**Experiment No.: 02**

**Title:** To Map EER diagram drawn in experiment no.1 to relational model.

**Batch: A3** **Roll No.:16010423099** **Experiment No.: 02**

**Aim:** To Map EER diagram drawn in experiment no.1 to relational model.

**Resources needed:** MS-office

**Theory:**

The relational model uses collection of tables to represent both data and the relationships among those data. Each table has multiple columns and each column has a unique name. The relational model is an example of record-based model. Each table contains records of a particular type. The columns of the table correspond to the attributes of the record type. The relational model is the most widely used data model.

**~~Procedure / Approach /Algorithm / Activity Diagram:~~**

**Steps for Reducing EER model into relational model**

1) Any strong entity set E having attributes a1, a2,…,an is reduced into a relation schema called E with n distinct attributes i.e. a separate relation with name E and n distinct columns.

2) Any weak entity set A having attributes a1, a2,..n and a strong entity set B on which A depends, having primary key attributes as b1, b2, …, bn is reduced into a relation schema called A with one attribute for each member of set
{ a1, a2,…, an} U {b1, b2, ……. , bn}

3) Any relationship set R having a1,a2,…,an as a set of attributes formed by union of the primary keys of each of the entity sets participating in R and b1, b2,….,bn as set of descriptive attributes is reduced into a relation schema called R with one attribute for each member of the set {a1, a2, …. ,an} U {b1, b2, …., bn}

**Primary key of relationship set is decided as follows**

For **binary many to many relationships** the union of primary key attributes from the participating entity sets is primary key.

For **binary one to one relationship set** the primary key of either of the participating entity set can be chosen as the primary key.

For **binary many to one or one to many relationship set** the primary key of the entity set on the many side of the relationship set serves as the primary key.

For **n-ary relationship sets without any arrows on its edges**, union of the primary key attributes of participating entity sets is a primary key.

For **n-ary relationship sets with an arrows on one of its edges**, union of the primary key attributes of participating entity sets is a primary key.

To remove redundancy we generally make separate relation schema for many to many relationship set with primary key and other attributes as mentioned above.

For one to one we combine relation schema of relationship set with relation schema of either sides of entity sets relation schema.

For one to many and many to one we combine relation schema of relationship set with relation schema of entity set on many side entity set.

We don't make separate relation schema for identifying relationship set.Every composite attribute A having subparts a1, a2,…,an is represented by separate column for each subpart in relation schema of the associated entity set.

For **multivalued attribute** separate schema is form having columns as attributes of primary key of associated entity set and a column for multivalued attribute

For **overlapping generalization/specialization** create separate relation schemas for higher level as well as lower level entity sets.

Also include the foreign key constraint in lower level entityset for the primary key attributes of higher level entity set.

For **disjoint generalization/specialization** create separate relation schemas only for every lower level entity set(higher level entity set's attributes are inherited so add columns for same) and not for higher level entity set.

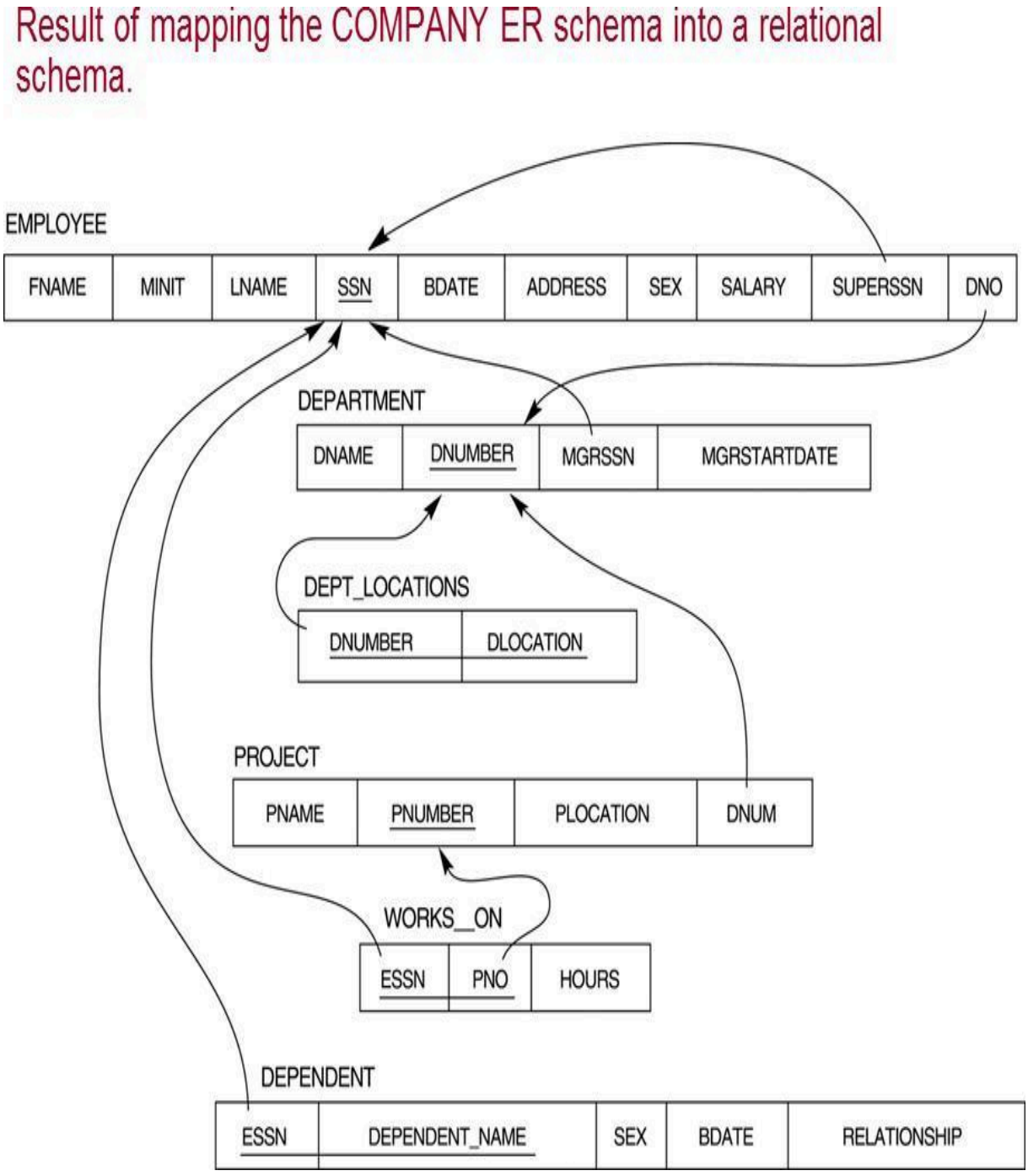
No separate relation is required to represent the **aggregation** the relation created from the defining relationship is used instead (design schema for relationship set treated as entity set carefully)

---

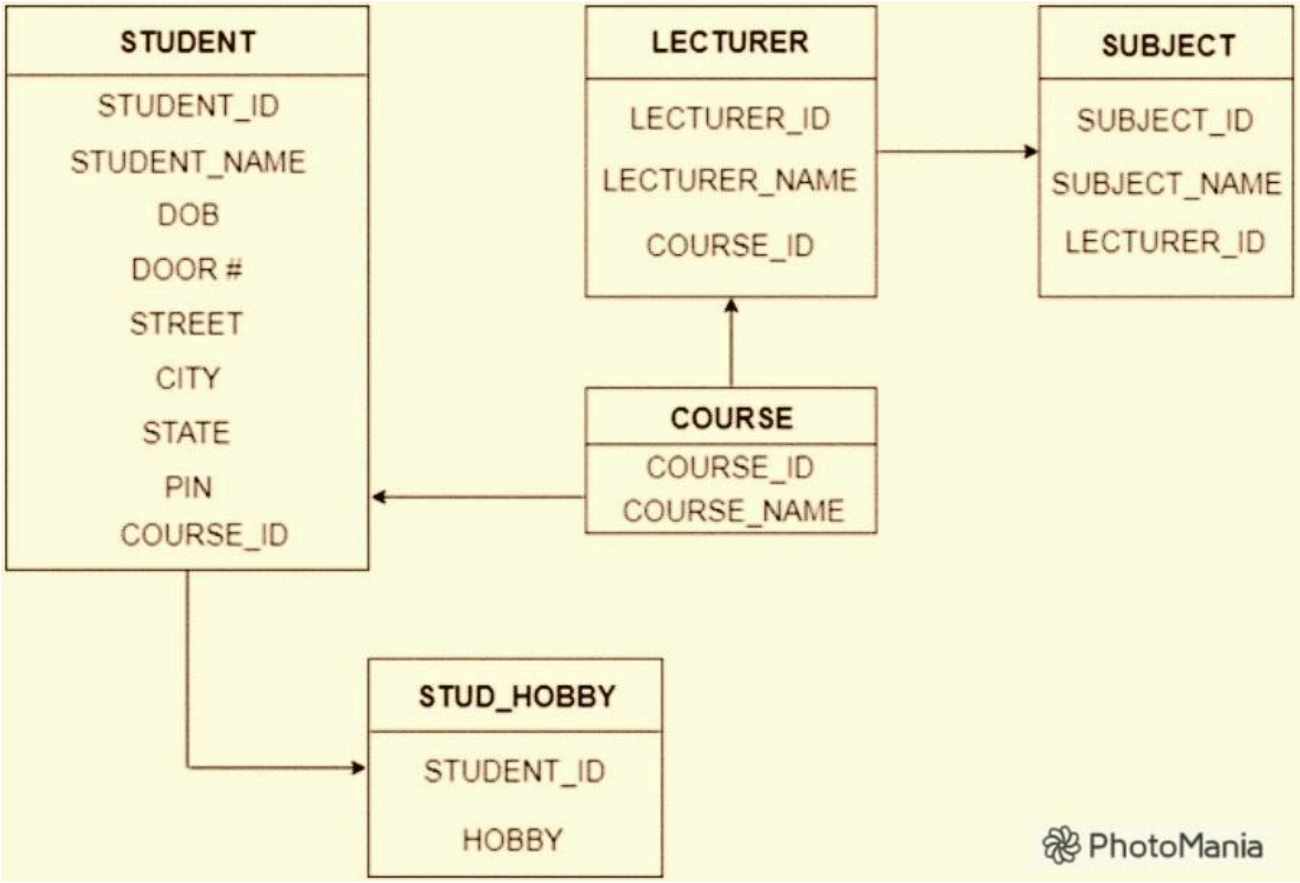
**Results: (Document printout/handwritten)**

1. Relational model

**Example:**



**Results: (Document printout/handwritten)**




**Outcomes:**

CO2: Apply data models to real world scenario.

**Questions:**

**Q1: Explain generalization and specialization with example**

Generalization is the process of extracting common characteristics from multiple entities to create a more abstract entity. For example, if we have entities like "Car" and "Truck," we can generalize them into a single entity called "Vehicle." Specialization is the opposite; it involves creating more specific entities from a general one. Using the previous example, we could specialize "Vehicle" back into "Car" and "Truck," adding specific attributes like "Cargo Capacity" for trucks.

**Q2: Explain generalization and specialization with example**

Physical data independence refers to the ability to change the physical storage of data without affecting the application programs. For example, you could switch from one storage device to another without rewriting your queries. Logical data independence, on the other hand, allows changes to the logical structure (like adding a new field) without altering how applications interact with the data.

**Conclusion:**

Applied concept of entity relation model.

**Reference books:**

1. Elmasri and Navathe, "Fundamentals of Database Systems", 6th Edition, Pearson Education
2. Korth, Slberchatz,Sudarshan, :"Database System Concepts", 6th Edition, McGraw – Hill