

CS 420 - Database Management Systems

KMA Solaiman

Topics

- design/implementation of database management systems (DBMSs)
- not a course about how to use a DBMS to build applications or how to administer a DBMS.

Relational Databases

Storage

Execution

Concurrency Control

Recovery

Distributed Databases

Potpourri

Query Planning

Operator Execution

Access Methods

Buffer Pool Manager

Disk Manager

Assignments

- You'll implement certain database components in the projects (such as buffer manager, query optimizer, transaction manager), using
 - MiniBase
 - SimpleDB
 - CMU DB Group BusTub academic DBMS
 - RedBase projects from Stanford
- Each project builds on the previous one.
- Students will complete multiple homework assignments during the course. These homeworks are designed to reinforce the lectures and reading materials.
- Homeworks and projects are counted equally when computing the homework and project portion of the final grade.

Minibase

- parser, optimizer, buffer pool manager, storage mechanisms (heap files, secondary indexes based on B+ Trees), and a disk space management system.
- The goal is not just to have a functional DBMS, but to have a DBMS where the individual components can be studied and implemented by students.

Homework and Projects

HOMEWORKS

Assignment

SQL

Storage & Indexes

Query Execution

Concurrency Control

Distributed Databases

PROJECTS

Assignment

C++ Primer

Buffer Pool Manager

B+Tree Index

Query Execution

Concurrency Control

Disk vs In-Memory vs Database

Relational Operators

Assignments: Grading

- Homework: **15%**
- Midterm exam: **20%**
- Final exam: **25%**
- Project: **35%**
- Class attendance: **5%**
- Extra credits/ same day quiz: **5%**

Face-to-face vs Online

Strengths of face-to-face instruction

- interactive discussions,
- hands-on activities, and
- collaborative projects.

Online asynchronous learning

- self-paced content consumption,
- learn-by-doing
- practice exercises, and
- online discussions.

Face-to-face: Content

- 1. Introduction to Database Systems:** Overview of database concepts, architecture, and data models.
- 2. Query Languages and SQL:** In-depth understanding of SQL syntax, query optimization, and advanced query techniques.
- 3. Database Design:** Entity-Relationship (ER) modeling, normalization, and schema refinement.
- 4. Transactions and Concurrency Control:** ACID properties, transaction processing, and concurrency control mechanisms.
- 5. Indexing and Query Optimization:** Index structures, indexing techniques, and query optimization strategies.
- 6. Database Security:** Access control, authentication, authorization, and data privacy considerations.
- 7. Case Studies and Real-world Examples:** Analysis of real-world database systems and applications.



Skills

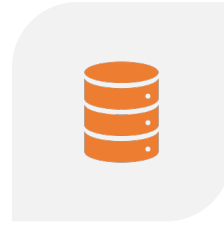
Problem-solving and critical thinking skills for designing and optimizing database schemas.

Collaborative skills for group discussions, case studies, and designing database systems.

Hands-on skills in writing and executing SQL queries, including complex joins and subqueries.

Communication skills for participating in class discussions, presenting case studies, and explaining database concepts.

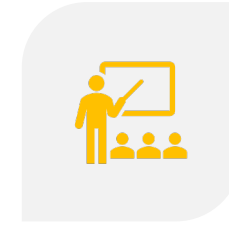
Online Learning: Content



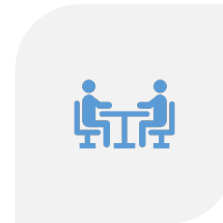
PRE-RECORDED VIDEO LECTURES: FOUNDATIONAL CONCEPTS, QUERY LANGUAGES, AND DATABASE DESIGN PRINCIPLES.



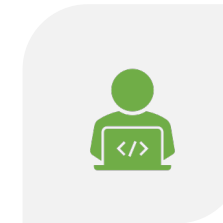
READING MATERIALS: TEXTBOOK CHAPTERS, RESEARCH PAPERS, OR ONLINE RESOURCES



ONLINE TUTORIALS: SHARING VIDEO TUTORIALS OR INTERACTIVE TUTORIALS THAT DEMONSTRATE SQL QUERIES AND DATABASE DESIGN TECHNIQUES.



ONLINE DISCUSSION FORUMS: STUDENTS CAN ASK QUESTIONS, DISCUSS CONCEPTS, AND COLLABORATE ASYNCHRONOUSLY.



PRACTICE EXERCISES AND QUIZZES: PROVIDING ONLINE EXERCISES AND QUIZZES TO ASSESS UNDERSTANDING OF DATABASE CONCEPTS, SQL QUERIES, AND NORMALIZATION.



Online Learning: Skills



Self-paced learning and time management skills to watch pre-recorded video lectures and complete reading assignments.



Self-guided problem-solving skills when working on practice exercises and quizzes independently.



Written communication skills for participating in online discussion forums, asking questions, and engaging in academic discourse.



Self-assessment and feedback incorporation skills for reviewing solutions to practice exercises and quizzes to identify areas for improvement.

ACCESS TIMES

Latency Numbers Every Programmer Should Know

1 ns	L1 Cache Ref	← 1 sec
4 ns	L2 Cache Ref	← 4 sec
100 ns	DRAM	← 100 sec
16,000 ns	SSD	← 4.4 hours
2,000,000 ns	HDD	← 3.3 weeks
~50,000,000 ns	Network Storage	← 1.5 years
1,000,000,000 ns	Tape Archives	← 31.7 years