

CSC056-Z1 – Database Management Systems

Hofstra University – Fall 2005

Instructor: Vinnie Costa

Phone: (212) 438-3269

E-Mail: vcosta@optonline.net (preferred)

vincent_costa@mcgraw-hill.com

Class Meets: Saturdays, 8:15-12:00PM, and Sundays, 9/11 & 10/2,
in Starr Hall, Rm: 0205

Office hours: After class in Adams 210 or by appointment

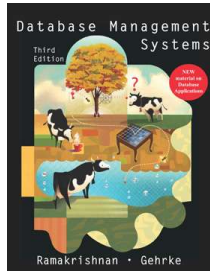
August 20, 2005

1.Course Overview and Description

This course is designed to provide individuals with an introduction to database concepts and the relational database model. Topics include SQL, normalization, design methodology, DBMS functions, database administration, and other database management approaches, such as client/server databases, object oriented databases, and data warehouses. At the completion of this course, students should be able to understand a user's database requirements and translate those into a valid database design. The emphasis will be on application development rather than system fundamentals.

2.Required Text

Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, 3/e, McGraw-Hill Higher Education, 2003, 1065pp., ISBN 0-07-246563-8



2.1Reference

Rasmus Lerdorf and Kevin Tatroe, *Programming PHP*, O'Reilly & Associates, Inc., 2002

Michael “Monty” Widenius, David Axmark, and MySQL AB, *MySQL Reference Manual*, O'Reilly & Associates, Inc., 2002

3.Grading

There will be several **programming/homework assignments** during the class, three of these will count towards your grade (you'll get advanced notice). There will also be a **mid-term** and an **end-term** exam. These will be take home exams assigned a week before the due date.

Class participation and involvement counts. This should be an interactive experience. Please feel free to share information and ideas. Be willing to assist others.

The will be a final project or paper due toward the end of the semester. The purpose of this is to encourage extensive research in the database field.

There will be no makeup tests (mid-term and end-term exams) or extended deadlines. Submitting the test on an alternative date is at the discretion of the instructor, but prior arrangements should be made (unless, in case of emergencies, in which case, proper documents should be provided).

3.1Point Allocation:

Assignments 1-3:	5% each
Final Project/Paper:	30%
Mid-Term:	25%
End-Term:	25%
Participation:	5%

4.Attendance

Attendance will be taken at each class but it is not mandatory. However, if you do not attend class regularly, you will have a high probability of failing. Participation is important to fully appreciate the subject. If you cannot make a class for some reason (travel, business commitments, etc.) try to let me know.

5.Course Outline

Table 1 is a rough outline of the course. This schedule may change depending on the pace of the class and threads of discussion. Assignment dates are not shown here. These will be provided at a later date.

Session	Date	Topic	Comments
1	08/20/05	Overview & Introduction to DBMS	Covers ER Model
2	08/27/05	Relational Model	Paper Assignment
3	09/10/05	Relational Algebra and Calculus	
4	09/11/05	SQL: Queries, Constraints, Triggers	Mid-term Handout
5	09/17/05	Database Application Development	Mid-term Due
6	09/24/05	Database-Internet Applications	
7	10/01/05	Database-Internet Applications	
8	10/02/05	Systems Basics: Storage, Transactions	
9	10/08/05	Schema Refinement, Normalization	End-term Handout; Paper Due
10	10/15/05	XML Data Management	End-term Due

Table 1: Course Outline

6.Programming Assignments

There may be some programming assignments but these will be to provide hands on experience with databases. The programs will be graded 80% on correctness and 20% on style (general structure, comments, etc.)

7.Slides, Links and News

I will try to have the slides for each class available on a web site at:

<http://www.cs.hofstra.edu/~cscvjc/Fall05>

These will be available in HTML and PowerPoint formats. There will also be helpful and interesting links along with news items.

8.Class Rules

- Unless specifically stated otherwise, *assignments are to be completed individually*. You are encouraged to discuss the understanding of a particular issue or class material with fellow students, but code and solutions have to be your own effort.
- *Academic honesty* is to be taken very seriously. If you submit work that references another person's efforts, then you must properly attribute it to that person, otherwise it is plagiarism and you will receive zero credit.
- This is not a course on how to crack systems, however, it is practically impossible for us to avoid discussing concrete security weaknesses in existing systems. *Any attempt* to use such information to gain *unauthorized access* to any system will be dealt with harshly.