COURSE CONTENT - STRUCTURED QUERY LANGUAGE

SQL Overview

- Outlining SQL as the cornerstone of database activity
- Applying the ANSI/ISO standards
- Describing the fundamental building blocks: tables, columns, primary keys, and foreign keys

Building the Database Schema

- Creating tables and columns
- Building tables with CREATE TABLE
- Modifying table structure with ALTER TABLE
- Adding columns to an existing table
- Removing tables with DROP TABLE

Protecting data integrity with constraints

- Guaranteeing uniqueness with primary key constraints
- Enforcing integrity with foreign key constraints
- Imposing business rules with check constraints
- Improving performance with indexes
- Expediting data retrieval with indexes
- Recommending guidelines for index creation

Manipulating Data

- Adding table rows with INSERT
- Changing row content with UPDATE
- Removing rows with DELETE
- Applying transactions
- Controlling transactions with COMMIT and ROLLBACK
- Deploying BEGIN TRANSACTION in SQL Server

COURSE CONTENT - STRUCTURED QUERY LANGUAGE

Working with the SELECT Statement

- Adding table rows with INSERT
- Changing row content with UPDATE
- Removing rows with DELETE
- Applying transactions
- Controlling transactions with COMMIT and ROLLBACK
- Deploying BEGIN TRANSACTION in SQL Server

Querying Multiple Tables

- Applying the ANSI/ISO standard join syntax
- Matching related rows with INNER JOIN
- Including nonmatched rows with OUTER JOIN
- Creating a Cartesian product with CROSS JOIN
- Combining results with set operators
- Stacking results with UNION
- Identifying matching rows with INTERSECT
- Utilizing EXCEPT to find nonmatching rows

Employing Functions in Data Retrieval

- Processing data with row functions
- Conditional formatting with the CASE expression
- Utilizing the CASE expression to simulate IF tests
- Dealing with NULL values
- Performing analysis with aggregate functions
- Summarizing data using SUM, AVG and COUNT
- Finding the highest/lowest values with MAX and MIN
- Refining the summary level with GROUP BY
- Applying filter conditions with HAVING