**DB ASSIGNMENT#01 THEORY ANSWERS**

**QUESTION#01**

**PART A)**

* Program–data dependence: programs must be modified if data storage format changes.
* No data security: no restriction on unauthorized access.
* Data inconsistency and lack of integrity: updates may apply to some files but not others.
* Not flexible: fixed queries, limited processing ability.
* Separation and isolation of data: hard to integrate across multiple files.
* Duplication of data: same data stored repeatedly, wastes storage, causes inconsistency.

**PART B)**

Composite key is a primary key made of two or more attributes, used when one attribute alone cannot uniquely identify a tuple.

**Example:** ENROLLMENT(Student\_ID, Course\_ID, Grade)

* A single student can enroll in multiple courses → Student\_ID alone not unique.
* A single course can have many students → Course\_ID alone not unique.
* Combination (Student\_ID, Course\_ID) uniquely identifies each enrollment record.

**PART C)**

* Send queries to DBMS to retrieve/manipulate data.
* Perform transactions: read and write data into the database.
* Insertions, deletions, modifications of content.
* Generate reports from database queries.
* Concurrent processing support while keeping data consistent.
* Protection functions: prevent hardware/software malfunction and unauthorized access.
* Maintenance: keep database updated over long period.

**PART D)**

Logical Data Independence is more difficult than physical because changes to the conceptual schema (entities, attributes, constraints) often affect application programs.

**Example:** Adding Date\_of\_birth to STUDENT schema requires modifying conceptual schema; queries and programs that use STUDENT must adapt. Even though external schema should insulate users, in practice applications break, so logical independence is hard.

**PART E)**

Superkey: Set of attributes SK of a relation R such that no two tuples in any valid state r(R) have the same SK value.

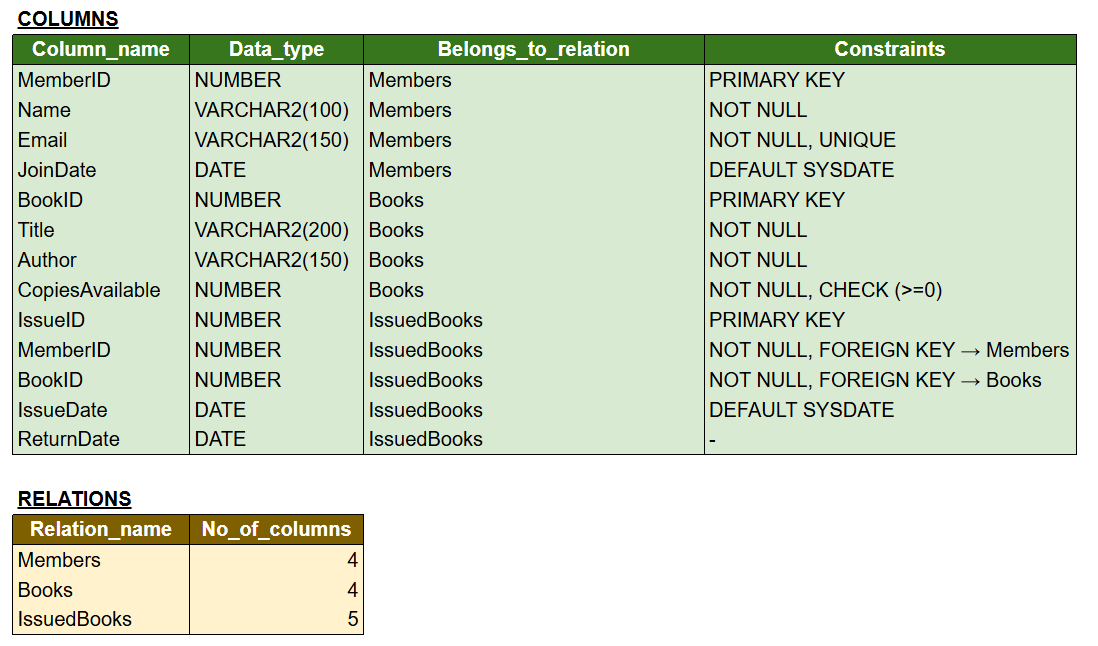
Key: A minimal superkey, if any attribute is removed, uniqueness is lost.

**Example:** CAR(State, Reg#, SerialNo, Make, Model, Year):

* {SerialNo} → Key (minimal, uniquely identifies each car).
* {State, Reg#} → Key (composite, minimal, together uniquely identify cars, since Reg# repeats across states).
* {SerialNo, Make} → Superkey (still unique, but not minimal because Make is unnecessary).
* {State, Reg#, SerialNo} → Superkey (unique but redundant, SerialNo alone is enough).

hence, every key is a superkey (uniqueness guaranteed), but not every superkey is a key (some have extra redundant attributes).

**QUESTION#02 – PART 2 CATALOG DIAGRAM**

****

**QUESTION#02 – PART 5 THEORY ANSWER**

**1.** **Due Date & Fine Management:** The system can be improved by introducing a due date for issued books and implementing an automatic fine calculation feature for late returns. This will encourage timely returns and improve circulation efficiency.

**2.** **Book Reservation System:** Another improvement would be to add a reservation module that allows members to reserve books currently unavailable. This ensures fair access and better member satisfaction.

**Adding Enhancements:**

* Normalize the database further to reduce redundancy and improve data integrity.
* Add proper indexing and optimization to improve query speed and scalability.