

Transact SQL Programming (TSQL)

Duration: 3 Days

OBJECTIVES

This course is aimed at SQL Server database developers, administrators and analysts who need a little bit more than SQL can do alone. SQL Server provides a powerful scripting language called Transact SQL, which enables developers to extend the capabilities of the SQL language. Using Transact SQL (TSQL) powerful scripts, stored procedures, functions and triggers can be produced to enable extracted information to be processed further, or to enable complex transformations of data prior to, during or after an insert, update or delete, or to implement automated updates and audit trails. Covers SQL Server 2005 through to SQL Server 2016.

PRE-REQUISITES

Delegates should have SQL Server SQL language experience to the level of our Advanced Querying of SQL Server Databases course.

COURSE OUTLINE

Transact SQL Scripting

- The USE Statement
- Using Variables
- Data Types
- Variable Scope
- Table Variables
- The PRINT Statement
- The IF Statement
- BEGIN and END
- The WHILE Statement

Error Handling

- The @@ERROR Variable
- The RETURN Statement
- The RAISERROR Statement
- Structured Error Handling
- TRY CATCH Blocks

Transaction Logging

- BEGIN TRAN
- COMMIT TRAN
- ROLLBACK TRAN

Locking

- Row Level Locking
- Key Level Locking
- Page Level Locking
- Extent Level Locking
- Table Level Locking
- Database Level Locking

- Transactions and Locks
- Transaction Isolation Levels
- Deadlocks

Implementing Cursors

- Transact-SQL Cursors
- Monitoring Transact-SQL Cursor Activity
- Declaring A Cursor With The DECLARE Statement
- Opening A Cursor With The OPEN Statement
- Fetching Records with The FETCH Statement
- Closing The Cursor With The CLOSE Statement
- Freeing A Cursor With The DEALLOCATE Statement
- Working With Cursors
- INSENSITIVE Cursor
- SCROLL Cursors
- Fetching Data With A SCROLL Cursor
- The @@CURSOR_ROWS Variable
- Working With Very Large Data Sets
- Changing Data Through Cursors
- READ ONLY Cursors
- Limited UPDATE Cursors
- Programming With Cursors
- Cursors and Locking

Stored Procedures

- Pre-Compiled Code
- Creating Stored Procedures
- Encrypting Stored Procedures
- Recompiling Stored Procedures
- Creating A Stored Procedure WITH RECOMPILE
- Altering Stored Procedures
- Deleting A Stored Procedure
- Stored Procedure Security
- Passing In Parameters
- Initialising Parameters
- Outputting Values
- The RETURN Statement
- Modular Design

User-Defined Functions

- What is a User-Defined Function?
- Usage of User-Defined Functions
- Creating a User-Defined Scalar Function
- Using a User-Defined Scalar Function
- Table-Valued Functions
- Creating a User-Defined Inline Table-Valued Function
- Multi-Statement Table-Valued Functions
- Viewing Existing User-Defined Functions
- Modifying and Dropping User-Defined Functions

Triggers

- Creating Triggers
- Update and Insert Triggers

- Altering Triggers
- Triggers and Constraints
- Realistic Application Of Triggers
- Automatic Updates
- Data Integrity
- Auditing
- Stored Procedures and Triggers