that person? A person may be an employee or customer which can be further categorized into Manager, Staff or Investor, Borrower and so on.

2.9.2 Specialization

Specialization includes subgrouping of entities within an entity set having some distinct nature then other entities. If deep information is needed then go towards specialization. In other words Specialization is a process by which any existing entity set is divided into smaller entity sets according to the distinct or different nature of entities.

Consider the example of Bank in Figure 2.15. Person is an entity set of all people who belongs to bank. Further Person is classified into Employees and Customers of bank. So, Person entity set is divided into Employee entity set and Customer entity set. Employees are further classified into two categories **full time** employees and **part time** employees and so on. Customers are also classified into **Investors** and **Borrowers** and so on.

2.9.3 Generalization

Generalization is a process by which two or more entity sets can be combined into a single entity set by determining similarities between the entities. Its an abstract view of any Enterprise. Generalization proceeds from the recognization that a number of entity sets share some common features. If an abstract view of information is needed then go towards generalization.

Consider the example in Figure 2.15. Here Investor and Borrower are two entity sets. They have common feature that both are Customer of the Bank. Similarly, Employee entity set and Customer entity set can be combined into Person entity set.

2.9.4 Attribute Inheritance

Specialization and generalization leads to attribute inheritance between higher level entity set and lower level entity set. Inheritance is a process by which lower level entity set inherits (or taken) some properties of its higher level entity set.

Consider the Figure 2.15. Here entity sets Employee and Customer inherits attributes Person_ID, Name, Address, Age from Person entity set.

2.9.5 Aggregation

Aggregation is an abstraction process in which a relationship set is considered as higher level entity set.

Consider an example of ternary relationship having three entity sets Employee, Job and Branch with relationship set works-on as shown in Figure 2.16. The information about Managers on employes, managers of particular jobs and of different branches can be taken easily.

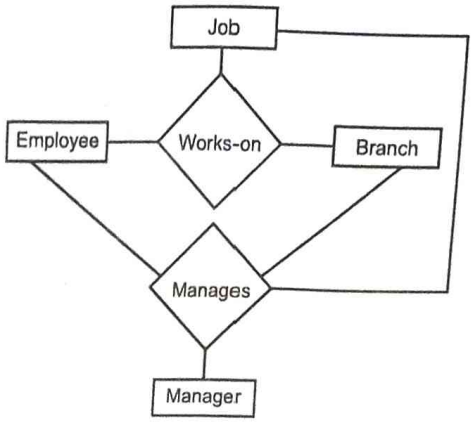

FIGURE 2.16. E-R model.

2.9.6 Specialization and Generalization Constraints

The following constraints are applied on specialization and generalization to capture important business rules of the relationships in an enterprise. There are **Two** types of constraints :

1. Participation Constraints : It tells the participation of entity set in relationship sets. There are two types of participations.

- *Partial participation* : If only some entities from entity set E is participated in relationships in set R then it is known as **Partial participation**. Partial participation is shown in Figure 2.17(*a*).
- *Total participation* : If every entity from entity set E is participated with at least one relation in relationship set R then it is known as **Total participation**. Consider the Figure 2.17(*b*).

Here Customer and Loan are two entity sets and Relationship set is Borrower.

— Every customer may or may not take the Loan so Customer entity set is partially participated.

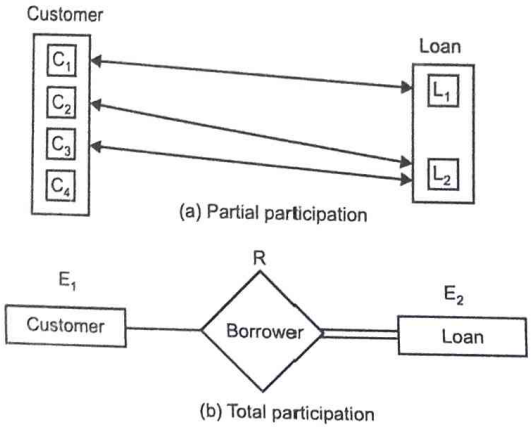

FIGURE 2.17. E-R model.

— But every loan is concerned with at least one customer of bank. So Loan entity set is totally participated.

2. Disjoint Constraints : Disjoint constraints describe the relationship between members of different subclasses. According to Disjoint constraint if the subclasses of a specialization/generalization are disjoint then an entity can be a member of only one subclass of that specialization/generalization. Consider Figure 2.15, subclasses Full Time Employee and Part Time Employee of superclass Employee (discussed earlier that a subclass may be further categorized) are disjoint. Suppose any employee 'Martin' works as part time employee for Bank then it can only belongs to subclass 'Part Time Employee'.

2.9.7 Categorization

Categorization is a modeling process of a single subclass having relationship with more than distinct superclasses. The subclass having more than one superclass is known as category and the process of defining a category is known as categorization. The symbol shown in Figure 2.18(*a*) represents categorization. Consider Figure 2.18(*b*).

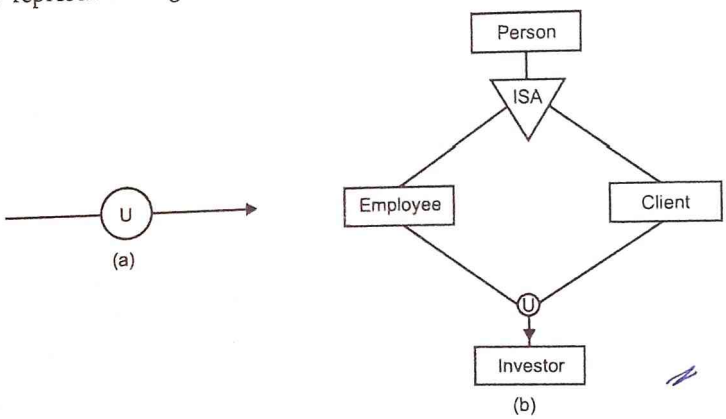

FIGURE 2.18. Categorization.

In a bank, a person can be either a employee or a client and both of them may be investors. So, here subclasses employee and client act as Disjoint Superclasses and Subclass Investor acts as Category.

You cannot combine works-on and managers relationship sets because some workers are not managers. Using aggregation, works-on relationship set acts as higher entity set and solve this drawback of E-R Model. E-R Model with Aggregation is shown in Figure 2.19.

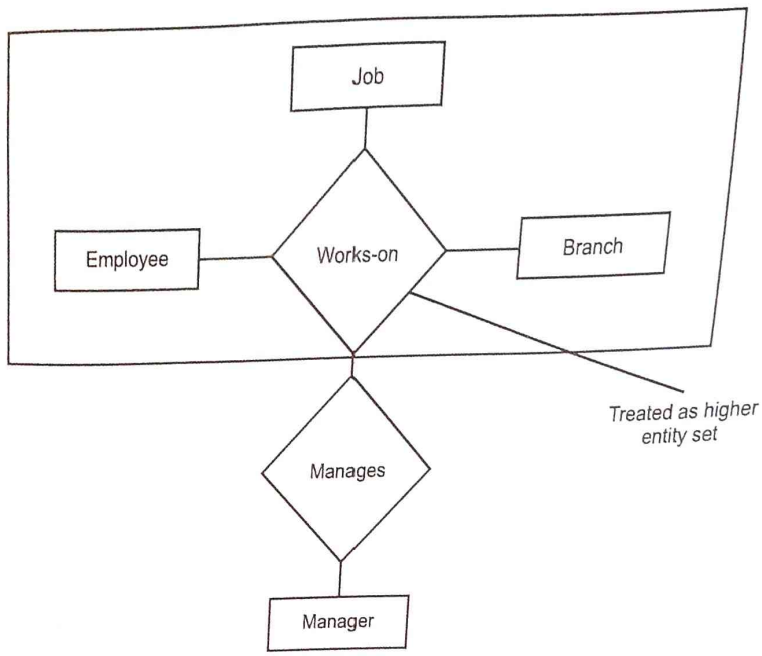

FIGURE 2.19. E-R model with aggregation.

2.10 REDUCTION OF AN E-R AND EER DIAGRAM INTO TABLES

To represent the database in tabular form, E-R diagrams have to be reduced in tables.

For each entity set, make different table and for each relationship set make a different table.

1. Reduction of Strong Entity Sets into Tables

For a strong entity set E with attributes a_1, a_2, ..., a_n, make a table having same name as of entity set E and having n number of columns or table name is equal to entity set name and number of columns is equal to number of attributes. Consider the Figure 2.20 having strong entity set Department with two attributes Dept-ID and Dept-name.

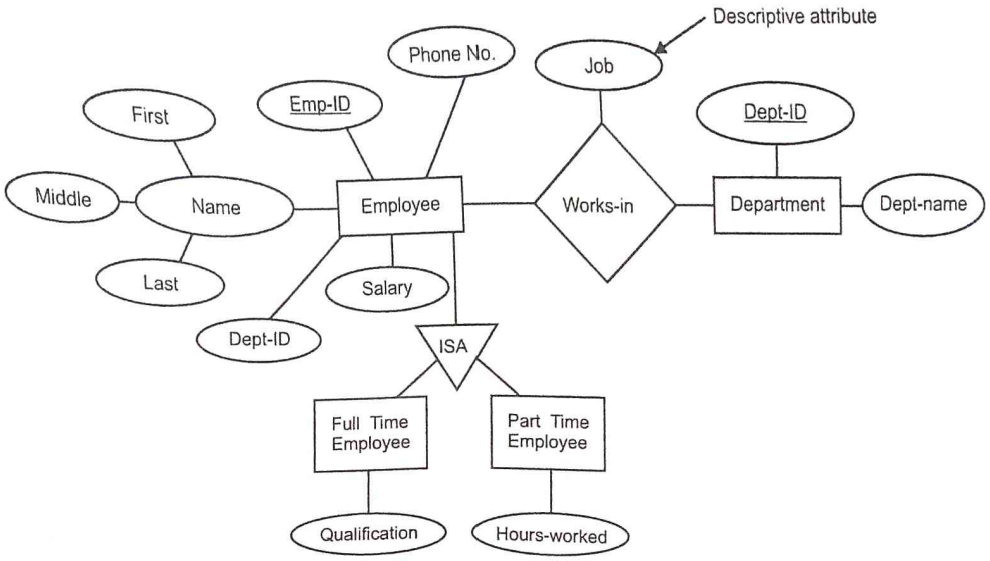

FIGURE 2.20. E-R model of employee and department entity sets.

The corresponding table is shown in Figure 2.21 with table name Department and two columns, Dept_ID and Dept_name.

Department

Dept-ID	Dept-name
10	Sales
20	Development
30	Testing
40	Accounts

FIGURE 2.21. The department table (Reduction of strong entity set).

2. Reduction of Composite Attributes

For a composite attribute, create a separate column for each component attribute or parts of composite attributes. Consider the example shown in Figure 2.20. The Name is a composite attribute with three component attributes First, Middle and Last. So, make three columns First-name, Middle-name and Last-name. The corresponding table is shown in Figure 2.22.

EID-ID	First-name	Middle-name	Last-name	Salary	Dept-ID
A12	Deepak	Kumar	Goyal	15,000	10
S50	Shivi	–	Goyal	75,000	20
51C	Anu	–	Parmar	8,000	10
67B	Ravi	–	–	5,000	40

FIGURE 2.22. The employee table (Reduction of composite attributes).

3. Reduction of Multi-valued Attributes

For multi-valued attributes, make a separate table with columns C1 which represent the primary key of entity set or relationship set and with columns C2 which represent the multi-valued attributes. Rows are equal to total number of values of that attribute. Consider Figure 2.20 in which Phone-No. is multi-valued attribute. So, make a table with two columns, one is Emp-ID (primary key of Employee) and second is Phone-No. (multi-valued attribute). Give any name to that table. The table is shown in Figure 2.23. If any employee has two phone numbers then it is possible to make two different entries in table and so on.

Emp-ID	Phone-No.
A-12	23896
A-12	23897
51-C	38976
51-C	23551
51-C	98941
67-B	23999

FIGURE 2.23. The phone-number table (Reduction of multi-valued attributes).

4. Reduction of Weak Entity Sets

Let A be the weak entity set and B be the strong entity set on which A depends. Then, it is possible to make a table with table name as of Weak Entity Set having columns equal to the

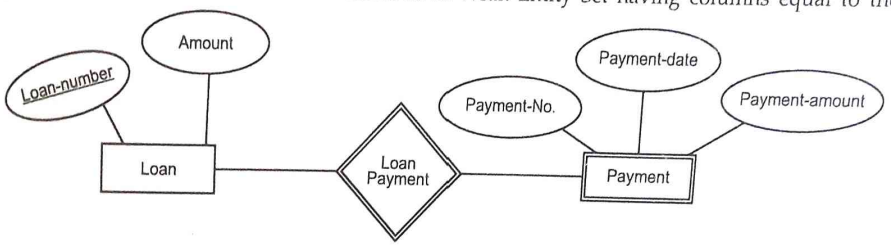

FIGURE 2.24. E-R diagram of weak entity set payment.

attributes of Weak Entity set plus Primary Key of the Strong Entity set on which Weak Entity Set depends. Consider the E-R diagram shown in Figure 2.24, in which Payment is a Weak entity set that depends upon Loan entity set. So, make a table with table name Payment having four columns as shown in Figure 2.25.

Payment

Loan-number	Payment-No.	Payment-date	Payment-amount
E-12	2	19-2-2004	6912
C-55	5	31-1-2005	5000
H-96	11	2-2-2005	2000
P-77	2	6-9-2005	2500

FIGURE 2.25. The payment table (Reduction of weak entity set).

5. Reduction of Relationship Sets

Let R be the relationship set and E1, E2, ..., EN be the entity sets participating in R. Make a table with table name as of Relationship Set having columns equal to number of attributes in relationship set (descriptive attributes) and primary keys of all participating entity sets.

Consider the ER diagram shown in Figure 2.20, having relationship set works-in having two participating entity sets, Employee and Department. The corresponding table is shown in Figure 2.26.

Works-in

Emp-ID	Dept-ID	Job
S-50	20	Engineer
A-12	10	Salesman
51-C	10	Salesman
67-B	40	Accountant

FIGURE 2.26. The works-in table (Reduction of relationship sets).

(i) Redundant Tables

The relationship set between weak and strong entity sets are treated specially. Consider the E-R diagram shown in Figure 2.24, where weak entity set, Payment depends on strong entity set Loan having relationship set loan-payment. Primary key of Entity set Loan is [loan-number] and of Weak entity set is [loan-number, payment-number]. Table of entity set Payment has four attributes [loan-number, payment-number, Payment-date, payment-amount]. If you make table of relationship set loan-payment then it contains attributes [loan-number, payment-number]. This combination is already present in table of Payment. Even, there are no descriptive attributes. So, this table is redundant and discard it.

(ii) Combination of Tables

Consider two entity sets X and Y connecting with relationship set XY. The *n*, three tables named X, Y and XY have to be made. If cardinality ratio between X and Y is many-to-many and X is totally participated then, combine tables X and XY. Consider the E-R diagram shown in Figure 2.27, having two entity sets, Customer and Loan. The relationship is many-to-many because a customer can take many loans and a single loan can be taken by more than one customer or joint loan. Loan entity set is totally participated because every loan refers to some customer. So, combine tables Loan and Borrower. But loan cannot exist with any customer so two tables are needed *i.e.*,

- Loan [loan-number, amount, customer-ID, Income]
- Customer [Customer-ID, Name]

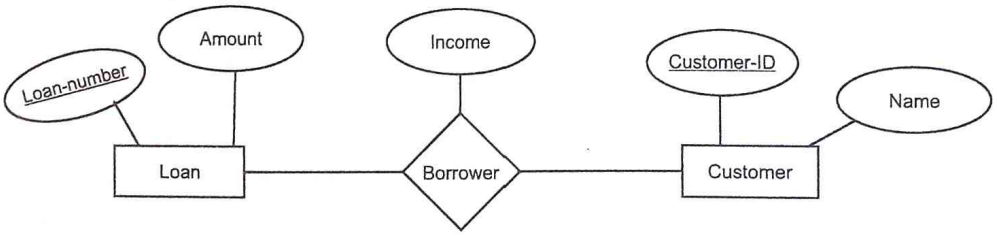

FIGURE 2.27. Combination of tables.

6. Reduction of Generalization

In generalizations, higher level entity sets and lower level entity sets are considered. Make a table for higher level entity set with all its attributes. For lower level entity set, make a table with all its attributes with primary key attributes of its higher level entity set. Consider E-R diagram shown in Figure 2.20, in which Employee is high level entity set and Full Time Employee and Part Time Employee are two lower level entity sets. So, make three tables as given below:

- Employee [Emp-ID, Dept-ID, First-Name, Middle-Name, Last-Name, Salary]
- Full Time Employee [Emp-ID, Qualification]
- Part Time Employee [Emp-ID, Hours-Worked]

7. Reduction of Aggregation

Reduction of aggregation into tables is simple. Consider the E-R diagram shown in Figure 2.19. For all entity sets, make tables as discussed earlier. For making tables for relationship sets, consider the same approach as discussed earlier. Take an example of relationship set Manages. Make a table manages with all descriptive attributes, primary key of entity set Manager and the relationship set works-on.

SOLVED PROBLEMS

Problem 1. Construct an E-R diagram for a hospital with a set of patients and a set of medical doctors. Associate with each patient, a log of various tests and examinations conducted. Construct the appropriate tables for this E-R diagram and list the tables with their attributes, primary key and foreign keys.

Solution. The E-R diagram is shown in Figure 2.28.

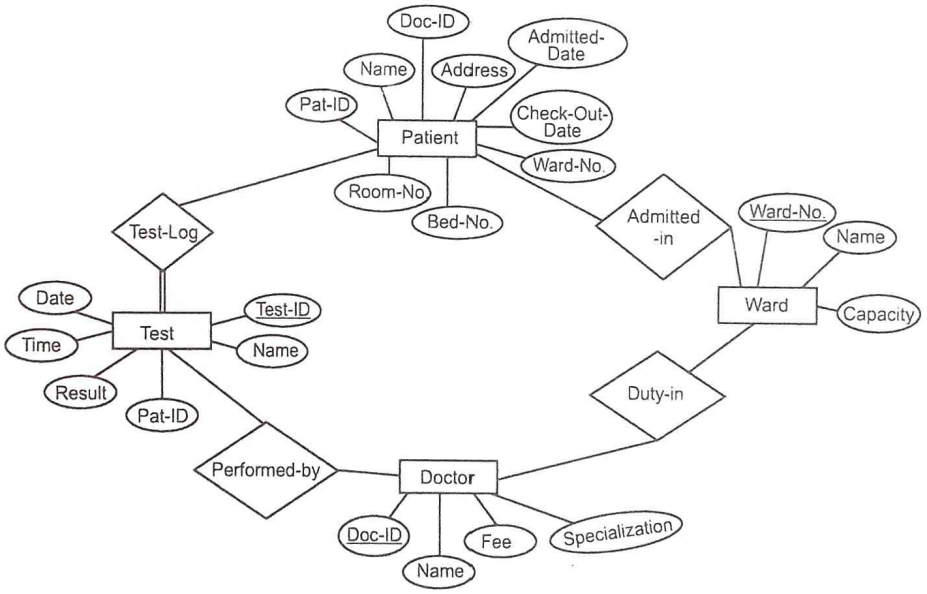

FIGURE 2.28. E-R diagram of hospital.