## Experiment No.: 1

**Title:** Draw E-R diagram for different applications and Convert them into tables

**Objectives:**

1. To learn different data models.
2. To learn about notations used to draw Entity-Relationship Diagram.
3. To get hands on experience with drawing E-R diagram.
4. To learn about conversion of E-R Diagram to table

**Key Concepts:** Data Models, E-R diagram.

## Theory:

## Data Models

Underlying the structure of a database is data model. It is a collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints.

**Different Data Models-**

- Entity – relationship (E-R) model
- Object-oriented data model
- Semi-structured data models
- Hierarchical data model
- Relational model
- Object-relational data model
- Network data model

The entity – relationship (E-R) model is a high-level data model. It is based on a perception of a real world that consists of a collection of basic objects, called entities, and of relationships among these objects.

The relational model is a lower-level model. It uses a collection of tables to represent both data and the relationships among those data. Its conceptual simplicity has led to its widespread adoption; today a vast majority of database products are based on the relational model. Designers often formulate database schema design by first modeling data at a high level, using the E-R model, and then translating it into the relational model.

The object-oriented data model is another data model that has seen increasing attention. The object-oriented model can be seen as extending the E-R model with notions of encapsulation, methods (functions), and object identity.

The object-relational data model combines feature of the object-oriented data model and relational data model.

Semi-structured data models permit the specification of data where individual data items of the same type may have different sets of attributes. This is in contrast with the other data models, where every data item of a particular type must have the same set of attributes. The extensible markup language (XML) is widely used to represent semi-structured data.

Historically, two other data models, the network data model and the hierarchical data model, preceded the relational data model.

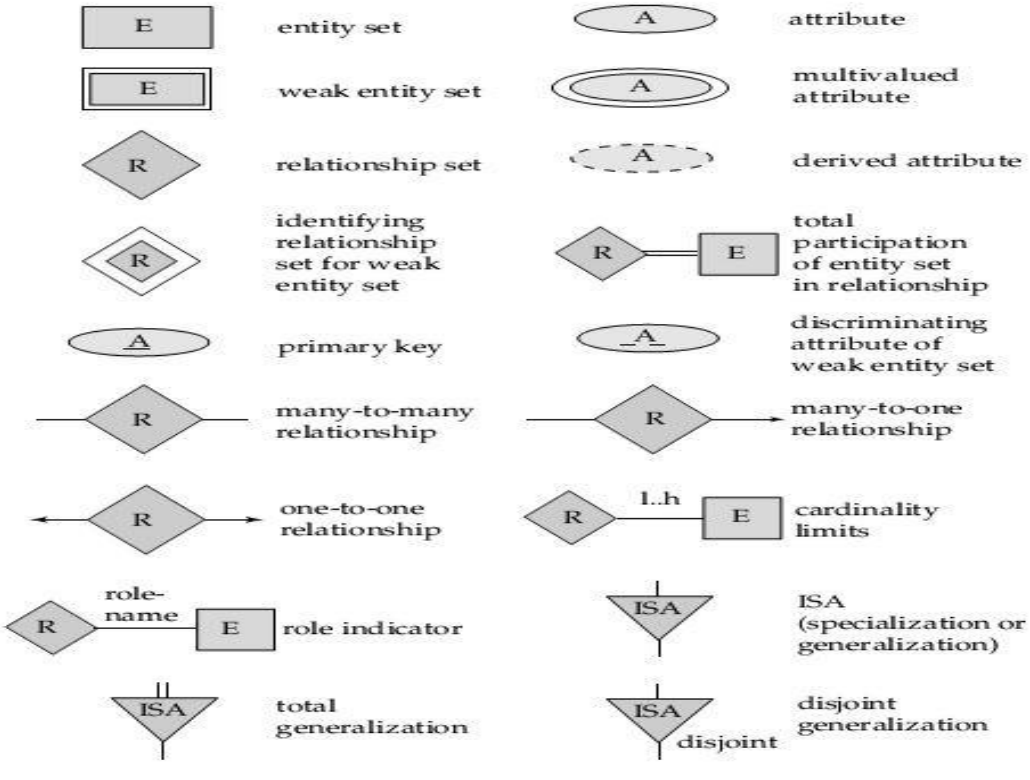
## Entity – relationship (E-R) Diagram

An entity is a "thing" or "object" in the real world that is distinguishable from all other objects. For example, each person in an enterprise is an entity. An entity has a set of properties, and the values for some set of properties may uniquely identify an entity.

An entity set is a set of entities of the same type that share the same properties, or attributes. An entity is represented by a set of attributes. Attributes are descriptive properties possessed by each member of an entity set. Each entity has a value for each of its attributes. For each attribute, there is a set of permitted values, called the domain, or value set, of that attribute.

A relationship is an association among several entities. A relationship set is a set of relationships of the same type.

E-R diagram consists of the following major components:




Symbols used in the E-R notation.

## Sample E-R Diagram

Consider two entity sets, customer and loan, related through a binary relationship set borrower. The attributes associated with customer are customer-id, customer-name, customer-street, and customer-city. The attributes associated with loan are loan-number and amount

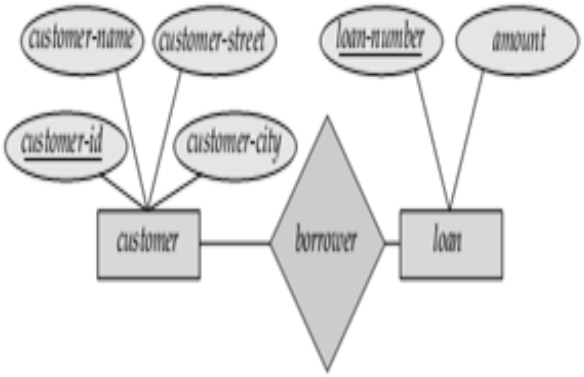



Fig: E-R Diagram correspond to loan & customer

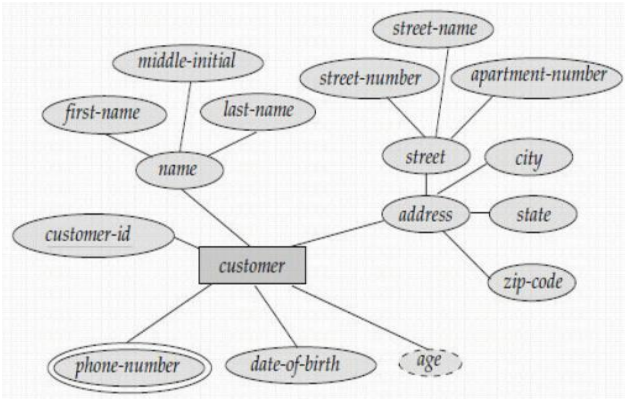



Fig: E-R diagram with composite, multivalued & derived attributes

**Q. Case Study**- Draw E-R Diagram for Banking Enterprise with following requirement:
1. There are multiple banks and each bank has many branches. Each branch has multiple customers
2. Customers have various types of accounts.
3. Some Customers also had taken different types of loans from these bank branches
4. One customer can have multiple accounts and Loans

**Step to Draw E-R Diagram for an application**

**Step 1 : Identify the Entities**

BANK BRANCH LOAN ACCOUNT CUSTOMER

**Step 2 : Identify the relationships**

One Bank has many branches and each branch belongs to only one bank, hence the cardinality between Bank and Branch is One to Many.
• One Branch offers many loans and each loan is associated with one branch, hence the cardinality between Branch and Loan is One to Many.
• One Branch maintains multiple accounts and each account is associated to one and only one Branch, hence the cardinality between Branch and Account is One to Many
• One Loan can be availed by multiple customers, and each Customer can avail multiple loans, hence the cardinality between Loan and Customer is Many to Many.
• One Customer can hold multiple accounts, and each Account can be held by multiple Customers, hence the cardinality between Customer and Account is Many to Many

**Step 3: Identify the key attributes**

BankCode (Bank Code) is the key attribute for the Entity “Bank”, as it identifies the bank uniquely.
• Branch# (Branch Number) is the key attribute for “Branch” Entity.
• Customer# (Customer Number) is the key attribute for “Customer” Entity.
• Loan# (Loan Number) is the key attribute for “Loan” Entity.
• Account No (Account Number) is the key attribute for “Account” Entity

**Step 4: Identify other relevant attributes**

For the "Bank" Entity, the relevant attributes other than "BankCode" would be "Name" and "Address".

• For the "Branch" Entity, the relevant attributes other than "Branch#" would be "Name" and "Address".

• For the "Loan" Entity, the relevant attribute other than "Loan#" would be "Loan Type".

• For the "Account" Entity, the relevant attribute other than "Account No" would be "Account Type".

• For the "Customer" Entity, the relevant attributes other than "Customer#" would be "Name", "Telephone#" and "Address".

**Step 5: Draw complete ER diagram**

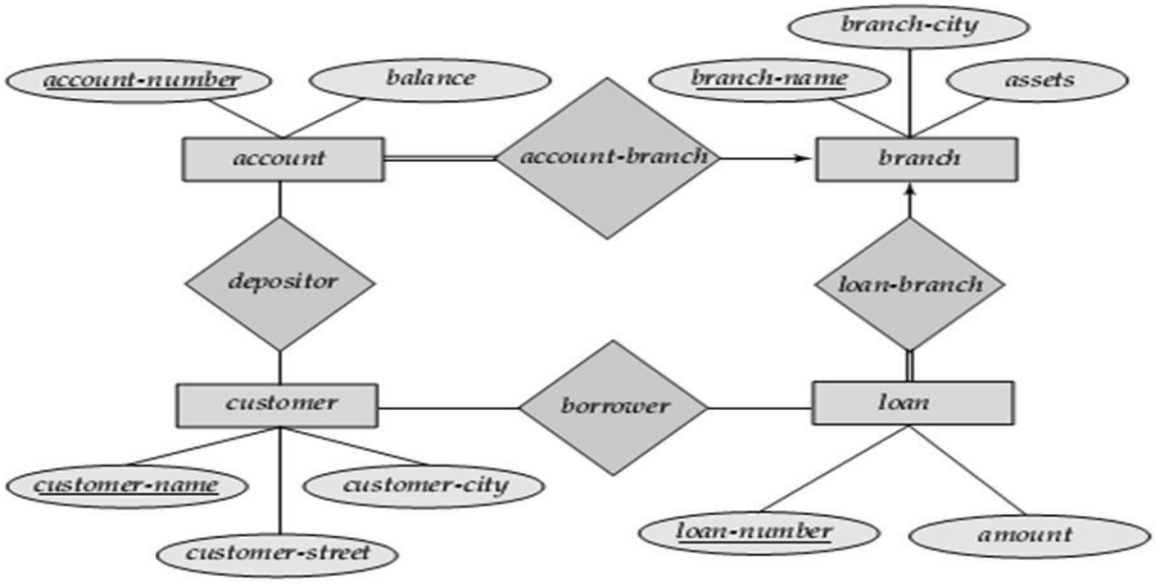
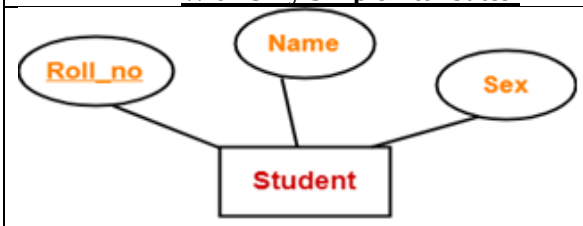
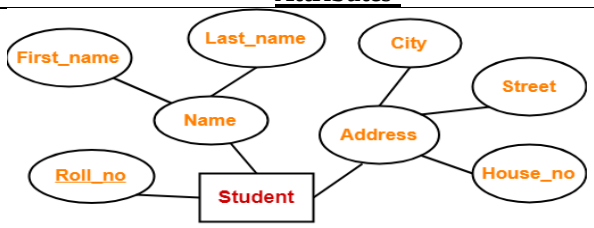
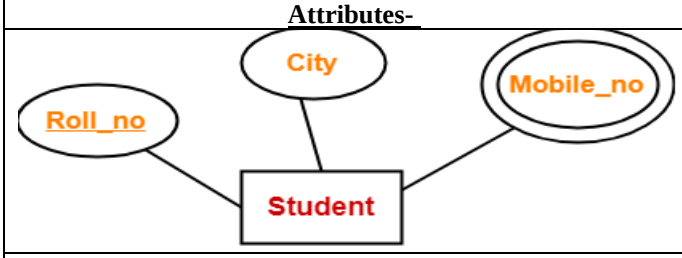
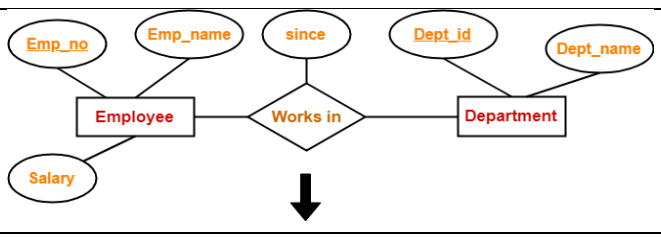


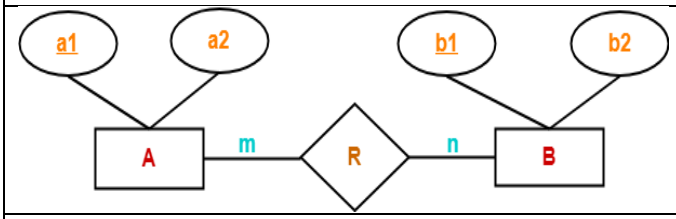

**Figure** E-R diagram for the banking enterprise.

## Conversion of E-R Diagram to Table:

| **Rule-01: For Strong Entity Set With Only Simple Attributes-** | **Rule-02: For Strong Entity Set With Composite Attributes-** |
|---|---|
|   |   |
| Student (Roll_no, Name, Sex) | Student (Roll_no, First_name, Last_name, House_no, Street, City) |
| **Rule-03: For Strong Entity Set With Multi Valued Attributes-** | **Rule-04: Translating Relationship Set into a Table-** |
|   |   |
| Student1 (Roll_no, City) Student2 (Roll_no, Mobile_no) | Works in (Emp_no, Dept_id, since) Employee (Emp_no, Emp_name, Salary) Department (Dept_id, Dept_name) |

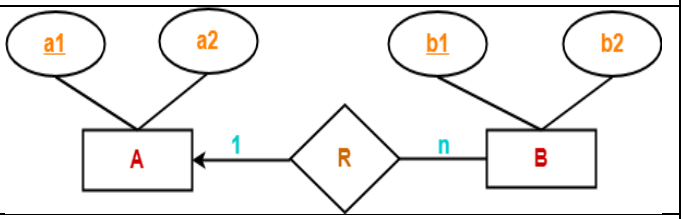
**Rule-05: For Binary Relationships With Cardinality Ratios-**

**Case-01: For Binary Relationship With Cardinality Ratio m:n**



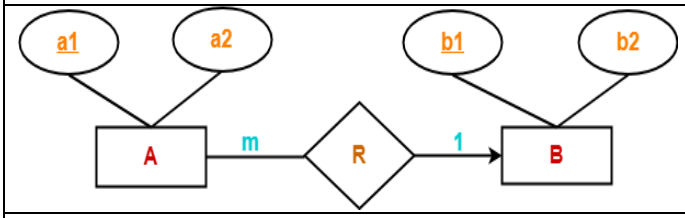

A ( a1 , a2 ) R ( a1 , b1 ) B ( b1 , b2 )

**Case-02: For Binary Relationship With Cardinality Ratio 1:n**



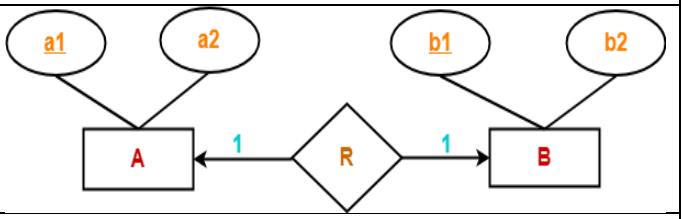

A ( a1 , a2 ) BR ( a1 , b1 , b2 )

**Case-03: For Binary Relationship With Cardinality Ratio m:1**




AR ( a1 , a2, b1 ) B ( b1 , b2 )

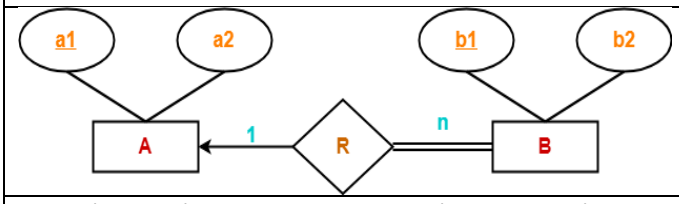
**Case-04: For Binary Relationship With Cardinality Ratio 1:1**




AR ( a1 , a2, b1 ) B ( b1 , b2 ) OR
A ( a1 , a2 ) BR ( a1 , b1 , b2 )

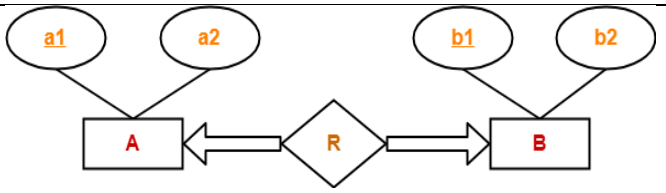
**Rule-06: For Binary Relationship With Both Cardinality Constraints and Participation Constraints-**

**Case-01: For Binary Relationship With Cardinality Constraint and Total Participation Constraint From One Side-**



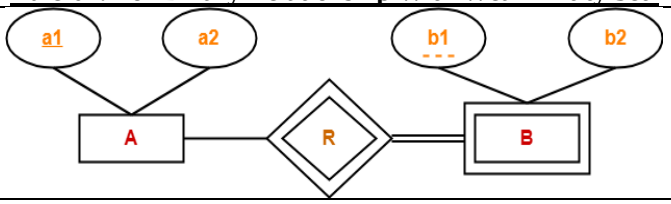

A ( a1 , a2) BR ( a1, b1 , b2 )

**Case-02: For Binary Relationship with Cardinality Constraint and Total Participation Constraint From Both Sides-**




ARB ( a1 , a2, b1 , b2)

**Rule-07: For Binary Relationship With Weak Entity Set-**




A ( a1 , a2) BR ( a1, b1 , b2 )

**Lab Work:**

Case Study – Draw ER Diagram for a college DB and convert them into table.

Requirement:

- A college contains many departments
- Each department can offer any number of courses
- Many instructors can work in a department
- An instructor can work only in one department
- For each department there is a Head
- An instructor can be head of only one department
- Each instructor can take any number of courses
- A course can be taken by only one instructor
- A student can enroll for any number of courses
- Each course can have any number of students