Course Outline | Querying with Transact SQL (T-SQL)

2 day(s)

Overview

This highly rated course is for those working with Microsoft SQL Server who need to extract data using Transact SQL (T-SQL). While attendees gain much experience designing complex T-SQL queries using standard functions, inner joins and aggregate functions, T-SQL action statements which facilitate data or database modification do not form part of this course. So as to master the syntax of the language, the course is highly practical in nature and the focus throughout is on coding T-SQL by hand. On completion, a comprehensive set of course notes, examples, tutor and attendee scripts are made available online for each attendee.

Prerequisites

No prior SQL training or relational database experience is assumed.

Course Outline

Relational Databases and SQL: An Overview

- The Role of the Database Server: SQL Server
- Interacting with a Database Server: The Client
- SQL and Transact SQL (T-SQL) in Context
- T-SQL: Its Role and Purpose
- Databases, Schemas, Tables, Rows and Columns

Introducing Management Studio: SQL Server's IDE

- Using an Integrated development Environment (IDE)
- Exploring Database Objects with Object Explorer
- Using the Query Editor to Create and Edit T_SQL
- Ways to Reduce Errors
- Minimising, Trapping and Finding Errors
- Viewing Results with the Results Pane
- Results as Text, Results in a Grid
- Making Sense of Error Messages

T-SQL Statements and Queries: Their Structure

- About Statements, Batches and Scripts
- Executing and Parsing T-SQL Scripts
- Using Comments
- T-SQL Syntax and The Rules of T-SQL
- About Keywords, Identifiers, Operators, Whitespace and Case
- Working with Clauses
- About the Semi Colon and GO
- T-SQL Conventions and Good Practice

Retrieving Data with T-SQL: First Steps

- Ways of Specifying the Database, Schemas and Table
- Working with the Default Schema
- Using SELECT to Extract Data
- Retrieving Entire Tables
- Retrieving Specific Columns
- Working with the Result Table
- Outputting Results to CSV
- Using TOP to Limiting the Number of Rows
- Using ORDER BY to Sort the Output
- Working with Calculations
- Using Column Aliases to Rename Columns

- Using Numeric and String Operators to Create Derived Output
- Ways of Working: Some Tips

Using WHERE to Filter Results

- Working with Comparison Operators (=, >= etc)
- Numeric and String Based Filtering
- Filtering Based on Calculations
- Eliminating Duplicate Results with DISTINCT
- Working with Execution Order
- Column Aliases: Where You Can and Cannot Use Them
- Extending Filters with AND and OR
- Solving AND/OR Difficulties with Brackets
- Excluding Results with NOT: Some Tips
- Range Filtering using BETWEEN and IN
- NULL and its Implications Explained
- Catering for NULL
- Matching Patterns with LIKE

Getting Results From Multiple Tables

- Qualifying Column Names
- Joins Explained
- The Different Types of Joins
- Mastering the Inner Join: WHERE Syntax
- Mastering the Inner Join: INNER JOIN Syntax
- Table Aliases: Why We Need Them
- Joining Two Tables
- Joining Multiple tables
- How to Simplify Joins: An Approach

Getting Confident with T-SQL Joins

- The Importance of the Database Diagram
- Creating a Database Diagram with Management Studio
- Primary Keys and Foreign Keys Explained
- Locating Primary Keys and Foreign Keys
- Do we Need Primary Keys and Foreign Keys?
- Approaching Joins the Right Way
- Avoiding Common Join Errors

Using Standard T-SQL Functions

- How to Use Standard T-SQL Functions to Modify Results
- How to Find the Right Function
- Mathematical, String and Conversion Functions
- Functions for Modifying and Calculating Dates
- Formatting Numbers to Two Decimal Places
- Replacing NULL with a Specific Value
- Using Standard Functions in WHERE
- Using CASE to Specify Output Conditions

Working with Dates

- Understanding How Dates are Stored
- Introducing Date Functions
- Converting Text Dates To Date Format
- Establishing Today's Date
- Extracting Parts of a Date (Day, Month, Year)
- Displaying Dates in Specific Date Formats
- Filtering with Dates

Grouping and Summarizing Results

- The difference Between Tabular and Scalar Results
- Using Aggregate Functions (MAX(), SUM(), AVG(), COUNT() etc)
- The Way Aggregate Functions Work
- Where to Use and Where Not to Use Aggregate Functions
- Using GROUP BY to Group Results
- The Need for HAVING: Filtering the Result Table