modeling basics

Introduction to dimensional modeling

Relationships and their effect on the evaluation context

Getting data from other tables using RELATED() and RELATEDTABLE()

Modifying the context using CALCULATE()

Looking up data without using relationships

Performing more advanced calculations

Handling errors gracefully

Using DAX iterator functions

Using table manipulation functions

Troubleshooting your calculations using the Performance Analyzer and DAX Studio

Working with time

Performing date calculations

Working with date tables

Generating a date table with the CALENDAR() function

Defining custom operating periods

YTD, QTD, and MTD calculations

Custom to-date calculations

Finding year-over-year change

Finding moving averages

Enhancing the user experience

Controlling visibility of your measures

Using What-if parameters

Adding banding

Using DAX to provide row-level security