

Course 6232A:

Implementing a Microsoft SQL Server 2008 Database

About this Course

Elements of this syllabus are subject to change.

This five-day instructor-led course provides students with the knowledge and skills to implement a Microsoft SQL Server 2008 database. The course focuses on teaching individuals how to use SQL Server 2008 product features and tools related to implementing a database.

Audience Profile

This course is intended for IT Professionals who administer and maintain SQL Server databases.

Course Details

Module 1: Creating Databases and Database Files

The students will learn one of the most fundamental tasks that a database developer must perform, the creation of a database and its major components, such as creating databases, setting database options, creating filegroups, schemas, and database snapshots.

Lessons

- Creating Databases
- Creating Filegroups
- Creating Schemas
- Creating Database Snapshots

Lab : Creating Databases and Database Files

- (Level 200) Creating a Database
- (Level 200) Creating Schemas
- (Level 300) Creating a Database Snapshot

After completing this module, students will be able to:

- Create databases
- Create filegroups
- Create schemas
- Create database snapshots

Module 2: Creating Data Types and Tables

The students will learn about the system-supplied data types in SQL Server 2008. They will learn how to define custom Transact-SQL data types and how to create tables and how to use partitioned tables to organize data into multiple partitions.

Lessons

- Creating Data Types

Creating Tables
Creating Partitioned Tables

Lab : Creating Data Types and Tables

(Level 200): Creating Data Types
(Level 200): Using New Date and Time Data Types
(Level 200): Creating Tables
(Level 300) Creating Partitioned Tables

After completing this module, students will be able to:

Create new data types.
Create new tables.
Create partitioned tables.

Module 3: Creating and Tuning Indexes

The students will learn how to plan, create, and optimize indexes to attain optimal performance benefits.

Lessons

Planning Indexes
Creating Indexes
Optimizing Indexes

Lab : Creating and Tuning Indexes

(Level 200) Creating Indexes
(Level 300) Tuning Indexes

After completing this module, students will be able to:

Plan indexes.
Create indexes.
Optimize indexes.

Module 4: Implementing Data Integrity by Using Constraints and Triggers

The students will learn about implementing data integrity in SQL Server 2008 by using constraints. They will also implement data integrity by using triggers.

Lessons

Data Integrity Overview
Implementing Constraints
Implementing Triggers

Lab : Implementing Data Integrity by Using Constraints and Triggers

(Level 200) Creating Constraints

(Level 200) Disabling Constraints

(Level 300) Creating Triggers

After completing this module, students will be able to:

Describe the options for enforcing data integrity in SQL Server 2008.

Implement data integrity in SQL Server 2008 databases by using constraints.

Implement data integrity in SQL Server 2008 databases by using triggers.

Module 5: Using XML

The students will learn how to work with XML, including use of the FOR XML clause, the OPENXML function, XQuery expressions, and the xml native data type. They will learn the considerations to be taken into account when creating XML indexes and the syntax used to create the XML indexes. They will also learn what XML schemas and XML schema collections are as well as how to use them to implement typed XML data.

Lessons

Using the XML Data Type

Retrieving XML by Using FOR XML

Shredding XML by Using OPENXML

Introducing XQuery

Creating XML Indexes

Implementing XML Schemas

Lab : Using XML

(Level 200) Mapping Relational Data and XML

(Level 200) Storing XML Natively in the Database

(Level 300) Using XQuery with XML Methods

(Level 200) Create XML Indexes

(Level 300) Implementing XML Schemas

After completing this module, students will be able to:

Use the xml data type.

Retrieve XML by using the FOR XML clause.

Shred XML by using the OPENXML function.

Use XQuery expressions.

Create XML indexes.

Implement data integrity in SQL Server 2008 databases by using XML schemas.

Module 6: Implementing Views

The students will be introduced to the different types of views available in Microsoft SQL Server 2008 which provide a convenient way to access data through a predefined query.

Lessons

Introduction to Views

Creating and Managing Views
Optimizing Performance by Using Views

Lab : Implementing Views

(Level 200) Creating Views
(Level 200) Creating Indexed Views
(Level 200) Creating Partitioned Views

After completing this module, students will be able to:

Describe the purpose of views.
Create and manage views.
Explain how to optimize query performance by using views.

Module 7: Implementing Stored Procedures

The students will learn the design and implementation of stored procedures to enforce business rules or data consistency, or to modify and maintain existing stored procedures written by other developers.

Lessons

Implementing Stored Procedures
Creating Parameterized Stored Procedures
Working With Execution Plans
Handling Errors

Lab : Implementing Stored Procedures

(Level 300) Creating Stored Procedures
(Level 300) Working with Execution Plans

After completing this module, students will be able to:

Implement stored procedures.
Create parameterized stored procedures.
Work with execution plans.
Handle errors in stored procedures.

Module 8: Implementing Functions

The students will learn the design and implementation of user-defined functions that enforce business rules or data consistency, or to modify and maintain existing functions written by other developers.

Lessons

Creating and Using Functions
Working with Functions
Controlling Execution Context

Lab : Implementing Functions

- (Level 300) Creating Functions
- (Level 300) Controlling Execution Context

After completing this module, students will be able to:

- Create and use functions.
- Work with functions.
- Control execution context.

Module 9: Implementing Managed Code in the Database

The students will learn to use managed code to implement database objects, such as stored procedures, user-defined data types, user-defined functions, and triggers.

Lessons

- Introduction to the SQL Server Common Language Runtime
- Importing and Configuring Assemblies
- Creating Managed Database Objects

Lab : Implementing Managed Code in the Database

- (Level 300) Importing an Assembly
- (Level 300) Creating Managed Database Objects

After completing this module, students will be able to:

- Identify appropriate scenarios for managed code in the database.
- Import and configure assemblies.
- Create managed database objects.

Module 10: Managing Transactions and Locks

The students will learn to use transactions and SQL Server locking mechanisms to meet the performance and data integrity requirements of their applications.

Lessons

- Overview of Transactions and Locks
- Managing Transactions
- Understanding SQL Server Locking Architecture
- Managing Locks

Lab : Managing Transactions and Locks

- (Level 300) Using Transactions
- (Level 300) Managing Locks
- (Level 300) Using Partition Locking

After completing this module, students will be able to:

- Describe how SQL Server 2008 transactions use locks.
- Execute and cancel a transaction.
- Describe concurrency issues and SQL Server 2008 locking mechanisms.
- Manage locks.

Module 11: Using Service Broker

The students will learn to use Service Broker to create secure, reliable, and scalable applications.

Lessons

- Service Broker Overview
- Creating Service Broker Objects
- Sending and Receiving Messages

Lab : Using Service Broker

- (Level 300) Creating Service Broker Objects
- (Level 300) Implementing the Initiating Service
- (Level 300) Implementing the Target Service

After completing this module, students will be able to:

- Describe Service Broker functionality and architecture.
- Create Service Broker objects.
- Send and receive Service Broker messages.

Before attending this course, students must have:

- Basic knowledge of the Microsoft Windows operating system and its core functionality.
- Working knowledge of Transact-SQL.
- Working knowledge of relational databases.
- Some experience with database design.