Experiment No.1
Design an EntityRelationship (ER) / Extended Entity-Relationship (EER) Model.
Date of Performance:
Date of Submission:

Aim :- Identify the case study and detailed statement of the problem. Design an EntityRelationship (ER) / Extended Entity-Relationship (EER) Model.

Objective :- To identify and explore a real world problem, and to design an Entity Relationship (ER) / Extended Entity-Relationship (EER) Model.

Theory:

1. **Entity:**

- An entity is a real-world object or concept that exists independently and has distinguishable attributes.
- In a database context, an entity represents a table, and each row in that table represents a unique instance of that entity.
- For example, in a university database, entities could include Student, Course, Professor, Department, etc.
- Each entity has a set of attributes that describe its properties.

2. **Attributes:**

- Attributes are the properties or characteristics that describe an entity.
- They represent the data we want to store about each instance of an entity.
- For example, attributes of a Student entity might include StudentID, Name, Age, GPA, etc.
- Attributes can be categorized as simple (atomic) attributes, which cannot be divided further, or composite attributes, which are made up of smaller sub-parts.

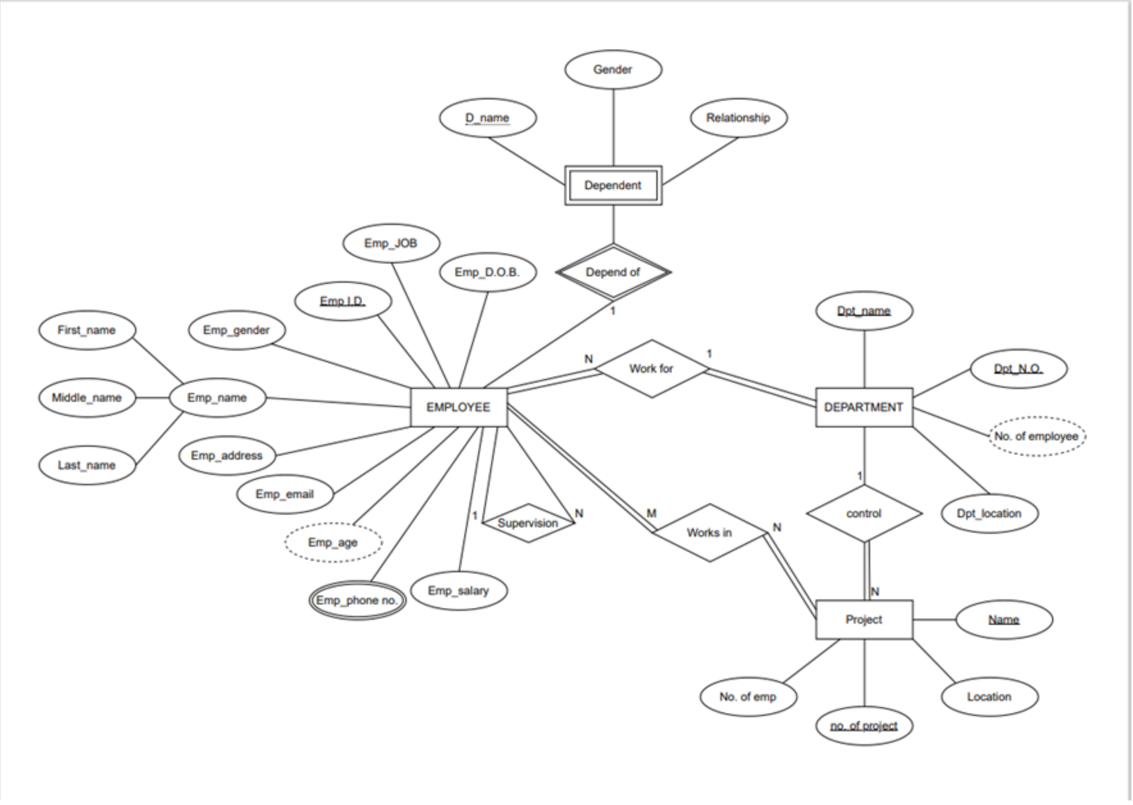
3. **Relationships:**

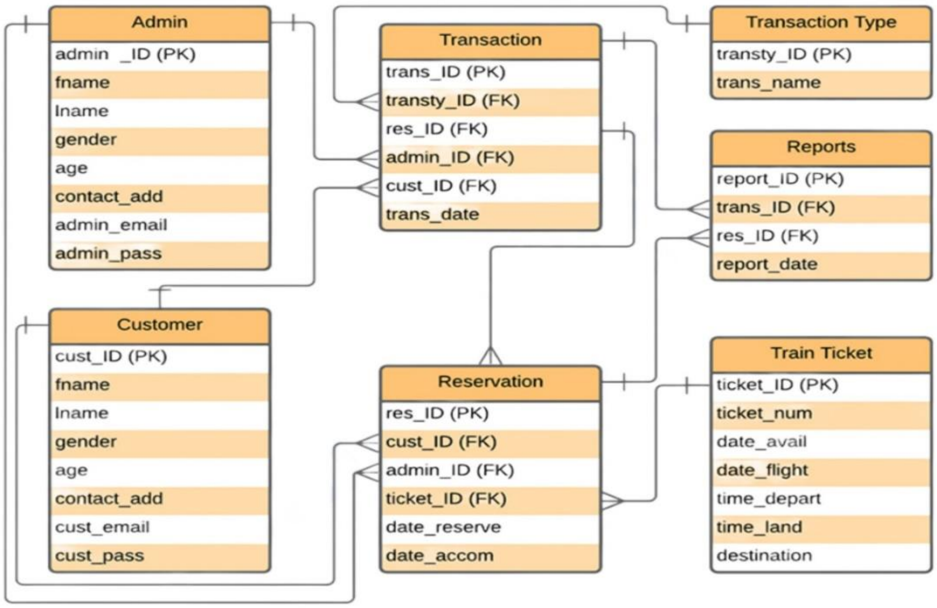
- Relationships describe how entities are related to each other or how they interact.
- They represent the associations between entities.
- Relationships are depicted as lines connecting related entities in the ER diagram.
- Each relationship has a degree, indicating the number of entities involved. It could be unary (involving one entity), binary (involving two entities), or ternary (involving three entities).
- Relationships also have cardinality, which defines the number of instances of one entity that can be associated with the number of instances of another entity through the relationship.

4. Cardinality:

- Cardinality specifies the number of instances of one entity that are related to the number of instances of another entity through a relationship.
- It defines the maximum and minimum number of occurrences of one entity that can be associated with the occurrences of another entity.
- Common cardinality constraints include:
  I. One-to-One (1:1): Each instance of one entity is associated with exactly one instance of another entity, and vice versa.
  II. One-to-Many (1:N): Each instance of one entity is associated with zero or more instances of another entity, but each instance of the second entity is associated with exactly one instance of the first entity.
  III. Many-to-One (N:1): The reverse of One-to-Many; many instances of one entity are associated with one instance of another entity.
  IV. Many-to-Many (N:N): Many instances of one entity can be associated with many instances of another entity.

Implementation:





Railway Management System ER Diagram

Conclusion:

1. Define Entity, Attributes(also types) and Relationship between entities
   Entity: An entity is a real-world object or concept that exists independently and can be uniquely identified. In database design, an entity typically corresponds to a table in a relational database. For example, in a database for a library system, entities could include "Book," "Author," "Member," and "Library Branch."
   Attributes: Attributes are the properties or characteristics that describe an entity. They represent the specific pieces of information we want to store about each instance of an entity. Attributes have types that define the kind of data they can hold, such as text, numeric, date, or boolean. For instance, attributes of a "Book" entity could include "Title," "Author," "Publication Date," and "ISBN."
   Relationships: Relationships define how entities are connected or associated with each other. They capture the interactions and dependencies between entities. Relationships can be one-to-one, one-to-many, or many-to-many. In a library database, for example,

there might be a one-to-many relationship between "Author" and "Book" (one author can write many books, but each book is written by one author), or a many-to-many relationship between "Book" and "Library Branch" (a book can be available in multiple library branches, and each library branch can have multiple books).

2. Write ER/EER diagram notations.

   a) Entities: Represented by rectangles, each entity denotes a distinct object, such as a person or a product.

   b) Attributes: Shown inside the entity rectangles, attributes describe properties of entities, like name or price.

   c) Relationships: Illustrated by lines connecting entities, relationships indicate connections between entities, such as a person buying a product.

   d) Cardinality: Depicted near the relationship lines, cardinality defines the number of instances of one entity that can be associated with another entity.

   e) Keys: Highlighted using underlines, keys uniquely identify instances of an entity, such as a primary key.