Experiment -1: To develop the ER Model for a given database

M. jayasree

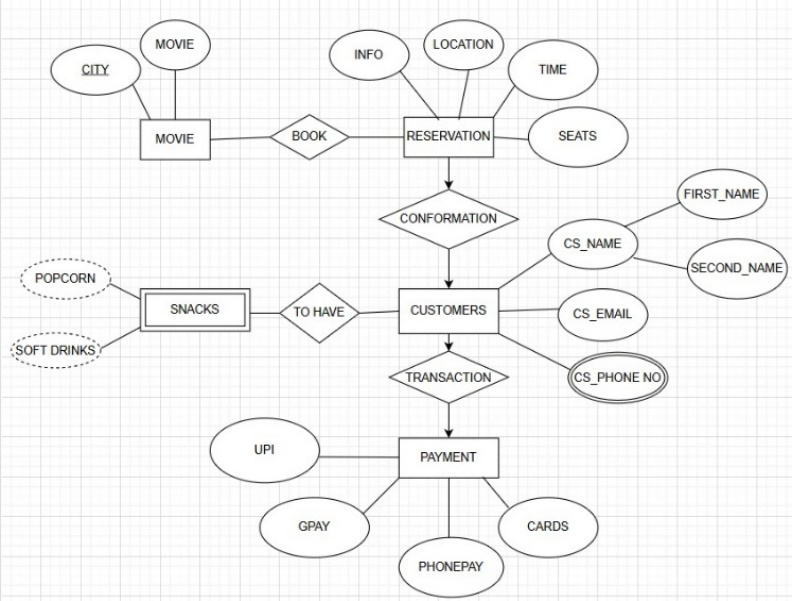
BU22CSEN0100142

Application name: Book My Show

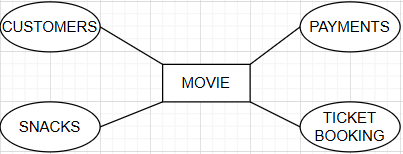
ER model

* ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.
* It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.
* In ER modeling, the database structure is portrayed as a diagram called an entity-relationship diagram.

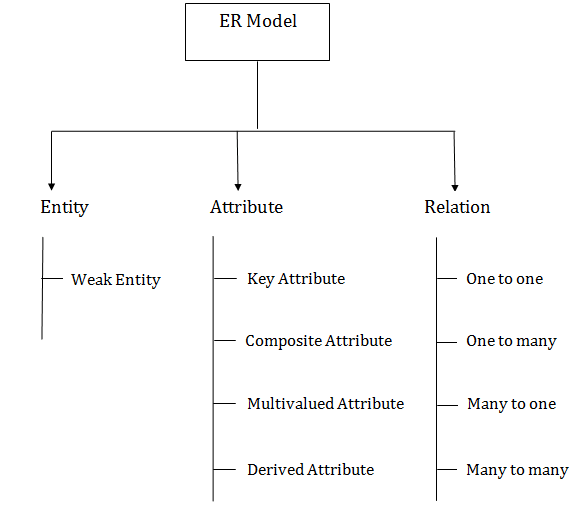
ER diagram



**For example:** Suppose we design a Book My Show database. In this database, the movie will be an entity with attributes like customers, ticket booking, etc.



Component of ER Diagram



1. Entity:

An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles.

Consider an organization as an example- movie, book, reservation etc. can be taken as an entity.



**a. Weak Entity**

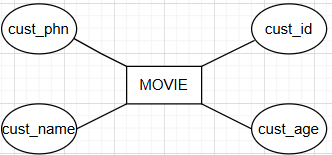
An entity that depends on another entity called a weak entity. The weak entity doesn't contain any key attribute of its own. The weak entity is represented by a double rectangle.



2. Attribute

The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.

**For example,** customer name, customer id, etc. can be attributes of a movie.



**a. Key Attribute**

The key attribute is used to represent the main characteristics of an entity. It represents a primary key. The key attribute is represented by an ellipse with the text underlined.

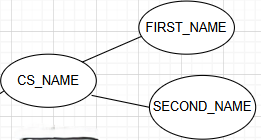


**b. Non-Key Attribute**

Attribute which are non-unique (can or cannot) is called as Non Key Attribute

**c. Composite Attribute**

An attribute that composed of many other attributes is known as a composite attribute. The composite attribute is represented by an ellipse, and those ellipses are connected with an ellipse.



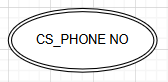
**d. Single Attribute**

An attribute has only one value. Eclipse is used to represent



**e. Multivalued Attribute**

An attribute can have more than one value. These attributes are known as a multivalued attribute. The double oval is used to represent multivalued attribute.



**f. Stored Attribute**

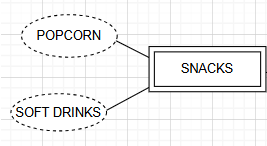
An attribute that cannot be derived, (that is stored) known as a stored attribute. It can be represented by ellipse.

**For example,** A person’s payment



**g. Derived Attribute**

An attribute that can be derived from other attribute is known as a derived attribute. It can be represented by a dashed ellipse.



**h. Required Attribute \***

An attribute that required (mandatory) is known as a required attribute.

**i. Optional Attribute**

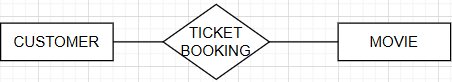
An attribute that is not required (not mandatory) is known as a optional attribute.

**j. Complex Attribute**

Composite + Multivalued

3. Relationship

A relationship is used to describe the relation between entities. Diamond or rhombus is used to represent the relationship.



Types of relationship are as follows:

**a. One-to-One Relationship**

When only one instance of an entity is associated with the relationship, then it is known as one to one relationship.

**For example,** A movie can have one city and time.



**b. One-to-many relationship**

When only one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then this is known as a one-to-many relationship.

**For example,** Customer can do booking with many payments, but the payments is done by the only specific customer.



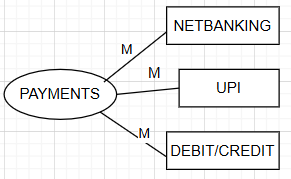
**c. Many-to-one relationship**

When more than one instance of the entity on the left, and only one instance of an entity on the right associates with the relationship then it is known as a many-to-one relationship.



**d. Many-to-many relationship**

When more than one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then it is known as a many-to-many relationship.

****