Database Design
CS 450

- Instructor:

Prof. Khaled M. Elleithy
Office: Tech Building, Room 229
Telephone: (203) 576-4703
Email: elleithy@bridgeport.edu

Textbook(s)

- **Text:**
  - An Introduction to Database Systems, Addison Wesley Publishing Company, C.J. Date, 2000.
- **Other useful textbooks:**
  - A First Course in Database Systems: by Jeff Ullman and Jennifer Widom
- **Course Page:**

  http://www.bridgeport.edu/~elleithy/cs450sp2002

Course Outline

1. An Overview of Database Management.
2. Database System Architecture.
3. An Introduction to Relational Databases.
4. An Introduction to SQL.
5. Domains, Relations, and Base Relvars.
6. Relational Algebra.

7. Integrity.
8. Views.
9. Functional Dependencies.
10. Normalization I: 1NF, 2NF, 3NF, BCNF.
11. Semantic Modeling

Real Business: What is a DBMS

- **Database management systems:**
  - Provide efficient and secure access to large amounts of data.
  - Address problems such as:
    - How to store the data
    - How to query data efficiently
    - How to update the data securely (by multiple users[1])
  - Contrast with using file systems for the same task

Relational Databases

- Based on the **relational model**

EMP#	ENAME	DEPT#	SALARY
E1	JOSEPH	D1	57K
E2	LARRY	D1	45K
E3	JOHN	D2	72K
E4	CATHY	D3	60K
E5	ROSE	D4	40K

Employee table

Querying a Database

- Find the names of employees who are in D1 and have salary > 40k.
- SQL(**S**tructured **Q**uery **L**anguage)

```
select  E.ENAME
    from employee E
    where E.DEPT# = D1 and
                        E.SALARY= 40K;
```

- Query processor figures out how to answer the query efficiently.

Database Industry

- Relational databases are a great success of theoretical ideas.
- “Big 3” DBMS companies are among the largest software companies in the world.
- IBM (with DB2) and Microsoft (SQL Server, Microsoft Access) are also important players.
- $20B industry
- Challenged by object oriented DBMS.

Functionality of a DBMS

- Storage management
- Abstract data model
- High level query and data manipulation language
- Efficient query processing
- Transaction processing
- Resiliency: recovery from crashes
- Interface with programming languages

Why Use a DBMS?

- Data independence and efficient access.
- Reduced application development time.
- Data integrity and security.
- Uniform data administration.
- Concurrent access and recovery from crashes.
- Data can be shared.
- Redundancy can be reduced.

The Study of DBMS

- Several aspects:
  - Modeling and design of databases
  - Database programming: querying and update operations
  - Database implementation
- DBMS study cuts across many fields of Computer Science: OS, languages, AI, Logic, multimedia, theory...