Experiment No.1

Design an EntityRelationship (ER) / Extended Entity-Relationship (EER) Model.

Aim :- Identify the case study and detailed statement of the problem. Design an Entity Relationship (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:**

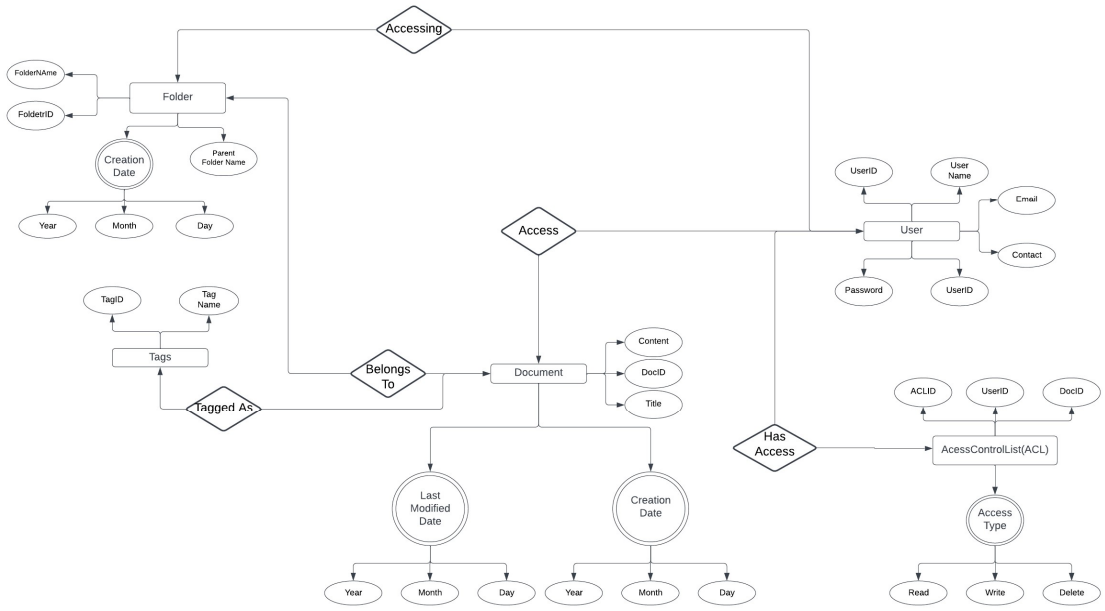
- Relationships describe how entities are related to each other or how they interact.

- 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:



Conclusion:

1. Define Entity, Attributes(also types) and Relationship between entities

   - **Entity**:
     - An entity represents a real-world object, concept, or thing that we want to store information about in a database.
     - Entities can be people, places, events, or any other significant item.
     - Example: In a university database, entities could be "Student," "Course," "Professor," etc.
   - **Attributes**:
     - Attributes are characteristics or properties of an entity.
     - They describe the data associated with an entity.
     - Example: For the "Student" entity, attributes could be "StudentID," "Name," "Date of Birth," etc.
     - Types of Attributes:
       - **Simple Attribute**: Represents a single value (e.g., "Name").
       - **Composite Attribute**: Composed of multiple sub-attributes (e.g., "Address" with sub-attributes "Street," "City," "Zip").
       - **Multivalued Attribute**: Can have multiple values (e.g., "Phone Numbers").
       - **Derived Attribute**: Calculated from other attributes (e.g., "Age" derived from "Date of Birth").
   - **Relationships**:
     - Relationships represent associations between entities.
     - They describe how entities are related to each other.

- Example: A "Student" entity can be related to a "Course" entity through the "Enrollment" relationship.

2. Write ER/EER diagram notations

ER Diagram Notations:

- ER diagrams use symbols to represent entities, attributes, and relationships:
  - **Rectangles**: Represent entities.
  - **Ellipses (Ovals)**: Represent attributes.
  - **Diamonds**: Represent relationships.
  - **Lines (Arrows)**: Connect entities and show relationships (one-to-one, one-to-many, many-to-many).
  - **Double Ellipses**: Represent multivalued attributes.
  - **Double Rectangles**: Represent weak entities.
  - **Lines with Crow's Feet (Three Lines)**: Indicate one-to-many relationships.
  - **Lines with Bars (One Line and One Bar)**: Indicate one-to-one relationships.
  - **Lines with Crow's Feet and Bars (Three Lines and One Bar)**: Indicate many-to-many relationships.

ERD Notations

- **Chen Notation**, **Crow's Foot Notation**, and **UML Notation** are commonly used notations for ER diagrams.