



The Definitive Guide to Power Query (M) : Mastering Complex Data Transformation with Power Query /

Deckler, Gregory,
author

Monografía

Learn how to use the Power Query M formula language and its functions effectively for better data modeling and impactful business intelligence reports. Purchase of the print or Kindle book includes a free PDF eBook

Key Features Get comprehensive coverage of fundamental and advanced Power Query concepts Gain hands-on experience with practical examples based on real-world problems Delve into the intricacies of Power Query M language for enhanced data manipulation

Book Description Data transformation is a critical step in building data models and business intelligence reports. Power Query is an invaluable tool for anyone who wants to master data transformation, and this book will equip you with the knowledge and skills to make the most of it. The Definitive Guide to Power Query (M) will help you build a solid foundation in the Power Query M language. As you progress through the chapters, you'll learn how to use that knowledge to implement advanced concepts and data transformations. This will set the stage for an uncompromisingly thorough exploration of the Power Query M Language. You'll also get to grips with optimizing performance, handling errors, and implementing efficient data processing techniques. As this is a hands-on guide, the practical examples in the chapters will help you gain the skills to apply Power Query to real-world problems and improve your data analysis capabilities. By the end of this book, you will be able to leverage all of Power Query's remarkable capabilities for data transformation. What you will learn

Gain a strong understanding of Power Query fundamentals Master various functions within Power Query to perform complex data operations Acquire knowledge about values, types, and control structures in Power Query Develop proficiency in error handling techniques Learn performance optimization strategies for Power Query Apply what you've learned to real-world scenarios, including common troublesome patterns

Who this book is for This book is for business analysts, business intelligence professionals, and power business users working with data who want to add Power Query mastery to their resume. This book will be beneficial for anyone who wants to automate their process of data cleaning and save huge amount of time. Having some basic experience in Power Query is recommended

<https://rebiunoda.pro.baratznet.cloud:28443/OpacDiscovery/public/catalog/detail/b2FpOmNlbGVicmF0aW9uOmVzLmJhcmF0ei5yZW4vMzU5NDE4ODk>

Título: The Definitive Guide to Power Query (M) Mastering Complex Data Transformation with Power Query
Gregory Deckler, Rick de Groot, Melissa de Korte

Edición: First edition

Editorial: Birmingham, England Packt Publishing [2024] 2024

Descripción física: 1 online resource (759 pages)

Mención de serie: Expert insight

Bibliografía: Includes bibliographical references and index

Contenido: Cover -- Copyright -- Foreword -- Contributors -- Table of Contents -- Preface -- Chapter 1: Introducing M -- The history of M -- Who should learn M? -- Where and how is M used? -- Experiences -- Products and services -- Why learn M? -- M language basics -- The let expression -- The characteristics of M -- Formal classification -- Informal characteristics of M -- Summary -- Chapter 2: Working with Power Query/M -- Technical requirements -- Touring the Power Query Desktop experience -- A brief tour -- Header -- Formula bar -- Ribbon -- Queries pane -- Query Settings pane -- Preview pane -- Status Bar -- Your first query -- Options and data source settings -- Options -- Data source settings -- Editing experience-generated code -- Creating custom columns -- Adding an index column -- Adding columns with examples -- Math operations -- Adding custom m code columns -- Using the Advanced Editor -- Summary -- Chapter 3: Accessing and Combining Data -- Technical requirements -- Accessing files and folders -- File.Contents -- Text/CSV -- Excel -- Folder -- PDF -- XML -- Xml.Tables -- Xml.Document -- Azure Storage -- Additional file formats -- Retrieving web content -- Investigating binary functions -- Lines functions -- Accessing databases and cubes -- Cube functions -- Working with standard data protocols -- Addressing additional connectors -- Popular software systems -- Identity functions -- Combining and joining data -- Table.Combine -- Table.NestedJoin and Table.Join -- Table.FuzzyNestedJoin and Table.FuzzyJoin -- Summary -- Chapter 4: Understanding Values and Expressions -- Introducing the types of values -- Binary values -- Structure -- Related functions -- Special considerations -- The Date/Time family of values -- Date values -- Time values -- DateTime values -- DateTimeZone values -- Duration values -- Logical values -- Structure -- Related Functions -- Special considerations -- Null values -- Structure -- Related functions -- Special considerations -- Number values -- Structure and examples -- Related functions -- Special considerations -- Text values -- Structure -- Related functions -- Special considerations -- List values -- Structure -- Related functions -- Special considerations -- Record values -- Structure -- Related functions -- Special considerations -- Table values -- Structure -- Related functions -- Special considerations -- Function values -- Structure -- Related functions -- Special considerations -- Type values -- Operators -- Expressions -- Nesting let expressions -- Coding best practices for expressions -- Control structures -- Enumerations -- Summary -- Chapter 5: Understanding Data Types -- What are data types? -- The type system -- Columns with mixed types -- Column data type versus value type -- Importance of types -- Clarity and consistency -- Data validation -- Other reasons -- Data types available in M -- Primitive types -- Abstract primitive types -- Nullable primitive types -- Using primitive types for data filtering -- Custom types -- List type -- Record type -- Table type -- Function Type -- Type detection -- Retrieving data types from a data source -- Automatically detecting types -- Data type conversion -- Converting value types -- Converting column types -- Avoiding data loss during conversion -- Effect of locale/culture -- Facets -- Type Claims -- Available Type Claims -- Converting values using type claims -- Inspecting Type Claims -- Ascribing types -- What is ascription? -- Functions that support ascribing types -- Ascribing types when creating records -- Ascribing types when creating tables -- Ascribing types when modifying tables -- Ascribing types to any value -- Errors when ascribing types -- The base type is incompatible with the value The Type Claim does not conform with the value -- Ascribing incompatible types to structured values -- Type equivalence, conformance, and assertion -- Type equality -- Type conformance -- Type assertion -- Summary -- Chapter 6: Structured Values -- Introducing structured values -- Lists -- Introduction to lists -- List operators -- Equal -- Not equal -- Concatenate -- Coalesce -- Methods to create a list -- Creating lists using the list initializer -- Creating lists using functions -- Referencing a table column -- Using the a..b form -- Accessing items in a list -- Accessing list values by index -- Handling non-existent index positions -- Common operations with lists -- Assigning data types to a list -- Records -- Introduction to records -- Record operators -- Equal -- Not equal -- Concatenation -- Coalesce -- Methods to create a record -- Creating records using the record initializer -- Creating records using functions -- Retrieving a record by referencing a table row -- Accessing fields in a record -- Field selection -- Record projection -- Common operations with records -- Structure for variables -- Referencing the current row -- Providing options for functions -- Keeping track of intermediary results -- Assigning a data type to records -- Tables -- Introduction to tables -- Table operators -- Equal -- Not equal -- Concatenation -- Coalesce -- Methods to create a table -- Retrieve data from a source -- Manually input data into functions -- Re-use existing tables/queries -- Using the Enter data functionality --

Accessing elements in a table -- Item access -- Field access -- Common operations with tables -- Assigning a data type to tables -- Summary -- Chapter 7: Conceptualizing M -- Technical requirements -- Understanding scope -- Examining the global environment -- Studying sections -- Creating your own global environment -- Understanding closures Query folding -- Managing metadata -- Summary -- Chapter 8: Working with Nested Structures -- Transitioning to coding -- Getting started -- Understanding Drill Down -- The trick to getting more out of the UI -- Methods for multistep value transformation -- Transforming values in tables -- Table.AddColumn -- Table.TransformColumns -- Table.ReplaceValue -- Working with lists -- Transforming a list -- List.Transform -- List.Zip -- Extracting an item -- Resizing a list -- List.Range -- List.Alternate -- Filtering a list -- List.FindText -- List.Select -- To-list conversions -- Column or field names -- A single column -- All columns -- All rows -- Other operations -- Expanding multiple list columns simultaneously -- Flattening inconsistent multi-level nested lists -- Working with records -- Transforming records -- Extracting a field value -- Resizing records -- Filtering records -- To-record conversions -- Table row to record -- Record from table -- Record from list -- Conditional lookup or value replacement -- Working with tables -- Transforming tables -- Extracting a cell value -- Resizing a table in length -- Resizing a table in width -- Filtering tables -- Approximate match -- To-table conversions -- Record-to-table conversion -- Creating tables from columns, rows, or records -- Table information -- Working with mixed structures -- Lists of tables, lists, or records -- Tables with lists, records, or tables -- Mixed structures -- Flatten all -- Unpacking all record fields from lists -- Extracting data through lookup -- Summary -- Chapter 9: Parameters and Custom Functions -- Parameters -- Understanding parameters -- Creating parameters -- Using parameters in your queries -- Putting it all together -- Parameterizing connection information -- Dynamic file paths -- Filtering a date range -- Custom functions -- What are custom functions? Transforming queries into a function -- What is the "create function" functionality? -- Simplifying troubleshooting and making changes -- Invoking custom functions -- Manually in the advanced editor or formula bar -- Using the UI -- The each expression -- Common usecases -- Refining function definitions -- Specifying data types -- Making parameters optional -- Referencing column and field names -- Debugging custom functions -- Function scope -- Top-level expression -- In line within a query -- Putting it all together -- Turning all columns into text -- Merging tables based on date ranges -- Summary -- Chapter 10: Dealing with Dates, Times, and Durations -- Technical requirements -- Dates -- M calendar table -- Other date formats -- Julian days -- Alternate date formats -- Additional custom date functions -- Working days -- Moving average -- Time -- Creating a time table -- Shift classification -- Dates and times -- Time zones -- Correcting data refresh times -- Duration -- Working duration -- Summary -- Chapter 11: Comparers, Replacers, Combiners, and Splitters -- Technical requirements -- Key concepts -- Function invocation -- Some common errors -- Closures -- Higher-order functions -- Anonymous functions -- Ordering values -- Comparers -- Comparer.Equals -- Comparer.Ordinal -- Comparer.OrdinalIgnoreCase -- Comparer.FromCulture -- Comparison criteria -- Numeric value -- Computing a sort key -- List with key and order -- Custom comparer with conditional logic -- Custom comparer with Value.Compare -- Equation criteria -- Default comparers -- Custom comparer -- Key selectors -- Combining key selectors and comparers -- Replacers -- Replacer.ReplaceText -- Replacer.ReplaceValue -- Custom replacers -- Combiners -- Combiner.CombineTextByDelimiter -- Functionality -- Example -- Combiner.CombineTextByEachDelimiter -- Functionality -- Example Combiner.CombineTextByLengths

ISBN: 1-83508-616-0

Materia Título preferido: Power Query (Computer file)

Materia: Database management- Computer programs

Autores: Groot, Rick de, author Korte, Melissa de, author

Enlace a formato físico adicional: 1-83508-972-0

Punto acceso adicional serie-Título: Expert insight

Baratz Innovación Documental

- Gran Vía, 59 28013 Madrid
- (+34) 91 456 03 60
- informa@baratz.es

