Introduction to DAX for Power BI (2nd edition) Courseware

Cursusduur: 2 Dagen **Cursuscode: M-55310AC**

Beschrijving:

This two-day instructor-led course is intended for business users who have been using Power BI and want to use DAX to create custom calculations in their data models. In this class, you 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 the DAX language, but the emphasis is on using DAX to solve common business problems. You will learn how to write your own calculated columns, measures, and tables, how to visualize the way Power BI computes DAX calculations, and how to troubleshoot custom code.

Doelgroep:

This course is intended for individuals who want to develop their skills in using DAX for data analysis in Power BI.

Doelstelling:

- DAX is fundamental to Power BI's calculation model. Although analysts don't initially need it to work successfully with Power BI, it's critically important for any kind of analytic work beyond the most basic applications.

Vereiste kennis en vaardigheden:

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.

Examens en certificering

Vervolgcursussen:

None

Cursusinhoud:

Module 1: MODULE 1: GETTING STARTED WITH DAX

LESSON 1: INTRODUCTION TO DAX

- What is DAX?
- Basics of DAX syntax
- Syntax Best Practices
- Common DAX Syntax Best Practices

LESSON 2: CREATING OBJECTS WITH DAX

- Creating a calculated column
- Creating a calculated measure
- Creating a calculated table
- Differences in evaluation context:

LESSON 3: CONTEXT AND RULES OF EVALUATION

- Rules of evaluation

LESSON 4: VARIABLES, COMMENTS, AND TESTING

- Working with variables
- Commenting your code
- Testing your code
- Where to look for help

LAB 1: WHAT IS DAX?

- Lab Introduction
- Exercise 1: Create calculated columns
- Exercise 2: Create calculated measures
- Exercise 3: Create calculated tables
- Exercise 4: Test your calculations
- Exercise 5: Get help from a Large Language Model (LLM)

LAB 1 CHALLENGE PROBLEMS

- CHALLENGE PROBLEM 1.1: CALCULATED [PROFIT COL] COLUMN
- CHALLENGE PROBLEM 1.2: CREATE A [# OF SALES] MEASURE

MODULE 2: PERFORMING BASIC CALCULATIONS

LESSON 1: GETTING STARTED

- Implicit measures
- Adding quick measures
- Working with DAX data types

LESSON 2: WORKING WITH NUMBERS

- Doing basic math

LESSON 3: LOGICAL FUNCTIONS AND

MODULE 3: WORKING WITH CONTEXT IN THE DATA MODEL

LESSON 1: EVALUATION CONTEXT

- Context defined
- A Row Context Example
- A Filter Context Example

LESSON 2: DIMENSIONAL MODELING

- Data modeling basics
- Introduction to dimensional modeling

LESSON 3: WORKING WITH RELATIONSHIPS

- Relationships and their effect on the evaluation context
- Getting data from other tables using RELATED() and RELATEDTABLE()
- Looking up data without using relationships

LESSON 4: MANIPULATING THE CONTEXT

- Modifying the context using CALCULATE()
- Using CALCULATE() with a filter
- Using more than one CALCULATE() in a single measure
- Using CALCULATE() with ALL() ; REMOVEFILTERS()
- Using CALCULATE() with ALLEXCEPT()

LAB 3: WORKING WITH CONTEXT IN THE DATA MODEL

- Lab Introduction
- Exercise 1: Creating a data model with a star schema
- Exercise 2: Enhance the data model with measures

LAB 3 CHALLENGE PROBLEMS

- CHALLENGE PROBLEM 3.1: CREATE A "TOP EARNER" MEASURE
- CHALLENGE PROBLEM 3.2: CREATE A [LEMONADE ONLY SALES] MEASURE

MODULE 4: PERFORMING MORE ADVANCED CALCULATIONS

LESSON 1: HANDLING ERRORS

- Types of errors
- Conversion errors
- Arithmetic errors
- Empty or missing values
- Handling errors gracefully

MODULE 5: WORKING WITH TIME

LESSON 1: PERFORMING CALCULATIONS WITH DATE AND TIME DATA

- Performing date calculations
- Date and Time Functions
- Time Intelligence Functions

LESSON 2: USING DATE TABLES IN YOUR DATA MODEL

- Working with date tables
- Generating a date table with the CALENDAR() function

LESSON 3: ADAPTING YOUR MODEL TO TIME-BASED BUSINESS RULES

- Defining custom operating periods

LESSON 4: WORKING WITH TIME INTELLIGENCE

- YTD, QTD, and MTD calculations
- Custom to-date calculations
- Finding year-over-year change
- Finding moving averages

LAB 5: WORKING WITH TIME

- Lab Introduction
- Exercise 1: Performing date calculations
- Exercise 2: Working with date tables
- Exercise 3: Working with to-date calculations
- Exercise 4: Finding year-over-year change
- Exercise 5: Finding moving averages

LAB 5 CHALLENGE PROBLEMS

- CHALLENGE PROBLEM 5.1: CREATE A RULES-BASED [OPENFORBUSINESS] COLUMN

MODULE 6: ENHANCING THE USER EXPERIENCE

LESSON 1: USING DAX TO CUSTOMIZE WHAT THE USER SEES

- Controlling visibility of your measures

LESSON 2: CREATING INTERACTIVE MEASURES WITH WHAT-IF PARAMETERS

- Using What-if parameters

LESSON 3: CONVERTING CONTINUOUS DATA TO CATEGORICAL DATA

OPERATORS

- Using logic in your calculations
- Using the IF() function
- Nested IF()
- Using the SWITCH() function
- Advanced SWITCH()
- Logical operators ; functions: ||, OR(), ;;, AND(), NOT()

LESSON 4: AGGREGATING AND SUMMARIZING DATA

- Aggregating and summarizing data
- The SUM() function

LAB 2: PERFORMING BASIC CALCULATIONS

- Lab Introduction
- Exercise 1: Work with mathematical operations
- Exercise 2: Working with text calculations
- Exercise 3: Working with dates and times
- Exercise 4: Working with logic

LAB 2 CHALLENGE PROBLEMS

- CHALLENGE PROBLEM 2.1: CREATE AN ENHANCED [CUSTOMER FULL NAME] COLUMN
- CHALLENGE PROBLEM 2.2: CREATE A CUSTOMER[ICONS] COLUMN
- ISERROR() and IFERROR()

LESSON 2: USING DAX ITERATOR FUNCTIONS

- The DAX Iterator Functions

LESSON 3: USING TABLE MANIPULATION FUNCTIONS

- Table manipulation functions

LESSON 4: TROUBLESHOOTING YOUR CALCULATIONS USING THE PERFORMANCE ANALYZER AND DAX STUDIO

LAB 4: PERFORMING MORE ADVANCED CALCULATIONS

- Lab Introduction
- Exercise 1: Work with errors in data
- Exercise 2: Using iterator functions
- Exercise 3: Using table manipulation functions
- Exercise 4: Get started with the built-in DAX Query View
- Exercise 5: Get started with DAX Studio

LAB 4 CHALLENGE PROBLEMS

- CHALLENGE PROBLEM 4.1: CREATE A [TOTAL SALES] COLUMN IN THE CUSTOMER TABLE
- CHALLENGE PROBLEM 4.2: USE THE SUMMARIZE() FUNCTION TO CREATE A TOTALSALESBYCOUNTRY TABLE
- Adding banding

LESSON 4: ADDING ROW-LEVEL SECURITY

- Using DAX to provide row-level security
- Standard Row-Level Security
- Dynamic Row-level Security

LAB 6: ENHANCING THE USER EXPERIENCE

- Lab Introduction
- Exercise 0: Enable Map and Filled Map visuals
- Exercise 1: Controlling visibility of your measures
- Exercise 2: Using numeric range parameters
- Exercise 3: Adding banding
- Exercise 4: Using DAX to provide row-level security

LAB 6 CHALLENGE PROBLEMS

- CHALLENGE PROBLEM 6.1: USE PARAMETERS TO CREATE AN RGB COLOR CODE AND USE IT TO FORMAT A VISUAL

Nadere informatie:

Neem voor nadere informatie of boekingen contact op met onze Customer Service Desk 030 - 60 89 444

info@globalknowledge.nl

www.globalknowledge.com/nl-nl/

Iepenhoeve 5, 3438 MR Nieuwegein