

REVIEW QUESTIONS:

- What are the main benefits of using a DBMS to manage data in applications involving extensive data access? (Sections 1.1, 1.4)
 - *Data Independence:* Application programs should not have data representation and storage details. DBMS abstract view hides such details.
 - *Efficient Data Access:* DBMS uses many techniques to store and retrieve data.
 - *Data Integrity and Security:* DBMS can enforce integrity constraints.
 - *Data Administration:* When several users share data, centralizing administration of data can offer significant improvements.
 - *Concurrent Access and Crash Recovery:* DBMS protects users from system failures effects. Users can think of only one access.
 - *Reduced Application Development Time:* High-level interface facilitates quick application development. DBMS Apps, more robust ones than stand-alone applications.
- When would you store data in a DBMS instead of in operating system files and vice-versa? (Section 1.3)

DBMS OVER FILES

- Store Many info
- Fast Access
- Concurrent Access
- Security Restrictions
- Rollbacks
- What is a data model? What is the relational data model? What is an independence, and how does a DBMS support it? (Section 1.5)
 - *Data Model:* Collection of high-level data description that hides low-level details. DBMS lets users define data to be stored according to data model.
 - *Relational data model:* the schema for a relation specifies its name, name for each field and field type. Each row is an instance, although the description is not complete, it is adequate for the desired purpose.

- *Data Independence*: Application programs are insulated from changes in the way data is stored and structured. Achieved through 3 levels of abstraction. Conceptual: describes all relations. Physical: storage details (indexes). External: allow data access to be customized.
- Explain the advantages of using a query language instead of custom programs to process data. (Section 1.6)

Relational model supports query languages. Relational calculus is a formal query language based on mathematical logic that gives intuitive and precise meaning. Relational algebra: collection of operators. Query evaluation is more efficient and allow inserting, delete and modify data.

- Identify the main components in a DBMS and briefly explain what they do. (Section 1.8)
 - Plan Executor / Parser: show interaction and receive all external instructions
 - Operator Evaluator / Optimizer: query evaluation and execution
 - Files and Access Methods/ Buffer Manager / Disk Space Manager: checks database files and returns asked information.
- Explain the different roles of database administrators, application programmers and end users of database systems? (Section 1.9)
 - *End-User*: from a diverse number of fields. Use apps developed by others.
 - *Application programmers*: develop packages to facilitate data access for end users, who usually are not professionals. Use external schema.
 - *Database administrators*: design conceptual and physical schemas, security & authorization, data availability & recovery from failures and database tuning.

Recovery

EXERCISES:

1.1) Why would you choose a database system instead of simply storing data in operating system files? When would it make sense not to use a database system?

Fast Access, Many Storage, Concurrent Access, Rollbacks, Security Restrictions

1.2) What is logical data independence, and why is it important?

Logical Data Independence: is used to change conceptual schema without having to change external schema. Allows to make changes like adding, modifying or deleting and attribute.

1.3) Explain the difference between logical and physical data independence

- Logical data independence: users can be shielded from changes in the logical structure of the data.
- Physical data independence: conceptual schema insulates users from changes in physical storage detail.

1.4) Explain the difference between external, internal, and conceptual schemas. How are these different schema layers related to the concepts of logical and physical data independence?

- External: allow data access to be customized.
- Conceptual: describes all relations.
- Physical: storage details (indexes).

Logical Data Independence: make changes in the middle level without affecting the highest-level.

Physical Data Independence: make changes in the lowest level without impacting the higher-level.

1.5) What are the responsibilities of a DBA? If we assume that the DBA is never interested in running his or her own queries, does the DBA still need to understand query optimization? Why?

A DBA should design schemas, restrict access to users, make sure data is available and check for versions. Although a DBA is not interested in his own queries, he should understand the query optimization to develop DB structure and make it the best possible.

1.6) Scrooge McNugget wants to store information (names, addresses, descriptions of embarrassing moments, etc.) about the many ducks on his payroll. Not surprisingly, the volume of data compels him to buy a database system. To save money, he wants to buy one with the fewest possible features, and he plans to run it as a stand-alone application on his PC clone. Of course, Scrooge does not plan to share his list with anyone. Indicate which of the following DBMS features Scrooge should pay for; in each case, also indicate why Scrooge should (or should not) pay for that feature in the system he buys.

1. A security facility: **No, because he won't share with anyone, and does not need to limit access to other users.**
2. Concurrency control: **No, because he won't share with anyone.**
3. Crash recovery: **Yes, to avoid losing data if not saved.**
4. A view mechanism: **Yes, because it contributes to data independence and data isolation.**
5. A query language: **Yes, to allow updates, queries, calculations, and deletions.**