COMPUTER ENGINEERING DEPARTMENT

Course: CMPE 226, Section 1
Course Title: Database Design
Semester: Spring 2007

---

1. Course Information

Instructor:
Dr. M.E. Fayad
Computer Engineering, College of Engineering, San Jose State University
Web page: http://www.engr.sjsu.edu/fayad

Course Title	**Advanced Database Design**
Course Code	**CmpE 226**
Section	**1**
Class Hours & Location	**Thursday 6:00 p.m. to 8:50 p.m. (18:00 to 20:50)** **Engr. Room 331**
Office Hours	**Thursday:** **3:50 p.m. – 5:50 p.m. (2 hrs)** **Friday:** **3:50 p.m. – 5:50 p.m. (2 hrs)** Other times: **Send an e-mail to schedule an appointment.**
Office Location	**ENG 283I**
Office Phone	**(408) 924-7364**
E-mail:	**m.fayad@sjsu.edu**
Preferred Contact	**Through e-mail**
Department Fax	**(408) 924-4153**
Course Web Page	**http://www.engr.sjsu.edu/~fayad /current.courses/cmpe226-spring2007**

2. Course Description

a. Course Overview and Description:

Course Catalog Description

Database concepts and architecture. Logical and physical data organization. Relational, hierarchical and network databases. Security/integrity, recovery/ concurrency. Design projects.

This course develops your understanding of database system **CONCEPTS** that are independent of any specific database system. As an analogy, if you took a driver's education class, you learned there driving, which is independent of any specific car. This distinction between concepts and tools is important to keep in mind in this course (especially by students who like to list on their resumes a long list of specific operating systems, programming languages, and database systems).

These concepts include: data abstraction levels; data independence; data models, including relational, constraint, spatial, and spatiotemporal data models; query languages, including relational algebra, and SQL; database design concepts, including integrity constraints; and interoperability, including data and query interoperability. Many of these concepts have a remarkable staying power, while the specific systems change all the time. In fact, the more basic the concept the more likely it is to endure. For example, the concept of data abstraction endured even as new data models were invented: hierarchical in the 1950s, network in the 1960s, relational in 1970, and various spatial data models used in GI in more recent years. As another example, the QL language, invented in the early 1970s,is still the primary language of relational database systems today, whether they come from Oracle, IBM, Microsoft, MySQL, ybase, Empress, etc.

By understanding well the concepts and not just a specific system on a specific platform, you should be able to adjust better to future changes in the database systems area. In particular, GI is emerging as a very important extension of relational databases, with some estimates putting the word-wide GIS market at about one billion dollars annually, while the relational database market is about six billion dollars annually. Hence we will learn about GIS too in this course. This course will contain two or three projects that are designed to illustrate a variety of uses of database systems (**all subject to change**):

b. Prerequisites:

CmpE126/CS046B (Algorithms & Data Structures), or CmpE 200 (Computer Arch.) **or Instructor Consent** -- Good background in the practical use of an object-oriented programming language is a plus.

c. Required & Recommended Texts:

Any book is fine. You can count on the information that I provide for you on the website.

d. Other Reading Materials:

Supporting Texts:

1. Ramez Elmasri and Shamkant Navathe. Fundamentals of Database Systems, 4th Edition, Addison-Wesley, July 2003, ISBN 0321122267
   Chapters: 1, 3, 7, 8, 11, and 12
   Chapters' sections of 2, 4, 19, 21, and 26.
   ** Good notes will be provided for the above chapters.
2. Peter Revesz, Introduction to Constraint Databases, Springer, New York, 2002, ISBN 0-387-98729-0
3. Paul DuBois. MySQL Cookbook, 1st Edition, O'Reilly & Associates, November 2002, ISBN 0596001452

Required Articles, Columns, Case Studies, and Patterns will be posted on the web later.

Other Resources: Instructor notes will be available on the course web page.

e. Student Learning Objectives:

On Successful completion of this course students shall be able to:

1. Model and design conceptual databases using Entity-Relationship model, the traditional and stable class diagrams
2. Understand and apply the relational algebra and SQL
3. Learn and work on database issues, such as normalization and security
4. Explore and examine database current trends, such as Constraints Database, Spatiotemporal Database, Data Visualization, and Data Animation
5. Explore and learn database emerging trends, such as web technologies, XML, data warehousing, and data mining

3. Course Requirements

a. Projects

Projects are team-oriented. The class will be divided into groups of **4-3 (three preferred)** for team projects. Students will be responsible for forming groups. **Students of the best 3 or 4 teams' projects will give final presentations of their project work if asked.** Grading criteria and project ideas will be posted in a project Web page.

Teams are responsible for two projects and the third project is optional for extra points. Team must deliver if they select project #3 otherwise each team member will be penalized for 4 whole points each.

Project 1: This project will focus on database modeling and design using UML Class Diagram and Normalization..