Developing Microsoft SQL Server 2012 Databases

36 Contact Hours

Course Overview

This 5-day instructor-led course introduces SQL Server 2012 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 transactions, concurrency, error handling, triggers, and SQL CLR are also covered in this course. Also this course helps you prepare for the Exam 70-464

All the labs for this course can be performed using the provided virtual machines. However, if you have a Microsoft Windows Azure account and the classroom virtual machines connect to the internet you may be able to connect to your Azure server and database from the classroom. Many of the labs in this course are enabled for you to perform the lab while connected to your own Azure database in the cloud. Your instructor should be able to provide a current list of Azure enabled labs .

Below are the some of the new features of SQL Server that has been discussed in this course

•Use of SC collations

◦Current versions of Windows support extended Unicode collations known as supplementary (or SC) collations. Earlier versions of SQL Server did not support the use of these collations. The course provides details on SQL Server support for these additional collations.

•Use of TRY_PARSE, TRY_CONVERT

◦SQL Server 2012 introduces a number of new functions. The demonstrations in the course utilize many of these new functions such as TRY_PARSE and TRY_CONVERT.

•Detailed discussion on IDENTITY constraints

◦A new more detailed topic on IDENTITY constraints has been added to the course.

•Working with SEQUENCES

◦SQL Server 2012 introduces the concept of sequences. The course now includes a topic and demonstration on the use of sequences.

•Slight course reordering to improve overall flow

◦The course has been reordered from 6232B to improve the overall flow. In particular, constraints are discussed immediately after the initial discussion on logical table design. The course layout offers a more unified experience for the students with each day having a particular flavor.

•Use of THROW and comparisons to RAISERROR

◦SQL Server 2012 enhances the available error-handling options in T-SQL. The module on error-handling has been expanded to show how to use THROW in place of RAISERROR and also to show how to rethrow errors (including system errors) within CATCH blocks using the THROW statement.

•Full-globe and arc support in spatial

◦SQL Server 2012 adds support for geography data-type objects that span more than a single hemisphere. The course discusses the new full globe support and along with new support for curved arc objects.

•Customizable nearness in full-text

◦The course covers how full-text indexing in SQL Server 2012 offers a more customizable and flexible proximity operation than earlier versions of SQL Server.

Audience Profile

The primary audience for this course is IT Professionals who want to become skilled on SQL Server 2012 product features and technologies for implementing a database.

At Course Completion

After completing this course, students 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.

Prerequisites

In addition to their professional experience, students who attend this training should already have the following technical knowledge:

- Knowledge of writing T-SQL queries.

- Knowledge of basic relational database concepts.

Students who attend this training can meet the prerequisites by attending the following courses, or obtaining equivalent knowledge and skills:

10774•A: Writing T-SQL Queries for Microsoft SQL Server 2012

Course Outline

Module 1: Introduction to SQL Server 2012 and its Toolset

This module stresses on the fact that before beginning to work with SQL Server in either a development or an administration role, it is important to understand the overall SQL Server platform. In particular, it is useful to understand that SQL Server is not just a database engine but it is a complete platform for managing enterprise data.

Lessons

•Introduction to the SQL Server Platform

•Working with SQL Server Tools

•Configuring SQL Server Services

Lab : Introduction to SQL Server and its Toolset

•Verifying SQL Server Component Installation

•Altering Service Accounts for New Instance

•Enabling Named Pipes Protocol for Both Instances

•Creating an Alias for AdvDev

•Ensuring SQL Browser is Disabled and Configure a Fixed TCP/IP Port (Only if time permits

After completing this lesson, you will be able to:

•Describe the SQL Server Platform.

•Work with SQL Server Tools.

•Configure SQL Server Services.

Module 2: Working with Data Types

This module explains how to use and convert data types. Also it focuses on how to work with specialized data types, and character data types.

Lessons

•Using Data Types

•Working with Character Data

•Converting Data Types

•Specialized Data Types

Lab : Working with Data Types

•Choosing Appropriate Data Types

•Writing Queries With Data Type Conversions

•Designing and Creating Alias Data Types (Only if time permits)

After completing this module, students will be able to:

•Work with data types.

•Work with character data.

•Convert between data types.

•Use specialized data types.

Module 3: Designing and Implementing Tables

This module explains how to design, create, and alter tables. Also it focuses 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

After completing this module, students will be able to:

• Design Tables.

• Work with Schemas.

• Create and Alter Tables.

Module 4: Ensuring Data Integrity through Constraints

This module explains how to enforce data integrity, and implement domain integrity to maintain high quality data. Also it focuses 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

After completing this module, students will be able to:

• Explain the available options for enforcing data integrity and the levels at which they should be applied.

• Implement domain integrity.

• Implement entity and referential integrity.

Module 5: Planning for SQL Server 2012 Indexing

This module explains the core indexing concepts and effectiveness of each data type commonly used in indexes.

Lessons

• Core Indexing Concepts

•Data Types and Indexes

•Single Column and Composite Indexes

Lab : Planning for SQL Server Indexing

•Exploring existing index statistics

•Designing column orders for indexes

After completing this module, students will be able to:

•Explain core indexing concepts.

•Describe the effectiveness of each data type common used in indexes.

•Plan for single column and composite indexes.

Module 6: Implementing Table Structures in SQL Server 2012

This module explains how tables can be structured in SQL Server databases. Also it focuses on designing and working with clustered indexes.

Lessons

•SQL Server Table Structures

•Working with Clustered Indexes

•Designing Effective Clustered Indexes

Lab : Implementing Table Structures in SQL Server

•Creating Tables as Heaps

•Creating Tables with Clustered Indexes

•Comparing the Performance of Clustered Indexes vs. Heaps

After completing this module, students will be able to:

•Explain how tables can be structured in SQL Server databases.

•Work with clustered indexes.

•Design effective clustered indexes.

Module 7: Reading SQL Server 2012 Execution Plans

This module explains how to design additional indexes. Also it focuses on how to read and interpret execution plans.

Lessons

•Execution Plan Core Concepts

•Common Execution Plan Elements

•Working with Execution Plans

Lab : Reading SQL Server Execution Plans

•Actual vs. Estimated Plans

•Identifying Common Plan Elements

•Querying Cost Comparison

After completing this module, students will be able to:

•Explain the core concepts related to the use of execution plans.

•Describe the role of the most common execution plan elements.

•Work with execution plans.

Module 8: Improving Performance through Nonclustered Indexes

This module explains how nonclustered indexes have the potential to significantly enhance the performance of applications and how to use a tool that can help design these indexes appropriately.

Lessons

•Designing Effective Nonclustered Indexes

•Implementing Nonclustered Indexes

•Using the Database Engine Tuning Advisor

Lab : Improving Performance through Nonclustered Indexes

•Nonclustered index usage review

•Improving nonclustered index designs

•Working with SQL Server Profiler and Database Engine Tuning Advisor

•Designing nonclustered index

After completing this module, students will be able to:

•Design effective nonclustered indexes

•Implement nonclustered indexes

•Use the database engine tuning advisor to design indexes

Module 9: Designing and Implementing Views

This module introduces Views, and explains how to create and manage Views. Also it focuses 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 Available Models View

After completing this module, students will be able to:

•Explain the role of views in database development.

•Implement views.

•Describe the performance related impacts of views.

Module 10: 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

After completing this module, students will be able to:

•Describe the role of stored procedures and the potential benefits of using them.

•Work with stored procedures.

•Implement parameterized stored procedures.

•Control the execution context of a stored procedure.

Module 11: Merging Data and Passing Tables

This module reviews the techniques that provide the ability to process sets of data rather than individual rows. Also it focuses on how these techniques can be used in combination with TABLE parameter types to minimize the number of required stored procedure calls in typical applications.

Lessons

•Using the MERGE Statement

•Implementing Table Types

•Using TABLE Types As Parameters

Lab : Passing Tables and Merging Data

•Creating a Table Type

•Using a Table Type Parameter

•Using a Table Type with MERGE

After completing this module, students will be able to:

•Use the MERGE statement.

•Implement table types.

•Use TABLE types as parameters.

Module 12: 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

•Resolve a Function-related Performance Issue

After completing this module, students will be able to:

•Design and implement scalar functions.

•Design and implement table-valued functions.

•Describe implementation considerations for functions.

•Describe alternatives to functions.

Module 13: Creating Highly Concurrent SQL Server 2012 Applications

This module explains how to use transactions and the SQL Server locking mechanisms to meet the performance and data integrity requirements of your applications.

Lessons

•Introduction to Transactions

•Introduction to Locks

•Management of Locking

•Transaction Isolation Levels

Lab : Creating Highly Concurrent SQL Server Applications

•Detecting Deadlockss

•Investigating Transaction Isolation Levels

After completing this module, students will be able to:

•Describe the role of transactions

•Explain the role of locks

•Manage locking

•Work with transaction isolation level

Module 14: Handling Errors in T-SQL Code

This module explores T-SQL error handling, looks at how it has traditionally been implemented, and how structured exception handling can be used

Lessons

•Understanding T-SQL Error Handling

•Implementing T-SQL Error Handling

•Implementing Structured Exception Handling

Lab : Handling Errors in T-SQL Code

•Replacing @@ERROR based error handling with structured exception handling

•Adding deadlock retry logic to the stored procedure

After completing this module, students will be able to:

•Design T-SQL error handling

•Implement T-SQL error handling

•Implement structured exception handling

Module 15: Responding to Data Manipulation via Triggers

This module, explains what DML triggers are and how they enforce data integrity. Also it focuses 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

After completing this module, students will be able to:

•Design DML triggers.

•Implement DML triggers.

•Explain advanced DML trigger concepts.

Module 16: Implementing Managed Code in SQL Server 2012

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 : Designing and Implementing Views

•Assessing Proposed CLR Code

•Implementing a CLR Assembly

•Implementing a CLR User-defined Aggregate and CLR User-defined Data Type

After completing this module, students will be able to:

•Explain the importance of SQL Server CLR Integration.

•Import and configure assemblies.

•Implement objects that have been created within .NET assemblies.

Module 17: Storing XML Data in SQL Server 2012

This module introduces XML and shows how XML data can be stored within SQL Server.

Lessons

•Introduction to XML and XML Schemas

•Storing XML Data and Schemas in SQL Server

•Implementing the XML Data Type

Lab : Storing XML Data in SQL Server

•Appropriate Usage of XML Data Storage in SQL Server

•Investigating the Storage of XML Data in Variables

•Investigating the use of XML Schema Collections

•Investigating the Creation of Database Columns Based on XML

After completing this module, students will be able to:

•Describe XML and XML schemas.

•Store XML data and associated XML schemas in SQL Server.

•Implement the XML data type within SQL Server.

Module 18: Querying XML Data in SQL Server

This module shows how XML data can be queried, including queries written in a language called XQuery.

Lessons

•Using the T-SQL FOR XML Statement

•Getting Started with XQuery

•Shredding XML

Lab : Querying XML Data in SQL Server

•Learning to query SQL Server data as XML

•Writing a stored procedure returning XML

•Writing a stored procedure that updates using XML

After completing this module, students will be able to:

•Use the T-SQL FOR XML Statement.

•Work with basic XQuery queries.

•Shred XML to a relational form.

Module 19: Working with SQL Server 2012 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

•Familiarity With Geometry Data Type

•Adding Spatial Data to an Existing Table

•Business Application of Spatial Data

After completing this module, students will be able to:

•Describe the importance of spatial data and the industry standards related to it.

•Explain how to store spatial data in SQL Server.

•Perform calculations on and query SQL Server spatial data.

Module 20: Working with Full-Text Indexes and Queries

This module introduces Full-Text Indexing and how to implement Full-Text Indexes in SQL Server.

Lessons

•Introduction to Full-Text Indexing

•Implementing Full-Text Indexes in SQL Server

•Working with Full-Text Queries

Lab : Working with Full-Text Indexes and Queries

•Implementing a full-text index

•Implementing a stoplist

•Creating a stored procedure to implement a full-text search

After completing this module, students will be able to:

•Describe why user interfaces in existing applications are not sufficient for end user search needs

•Implement full-text indexes in SQL Server

•Query SQL Server using full-text queries