Course 55232-A: Writing Analytical Queries for Business Intelligence

Duration : 03 Days

Course outline

Module 1: Introduction to TSQL for Business Intelligence

This module discusses writing analytical queries vs. transactional DML queries, and describes the typical architecture of a business intelligence environment. It discusses the role of SELECT queries in retrieving data for analysis from relational databases. It introduces the sample database to be used in the course, and begins a presentation of the SELECT query.

Lessons

- Two Approaches to SQL Programming
- TSQL Data Retrieval in an Analytics Environment
- The Database Engine
- SQL Server Management Studio and the CarDeal Sample Database
- Identifying Variables in Tables
- SQL is a Declarative Language
- Introduction to the SELECT Query

Lab : Lab 1

- Create a database diagram
- Create and execute basic SELECT queries

After completing this module, students will be able to:

- Describe the purpose of analytical queries
- Describe the function of TSQL data retrieval in an analytics / business intelligence environment
- Describe the primary functions of the database engine
- Discuss TSQL as a declarative language
- Identify variables of interest in database tables
- Write basic SELECT queries

Module 2: Turning Table Columns into Variables for Analysis: SELECT List Expressions, WHERE, and ORDER BY

This module covers the identification of and relationship between levels of measurement and column data types. It continues a discussion of the SELECT query and adds the WHERE and ORDER BY clauses.

Lessons

- Turning Columns into Variables for Analysis
- Column Expressions, Data Types, and Built-in Functions
- Column aliases
- Data type conversions
- Built-in Scalar Functions
- Table Aliases
- The WHERE clause
- ORDER BY

Lab : Lab 2- Write queries using:

- Column and table aliases
- DISTINCT
- WHERE
- ORDER BY
- Built-in functions
- Explicit and implicit data type conversion

After completing this module, students will be able to:

- Implement column expressions in SELECT queries
- Implement column and table aliases
- Describe data types and Implement data type conversions
- Implement built-in functions
- Use WHERE and ORDER BY clauses in SELECT queries

Module 3: Combining Columns from Multiple Tables into a Single Dataset: The JOIN Operators

Module 3 discusses creating single datasets for analysis by combining results from multiple database tables using JOIN.

Lessons

- Primary Keys, Foreign Keys, and Joins
- Understanding Joins, Part 1: CROSS JOIN and the Full Cartesian Product
- Understanding Joins, Part 2: The INNER JOIN
- Understanding Joins, Part 3: The OUTER JOINS
- Understanding Joins, Part 4: Joining more than two tables
- Understanding Joins, Part 5: Combining INNER and OUTER JOINs

- Combining JOIN Operations with WHERE and ORDER BY

Lab : Lab 3- Write SELECT queries using:

- Inner join
- Left, right, and full join
- Joins of more than two tables
- Join operators, in addition to WHERE and ORDER BY

After completing this module, students will be able to:

- Discuss and describe the conceptual flow of JOIN operations
- Implement INNER and OUTER JOIN operations on two or more tables
- Describe the order of operations of SELECT clauses
- Combine JOIN operations with WHERE and ORDER BY

Module 4: Creating an Appropriate Aggregation Level Using GROUP BY

This module covers the aggregation of quantitative column values across grouping factors for the purpose of group-wise comparisons and/or changing the granularity of a dataset.

Lessons

- Identifying required aggregation level and granularity
- Aggregate Functions
- GROUP BY
- HAVING
- Order of operations in SELECT queries

Lab : Lab 4- Write queries using:

- Aggregate functions
- Aggregate function with HAVING
- Aggregate function with GROUP BY and HAVING
- Aggregate function with GROUP BY, HAVING, WHERE, and ORDER BY

After completing this module, students will be able to:

- Describe row granularity of result sets
- Discuss and implement aggregate functions to achieve required row granularity
- Use GROUP BY to calculate aggregate values for groups
- Use HAVING to filter records in the result set by aggregate value
- Combine GROUP BY and HAVING with WHERE and ORDER BY

Module 5: Subqueries, Derived Tables and Common Table Expressions

This module covers the use of subqueries, derived tables, and common table expressions in SELECT queries as techniques for creating intermediate result sets.

Lessons

- Non-correlated and correlated subqueries
- Derived tables
- Common table expressions

Lab : Lab 5- Write queries using:

- Non-correlated subqueries
- Correlated subqueries
- Derived tables
- Common table expressions
- Subqueries, derived tables, and common table expressions and other topics

After completing this module, students will be able to:

- Describe and discuss the rationale of creating intermediate results sets within SELECT queries
- Implement non-correlated and correlated subqueries
- Implement derived tables
- Implement Common Table Expressions
- Create intermediate to advanced TSQL queries to retrieve result sets for analysis

Module 6: Encapsulating Data Retrieval Logic

This module discusses the encapsulation of data retrieval logic in views, table-valued functions, and stored procedures. It also describes scenarios in which these techniques are useful for producing datasets for analysis. Finally, it describes the database security issues involved, and techniques for creating and using these database objects while maintaining current permission sets on source data.

Lessons

- Views
- Table-valued functions
- Stored procedures
- Creating objects for read-access users
- Creating database accounts for analytical client tools

Lab : Lab 6

- Create a SQL login
- Create a database user and assign required permissions

- Create a database schema for views, functions, and stored procs
- Create a view
- Create a table-values function
- Create a stored procedure
- Allow a user with read-only access to use views, table-valued functions, and store procedures

After completing this module, students will be able to:

- Identify scenarios in which views, table-valued functions, and stored procedures simply data retrieval
- Compare and contrast views, table-valued functions, and stored procedures
- Create views, table-valued functions, and stored procedures
- Describe the security requirement for creating database objects
- Implement views, table-valued functions, and stored procedures for users with read-only access to source data

Module 7: Getting Your Dataset to the Client

This module covers common techniques for making datasets produced by SELECT queries available to analytical client tools such as SQL Server Reporting Services, PowerBI, Excel, and R. It discusses running queries directly from the client tool, in addition to exporting datasets to text files which can then be accessed by the client tool.

Lessons

- Connecting to SQL Server and Submitting Queries from Client Tools
- Connecting and running SELECT queries from:
- Excel
- PowerBI
- RStudio
- Exporting datasets to files using
- Results pane from SSMS
- The bcp utility
- The Import/Export Wizard

Lab : Lab 7

- Retrieving the results of a view in Excel
- Running an ad-hoc SELECT query from Excel
- Running an ad-hoc query from PowerBI
- Running an ad-hoc query from RStudio
- Using the Import/Export wizard to write the results of a query to a text file

After completing this module, students will be able to:

- Describe the properties of database connection strings
- Run queries from, and return results to, Excel, PowerBI, and RStudio
- Export query results to external text files using the SSMS results pane, the bcp utility, and the Import/Export Wizard