Database Design, CSCI 340, Spring 2015
Review for Final, April 4, 8-10am (11 if needed), S&E 308

Introduction to Databases

Chapter 1, Introduction
Know what is meant by meta data

Know that Entity-Relationship Diagram (ERD) show:

- Entity – distinct object or concept to be represented in the database
- Attribute – property of an object
- Relationship – association between two entities

Know what views are and that they can be used to provide:

- Security
- Ways to customize the look of a database
- Consistent picture of the database, even when the database changes

Chapter 2, Database Environment (Skip 2.3), pages 3-45 & 49-55
3-Level Architecture (Figure 2.1, page 37) - know what is meant by external schemas or sub schemas, conceptual schemas and internal schemas

Know the importance of both logical and physical data independence

Know the difference between the intension and extension of a db

Know what a logical record is

Be able to describe the following and know why they are needed in a DBMS.

1. Data storage, retrieval and update
2. User-accessible catalog
3. Transaction support
4. Concurrency control
5. Recovery
6. Authorization
7. Support for data communications
8. Integrity
9. Data independence
10. Utilities – importing, monitoring

(I will not ask you to list these services)

Relational Model and Languages

Chapter 4, Relational Model
Know the vocabulary of the relational model

- Relation
- Attribute
- Tuple
- Degree of a relation (# attributes)
- Cardinality of a relationship (# records)
- Degree of a relationship (# relations which participate)
- Cardinality of a relationship (the minimum and maximum records that participate in the relationship)

Know the vocabulary of keys:

- Composite key
- Candidate key
- Primary key
- Foreign key

Know the vocabulary of constraints:

- Entity constraint (no value in the key can be null)
- Referential constraint
- General constraint

Base relations – those relations given in the conceptual schema
View – relations derived from base relations (also call these virtual relations)

Chapter 5, Relational Algebra
Know the difference between imperative and declarative languages
Know that SQL is declarative while relational algebra is imperative
Be able to write queries involving:

Projection (Π), Select (σ), Natural Join (⋈),Theta join - $\bowtie_F$, Equijoin join - $\bowtie_F$, aggregation ($\mathcal{F}_{ag}$), grouping ($_{gr}\mathcal{F}$), rename (ρ), Cartesian product (x), union (∪), intersection(∩), set difference (-)

Be able to translate simple SQL statement into an equivalent relational algebra statement, and vice versa

Chapter 6, SQL Data Manipulation
Know that SQL includes a DDL and a DML and the purpose of each
Be able to write queries involving:

- SELECT, FROM, WHERE, GROUP BY, ORDER BY
- LIKE
- aggregate functions
- sub queries with equality
- INSERT, UPDATE, DELETE
- HAVING
- sub queries with IN, ANY, ALL, SOME, EXISTS, NOT EXISTS
- UNION, INTERSECT, EXCEPT
- Joins – outer, LEFT, RIGHT, FULL

Chapter 7, SQL Data Definition, Sections 7.1 & 7.3
Be able to write queries involving:

- CREATE TABLE enforcing referential integrity
- CREATE VIEW

Chapter 12, Entity-Relationship Modeling
Know the difference between intentional and extension of a database.
Know the vocabulary (some of this was introduced in Chapter 4): entity, relationship, degree, binary relationship, recursive relationships, composite attributes, derived attributes, candidate key, primary key and composite key, id-dependent entities, multiplicity of relationships (cardinalities), minimum cardinality, maximum cardinality,
Know to recommend that relationships of degree greater than two be split into multiple binary relationships.
Know to recommend that 1-1 relationships be combined.

Chapter 14, Normalization, pages 378-389
Know the purpose of normalization
Know the meaning of a functional dependency and be able to identify likely functional dependencies within a relation.
Know the definition of BCNF
Be able to recognize when a relation is not in BCNF and to decompose the relation into a set of relations which are in BCNF

Methodology

Chapter 17, Methodology-Logical Database Design for the Relational Model, pages 439-448
Be able to develop a logical data model which corresponds to a given conceptual data model

Chapter 18, Methodology-Physical Database Design for Relational Databases, pages 485-494
Know what an index is and why they are used
Know to consider creating an index for attributes which are used often in JOINS, are accessed often
Know that indicies speed up reads but slow down writes

Chapter 19, Denormalization, pages 495-508
Know and be able to describe how sometimes tables storing addresses are not fully normalized (due to zip codes)
Know the idea "normalization is about design, denormalization is about optimization"
Know that one reason not to normalize is that joins are expensive
Know that denormalization speed up reads but slows down writes

Chapter 20, Security and Administration - **SQL injection**
Be able to define, explain and give an example of sql injection.
Be aware of sql-injection prevention techniques, that it is important to code with security in mind and not to trust any kind of input, especially not that which comes from the client side
Know what PDO stands for, why it is used and approximately, how to use it

Chapter 21, Professional, Legal, and Ethical Issues in Data Management

Be able to discuss the consequences of data retention policies on privacy

Know

- FERPA – Family Educational Rights and Privacy Act, 1974
    - As first enacted, FERPA provided parents with the right to inspect and review "any and all official records, files, and data directly related to their children"
- European Union (EU) Directive 95/46/EC (1995)
    - Protection of individuals with regard to the processing of personal data and on the free movement of such data
- Health Insurance Portability and Accountability Act (HIPAA, 1996)
    - Standards for the security of patient data and transactions involving this data
- United Kingdom's Data Protection Act of 1998
    - Personal data processed for any purpose or purposes shall not be kept for longer than is necessary for that purpose or those purposes.

Chapter 22, Transactions and Concurrency Control, pages 569-597
Be able to define transactions and describe the purpose of transactions
Know the three keywords of transactions and the two possible outcomes of transactions
Keywords:

- **BEGIN TRANSACTION**
- **COMMIT**
- **ROLLBACK**

Be able to list and explain each of the ACID properties of transactions
Know the purpose of concurrency control and when concurrency causes problems
Be able to describe the "lost update", "uncommitted dependency (dirty read) problem" and the "inconsistent analysis problem" and tell how they are fixed.
Know what a check point is and why they are used

Chapter 23, Query Processing, pages 627-630, 641 (material on Query Optimization)
Know information which may be stored to help create an optimized query
Given a query and relevant statistics, be able to optimize the query
Know heuristic strategies for optimizing queries

Chapter 31, Semistructured Data and eXtensible Markup Language, XML, pages 1055-58, 1063-74, 1081-87 & 1091-101
Know what is meant by semistructured data
Know that XML is a metalanguage (a language for describing other languages) that enables creating customized tags not available with HTML
Know advantages of XML are its simplicity, extensiblity, separation of content and presentation
Know the difference between horizontal and vertical scaling
Know that relational DBMS are designed for vertical scaling, while NoSQL DBMS are designed for horizontal scaling
Know that relational DBMS are designed to support the ACID properties while NoSQL products typically only support "eventual consistency"
Know that relational DBMS generally have an impedance mismatch with programming which NoSQL products typically don't
Know that NoSQL databases can be document, key-value, columnar or graph based

Big data and data mining
Know what is meant by "big data"
Know that characteristics of big data are volume, variety, velocity, variability, veracity and complexity