School of Undergraduate Studies

COURSE SYLLABUS

Course Number:	**CIS 206**
Course Name:	**Introduction to Relational Database Management Systems**
Date:	**March 2011**
Quarter:	**q2**
Number of Credits:	**4.5 Credits**
Prerequisite:	**None**
Instructor(s):	**Myra Anson Nicholas**
E-mail:	**coyote@nanosite.com**
Office Phone:	**(703) 536-3013 (cell-703-402-9065)**

COURSE DESCRIPTION

This course is designed for students with limited or no previous database experience. Course outcomes include a solid understanding of fundamental database terms and concepts such as tables, queries, forms and reports, and their application using a popular database. This course also introduces database analysis, database design, and N-tiered client-server database systems. A problem-based approach using SQL is used in this course.

EXPECTED LEARNING OUTCOMES (IN BEHAVIORAL/MEASUREMENT TERMS)

After completing this course, students will be able to:

- Outline the fundamentals of the relational model;
- Construct Structured Query Language (SQL)
- Analyze Data modeling, Database design and Database administration

TEXTBOOKS & RESOURCES

Required:

Adamski, J.J. ***New Perspectives on Microsoft Office Access 2003, Comprehensive* (2nd Ed.).**	**ISBN:** **9780619268114**

In addition to your textbook, you are strongly encouraged to take advantage of available resource materials to enrich your knowledge.

LIBRARY USAGE

Students are encouraged to use the library resources throughout the course of their study program. There are multiple research resources available to Stratford University students to supplement classroom instruction and assigned textbooks.

Each Stratford University campus has a large library with computer terminals and dedicated library staff. Both shelf racked books and online texts are available for students, staff, and faculty.

The resources along with the information about library holdings can also be accessed online through Stratford University website www.stratford.edu

Click on: Current Student > Library Online > EBooks Online > Net Library

Username: student
Password: stratford

EBSCOhost is a powerful online reference system accessible via the Internet. It offers a variety of proprietary full text databases and popular databases from leading information providers. EBSCOhost is accessed through the Stratford home page at www.stratford.edu

Click on: Current Student > Library Online > Research Database, follow the link

Username: gradmba
Password: gradmbasu1

COURSE OUTLINE

March 18	**The textbook covers MS Access and uses wizards which you can follow easily for assignments.**	**In this class we are also going to expand on this knowledge, add theory, use real-world examples to explore other RDBMS systems, and cover more SQL principles.**
Week One	**Course Introduction** **Ch 1 – Getting Started** A. What is database B. Why use it C. Summary A DBMS, has to be persistent, that is it should be accessible when the program created the data ceases to exist or even the application that created the data restarted. A DBMS also has to provide some uniform methods independent of a specific application for accessing the information that is stored. RDBMS adds the additional condition that the system supports a tabular structure for the data, with enforced relationships between the tables. This excludes the databases that don't support a tabular structure or don't enforce relationships between tables.	Using data: the right application for result you need: • Spreadsheets (for if/or decisions) • Flat Database (for lists sorted by 1 field in a record.. limited by sort fields) • Relational Database (RDBMS) Ask many different questions using data/generate relevant results with alternate views. Used for strategy. http://databases.about.com/od/specificproducts/a/whatisadatabase.htm Read definition of Databases A database is a collection of data that is organized so that its contents can be easily accessed, managed and updated. The software used to manage and query a database is known as a Database Management System (DBMS). A Relational Database Management System(RDBMS) is a database where data are stored in more than one table, each one containing different types of data. www.hscripts.com/tutorials/mysql/overview.php
Week Two	**The Relational Model** A. Relations and Keys B. Normalization C. Dependencies D. Null values *Access: AC5 & AC6* *Field Types: AC 39* *Primary Key AC 50*	Slightly deeper into DBMS and RDBMS/explain how strategy can be extracted and used for corporate planning. Logical Data Modeling How model works Show primary key/foreign keys/how relations displayed/tables/why null values/ Begin CD Model http://students.anson-bowles.com/database-design.html
Week Three	**Creating and maintaining a Database** Read Tutorial 2 (AC 35-AC 80)	Using normalize to eliminate redundancies The use of relationships in a RDBMS. Creating Database from Data Model Using Primary Keys, Foreign Keys http://databases.about.com/od/specificproducts/a/keys.htm Properties of entities, attributes

Week Four	**Structured Query Language** A. SQL data definition B. SQL Relational Query **C.** Relational Data Modification **Read SQL Tutorial (p 517)**	Introduce SQL as base for DBs • Relational operators • SQL Comands/Tables • Inserts/Joins/etc. SQL Tutorials: www.w3schools.com/sql/sql_intro.asp
Week Five	**Mid-term exam** **Data Modeling and Entity – Relationship Model** A. Requirement Stage B. Entity – Relationship Model	SQL basics/RDBMS definitions/ Using Model from week 3: Using CD collection with multiple artists/songs/labels as subject create model/what are entities/attributes Read Top Five Things Beginners Need to Know About Databases http://databases.about.com/od/administration/tp/beginners_faq.htm
Week Six	**Database Design** A. Representing entities B. Representing relationships C. Example of database design	Using model: show relationships Convert model into tables: how many tables are needed to eliminate redundancies? Basics of Database Normalization: http://databases.about.com/od/specificproducts/a/normalization.htm
Week Seven	**Dynamic Database pages** Read Tutorial 8 A. Data Access tables (p. 369) B. Pivot Tables (p. 392)	Creating web-based data tables for review, updating and analysis in Access.
Week Eight	**Database Environment** A. SQL Commands B. Using Query with Calculation C. Representing Calculations	If you want your web site to be able to store and display data from a database, your web server should have access to a database system that uses the SQL language.
Week Nine	**Advanced topics** A. Web database processing B. How is a database used in content management systems and other web apps?	Show PHP and content management systems for web applications. White paper on data mining and distributed processing
Week Ten	**Final Exam** Review RDBMS Structure and concepts: RDBMS RULES	

Information Rule

All information in the database should be represented in one and only one way -- as values in a table.

Guaranteed Access Rule

Each and every datum (atomic value) is guaranteed to be logically accessible by resorting to a combination of table name, primary key value, and column name.

Systematic Treatment of Null Values

Null values (distinct from empty character string or a string of blank characters and distinct from zero or any other number) are supported in the fully relational DBMS for representing missing information in a systematic way, independent of data type.

Dynamic Online Catalog Based on the Relational Model

The database description is represented at the logical level in the same way as ordinary data, so authorized users can apply the same relational language to its interrogation as they apply to regular data.

Comprehensive Data Sublanguage Rule

A relational system may support several languages and various modes of terminal use. However, there must be at least one language whose statements are expressible, per some well-defined syntax, as character strings and whose ability to support all of the following is comprehensible:

a. data definition
b. view definition
c. data manipulation (interactive and by program)
d. integrity constraints
e. authorization
f. transaction boundaries (begin, commit, and rollback).

View Updating Rule

All views that are theoretically updateable are also updateable by the system.

High-Level Insert, Update, and Delete

The capability of handling a base relation or a derived relation as a single operand applies not only to the retrieval of data, but also to the insertion, update, and deletion of data.

Physical Data Independence

Application programs and terminal activities remain logically unimpaired whenever any changes are made in either storage representation or access methods.

Logical Data Independence

Application programs and terminal activities remain logically unimpaired when information preserving changes of any kind that theoretically permit unimpairment are made to the base tables.

Integrity Independence

Integrity constraints specific to a particular relational database must be definable in the relational data sublanguage and storable in the catalog, not in the application programs.

Distribution Independence

The data manipulation sublanguage of a relational DBMS must enable application programs and terminal activities to remain logically unimpaired whether and whenever data are physically centralized or distributed.

Nonsubversion Rule

If a relational system has or supports a low-level (single-record-at-a-time) language, that low-level language cannot be used to subvert or bypass the integrity rules or constraints expressed in the higher-level (multiple-records-at-a-time) relational language.