



Broward Community College

Course Outline

STATUS: A

COMMON COURSE NUMBER: CTS2434C

COURSE TITLE: Programming a Microsoft SQL Server Database

CREDIT HOURS: 4

CONTACT HOURS BREAKDOWN:

Lecture/Discussion 48

Lab 16

Other

Contact Hours/Week 4

CATALOG COURSE DESCRIPTION:

This course provides students with the technical skills required to program a database solution by using Microsoft SQL Server. The skills developed by students completing this course will help prepare them for the Microsoft Programming a SQL Server Database certification exam.

Prerequisite: CTS1433C and (CEN1301C or CEN1509C)
(with grades of C or higher)

Corequisite: None

UNIT TITLES:

1. SQL Server Overview
2. Overview of Programming SQL Server
3. Creating and Managing Databases
4. Creating Data Types and Tables
5. Implementing Data Integrity
6. Planning Indexes
7. Creating Maintaining Indexes
8. Implementing Views
9. Implementing Stored Procedures
10. Implementing User-Defined Functions
11. Implementing Triggers
12. Programming Across Multiple Servers
13. Optimizing Query Performance
14. Analyzing Queries

15. Managing Transactions and Locks

I. Course Overview:

Upon successful completion of this course, the students should be able to describe the elements of SQL Server; design a SQL Server enterprise application architecture; describe the conceptual basis of programming in Transact-SQL; create and manage databases and their related components; implement data integrity by using the IDENTITY column property, constraints, defaults, rules, and unique identifiers; plan for the use of indexes, create and maintain indexes; create, use, and maintain data views; implement user-defined functions; design, create, and use stored procedures; create and implement triggers; program across multiple servers by using distributed queries, distributed transactions, and partitioned views; optimize query performance; analyze queries; and manage transactions and locks to ensure data concurrency and recoverability.

II. Units:

Unit 1. 1. SQL Server Overview

General Outcome:

1.0 The students should be able to discuss the concepts underlying SQL Server.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 1.1 Describe SQL Server and its supported operating system platforms.
- 1.2 Describe SQL Server Integration with Microsoft Windows and other server applications.
- 1.3 Describe SQL Server databases.
- 1.4 Describe SQL Server security.
- 1.5 Describe SQL Server administration and implementation activities, as well as SQL Server application design options.

Unit 2. 2. Overview of Programming SQL Server

General Outcome:

2.0 The students should be able to discuss programming in a SQL Server environment.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 2.1 Describe the concepts of enterprise-level application architecture.
- 2.2 Describe the primary SQL Server programming tools.
- 2.3 Explain the difference between the two primary programming tools in SQL Server.
- 2.4 Describe the basic elements of Transact-SQL.
- 2.5 Describe the use of local variables, operators, functions, control of flow statements, and comments.
- 2.6 Describe the various way to execute Transact-SQL statements

Unit 3. 3. Creating and Managing Databases

General Outcome:

3.0 The students should be able to create and manage a SQL Server database.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 3.1 Create a database.
- 3.2 Create a filegroup.
- 3.3 Manage a database.
- 3.4 Describe data structures.

Unit 4. 4. Creating Data Types and Tables

General Outcome:

4.0 The students should be able to create data types and tables.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 4.1 Create and drop user-defined data types.
- 4.2 Create and drop user tables.
- 4.3 Generate column values.
- 4.4 Generate a script.

Unit 5. 5. Implementing Data Integrity

General Outcome:

5.0 The students should be able to discuss and employ data integrity techniques.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 5.1 Describe the types of data integrity.
- 5.2 Describe the methods to enforce data integrity.
- 5.3 Determine which constraint to use and create constraints.
- 5.4 Define and use DEFAULT, CHECK, PRIMARY KEY, and UNIQUE.
- 5.5 Disable constraints.
- 5.6 Describe and use defaults and rules.
- 5.7 Determine which data integrity enforcement methods to use.

Unit 6. 6. Planning Indexes

General Outcome:

6.0 The students should be able to explain the purposes of indexes.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 6.1 Describe why and when to use and index.
- 6.2 Describe how SQL Server uses clustered and nonclustered indexes.
- 6.3 Describe how SQL Server index architecture facilitates the retrieval of data.
- 6.4 Describe how SQL Server maintains indexes and heaps.
- 6.5 Describe the importance of selectivity, density, and distribution of data when deciding which columns to index.

Unit 7. 7. Creating and Maintaining Indexes

General Outcome:

7.0 The students should be able to create and employ indexes.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 7.1 Create indexes and indexed views with unique or composite characteristics.
- 7.2 Use the CREATE INDEX options.
- 7.3 Describe how the query optimizer creates, stores, maintains, and uses statistics to optimize queries.
- 7.4 Query the sysindexes table.
- 7.5 Describe how the Index Tuning Wizard works and when to use it.
- 7.6 Describe performance considerations that affect creating and maintaining indexes.

Unit 8. 8. Implementing Views

General Outcome:

8.0 The students should be able to effectively use views.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 8.1 Describe the concept of a view.
- 8.2 List the advantages of views.
- 8.3 Define a view with the CREATE VIEW statement.
- 8.4 Modify data through views.
- 8.5 Optimize performance by using views.

Unit 9. 9. Implementing Stored Procedures

General Outcome:

9.0 The students should be able to effectively create and use stored procedures.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 9.1 Describe how a stored procedure is processed.
- 9.2 Create, execute, modify, and drop a stored procedure.
- 9.3 Execute extended stored procedures.
- 9.4 Create custom error messages.

Unit 10. 10. Implementing User-defined Functions

General Outcome:

10.0 The students should be able to create user-defined functions.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

10.1 Describe the three types of user-defined functions.

10.2 Create and alter user-defined functions.

10.3 Create each of the three types of user-defined functions.

Unit 11. 11. Implementing Triggers

General Outcome:

11.0 The students should be able to create and effectively use triggers.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

11.1 Create a trigger.

11.2 Drop a trigger.

11.3 Alter a trigger.

11.4 Describe how various triggers work.

11.5 Evaluate the performance considerations that affect using triggers.

Unit 12. 12. Programming Across Multiple Servers

General Outcome:

12.0 The students should be able to work in a distributed database environment.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 12.1 Describe distributed queries.
- 12.2 Write ad hoc queries that access data that is stored in a remote SQL Server or in an OLE DB data source.
- 12.3 Set up a linked server environment to access data that is stored in a remote SQL Server or in an OLE DB data source.
- 12.4 Write queries that access data from a linked server.
- 12.5 Execute stored procedures on a remote server or linked server.

Unit 13. 13. Optimizing Query Performance

General Outcome:

13.0 The students should be able to effectively optimize the performance of queries.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 13.1 Explain the role of the query optimizer and how it works to ensure that queries are optimized.
- 13.2 Use various methods for obtaining execution plan information so that they can determine how the query optimizer processed a query and validate that the most efficient query plan was generated.
- 13.3 Create indexes that cover queries.
- 13.4 Identify indexing strategies that reduce page reads.
- 13.5 Evaluate when to override the query optimizer.

Unit 14. 14. Analyzing Queries

General Outcome:

14.0 The students should be able to analyze the performance of optimized operations.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

14.1 Analyze the performance gain of writing efficient queries and creating useful indexes for queries that contain the AND logical operator.

14.2 Analyze the performance gain of writing efficient queries and creating useful indexes for queries that contain the OR logical operator.

14.3 Evaluate how the query optimizer uses different join strategies for query optimization.

Unit 15. 15. Managing Transactions and Locks

General Outcome:

15.0 The students should be able to effectively use locks on database resources.

Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 15.1 Describe transaction processing.
- 15.2 Execute, cancel, or roll back a transaction.
- 15.3 Identify locking concurrency issues.
- 15.4 Identify resource items that can be locked and the types of locks.
- 15.5 Describe lock compatibility.
- 15.6 Describe how SQL Server uses dynamic locking.
- 15.7 Set locking options and display locking information.