

## Course Competencies Template - Form 112

GENERAL INFORMATION			
Name:	Phone #:		
Course Prefix/Number: CTS2433	Course Title: Microsoft SQL Implementation		
Number of Credits: 4			
Degree Type	<input type="checkbox"/> B.A. <input type="checkbox"/> B.S. <input type="checkbox"/> B.A.S <input type="checkbox"/> A.A. <input checked="" type="checkbox"/> A.S. <input type="checkbox"/> A.A.S. <input type="checkbox"/> C.C.C. <input type="checkbox"/> A.T.C. <input type="checkbox"/> C.T.C.(V.C.C.)		
Date Submitted/Revised: 10-05-2010	Effective Year/Term: 2011-1		
<input type="checkbox"/> New Course Competency <input checked="" type="checkbox"/> Revised Course Competency			
Course to be designated as a General Education course (part of the 36 hours of A.A. Gen. Ed. coursework): <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No			
The above course links to the following Learning Outcomes:			
<table style="width: 100%; border: none;"> <tr> <td style="width: 50%; vertical-align: top;"> <input type="checkbox"/> Communication  <input checked="" type="checkbox"/> Numbers / Data  <input checked="" type="checkbox"/> Critical thinking  <input type="checkbox"/> Information Literacy  <input type="checkbox"/> Cultural / Global Perspective               </td> <td style="width: 50%; vertical-align: top;"> <input type="checkbox"/> Social Responsibility  <input type="checkbox"/> Ethical Issues  <input checked="" type="checkbox"/> Computer / Technology Usage  <input type="checkbox"/> Aesthetic / Creative Activities  <input type="checkbox"/> Environmental Responsibility               </td> </tr> </table>		<input type="checkbox"/> Communication <input checked="" type="checkbox"/> Numbers / Data <input checked="" type="checkbox"/> Critical thinking <input type="checkbox"/> Information Literacy <input type="checkbox"/> Cultural / Global Perspective	<input type="checkbox"/> Social Responsibility <input type="checkbox"/> Ethical Issues <input checked="" type="checkbox"/> Computer / Technology Usage <input type="checkbox"/> Aesthetic / Creative Activities <input type="checkbox"/> Environmental Responsibility
<input type="checkbox"/> Communication <input checked="" type="checkbox"/> Numbers / Data <input checked="" type="checkbox"/> Critical thinking <input type="checkbox"/> Information Literacy <input type="checkbox"/> Cultural / Global Perspective	<input type="checkbox"/> Social Responsibility <input type="checkbox"/> Ethical Issues <input checked="" type="checkbox"/> Computer / Technology Usage <input type="checkbox"/> Aesthetic / Creative Activities <input type="checkbox"/> Environmental Responsibility		
Course Description (limit to 50 words or less, <b>must correspond with course description on Form 102</b> ):  A comprehensive course in learning how to design and implement enterprise database solutions using SQL. Working through a system of modular lessons and hands-on labs to comprehend SQL Architecture. Prerequisite: CTS1437. Laboratory fee. A.S. degree credit only. (3 hr. lecture; 2 hr. lab).			
Prerequisite(s): CTS1437	Co requisite(s):		

**Course Competencies:** (for further instruction/guidelines go to: <http://www.mdc.edu/asa/curriculum.asp>)

Competency 1: The student will demonstrate an understanding of how to implement tables and views by:

1. Creating and altering tables.
2. Implementing computed and persisted columns, schemas and scripts to deploy changes to multiple environments, (e.g., dev, test, production; Managing permissions (GRANT, DENY, REVOKE).
3. Creating and altering views using encryption, schema, check options and permissions (GRANT, DENY, REVOKE).
4. Creating and altering indexes.
5. Implementing filtered index, including columns, unique, clustered, non-clustered, fill factor, creating statistics and indexing views.
6. Creating and modifying constraints.
7. Implementing PRIMARY KEY, FOREIGN KEY, UNIQUE, CHECK, cascading referential integrity, enabling/disabling, NOCHECK; SET IDENTITY\_INSERT.
8. Using FILESTREAM for unstructured data storage.
9. Differentiating between structured, and semi-structured FILESTREAM.
10. Explaining when and why to keep *BLOB* in the database with all structured relational data or store them outside the database.
11. Implementing partitioning solutions.
12. Explaining how to use partitioned tables and indexes (constraints, partition functions, partition schemes, *MERGE, SPLIT, SWITCH*), and distributed partitioned views (constraints, linked servers).

**Revision Date** 06-28-2011

Approved By Academic Dean Date: \_\_\_\_\_

Reviewed By Director of Academic Programs Date: \_\_\_\_\_

Competency 2: The student will demonstrate an understanding of how to program with T-SQL by:

1. Creating and altering stored procedure, including table-valued parameters (TVPs), *EXECUTE AS*, *RECOMPILE*, parameter direction (output); *WITH ENCRYPTION*; Managing permissions (*GRANT*, *DENY*, *REVOKE*).
2. Creating and altering user-defined functions (UDFs).
3. Managing permissions (*GRANT*, *DENY*, *REVOKE*).
4. Creating and altering DML triggers, including *INSERTED*, *DELETED*, *INSTEAD OF*, *EXECUTE AS*.
5. Creating and altering DDL triggers, including enabling/disabling; returning event data.
6. Creating and deploying CLR-based objects, including permission sets, such as *SAFE*, *UNSAFE*, *EXTERNAL\_ACCESS*, *SET TRUSTWORTHY*.
7. Implementing error handling, including *TRY/CATCH*, *RAISERROR*, retrieving error information, custom error messages, *@@ERROR*.
8. Managing transactions, including *BEGIN TRANSACTION*, *COMMIT*, *ROLLBACK*, *SET TRANSACTION ISOLATION LEVEL*.

Competency 3. The student will demonstrate an understanding of how to work with query fundamentals by:

1. Using *SELECT* statements to retrieve data, including *LIKE*, *WHERE*, *ORDER BY*, *INTO*.
2. Modifying data by using *INSERT*, *UPDATE*, and *DELETE* statements.
3. Using the *OUTPUT* clause to return data.
4. Modifying data by using *MERGE* statements.
5. Implementing aggregate queries, including built-in aggregate functions, *GROUPING SETS*, *GROUP BY*, *HAVING*.
6. Combining datasets using functions including *CROSS APPLY*, *OUTER APPLY*, all join types; *UNION*, *UNION ALL*, *INTERSECT*, *EXCEPT*.
7. Applying built-in scalar functions, such as *CAST* and *CONVERT*; *REPLACE*; *LEN* and *DATALength*; *PATINDEX* and *CHARINDEX*.

Competency 4. The student will demonstrate an understanding of additional query techniques by:

1. Implementing subqueries including simple, correlated, scalar, list, table valued.
2. Implementing CTE (common table expression) queries including recursive and non-recursive.
3. Applying ranking functions, such as *RANK*, *PARTITION BY*, *DENSE\_RANK*, *OVER*, *ROW\_NUMBER*, *NTILE*.
4. Controlling execution plans, including table hints and query hints.
5. Managing international considerations such as collations, defining custom errors, filtering data, sort order, nvarchar, database collation, column collation.

Competency 5. The student will demonstrate an understanding of how to work with additional SQL server components by:

1. Integrating Database Mail.
2. Implementing full-text search utilizing *CONTAINS*, *CONTAINSTABLE*, *FREETEXT*, *FREETEXTTABLE*, *STOPLIST*, etc.
3. Implementing scripts by using Windows PowerShell and SQL Server Management Objects (SMOs).
4. Implementing Service Broker solutions, including services, queues, messages, message types, message validation, contracts, and activation procedures.
5. Tracking data changes using change tracking, database audit specification, *CHANGETABLE*, etc.

Revision Date 06-28-2011

Approved By Academic Dean Date: \_\_\_\_\_

Reviewed By Director of Academic Programs Date: \_\_\_\_\_

Competency 6. The student will demonstrate an understanding of how to work with XML data by:

1. Retrieving relational data as XML.
2. Transforming XML data into relational data.
3. Querying XML data.
4. Managing XML data.

Competency 7. The student will demonstrate an understanding of how to gather performance information by:

1. Capturing execution plans, including graphical execution plans and using *SHOWPLAN*.
2. Gathering trace information by using the SQL Server Profiler.
3. Collecting output from the Database Engine Tuning Advisor, including preparing a workload.
4. Collecting information from system metadata utilizing Dynamic Management Views (DMVs), catalog views, etc.

**Revision Date** 06-28-2011

Approved By Academic Dean Date: \_\_\_\_\_

Reviewed By Director of Academic Programs Date: \_\_\_\_\_