I. **OVERVIEW**
The following information will appear in the 2010 - 2011 catalog

**CMPSC 225 Database Programming with SQL** 3 Units

*Formerly listed as:* CMPSC - 225: SQL Database Implementation  
*Prerequisite:* Satisfactory completion of CMPSC 275 or CMPSC 204.

Provides students with the technical skills required to implement a database solution with SQL Server. Topics include: architecture, key features of SQL Server, reviewing SQL Server programming tools, Transact-SQL, creating databases, data integrity, planning and creating indexes, advanced query techniques, summarizing data, managing transactions and locks, implementing views, stored procedures and triggers, working with distributed data, and advanced text queries.

Three maximum completions.  
Field trips are not required. (A-F or P/NP - Student choice) Lecture /Lab  
Transfer: (CSU)

II. **LEARNING CONTEXT**
Given the following learning context, the student who satisfactorily completes this course should be able to achieve the goals specified in Section III, Desired Learning:

A. **COURSE CONTENT**

1. **Required Content:**

   a. Developing a logical data model
      i. Types of integrity
      ii. Enforcing data integrity
      iii. Using constraints, defaults and rules
      iv. Relations and normalization

   b. Using Transact-SQL
      i. Manipulate data by using Transact-SQL cursors
      ii. Elements of Transact-SQL
      iii. Type of cursor Sequence
      iv. Scope of cursor
      v. Ways to execute Transact-SQL statements
      vi. Processing queries and subqueries
      vii. SQL commands that retrieve and modify data (insert, select, delete, update)
      viii. Error checking
c. Planning and creating indexes
   i. Index architecture
   ii. Creating and maintaining indexes

d. Creating the physical database
   i. Disk allocation
   ii. Extent table size allocation
   iii. Redo/rollback log file space allocation

e. Create and manage files, filegroups, and transaction logs that define a database.

f. Working with distributed data
   i. Setting up a distributed environment
   ii. Querying a linked server
   iii. Executing a stored procedure on a linked server
   iv. Modifying distributed data
   v. Managing distributed data

g. Views, implementing and altering

h. Implementing stored procedures
   i. Defining triggers and performance considerations
   j. Working with distributed data and linked servers

2. **Required Lab Content:**

   In lab, students perform the functions listed in the lecture content section. Assignments are given that require application of the content.

   a. Developing a logical data model
      i. Types of integrity
      ii. Enforcing data integrity
      iii. Using constraints, defaults and rules
      iv. Relations and normalization

   b. Using Transact-SQL
      i. Manipulate data by using Transact-SQL cursors
      ii. Elements of Transact-SQL
      iii. Type of cursor Sequence
iv. Scope of cursor
v. Ways to execute Transact-SQL statements
vi. Processing queries and subqueries
vii. SQL commands that retrieve and modify data (insert, select, delete, update)
viii. Error checking
c. Planning and creating indexes
   i. Index architecture
   ii. Creating and maintaining indexes
d. Creating the physical database
   i. Disk allocation
   ii. Extent table size allocation
   iii. Redo/rollback log file space allocation
e. Create and manage files, filegroups, and transaction logs that define a database.
f. Working with distributed data
   i. Setting up a distributed environment
   ii. Querying a linked server
   iii. Executing a stored procedure on a linked server
   iv. Modifying distributed data
   v. Managing distributed data
g. Views, implementing and altering
h. Implementing stored procedures
   i. Defining triggers and performance considerations
   j. Working with distributed data and linked servers

B. ENROLLMENT RESTRICTIONS

1. Prerequisites
   Satisfactory completion of CMPSC 275 or CMPSC 204.

2. Requisite Skills
   Before entering the course, the student will be able to:
   
   a. Explain the components of a micro computer system.
   
   b. Describe the functional difference between systems software and applications software.
c. Demonstrate how the Windows operating system is used to manage the computer system.

d. Describe the types of software used for programming development such as editors, translators/compilers, linkers, debuggers, and demonstrate operating proficiency with each type of development tool

C. HOURS AND UNITS

<table>
<thead>
<tr>
<th>INST METHOD</th>
<th>TERM HOURS</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lect</td>
<td>36</td>
<td>2.00</td>
</tr>
<tr>
<td>Lab</td>
<td>54</td>
<td>1.00</td>
</tr>
</tbody>
</table>

D. METHODS OF INSTRUCTION (TYPICAL)

Instructors of the course might conduct the course using the following method:

1. Formal lectures and/or internet assisted presentation
2. Assigned reading and discussion of required text
3. Assigned reading of supplemental reference materials
4. Implementation of computer programming laboratory projects
5. Create scripts by using Transact-SQL. Design, create, use, and alter views
6. Group discussions and in-class projects

E. ASSIGNMENTS (TYPICAL)

1. EVIDENCE OF APPROPRIATE WORKLOAD FOR COURSE UNITS

Time spent on coursework in addition to hours of instruction (lecture hours)

On a weekly basis students will:

a. Read assigned textbook content
b. Read online assigned or research content
c. Complete computer exercises

Periodically students will:

a. Complete hands-on computer lab assignments
b. Research pertinent topics and report on findings
c. Take quizzes and tests to evaluate retention
d. Execute group activities designed to build teamwork skills

2. EVIDENCE OF CRITICAL THINKING

Assignments require the appropriate level of critical thinking

Assignment Example:
Develop a complete Entity Relationship Diagram for a fictitious business that sells pizzas, drinks and...
a few other entries like spaghetti, lasagna, and garlic bread. We will design the database that takes care of the central business of making and selling pizza. The database needs to be able to:

Track our products (pizzas, spaghetti, etc.)
Track types of products sold (drinks, sides, pizzas, entrees, etc.)
Track the sales of our products to customers (sales)
Track customers and their purchases for future direct-advertising
Track delivery data about customers that holding customer delivery location(s)
Track raw inventory (the products used to produce pizzas and other food)

Using the same method we used in class create a basic sketch on paper listing the different entities required to solve the above problem. Draw lines between the entities and label how they relate to one another. Remember that primary keys are required for each entity and that tables that relate will share their primary key as a foreign key in the related table.

Evaluate what kind of data will be stored for each entity property and select data type that best suits the value.

Exam Question Examples:

1. ___ is an example of the use of DDL in SQL.
   a. Select * From tableOne
   b. Delete from tableOne where field1 = 0
   c. Create table tableOne...
   d. none of the above

2. ___ is the process of removing redundant data from database design.
   a. indexing
   b. relating
   c. normalizing
   d. rationalizing

F. TEXTS AND OTHER READINGS (TYPICAL)


III. DESIRED LEARNING

A. COURSE GOAL
   As a result of satisfactory completion of this course, the student should be prepared to:

   articulate the rules of normalization; create and manage explicit, implicit, and distributed transactions to ensure data consistency and recoverability; write INSERT, DELETE, UPDATE, and SELECT statements that retrieve and modify data; formulate Transact-SQL statements that use joins or sub queries to combine data from multiple tables; and construct tables that enforce data integrity and referential integrity.

B. STUDENT LEARNING GOALS
   Mastery of the following learning goals will enable the student to achieve the overall course goal.

   1. Required Learning Goals
      Upon satisfactory completion of this course, the student will be able to:

      a. Construct a Logical Data Model by doing the following:
         i. Group data into entities by applying normalization rules
         ii. Identify primary keys
         iii. Choose the foreign key that will enforce a relationship between entities and that will ensure referential integrity Systems variables
         iv. Identify the business rules that relate to data integrity.
         v. Choose whether denormalization is appropriate
b. Apply the rules of normalization

c. Manage data by using Transact_SQL cursors

d. Create and manage explicit, implicit, and distributed transactions to ensure data consistency and recoverability.

e. Write INSERT, DELETE, UPDATE, and SELECT statements that retrieve and modify data.

f. Formulate Transact-SQL statements that use joins or subqueries to combine data from multiple tables.

g. Create and execute stored procedures to enforce business rules

h. Implement error handling by using return codes and the RAISERROR statement

i. Create and execute stored procedures and triggers

j. Evaluate where processing occurs when using OPENQUERY and access data from static or dynamic sources.

k. Define a database
   1. Create and manage files
   2. Filegroups
   3. Transaction logs

l. Create tables that enforce data integrity and referential integrity.

m. Create and maintain indexes.

n. Evaluate and optimize the performance of an execution plan by using DBCC (Database Consistency Checker).

o. Use distributed data

2. Lab Learning Goals

   Upon satisfactory completion of the lab portion of this course, the student will be able to:

   a. Construct a Logical Data Model by doing the following: i. Group data into entities by applying normalization rules ii. Identify primary keys iii. Choose the foreign key that will enforce a relationship between entities and that will ensure referential integrity Systems variables iv. Identify the business rules that relate to data integrity. v. Choose whether denormalization is appropriate

   b. Apply the rules of normalization

   c. Manage data by using Transact_SQL cursors

   d. Create and manage explicit, implicit, and distributed transactions to ensure data consistency and recoverability

   e. Write INSERT, DELETE, UPDATE, and SELECT statements that retrieve and modify data

   f. Formulate Transact-SQL statements that use joins or subqueries to combine data from multiple tables

   g. Create and execute stored procedures to enforce business rules

   h. Implement error handling by using return codes and the RAISERROR statement

   i. Create and execute stored procedures and triggers

   j. Evaluate where processing occurs when using OPENQUERY and access data from static or dynamic sources
k. Define a database i. Create and manage files ii. Filegroups iii. Transaction logs
l. Create tables that enforce data integrity and referential integrity
m. Create and maintain indexes
n. Evaluate and optimize the performance of an execution plan by using DBCC (Database Consistency Checker)
o. Use distributed data
p. SECOND COMPLETION:
q. demonstrate updated skills reflecting current industry standards as software tools, interface and functions evolve in new versions.
r. THIRD COMPLETION:
s. demonstrate updated skills reflecting current industry standards as software tools, interface and functions evolve in new versions.

IV. METHODS OF ASSESSMENT (TYPICAL)

A. FORMATIVE ASSESSMENT
   1. Assignments
   2. Projects/Labs
   3. Exams/Quizzes

B. SUMMATIVE ASSESSMENT
   1. Assignments
   2. Projects/Labs
   3. Exams/Quizzes