Hands-On

Microsoft SQL Programming

Course Description

This SQL programming course teaches students relational database fundamentals and SQL programming skills. Topics covered include relational database architecture, database design techniques, and simple and complex query skills. This class is intended for analysts, developers, designers, administrators, and managers new to the SQL programming language. Upon completion, participants will understand SQL functions, join techniques, database objects and constraints, and will be able to write useful SELECT, INSERT, UPDATE and DELETE statements. Extensive hands on exercises are used throughout to reinforce learning and develop real competency.

Students Will Learn

- **Database Design Issues**
- **Data Normalization**
- **SQL Statement Syntax**
- **Modifying Tables with DML**
- **Query Analysis**
- **Table Indexing**
- **Triggers and Stored Procedures**

Prerequisites

General computer knowledge. Familiarity with relational database concepts is helpful but not required.

Course Outline

Relational Database Fundamentals

- Overview of Relational Database Concepts
- Relational Databases and Relational Database Management Systems
- SQL, DML and DDL Languages

Manipulating Query Results

- Using Row Functions
    - Character
    - Numeric
    - Date and Time
    - Data Conversion
- Using Column Functions
- Handling NULLs
- Aggregating Results using GROUP BY
- Restricting Groups with the HAVING Clause

Writing Basic SQL Queries

- Displaying Table Structures
- Column Datatypes
- SELECTing Column Data FROM a Table
- SELECT DISTINCT Values
- COUNTing Rows
- Restricting Rows using the WHERE Clause
    - Using Operators
    - Mathematical
    - Logical
    - Comparison
    - Concatenation
    - Set Operators
- Sorting Results Using ORDER BY
- Joining Multiple Tables
- Using Column and Table Aliases
- Pseudo Columns

Advanced Query Techniques

- Outer Joins
- Joining a Table to Itself
- Subqueries
- Hierarchical Queries
- Using the Data Dictionary
- Tips for Developing Complex SQL Queries

Manipulating Table Data Using SQL's Data Manipulation Language (DML)

- INSERT into Tables
- UPDATE Table Data
- DELETE from Tables
- TRUNCATE Tables
- Controlling Transactions
    - COMMIT
    - ROLLBACK
    - SAVEPOINT

Creating a Database

- Database Development Methodology Overview
- Building a Logical Data Model
    - Identifying Entities and Attributes

        - Isolating Keys
        - Relationships between Entities
        - Creating Entity-Relationship Diagrams
    - Transforming to Physical Design
        - Migrating Entities to Tables
        - Selecting Primary Keys
        - Defining Columns
        - Enforcing Relationships with Foreign Keys
    - Constructing the Database using DDL
        - Creating Tables, Indexes, Constraints and Views
        - Using Table and Column COMMENTs
        - Using SYNONYMs
    - Facilitating Development Using CASE Tools

Managing Database Objects

- Dropping Tables, Indexes and Views
- ALTERing Tables, Indexes and Views
- Enabling and Disabling Constraints
- Managing Security
- Backup and Recovery Planning

Performance Tuning

- Writing Efficient SQL
- EXPLAIN Query Analysis
- Indexing for Performance
- Index-Organized Tables
- Clustering

Stored Procedures

- Definition and Benefits of Use
- Perils and Pitfalls
- CREATE PROCEDURE
    - Syntax
    - Variables and Parameters
    - Code Examples
- Temporary Tables
- Control of Program Flow
- ALTER and DROP PROCEDURE
- Implementation Differences

User-Defined Functions

- Definition and Benefits of Use
- CREATE FUNCTION
    - Syntax
    - RETURN Clause and the RETURNS Statement
    - Scalar vs. Table Functions
- Comparison with Stored Procedures
- Returning Scalar Values and Tables
- ALTER and DROP FUNCTION

Triggers

- Definition and Benefits of Use

- Alternatives (e.g., Constraints)
- CREATE TRIGGER
    - Syntax
    - Trigger Types
- "Inserted" (or "NEW") and "Deleted" (or "OLD") Tables
- Event Handling and Trigger Execution
- Nested and Recursive Triggers
- ALTER and DROP TRIGGER

Equipment Requirements

(This apply's to our hands-on courses only)

BTS always provides equipment to have a very successful Hands-On course. BTS also encourages all attendees to bring their own equipment to the course. This will provide attendees the opportunity to incorporate their own gear into the labs and gain valuable training using their specific equipment.

Course Length

3 Days