

## Course Details

**Duration:** 5 Days

**Course code:** M20464

### Overview:

This 5-day instructor-led course introduces SQL Server 2014 and describes logical table design, indexing and query plans. It also focusses on the creation of database objects including views, stored procedures, along with parameters, and functions. Other common aspects of procedure coding, such as indexes, concurrency, error handling, and triggers are also covered in this course. Also this course helps you prepare for the Exam 70-464.

### Target Audience:

The primary audience for this course is IT Professionals who want to become skilled on SQL Server 2014 product features and technologies for implementing a database.

### Prerequisites:

- ◆ Knowledge of writing T-SQL queries.
- ◆ Knowledge of basic relational database concepts.

### Delegates will learn how to

At the end of this course you will be able to:

- ◆ Introduce the entire SQL Server platform and its major tools. It will cover editions, versions, basics of network listeners, and concepts of services and service accounts.
- ◆ Determine appropriate data types to be used when designing tables, convert data between data types, and create alias data types.
- ◆ Be aware of good design practices regarding SQL Server tables and be able to create tables using T-SQL. (Note: partitioned tables are not covered).
- ◆ Implement PRIMARY KEY, FOREIGN KEY, DEFAULT, CHECK and UNIQUE constraints, and investigate cascading FOREIGN KEY constraints.
- ◆ Determine appropriate single column and composite indexes strategies.
- ◆ Create tables as heaps and tables with clustered indexes. Also consider the design of a table and suggest an appropriate structure.
- ◆ Read and interpret details of common elements from execution plans.
- ◆ Design effective non-clustered indexes.
- ◆ Design and implement views
- ◆ Design and implement stored procedures.
- ◆ Work with table types, table valued parameters and use the MERGE statement to create stored

procedures that update data warehouses.

- ◆ Design and implement functions, both scalar and table-valued. (Also describe where they can lead to performance issues).
- ◆ Perform basic investigation of a deadlock situation and learn how transaction isolation levels affect application concurrency.
- ◆ Use both traditional T-SQL error handling code and structured exception handling.
- ◆ Design and implement DML triggers
- ◆ Learn appropriate uses for SQL CLR integration and implement an existing .NET assembly within SQL Server.
- ◆ Store XML data and schemas in SQL Server.
- ◆ Perform basic queries on XML data in SQL Server.
- ◆ Work with the GEOGRAPHY and GEOMETRY data types
- ◆ Implement and query a full-text index.

## Course Outline

### Module 1: Introduction to Database Development

This module introduces database development and the key tasks that a database developer would typically perform.

#### Lessons

- ◆ Introduction to the SQL Server Platform
- ◆ Working with SQL Server Tools
- ◆ Configuring SQL Server Services

#### Lab : Introduction to Database Development

- ◆ Start SQL Server Management Studio

### Module 2: Designing and Implementing Tables

This module explains how to design, create, and alter tables. Also it focusses on working with schemas.

#### Lessons

- ◆ Designing Tables
- ◆ Working with Schemas
- ◆ Creating and Altering Tables

#### Lab : Designing and Implementing Tables

- ◆ Improving the Design of Tables
- ◆ Creating a Schema
- ◆ Creating the Tables

### **Module 3: Ensuring Data Integrity through Constraints**

This module explains how to enforce data integrity, and implement domain integrity to maintain high quality data. Also it focusses on implementing Entity and Referential Integrity.

#### **Lessons**

- ◆ Enforcing Data Integrity
- ◆ Implementing Domain Integrity
- ◆ Implementing Entity and Referential Integrity

#### **Lab : Ensuring Data Integrity through Constraints**

- ◆ Designing Constraints
- ◆ Testing the constraints

### **Module 4: Introduction to Indexing**

This module describes the concept of an index and discusses selectivity, density and statistics. It covers appropriate data type choices and choices around composite index structures.

#### **Lessons**

- ◆ Core Indexing Concepts
- ◆ Single Column and Composite Indexes
- ◆ SQL Server Table Structures
- ◆ Working with Clustered Indexes

#### **Lab : Creating Indexes**

- ◆ Creating Tables with Clustered Indexes
- ◆ Improving Performance through Nonclustered Indexes

### **Module 5: Advanced Indexing**

This module explains covering indexes and the INCLUDE clause as well as the use of padding, hints and statistics. The module also covers the use of the Database Engine Tuning Advisor and index-related dynamic management views to assess indexing strategies.

#### **Lessons**

- ◆ Execution Plan Core Concepts
- ◆ Common Execution Plan Elements
- ◆ Working with Execution Plans
- ◆ Designing Effective Nonclustered Indexes
- ◆ Performance Monitoring

#### **Lab : Planning for SQL Server 2014 Indexing**

- ◆ Exploring Existing Index Statistics
- ◆ Creating Covering Indexes

## **Module 6: Columnstore Indexes**

This module explains columnstore indexes, introduces clustered and nonclustered columnstore indexes, and discusses considerations for using columnstore indexes.

### **Lessons**

- ◆ Columnstore Indexes
- ◆ Best Practices for Columnstore Indexes

### **Lab : Using In-Memory Database Capabilities**

- ◆ Creating Columnstore Indexes

## **Module 7: Designing and Implementing Views**

This module introduces Views, and explains how to create and manage Views. Also it focusses on the performance consideration for Views.

### **Lessons**

- ◆ Introduction to Views
- ◆ Creating and Managing Views
- ◆ Performance Considerations for Views

### **Lab : Designing and Implementing Views**

- ◆ Designing, Implementing and Testing the WebStock Views
- ◆ Designing and Implementing the Contacts View
- ◆ Modifying the AvailableModels View

## **Module 8: Designing and Implementing Stored Procedures**

This module describes the potential advantages of the use of stored procedures along with guidelines on creating them.

### **Lessons**

- ◆ Introduction to Stored Procedures
- ◆ Working With Stored Procedures
- ◆ Implementing Parameterized Stored Procedures
- ◆ Controlling Execution Context

### **Lab : Designing and Implementing Stored Procedures**

- ◆ Creating stored procedures
- ◆ Creating a parameterized stored procedure
- ◆ Altering the execution context of stored procedures

## **Module 9: Designing and Implementing User-Defined Functions**

This module explains how to design and implement user-defined functions that enforce business rules or data consistency, and modify and maintain existing functions written by other developers.

## Lessons

- ◆ Overview of Functions
- ◆ Designing and Implementing Scalar Functions
- ◆ Designing and Implementing Table-Valued Functions
- ◆ Implementation Considerations for Functions
- ◆ Alternatives to Functions

## Lab : Designing and Implementing User-Defined Functions

- ◆ Formatting Phone Numbers
- ◆ Modifying an Existing Function
- ◆ Resolving a Function-Related Performance Issue

## Module 10: Responding to Data Manipulation via Triggers

This module, explains what DML triggers are and how they enforce data integrity. Also it focusses on the different types of triggers available, and how to define triggers in a database.

## Lessons

- ◆ Designing DML Triggers
- ◆ Implementing DML Triggers
- ◆ Advanced Trigger Concepts

## Lab : Responding to Data Manipulation via Triggers

- ◆ Creating and Testing the Audit Trigger
- ◆ Improving the Audit Trigger

## Module 11: Using In-Memory Tables

This module covers the creation of in-memory tables and native stored procedures and discusses the advantages and disadvantages of using in-memory tables.

## Lessons

- ◆ Memory-Optimized Tables
- ◆ Native Stored Procedures

## Lab : Using In-Memory Database Capabilities

- ◆ Working with Memory Optimized Tables
- ◆ Working with Natively Compiled Stored Procedures

## Module 12: Implementing Managed Code in SQL Server

This module explains how to use CLR integrated code to create user-defined database objects that are managed by the .NET Framework.

## Lessons

- ◆ Introduction to SQL CLR Integration
- ◆ Importing and Configuring Assemblies

- ◆ Implementing SQL CLR Integration

#### **Lab : Implementing Managed Code in SQL Server**

- ◆ Assessing Proposed CLR Code
- ◆ Implementing a CLR Assembly
- ◆ Implementing a CLR User-defined Aggregate and CLR User-defined Data Type

#### **Module 13: Storing and Querying XML Data in SQL Server**

This module introduces XML and shows how XML data can be stored within SQL Server and then queried, including queries written in a language called XQuery.

##### **Lessons**

- ◆ Introduction to XML and XML Schemas
- ◆ Storing XML Data and Schemas in SQL Server
- ◆ Implementing the XML Data Type
- ◆ Using the T-SQL FOR XML Statement
- ◆ Getting Started with XQuery
- ◆ Shredding XML

#### **Lab : Storing and Querying XML Data in SQL Server**

- ◆ Assessing appropriate Use of XML Data in SQL Server
- ◆ Testing XML Data Storage in Variables
- ◆ Retrieving Information about XML Schema Collections
- ◆ Querying SQL Server Data as XML
- ◆ Write a Stored Procedure Returning XML

#### **Module 14: Working with SQL Server Spatial Data**

This module introduces Spatial Data, and explains how to work with SQL Server Spatial Data Types.

##### **Lessons**

- ◆ Introduction to Spatial Data
- ◆ Working with SQL Server Spatial Data Types
- ◆ Using Spatial Data in Applications

#### **Lab : Working with SQL Server Spatial Data**

- ◆ Querying the Geometry Data Type
- ◆ Adding Spatial Data to an Existing Table