23K-0074
Ibrahim Johar Farooqi
BAI-5A
Database Systems - Assignment 1 Date ____ 20__

Q.1)

a) what are the problems with file system data management?

↳ The file-system data management approach suffers from program data dependence, weak security and high redundancy which leads to data inconsistency. It is also not flexible because queries must be hard coded, essentially fixed, and the separation of data in different files make sharing and any sort of integration difficult.

b) explain the usage of the Composite Primary Key w/ an example

↳ A composite primary key is a primary key made up of two or more columns together that would allow each row to be uniquely identified in a table. Its used when no single column can ensure uniqueness on its own. For example in a hospital room assignments table, a patient (PatientID) might be admitted to different rooms, a room (RoomID) can host different patients at different times. Only the combination of (PatientID, AdmitDate) uniquely identifies each specific stay

PatientID	RoomID	AdmitDate
P53	R75	2025-09-19
P54	R76	2025-09-18
P55	R77	2025-09-17
⋮	⋮	⋮

↳ the same patient cannot have two identical records for the same room on the same date, PatientID or RoomID alone on their own are not unique, but the set of columns together forms a unique key, composite primary key.

Date ____ 20 ____

c) what operations are performed by the application program when DBMS is used.

↳ When a DBMS is used, the application program can create; define tables, columns, schema and set constraints. Program can store (insert new records into tables), retrieve (read/query data using SQL, modify (update or delete existing records), control access (handle authentication & grant/revoke user permissions), and finally maintain integrity, by enforcing rules to ensure accuracy & consistency of data

d) which independence is difficult to achieve in three schema architectures & why, elaborate w/ an example.

↳ Logical data independence is more difficult to achieve in the three schema architecture because application programs and user views depend tightly on the conceptual schema. Any little logical change could break queries, reports or forms in the architecture. Like, a hospital database has a single Patient table that stores personal information and contact details. If the contact details are moved to a new Patient_Contact table to reduce redundancy, the external schema will no longer match the old structure, and all user views or applications that expected to follow one single table must be updated.

e) A key is a superkey but not vice-versa, explain w/ an example

↳ A key is a minimal set of attribute(s) that uniquely identifies each record in a table, while a superkey is any set of attribute(s) that can uniquely identify a record, even if it contains extra attributes, hence every key is automatically a superkey but not every superkey is a key.

Like in a Hospital Patient Table with attributes PatientID, Name, & Room No, the attribute PatientID alone is a key because it uniquely identifies each patient & is minimal.

Date_____20___

However the combinations of (PatientID, Name) or (PatientID, RoomNo) are superkeys because they still uniquely identify each patient but include extra attributes, hence they are not keys.