50401-SQL Server 2008

Designing and Optimizing Database Solutions with Microsoft SQL Server 2008

Introduction

This **five-day** instructor-led course provides the knowledge and skills that IT Professionals need to design, optimize, and maintain SQL Server 2008 database.

Audience

The audience of this course is developers who implement database solutions or perform development utilizing the programming features and functionality of SQL Server. Students taking this course are expected to have three or more years of experience working on databases for two or more of the following phases in the product lifecycle - design, development, deployment, optimization, maintenance, or support. They should possess a four-year college degree, BS or BA, in the computer field. The students should have experience in the following areas:

- Developing databases
- Writing Transact-SQL queries
- Designing, implementing and troubleshooting programming objects
- Doing database performance tuning and optimization
- Designing databases at both the conceptual and logical levels
- Implementing databases at the physical level
- Designing and troubleshooting the data access layer of an application
- Gathering business requirements

At Course Completion

After completing this course, students will be able to:

- Design a database design strategy
- Design a database for optimal performance
- Design security for a database
- Design programming objects
- Design queries for performance
- Design a transaction and concurrency strategy
- Design an XML strategy

50401-SQL Server 2008

Prerequisites

Before attending this course, it is recommended that students have:

- Working knowledge of data storage. Specifically, they should know about row layout, fixed length field placement and varying length field placement.
- Knowledge about index structures and index utilization. Specifically, they must understand the interaction between non-clustered indexes, clustered indexes and heaps. They must know why a covering index can improve performance.
- Hands-on database developer experience. Specifically, they should have three years of experience as a full-time database developer in an enterprise environment.
- Knowledge about the locking model. Specifically, students should have an understanding of lock modes, lock objects and isolation levels and be familiar with process blocking.
- Understanding of Transact-SQL syntax and programming logic. Specifically, students should be completely fluent in advanced queries, aggregate queries, subqueries, user-defined functions, cursors, control of flow statements, CASE expressions, and all types of joins.
- Knowledge about the trade offs when backing out of the fully normalized design and designing for performance and business requirements in addition to being familiar with design models, such as Star and Snowflake schemas. They should be able to design a database to third normal form (3NF).
- Strong monitoring and troubleshooting skills, including usage of monitoring tools.
- Basic knowledge of the operating system and platform. That is, how the operating system integrates with the database, what the platform or operating system can do, and how interaction between the operating system and the database works.
- Basic knowledge of application architecture. That is, how applications can be designed in three layers, what applications can do, how interaction between the application and the database works, and how the interaction between the database and the platform or operating system works.
- Knowledge of using a data modeling tool.
- Knowledge of SQL Server 2005 features, tools, and technologies.
- Have a Microsoft Certified Technology Specialist: Microsoft SQL Server 2005 credential - or equivalent experience.

In addition, it is recommended, but not required, that students have completed:

- Course 2779, Implementing a Microsoft SQL Server 2005 Database.
- Course 2780, Maintaining a Microsoft SQL Server 2005 Database.

Course Outline

Module 1: Designing a Conceptual Database Model

This module explains the guidelines for designing a conceptual database model with a systematic perspective. A systematic approach involves formulating your database design process, following guidelines on how to gather and document database requirements, and following best practices when formulating a conceptual design. Finally, you will learn the guidelines for using Entity Framework.

Lessons

- Overview of Database Design
- Gathering Database Requirements
- Creating a Conceptual Database Design
- Overview of Entity Framework

Lab: Designing a Conceptual Database Model

- Formulating a Conceptual Database Design
- Creating a Conceptual Database Design

After completing this module, students will be able to:

- Explain the key steps in the database design process.
- Gather database requirements.
- Describe the guidelines for creating a conceptual database design.
- Explain the guidelines for creating a conceptual database design by using the Entity Framework.

Module 2: Designing a Logical Database Model

This module explains the best practices followed when you build a new logical database model. You will also learn the guidelines for normalization when designing an OLTP model and when designing a data warehouse database. Finally, you will learn to evaluate the existing logical model of a database.

Lessons

- Guidelines for Building a Logical Database Model
- Planning for OLTP Activity
- Evaluating Logical Models

Lab: Designing a Logical Database Model

- Creating a Logical Database Model
- Normalizing the Logical Database Model

After completing this module, students will be able to:

- Explain the guidelines for building a logical database model.
- Plan for OLTP activity.
- Evaluate Logical models.

Module 3: Designing a Physical Database Model

This module explains the guidelines to be followed when designing physical database objects and constraints. The module also covers the best practices for designing database tables and for designing data integrity.

Lessons

- Selecting Data Types
- Designing Database Tables

- Designing Data Integrity

Lab: Designing a Physical Database Model

- Specifying Database Object Naming Standards
- Converting a Logical Database Model into a Physical Database Model

After completing this module, students will be able to:

- Design column data types.
- Design database tables.
- Design data integrity.

Module 4: Designing Databases for Optimal Performance

This module explains the considerations for designing indexes. The module also covers the guidelines for designing scalable databases, and choosing additional optimization techniques, including designing for plan guide and partition.

Lessons

- Guidelines for Designing Indexes
- Designing a Partitioning Strategy
- Designing a Plan Guide
- Designing Scalable Databases

Lab: Designing Databases for Optimal Performance

- Applying Optimization Techniques
- Creating Plan Guides
- Designing a Partitioning Strategy

After completing this module, students will be able to:

- Design indexes.
- Design a partitioning strategy.
- Design a plan guide.
- Design scalable databases.

Module 5: Designing Security for SQL Server 2008

This module explains the best practices to be followed when designing for security in a database. The module will cover the guidelines for designing security for identity and access control, SQL development, database deployment. Finally, this module explains the guidelines for designing secure operations.

Lessons

- Exploring Security in SQL Server 2008
- Implementing Identity and Access Control
- Guidelines for Secure Development in SQL Server 2008
- Guidelines for Secure Deployment of SQL Server 2008
- Guidelines for Secure Operations

Lab: Designing Security for SQL Server 2008

- Designing Secure Development
- Implementing Secure Operations
- Copying an Unencrypted Database
- Implementing Transparent Data Encryption
- Attempting to Copy an Encrypted Database

After completing this module, students will be able to:

- Explore different aspects of security in SQL Server 2008.
- Implement Identity and Access Control.
- Describe the guidelines for Secure development in SQL Server 2008.
- Describe the guidelines for secure deployment of SQL Server 2008.
- Describe the guidelines for secure operations.

Module 6: Designing a Strategy for Database Access

This module explains the best practices to be followed when designing a database access strategy. The module will cover the guidelines for designing views, stored procedures, and user defined function. Finally, this module explains the guidelines for CLR development.

Lessons

- Guidelines for Designing Secure Data Access
- Designing Views
- Designing Stored Procedures
- Designing User-Defined Functions

Lab: Designing a Strategy for Database Access

- Designing Security for Data Retrieval Objects
- Designing Data Retrieval Objects

After completing this module, students will be able to:

- Describe the guidelines for designing secure data access.
- Design views.
- Design T-SQL stored procedures.
- Design user-defined functions.

Module 7: Designing Queries for Optimal Performance

This module will explain the considerations for optimizing and tuning queries to improve performance. The module will cover the design considerations to refactor cursors into queries.

Lessons

- Considerations for Optimizing Queries for Performance
- Refactoring Cursors into Queries
- Extending Set-Based Operations

Lab: Designing Queries for Optimal Performance

- Optimizing Query Performance

- Refactoring Cursors into Queries

After completing this module, students will be able to:

- Explain the considerations for optimizing queries for performance.
- Refactor cursors into queries.
- Explain Set-Based Operations.

Module 8: Designing a Transaction and Concurrency Strategy

This module describes considerations and guidelines for defining a transaction strategy for a solution. It also shows the guidelines to specify isolation levels for data stores.

Lessons

- Guidelines for Defining Transactions
- Defining Isolation Levels
- Guidelines for Designing a Resilient Transaction Strategy

Lab: Designing a Transaction and Concurrency Strategy

- Determining the Database Isolation Level
- Determining the Order of Object Access
- Designing Transactions

After completing this module, students will be able to:

- Describe the guidelines for defining transactions.
- Define isolation levels.
- Describe the guidelines for designing a resilient transaction strategy.

Module 9: Designing an XML Strategy

This module describes the guidelines to design XML techniques. This module will cover the design considerations for XML storage, XQuery, XPath, and FOR XML clause. Finally, this module will cover the considerations for converting data between XML and relational formats.

Lessons

- Designing XML Storage
- Designing an XML Query Strategy
- Designing a Data Conversion Strategy

Lab: Designing an XML Strategy

- Designing an XML Data Storage Model
- Converting Data Between XML and Relational Forms

After completing this module, students will be able to:

- Design XML storage.
- Design an XML query strategy.
- Design data conversion between XML and Relational forms.

50401-SQL Server 2008

Module 10: Designing SQL Server 2008 Components

This module provides an overview of SQL Server 2008 architecture and the various considerations for choosing SQL Server components to be included in a solution. The module will also cover the considerations for designing service broker and full text search.

Lessons

- Overview of SQL Server 2008 Components
- Designing a Service Broker Architecture
- Designing the Service Broker Data Flow
- Designing the Service Broker Availability
- Exploring Full-Text Search
- Designing a Full-Text Search Strategy

Lab: Designing SQL Server 2008 Components

- Analyzing the Organizational Needs
- Designing a Service Broker Solution Model
- Designing a Detailed Service Broker Solution
- Implementing the Service Broker Solution

After completing this module, students will be able to:

- Examine the SQL Server 2008 architecture.
- Design a Service Broker architecture.
- Design the Service Broker data flow.
- Explore full-text search.
- Design a full-text search strategy.

Course Duration

- 5 Days
- include equipment, training material, lunch and refreshments

Tel: 0861 LGIT SS / 0861 544 877 • **Fax:** (011) 804 6483 • **E-mail:** info@lgit.co.za • www.lgit.co.za
3rd Floor, Bldg 27, "The Woodlands", Western Services Road, Woodmead • P.O. Box 4824, Rivonia 2128