Syllabus

Instructor Information

Name	Raymond J. Curts, PhD
Email	rcurts@gmu.edu
Office location	Science & Tech II, Room 435
Office hours	By Appointment
Phone	703-395-9143
Teaching assistants	TBD

Course Information

Course title	Database & Object Oriented Concepts
Course number	CS 450 S02 - Spring 2002
Course discipline	Computer and Information Science
Course description	This is an introductory course covering basic to intermediate design, implementation and use of database systems. The main topics include the Entity-Relationship (ER) and Enhanced Entity-Relationship (EER) models, database design, Relational Algebra (RA), Structured Query Language (SQL), functional dependencies, normalization, and a brief introduction to object-oriented databases, data warehousing and data mining. Students practice designing, developing and implementing a relational database and use the database for transaction processing, report generation, and queries. This course will provide a good understanding of data modeling and mapping from a data model to both relational and object-oriented models. Over half of the course focuses on the relational model as it provides the most rigorous presentation of data analysis. We will examine the various data modeling methods against some basic software engineering metrics. This will show why this course has such a focus on the relational model. It helps explain why knowledge of relational technology is so much in demand by industry. By the end of the course, the student will be able to: · Understand the relational model · Extract a data model from a real-world system · Design a relational implementation of the data model · Competently navigate using SQL You will have a basic understanding of: · Oracle · Object-oriented databases · Distributed databases · Data Warehouses · Relational database optimization Implementation will be covered to a level that you have an understanding of

	how commercial databases work and why they work the way they do. You will not be expected to learn all of the nuances of SQL, have an instant recall of relational algebra, nor an in-depth understanding of the third-order implementation effects. You won't get a job at Oracle as a Senior Architect or at Fannie Mae as a Principal DBA. On the other hand, with hands-on database experience, you will be prepared to interview for a job as an analyst, data engineer, programmer, or DBA.
Course date	Jan 23, 2002 through May 8, 2002
Location	Science & Tech, Room 122
Meeting day(s)	Wednesday
Meeting time(s)	7:20 PM - 10:00 PM
Prerequisite(s)	You must be comfortable in set theory and mathematical logic. If equations with lots of curly braces and Greek symbols frighten you, you will struggle with the first quarter of the course. You must be a competent programmer. You will have to be able to navigate such software as Windows, Unix, MS Access, Oracle, JAVA, Visual Basic and C/C++. You will become familiar with Oracle as needed through the class. You need a basic understanding of how operating systems work. In particular, you need to have some idea of how scheduling works and how CPU's interact with disk subsystems.

Textbooks

Required reading	*Fundamentals of Database Systems*, Elmasri, R. and Navathe, S., Addison-Wesley, 3rd Ed, 2000, 0-8053-1755-4
Required reading	*Oracle™ Programming: a Primer*, Sunderraman, R., Addison-Wesley, 2nd Printing, 1999.

Recommended Reading

Recommended reading	*Relational Database Design Clearly Explained*, Harrington, Jan, AP Professional, 1998
Recommended reading	*An Introduction to Database Systems*, Date, C. J., Addison Wesley, 7th Ed., 2000
Recommended reading	*SQL Clearly Explained*, Harrington, Jan, AP Professional, 1997

Recommended Reading Continued

Recommended reading	*Database System Concepts*, Silbershatz, Abraham, Korth, Henry F. and Sudarshan, S., 4th Edition
Recommended reading	*A First Course in Database Systems*, Ulman, J.D. and Widom, J., Prentice Hall, 1997

Links

Recommended	*Oracle Primer*, http://ite.gmu.edu/machines/oracle.html

reading	
Recommended reading	*Oracle Account Setup*, oracle@ite.gmu.edu

Schedule

The course schedule will follow the outline below.

There may be a guest lecturer from industry for one or two classes as time permits.

The assignment schedule may be rearranged as the term progresses. Homework is assigned throughout the semester as indicated.

Lesson 1

Lesson	Introduction to Databases
Date	Jan 23, 2002
Topics	Introduction to CS 450. Overview and basic terminology.
Readings	Elmasri, Chapters 1 & 2.

Lesson 2

Lesson	Data Modeling: ERD
Date	Jan 30, 2002
Topics	Introduction to the Entity Relationship Model and ERDs.
Readings	Elmasri, Chapter 3.

Lesson 3

Lesson	Data Modeling: EERD
Date	Feb 6, 2002
Topics	More on the Entity Relationship Model and ERDs. Introduction to the Enhanced Entity Relationship Model and EERDs.
Readings	Elmasri, Chapters 4, 9.1 & 9.2
Assignments	Self Study: Storage, indexing and primary file organization. Elmasri, Chapters 5 & 6

Lesson 4

Lesson	Relational Data Model & Algebra
Date	Feb 13, 2002
Topics	The Relational Data Model and Relational Algebra.
Readings	Elmasri, Chapter 7. Sunderraman, Chapter 1
Assignments	Homwork #1 due.

Lesson 5

Lesson	Relational Alegbra and SQL
Date	Feb 20, 2002
Topics	Relational Algebra SQL: DML
Readings	Elmasri, Chapter 8

Lesson 6

Lesson	SQL: DML I & II
Date	Feb 27, 2002
Topics	SQL: DML I & II
Readings	Elmasri, Chapter 8 Sunderraman, Chapter 2
Assignments	Homwork #2 due.

Lesson 7

Lesson	Embedded SQL
Date	Mar 6, 2002
Topics	Embedded SQL and Oracle Programming Review for Mid Term Exam
Readings	Elmasri, Chapter 10 Sunderraman, Chapter 3

Spring Break

Date	Mar 13, 2002
Topics	Spring Break No Class

Lesson 8

Lesson	Mid Term Examination
Date	Mar 20, 2002
Topics	Mid Term Examination
Readings	Elmasri, Chapters 1 - 10 Lectures 1 - 7
Assignments	Homework #3 due.

Lesson 9

Lesson	Functional Dependencies
Date	Mar 27, 2002
Topics	Functional Dependencies Normalization
Readings	Elmasri, Chapter 14

Lesson 10

Lesson	Object Oriented Concepts
Date	Apr 3, 2002

Topics	Object Oriented Database Concepts Object Database Standards, Languages and Design
Readings	Elmasri, Chapters 4.6, 4.8, 11, 12, 13
Assignments	Homework #4 due.

Lesson 11

Lesson	Database Design / Architecture
Date	Apr 10, 2002
Topics	Database Design Tuning System Catalog Introduction to Query Processing and Optimization
Readings	Elmasri, Chapters 16, 17 & 18
Assignments	Homework #5 due.

Lesson 12

Lesson	Distributed Databases
Date	Apr 17, 2002
Topics	Distributed Databases and Client Server Architecture
Readings	Elmasri, Chapter 18

Lesson 13

Lesson	Data Warehousing & Data Mining
Date	Apr 24, 2002
Topics	Introduction to data warehousing and data mining.
Readings	Elmasri, Chapter 26
Assignments	Homework #6 due.

Lesson 14

Lesson	Review
Date	May 1, 2002
Topics	Review for Final Examination
Readings	Elmasri, Chapters 11 - 14, 16 - 18, 26

Lesson 15

Lesson	Final Examination
Date	May 8, 2002
Topics	Final Examination
Readings	Elmasri, Chapters 11 - 14, 16 - 18, 26 Lecture 7 - 14

Classroom Conduct

Introduction	You are expected to be punctual, alert, and prepared for each class. Be considerate of other students, i.e., be quiet for the duration of the class period, except when you have something to contribute.
Additional information	Please feel free to ask questions in class. If you are confused, more than likely, someone else is too. If you need extra help, please schedule an appointment in advance.

Policies & The Honor Code

Introduction	The project and all other assignments in this course represent individual work. As always the GMU Honor Code holds for all work in this course. Stated in English, do the work by yourself. If you need help, see me.
Additional information	The most frequent violations of the university honor code and policies include but are not limited to: ? Copying or sharing a file/homework or any portion of a file/homework from another student. ? Sharing or allowing another student to copy your files/homeworks or any portion of a file/homework. ? Duplicating or distributing copies of copyrighted software program. ? Unauthorized access or use of university computers, computer systems, or computer network. ? Creating, sending or distributing electronic chain letters. ? Using a disk containing a virus in the computer lab or computer system, or distributing the virus on the computer network. ? Using the university computers, computer system, or computer network to view or distribute profanity or objectionable material.

Project / Homework

Introduction	Students are provided with the requirements for an enterprise database system. The project is implemented stepwise as part of regular homework assignments. Students have to design the database (HW#1), create the database and load the system (HW #3), write SQL statements to implement queries and views (HW#4), and use embedded SQL to generate a report to support a decision-making system (HW#6). HW#2 and HW#5 are related to Relational Algebra and functional normalization, respectively.
Requirements	Either MS Access 2000 or Oracle may be used for implementation. Access is adaquate for most database applications; it is simple to use and readily available in the business environment. Oracle is more robust but it is also complex, expensive and less readily available to many businesses. Access may be easier to use of the uninitiated especially in the beginning of the course but it lacks Oracles utilities for embedded SQL and other functions

in later exercises. All assignments can be completed in either with adaquate understanding of SQL and programming. You may choose either but you will not be permitted to switch in the middle of the term.

All support materials (charts, diagrams, text, etc.) may be generated in MS Office and / or Visio as appropriate.

All homework will be collected at the beginning of class in hardcopy. Students are required to upload softcopy (the files from which your hardcopy was printed) to WebCT prior to class.

Homework #1

Database Design

Consider the following set of requirements for a university database (UNIVERSITY) that is used to keep track of students' transcripts.

(a) The university keeps track of each student's name, social security number, current address and phone number, permanent address and phone number, birthdate, sex, class (freshman, sophomore,.., graduate), major department, minor department (if any), and degree program (B.A., B.S., .., Ph.D.). Some user applications need to refer to the city, state, and zip code of the student's permanent address and to the student's last name.

(b) Each department is described by a name, department code, office number, office phone, and college. Both name and code have unique values for each department.

(c) Each course has a course name, description, course number, number of semester hours, level, and offering department. The value of course number is unique for each course.

(d) Each section has an instructor, semester, year, course, and section number. The section number distinguishes sections of the same course that are taught during the same semester/year; its values are 1, 2, 3, .., up to the number of sections taught during each semester.

(e) A grade report has a student, section, letter grade, and numeric grade (0, 1, 2, 3, or 4).

Draw the ER diagram (see Fig. 3.15, pg 65 in the textbook for an example) for the UNIVERSITY enterprise. Specify key attributes for each entity type and relationship. Use (min, max) notation to indicate type of participation and cardinality corresponding to structural constraints on each relationship type.

Homework #2

Relational Model & SQL

Solve Exercises 7.18, 7.19, 7.20, 7.21, and 7.23 (pgs 235 - 237) and 8.8, 8.9, and 8.16 (pg 286) from the textbook.

Homework #3

Relational DB Schema

1. Show the Relational DB schema diagram (see Fig. 7.7, pg 208 in the textbook for an example) for the UNIVERSITY ER diagram from Homework #1.

2. Create the UNIVERSITY database and populate ('load') the database.

Your report should include the relational UNIVERSITY database model and a printout of both your session (DDL statements) and the contents of the loaded UNIVERSITY (populated) database.

Homework #4

SQL Queries

1. Query and modify the UNIVERSITY database using SQL. Include several of the following (at least one of each):

retrieval of information (select-from-where),
set operations,
join operations,
aggregate functions,
nested subqueries,
and views -- see Ch. 4.

The modification of the database must include several (>3) deletions, insertions, and updates (at least one of each).

2. Modify the UNIVERSITY database to include integrity constraints of your own choice. Use several (>3) domain constraints, referential integrity (also on delete/update cascade), and assertions / triggers (at least one of each).

Run several (>3) appropriate SQL queries to validate the usefulness of each of the constraints you have just implemented.

Homework #5

Normalization

1. Normalize your UNIVERSITY relational DB (database) design to the BCNF level and update the database from Homework #3 accordingly.

2. Solve problem 14.17, pg 497 from the textbook.

Homework #6

Advanced Data Modeling & Embedded SQL

1. Use the Advanced Data Modeling concepts discussed in the class, the EER design and the appropriate mapping procedures of EER into relational database in order to augment your UNIVERSITY original database design. In particular, include information on (i) students and faculty and the roles they play within the UNIVERSITY framework, and (ii) the physical space available

for regular UNIVERSITY activities.

2. Call the new design UNIVERSITY1 and implement it - similar to Homework #3.

3. Use Embedded SQL to generate reports of your own choice on the state of the UNIVERSITY1 database.

Grading

Introduction

Grades will be assigned as follows:

Requirements

Homework / Project - 40%
Mid Term Exam - 25%
Final Exam - 25%
Class attendance, preparation and participation - 10%