## Q3. Give detailed answers of the following questions.

i. What are the advantages and disadvantages of wireless networks?

ii. Explain different types of short distance communications technologies.

iii. Describe long distance communications technologies in detail.

iv. Explain different classifications of Satellite Systems.

v. What are the Requirements of Mobile Communication System?

vi. Give important features and limitations of Mobile Communication System.

# UNIT 7 DATABASE FUNDAMENTALS

**After the completion of Unit - 7, the Students will be able to:**

- explain the difference between data and information.
- explain the file management system.
- define Database.
- explain Database Management System (DBMS).
- identify advantages of DBMS over the File Management System.
- identify the role of Database Administrator (DBA).
- describe the types of Database Models.
- explain the types of database languages for relational databases.
- define the basic database terminologies (Field/Attribute/Column, Record/Tuple/Row, Table/Relation, View, Data Type, Key).
- explain the steps for designing a database.
- explain with examples (Entity, attributes, relationships, keys).
- explain the cardinalities and modalities with the help of pictorial examples.
- draw Entity-Relationship (E-R) diagram for library management system, student management system, ticket booking system.
- transform the E-R diagram to the Relational schema (Transforming entities, transforming attributes, transforming relationships).
- normalize relations up to third normal form including integrity rules.

## 7.1 INTRODUCTION

In past, databases were found only in special research laboratories where computer scientists struggled with ways to make them efficient and useful, and published their findings in countless research papers. Today databases are a ubiquitous part of the information technology (IT) industry and business in general. We directly and indirectly use databases every day banking transactions, travel reservations, employment relationships, web site searches, purchases, and most other transactions are recorded in and served by databases.

A database is a collection of related files that are usually integrated, linked or cross-referenced to one another. The advantage of a database is that data and records contained in different files can be easily organized and retrieved using specialized database management software called database management system (DBMS).

### 7.1.1 Data and Information

#### Data

The word Data is the Latin plural of Datum which means "to give" or "something given" (The word data is usually used as a singular term). Data is a collection of facts, figures, numbers or ideas that can be organized and processed. Data may or may not be meaningful and cannot be used for decision making process

#### Information

When facts, figures or numbers (data) are processed and converted into meaningful form that can be used for decision making or any other useful activity, it is called Information.

#### Examples of data and information

1. Students' names in a class are **data** while names of students' in alphabetic order is an **information**.
2. A day's temperature, humidity, wind speed recorded are **data** while weather prediction as cold or warm is **information.**
3. A student's subject marks are **data** while his percentage of marks, grade and position are **information**.

#### Difference between Data and Information

| | Data | Information |
|---|---|---|
| 1 | Data is raw, unorganized facts that need to be processed. Data can be something simple and seemingly random and useless until it is organized. | When data is processed, organized, structured or presented in a given context so as to make it useful, it is called Information. |
| 2 | Data is used as input in the computer. | Information is the output of computer. |
| 3 | Data alone is meaningless and valueless. | Information is useful, meaningful and valuable. |
| 4 | Data is difficult or even impossible to reproduce if lost. For examples in an Exam result all the subject marks award lists are lost or damaged, it would be very difficult to reproduce them. | Information is easier to reproduce if lost. For example, if the obtained marks of the students' are lost, these can easily be recalculated from the award lists |
| 5 | Data is an independent entity. | Information depends on data. |

### 7.1.2 File Management System

File Management system also known as Conventional file system or simply file system is a method of storing and organizing collection of data in the form of files on the secondary storage devices. These files are accessed with the help of certain file handling program developed in programming languages like C, BASIC and Pascal. Conventional file system is no longer used due to its demerits and limitations as mentioned below.

## Limitations of File Management System

### i. Data Redundancy

Independent data files include a lot of duplicated data; the same data is recorded and stored in several files. This data redundancy causes problems when data is to be updated. Hence data redundancy is more in case of file approach.

### ii. Inconsistency

In this system, data is not consistent due to redundant storage. If a data item is changed all the files containing that data item need to be changed and updated properly. If all the files are not updated properly there may be high risk of inconsistency.

### iii. Intensive Coding

The processing tasks like searching, editing, deletion and updating should have separate programs. It means there are no built in functions available and needs coding every time to perform these operations.

### iv. Data Isolation

Data is scattered in various files and the files may be in different format. The users have to write new application program to retrieve data from these files. This is difficult, time-consuming, and costly.

### v. Data Program Dependence

In file management systems, changes in the format and structure of data and records in a file require that changes be made to all of the programs that use that file. This program maintenance effort is a major burden of file management systems.

### vi. Difficulty in Accessing Data

It is not easy to retrieve information using a file management system. The simplest data retrieval task from file requires extensive programming. Also this is a time consuming and a high skill activity.

### vii. Integrity Problems

Integrity means reliability and accuracy of data. Data values may need to satisfy some integrity constraints. For example the balance field value of a bank account must be greater than 5000. We have to handle this through program code in file management systems.

### viii. Atomicity Problem

Atomicity means that either one transaction or another should take place as a whole. It is difficult to ensure atomicity in file management system. For example while transferring an amount of Rs. 1000/- from Account A to account B, if a failure occurs during execution there could be situation like Rs. 1000/- are deducted from Account A and not credited in Account B.

### ix. Poor data security

All the files are stored as text files. These files can be easily located and trapped because file management system has no centralized data security. Enforcing security constraints such as effective password protection, locking parts of file in file processing system are very difficult to program.

## 7.1.3 Database

Database is a shared collection of logically related data (and a description of this data i.e. metadata), designed to meet the information needs of multiple users in an organization. It usually refers to data organized and stored in a computer that can be searched and retrieved by a computer program called DBMS.

A database combines records previously stored in separate files into a shared pool of data records that provides data for many applications. The data stored in a database is independent of the application programs using it and the type of secondary storage devices on which it is stored.

## 7.1.4 Database Management System (DBMS)

A database management system (DBMS) is a set of programs that allow users to create, maintain and manipulate database, and store or retrieve data from those database files. Manipulation of data includes the following.

- Adding new data, for example adding details of new student.
- Deleting unwanted data, for example deleting the details of students who have completed course.
- Changing existing data, for example modifying the fee paid by the student.

The DBMS helps to create an environment in which users have better access to data. DBMS helps to give an integrated view of the organization's operations. The DBMS makes it possible to share the data in the database among multiple applications and users.

The following are few examples of the database systems, managed by DBMS.

- ➢ Customer information system
- ➢ Inventory information
- ➢ Library management
- ➢ Accounting and bookkeeping

Typical DBMS include Microsoft Access, Microsoft SQL Server, Sybase, IBM DB2, Oracle, Ingres and MySQL.

## 7.1.5 Advantages of DBMS over the File Management System

The DBMS is preferred over Conventional File Management System due to the following advantages:

### i. Controlled Data Redundancy

In the Conventional File Management System, every user maintains its own files for handling data. The database approach combines redundant data into a single, logical structure. Every primary fact is stored at only one place in the database.

### ii. Data Consistency

Minimal data redundancy leads to consistent data. By controlling data redundancy, data inconsistency is greatly reduced. If a data item appears only at one place it is easy to maintain. Any change will automatically take effect at all places wherever this data is used.

### iii. Data Sharing

A database is designed as shared .resource compared to Conventional File System. Authorized users are granted access to use the shared database according to their needs and rights.

### iv. Enforcement of Standards

Since all access to database must be through DBMS so standards are easier to enforce. Standards may relate to the naming of data, format of data, and structure of data.

### v. Security

In Conventional filing systems there is no centralized security system which restricts users according to their role in the organization. DBMS makes it easier to enforce security restrictions since database is centralized. Users are provided permissions to access data according to their rights.

### vi. Reduced Program Maintenance

Many DBMS provide several tools to use in program development and maintenance. It reduces the overall time for developing and maintenance of programs.

### vii. Backup and Recovery

DBMS provides facilities for backup and recovery from failures including disk crash, power failure, software errors, which may bring the database from the inconsistent state to a state prior to the failure.

### viii. Program-Data Independence

The separation of data from the application programs that use the data are called data independence. In database approach data descriptions (metadata) is stored in a central location called the repository. With data independence, user can change data without changing the application programs that process the data.

### ix. Support for multiple views

DBMS may allow different users to see different "views" of the database, according to the perspective each one requires. For example, the people using the payroll system may not require to see data about students and class schedules.

### x. User-friendliness

Using DBMS user can easily access and manipulate data in database. DBMS provide a user friendly interface for interaction with database. In most cases, DBMSs also reduce the reliance of individual users on computer specialists to meet their data needs.

## 7.1.6 Role of the Database Administrator (DBA)

DBA is a person responsible for managing the overall database management and physical realization of the database system. The primary role of the DBA

is to administer, develop and implement policies and procedures necessary to ensure the security and integrity of the database. Other primary responsibilities of a DBA include:

- Implementation of Data Models
- Database design
- Performance issues
- Storage structure and access-method definition
- Granting of authorization for data access
- Routine maintenance which may include:
    - Periodically backup the database
    - Ensuring that enough free disk space is available
    - Monitoring jobs running on the database and performance

## 7.1.7 Database Models

A database model is a set of rules or specifications which state that how data can be stored, organized and manipulated in a database system. It provides tools for describing data, data relationships, data semantics and consistency constraints. Several models have been suggested. Few common models include:

a. Hierarchical Database Model

b. Network Database Model

c. Relational Database Model

d. Object Oriented Database Model

e. Object Relational Database Model

### a. Hierarchal Database Model

In this type of model, data is organized into a tree-like structure. There is a hierarchy of parent and child segments. This structure implies that a record can have repeating information, usually in the child data segment. Each parent can have many children but each child has only one parent. Figure 7.1 depicts a hierarchical database model.

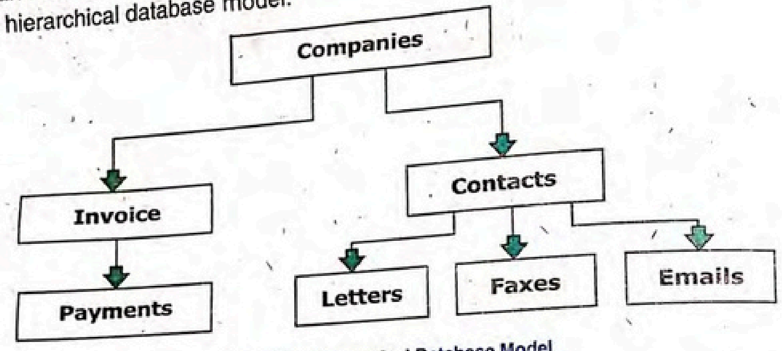


Figure 7.1 Hierarchical Database Model

## b. Network Database Model

A network database is similar to a hierarchical database model except that each child can have more than one parent record. A child record is referred to as a "member" and a parent record is referred to as an "owner". The advantage of the network database is its ability to establish relationships between different branches of data records and thus offer increased access capability for the manager. Figure 7.2 shows a network database model.

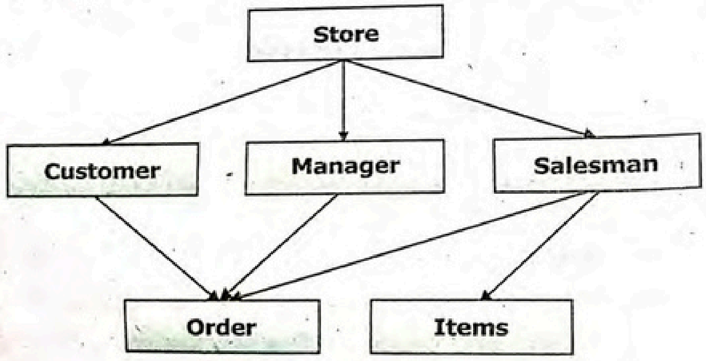


Figure 7.2 Network Database Model

## c. Relational Database Model

A relation is a table with columns and rows. The Relational model uses a collection of tables/relations to represent both data and the relationship among those data. Each table has multiple columns and each column has a unique name. Information about particular entity is represented in rows (also called tuples) and columns. Figure 7.3 illustrates an example showing an Entity-Relationship (E-R) diagram that represents entities (tables) and their relationships for a sample relational model.

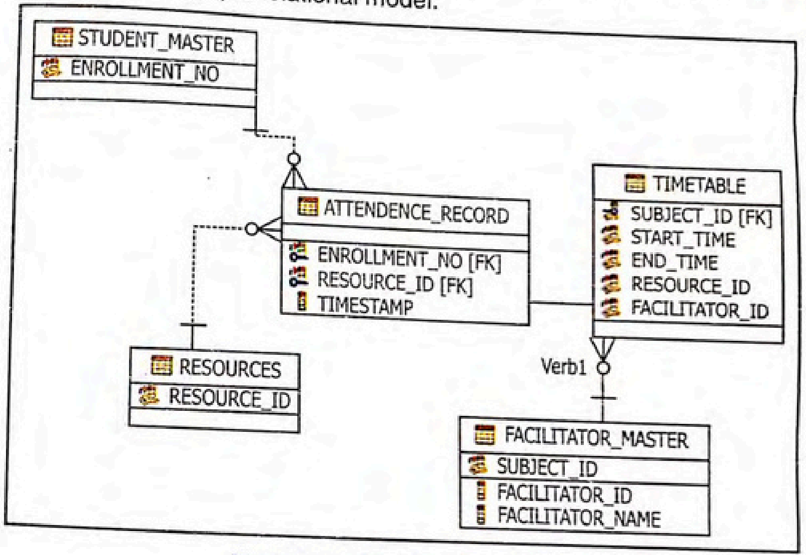


Figure 7.3 Relational Database Model

## d. Object-Oriented Database Model

Object-Oriented database Model is a database model in which information is represented in the form of objects as used in object-oriented programming. Object oriented databases are different from relational databases and belong together to the broader database management system. When database capabilities are combined with object-oriented programming language capabilities, the result is an object-oriented database model.

### e. Object Relational Database Model

Object relational database model add new object storage capabilities to the relational database systems at the core of modern information systems. These new facilities integrate management of traditional fielded data, complex objects such as time-series and geospatial data and diverse binary media such as audio, video, images, and applets. By encapsulating methods with data structures, an ORDB Model server can execute complex analytical and data manipulation operations to search and transform multimedia and other complex objects.

## 7.1.8 Database Languages for Relational Databases

To manipulate data, one approach is to interact directly with the DBMS using a special language called a query language.

### Structured Query Language (SQL)

Almost all relational database management systems use SQL (Structured Query Language) for data manipulation and retrieval. SQL is the standard language for relational database systems. SQL is further divided into the following languages:

a. DDL (Data Definition Language)

b. DML (Data Manipulation Language)

c. DCL (Data Control Language)

### a. Data Definition Language (DDL)

Data definition language (DDL) is a database language that defines the structure in which data are stored. These structures may include database, table, query fields and records.

### b. Data Manipulation Language

A data manipulation language (DML) is a language that enables users to access or manipulate data. The types of access are:

- Retrieval of information from database
- Insertion of new data into the database
- Deletion of data from database
- Modification of data stored in the database

### c. Data Control Language

Data Control Language is a database language used to control access to the data in a database. Examples of DCL are:

- Giving rights to the users
- Revoking of the already given rights

## 7.2 BASIC DATABASE TERMINOLOGIES

The following are some basic database terminologies.

### a. Field/Attribute/Column

An attribute is a property or characteristic of an entity that is of interest to the organization. Following are some attributes:

- STUDENT: student no, name, address, phone no
- EMPLOYEE: employee no, name, address, skill

In Relational Database Model, an attribute is represented by column. Whereas the term field is usually used for a column but it is more correct to use field to specify a single item that exists at the intersection of a row and column.

### b. Record/Tuple/Row

A collection of related fields treated as a single unit is called a record. For example a student's record includes a set of fields that contains Roll No, Name, Class, Date of Birth and Address.

In Relational database, a Row or a Record or Tuple represents a single occurrence of an entity.

### c. Table/Relation

A table is set of values that are organized using vertical named columns and horizontal rows. A table has a specified number of columns, but may have any number of rows. Whereas in terms of relational database, a table can be considered as relation, but the two are not equivalent. For example a table may contain duplicate rows, but a true relation cannot contain duplicate rows.

### d. View

A view is a dynamic and virtual table that may not exist in the database but is derived from one or more base tables. In simple words there is no stored table that represents the view instead a definition or query representing the view is stored in the database.

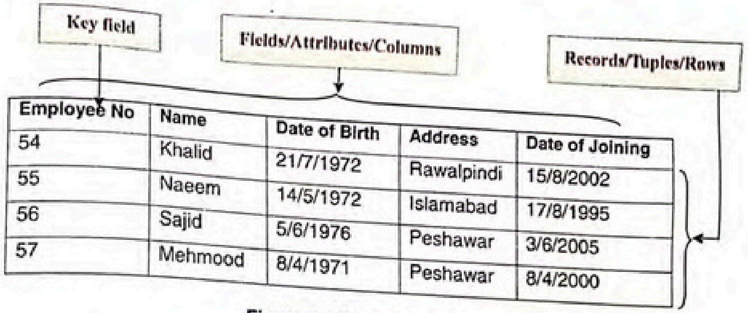
### e. Data Type

Data type is the classification of a particular type of information. It is easy for human beings to distinguish among different types of data but most database systems require the user to specify the type of each data field. Integer, floating-point and character (text) are the common data types.

### f. Key

In order to track and analyze data effectively, each record in a table requires an identifier that uniquely indentifies it, is called a key. The key must be completely unique to a particular record. For example student's roll number, employee number, and customer number, etc. are key fields.

Figure 7.4 shows a complete Table or Relation.



| Employee No | Name | Date of Birth | Address | Date of Joining |
|---|---|---|---|---|
| 54 | Khalid | 21/7/1972 | Rawalpindi | 15/8/2002 |
| 55 | Naeem | 14/5/1972 | Islamabad | 17/8/1995 |
| 56 | Sajid | 5/6/1976 | Peshawar | 3/6/2005 |
| 57 | Mehmood | 8/4/1971 | Peshawar | 8/4/2000 |

Figure 7.4 Table/Relation

## 7.3 PLANNING A DATABASE

Database planning is a systematic approach to the development of database that moves from concept to design and development to implementation. A well-designed database promotes consistent data entry and retrieval.

### 7.3.1 Steps for Designing a Database

The process of database design is divided into different steps. These are:

a. Problem Identification/definition
b. Feasibility Study
c. Requirement Analysis
d. Identifying Entities and Attributes
e. Assigning Names to Tables and Columns

### a. Problem Identification/definition

One must know what the problem is before it can be solved. In this step a statement is prepared specifying the scope and objective of the problem. For example, the Admission Section Head has been getting complaints of poor service from student. This may lead an initial investigation to find whether a new system can solve the problem. If the report suggests a new system, this leads to the next phase which is the feasibility study of the new system.

### b. Feasibility Study

Feasibility Study also known as pilot study is conducted to determine whether new database system should be developed or not. Feasibility study depends on various factors which may include time needed for development, cost of development, resources needed in and after development, training of personals, and maintenance of the new system after implementation.

### c. Requirement Analysis

Requirements analysis is the process of understanding the customer needs and expectations from the proposed system. Requirements are a description that how new system should behave or features to be included in it. In order to collect all these required information, a database analyst need to spend a lot of time within the business organization observing the current system, talking to the end-users and examine the overall system.

### d. Identifying Entities and Attributes

After requirements identification, the next step is to identify the entities and its attributes. An entity is the main data object that is of significant interest to the organization. It is usually a person, place, thing, or event to be recorded in the database.

An Attribute is a property that describes an entity. For example if employee is an entity then, the employee's name, age, address, salary and job etc. are the attributes.

### e. Assigning names to Tables and Columns

Entities are converted to tables and attributes to columns of the tables, once entities and attributes are identified. There are no standard conventions for naming tables and columns but all names should be meaningful and consistent throughout the database. Table names represent business object, so meaningful words should be used for naming tables, for example "Customers" for customer's table. In case of attributes, meaningful names should be used wherever possible.

## 7.4 DATA MODELING AND ENTITY RELATIONSHIP DIAGRAM

### a. Data Modeling

A data model is a logical representation of data in an organization. It is used to describe the elements of a database system and their relationships. A data model can be thought of as an entity relationship diagram that illustrates the relationships between data. Capturing all the possible relationships in a data model can be very time-intensive. It is an important step and should not be avoided. Data modelers often use multiple models to view the same data and ensure that all processes, entities, relationships and data flows have been identified.

### b. ENTITY RELATIONSHIP DIAGRAM (ERD)

Entity Relationship model is expressed in terms of entities, the relationships (or associations) among those entities, and the attribute (or properties) of both the entities and their relationships in the business environment. Diagrams created by this process are called **Entity-Relationships Diagrams (ERDs).**

An entity-relationship (ER) diagram is a graphical representation of entities and their relationships to each other. It is typically used in computing for organization of data within databases or information systems. The following are a few symbols used in ERD.

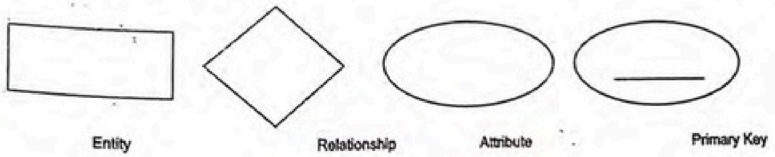


Figure 7.5 ERD symbols

## 7.4.1 Elements of Entity Relationship Diagram

### a. Entity

An entity is a person, place, object, event or concept in user environment about which the organization wishes to maintain data. Examples of entities are as under:

- PERSON: employee, student, patient
- PLACE: city, state, country
- OBJECT: machine, building, automobile
- EVENT: sale, admission, exams
- CONCEPT: account, course, work center

#### Entity Instance

A single occurrence of an entity is called entity instance. For example the Instances of Entity STUDENT are as under:

| | |
|---|---|
| RollNo: 454 | RollNo: 343 |
| Name: Ahmad Ali | Name: Nauman |
| Address: Peshawar | Address: Mardan |
| PhoneNo: 0454450454 | PhoneNo: 0445666454 |

### b. Attribute

An attribute is a property or characteristic of an entity that is of interest to the organization. Following are some attributes:

- STUDENT: student no, name, address, phone no
- EMPLOYEE: employee no, name address, skill

### c. Relationship

A meaningful association among entities is called a relationship. A relationship describes how the data is shared between entities. Relationships are represented by a diamond symbol connected to the related entities.

In the original Chen notation, the relationship is placed inside a diamond, e.g. managers manage employees as shown in Figure 7.6.

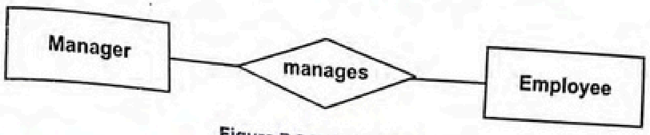


Figure 7.6 Relationship

#### Degree of a Relationship

The number of participating entities in a relationship is known as the degree of the relationship. It has three types.

i. A **unary relationship** exists when an association is maintained within a single entity. Unary relationships are also known as a ***recursive* relationship**. It is a relationship where the same entity participates more than once in different roles.

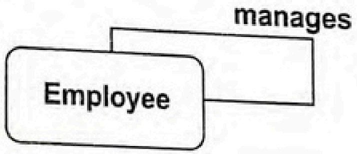


Figure7.7 Unary/Recursive relationship

In the above example employees are managed by employees.

ii. A **binary relationship** exists when two entities are associated with each other through a relationship. OR If there are two entity types involved, it is a *binary* relationship type.

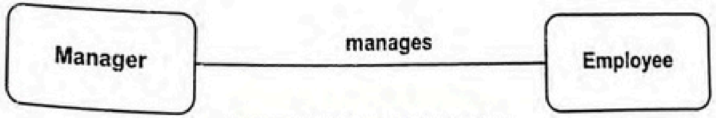


Figure7.8 Binary Relationship

Like unary relationship, in binary relationship the associations between two entity types may be described as one-to-one, one-to-many or many-to-many.

**one-to-one:** one instance of an entity (A) is associated with only one instance of another entity (B). For example, in a database one instance of entity type Country (A) is associated with only one instance of entity type President (B).

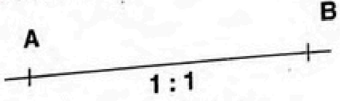


Figure7.9 one-to-one

**one-to-many:** one instance of an entity (A) is associated with zero, one or many instances of another entity (B), but for one instance of entity B there is only one instance of entity A. For example, for a company with all employees working in one building, the building name (A) is associated with many different employees (B), but those employees all share the same singular association with entity A.

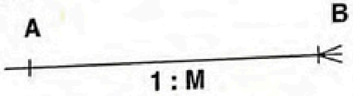


Figure7.10 one-to-many

**many-to-many:** one instance of an entity (A) is associated with one, zero or many instances of another entity (B), and one instance of entity B is associated with one, zero or many instances of entity A. For example, for a company in which all of its employees work on multiple projects, each instance of an employee (A) is associated with many instances of a project (B), and at the same time, each instance of a project (B) has multiple employees (A) associated with it.

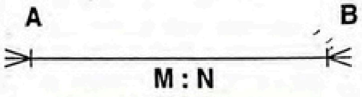


Figure7.11 many-to-many

iii. A **ternary relationship** exists when three entities are associated. OR If there are three entity types involved it is a ternary relationship type.

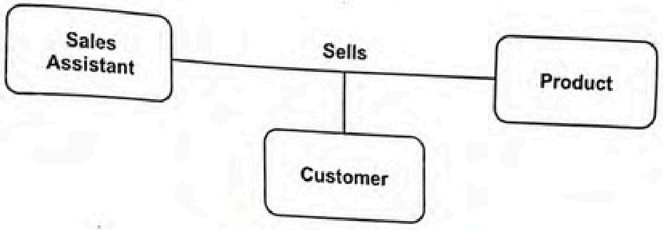


Figure 7.12 Ternary relationship

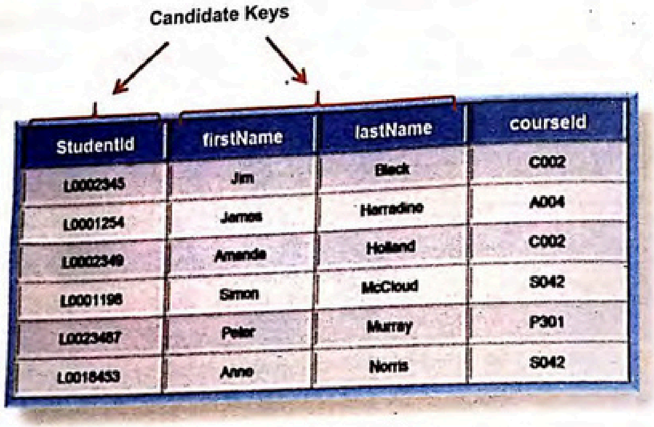
## d. Keys

A key is the relational means of specifying uniqueness. A key is an important part of relational database and a vital part of the structure of a table. A Key ensures each record within a table can be uniquely identified by one field or a combination of fields within the table.

There are different types of keys.

### i. Candidate Key

A Candidate key is any set of one or more columns whose combined values are unique among all occurrences. In other words a candidate key can be used to uniquely identify each record in the table. Every table must have at least one candidate key but at the same time can have several.



| StudentId | firstName | lastName | courseId |
|---|---|---|---|
| L0002345 | Jim | Black | C002 |
| L0001254 | James | Harradine | A004 |
| L0002349 | Amanda | Holland | C002 |
| L0001198 | Simon | McCloud | S042 |
| L0023487 | Peter | Murray | P301 |
| L0018453 | Anne | Norris | S042 |

Figure 7.13 Candidate keys

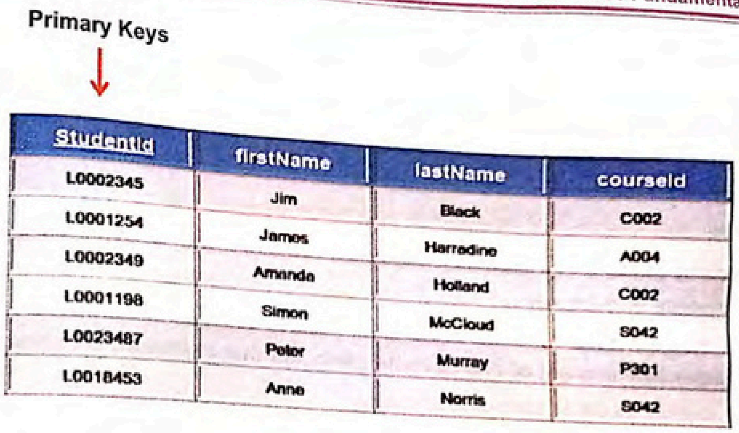
In the above example, student_id can be a candidate key which uniquely identifies the students in a student table. But at the same time, the combination of student's first name and last name also form a candidate key. These both can be candidate keys for student table.

A candidate key must have the following properties:

- It must contain unique values
- It must not contain null values
- It contains the minimum number of fields to ensure uniqueness
- It must uniquely identify each record in the table

### ii. Primary Key

A primary key is a field, or a combination of fields used to identify particular rows in a relation. Primary key, as its name shows is the primary reference for the table and is used throughout the database to refer this table. Primary key must contain unique value and never be null.

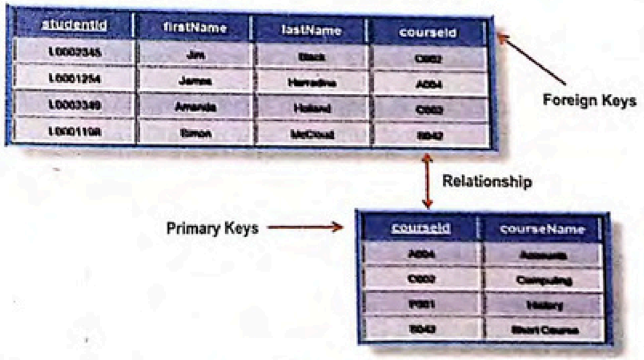


| StudentId | firstName | lastName | courseId |
|---|---|---|---|
| L0002345 | Jim | Black | C002 |
| L0001254 | James | Harradine | A004 |
| L0002349 | Amanda | Holland | C002 |
| L0001198 | Simon | McCloud | S042 |
| L0023487 | Peter | Murray | P301 |
| L0018453 | Anne | Norris | S042 |

Figure 7.14 Primary keys

### iii. Foreign Key

A foreign key is generally a primary key from one table that appears as a field in another table where the first table has relationship to the second. For example, if we have a table A with a primary key X that is linked to a table B where X is a field in B, then X would be a foreign key in B.



| studentId | firstName | lastName | courseId |
|---|---|---|---|
| L0002345 | Jim | Black | C002 |
| L0001254 | James | Harradine | A004 |
| L0002349 | Amanda | Holland | C002 |
| L0001198 | Simon | McCloud | S042 |

| courseId | courseName |
|---|---|
| A004 | Accounts |
| C002 | Computing |
| P301 | History |
| S042 | Short Course |

Figure 7.15 Foreign keys

In the above example, courseld is primary key in courses table, whereas it is foreign key in the student table. The two tables are linked through courseld.

### iv. Composite Primary Key

In some tables a single column cannot uniquely identify entities (rows). In that case we have to use two or more columns to uniquely identify rows of the table.

When a primary key contains two or more columns or fields, it is called as composite primary key.

### v. Super Key

A Super Key is a set of one or more attributes that are taken collectively and can identify all other attributes uniquely.

If we add additional attributes to a primary key, the resulting combination would still uniquely identify an instance of the entity set Such keys are called super keys A primary key is therefore a minimum super key For example, if DOB (date of birth field or attribute) is the primary key, then by adding some additional information about the day of the month key in the DOB field, this field or attribute becomes more powerful and useful Such type of key is called super key.

### vi. Secondary Key

A secondary key is an attribute or combination of attributes that is not a primary key and can have duplicate data i.e. it can identify a group of records. In other words secondary key is used after the identification of the primary key. e.g. in a STUDENT table if Roll Number is a primary key, then Name of the student, address of the student, Phone number of the student and the fees paid by the student are secondary keys. Secondary keys are used to speed up searching process in databases.

### vii. Alternate Key

A candidate key which is not selected as the primary key is called alternate key. For example if Roll No. and Registration No. fields are the candidate keys and if Roll No. is selected as the primary key then Registration No. will work as the alternate key.

## 7.4.2 Cardinality and Modality

**Cardinality** refers to the maximum number of times an instance in one entity can be associated with instances in the related entity.

**Modality** refers to the minimum number of times an instance in one entity can be associated with an instance in the related entity.

Cardinality can be 1 or Many and the symbol is placed on the outside ends of the relationship line, closest to the entity. Modality can be 1 or 0 and the symbol is placed on the inside, next to cardinality symbol.

For a cardinality of 1 a straight line is drawn. For a cardinality of Many a foot with three toes is drawn. For modality of 1 a straight line is drawn. For a modality of 0 a circle is drawn.

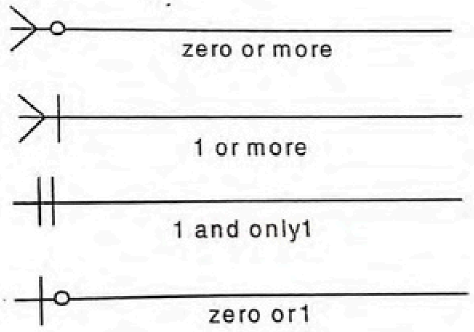
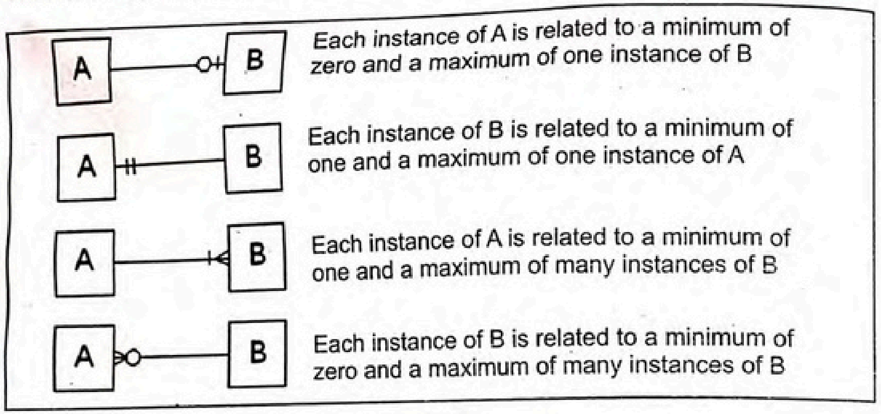


Figure 7.16

*The following diagram indicates all of the possible combinations:*



Each instance of A is related to a minimum of zero and a maximum of one instance of B

Each instance of B is related to a minimum of one and a maximum of one instance of A

Each instance of A is related to a minimum of one and a maximum of many instances of B

Each instance of B is related to a minimum of zero and a maximum of many instances of B

**Figure 7.17**

## 7.4.3 Sample E-R Diagrams

The following are few examples of E-R Diagrams for some systems like Library Management System, Student Management System and Ticket Booking System

### E-R Diagram for Library Management System

In the library Management system shown in Figure 7.18, the following entities and attributes can be identified.

- ***Book:*** The set of all the books in the library. Each book has a Book-id, Title, Author and subject as its attributes.
- ***Member:*** The set of all the library members. The member is described by the attributes Memb_id, Name, Address and Telephone.

- ***Publisher:*** The set of all the publishers of the books. Attributes of this entity are Pub_ID, Address and Name.
- The relationship between Publisher and Book is one-to-many because a Publisher publishes many books.
- The relationship between Member and Book is also one-to-many because a Member can borrow one or more books.

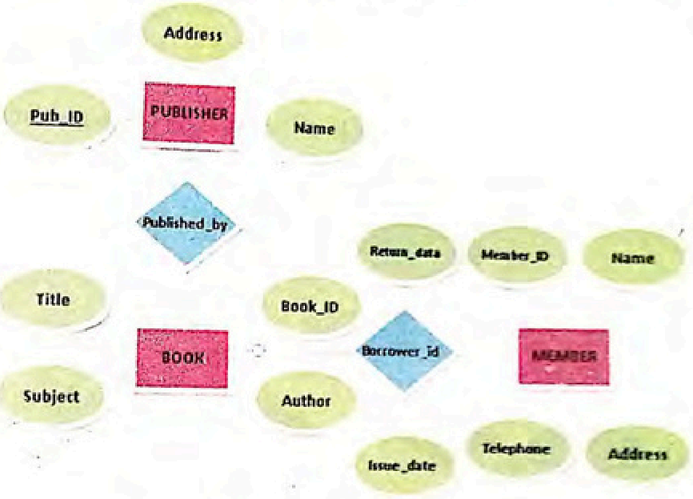


**Figure 7.18 E-R Diagram for Library Management System**

### E-R Diagram for Student Management System

In the student Management system shown in Figure 7.19, the following entities and attributes can be identified.

- ***Department***: The set of all the departments in an organization. Each department has a DepartmentName and Location as its attributes.
- ***Instructor***: The set all the instructors in a department. The instructor is described by the attributes Instructor_ID, first_name, last_name and phone.

- ***Course***: The set of all the courses offered by a department. The course is described by the attributes Course_ID, duration and course_name.
- ***Student:*** The set of all the students of the institution. Attributes of this entity are Student_ID first_name, last_name and phone.
- The relationship between Department and Course is one to many because a department offers many courses for study.
- The relationship between Department and Instructor is one to many because a department can hire many instructors to teach different courses in the institution.
- The relationship between Instructor and Course is one to many because an instructor teaches one or many courses.

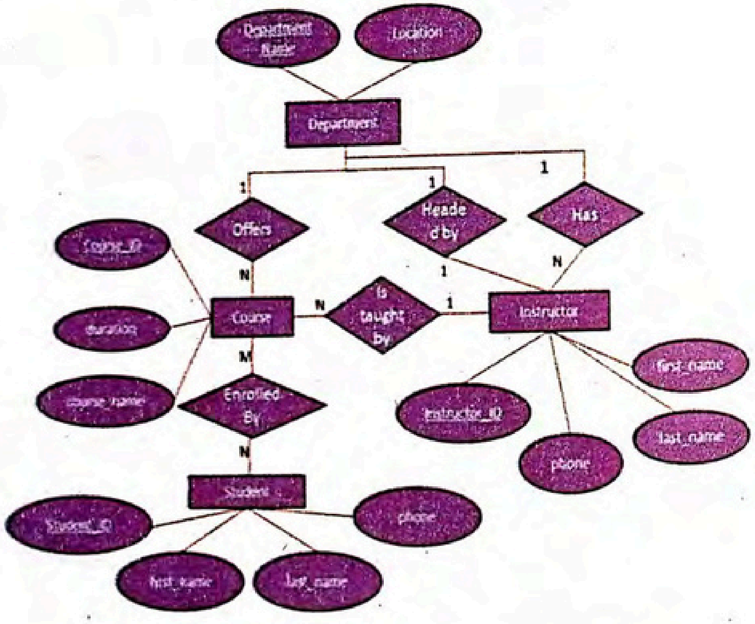


Figure 7.19 E-R Diagram for Student Management System

- The relationship between Student and Course is many-to-many because a student can be enrolled for one or more courses and a Course can be taken by one or many Students.

## E-R Diagram for Ticket Booking System

In the Ticket Booking System shown in Figure 7.17, the following entities and attributes can be identified.

- ***Airline:*** The entity Airline is set of all the airlines and its attributes are Airline_Code and Airline Name.
- ***Passenger:*** The set of all the passengers who want to travel by an airplane. Each passenger has a TicketNo, Name, Address and PhoneNo as its attributes.
- ***Seat:*** The set of all the seats available for reservation to passengers. The seat is described by the attributes SeatNo, Class and Name.
- ***Flight***: The set of all the flights offered by an airline. The flight is described by the attributes FlightNo, DepartureDate, DepartureTime, ArrivalDate, ArrivalTime, From and To.
- The type of relationship between Airline and Flight is one-to-many because one airline has many flights.
- The type of relationship between Passenger and Seat is one to one because each passenger occupies a single seat for travelling in a flight.

- The type of relationship between Flight and Passengér is one to many because each flight has many passengers.
- The relationship between Flight and Seat is one to many because there may be available one or many seats available for reservation in each flight.

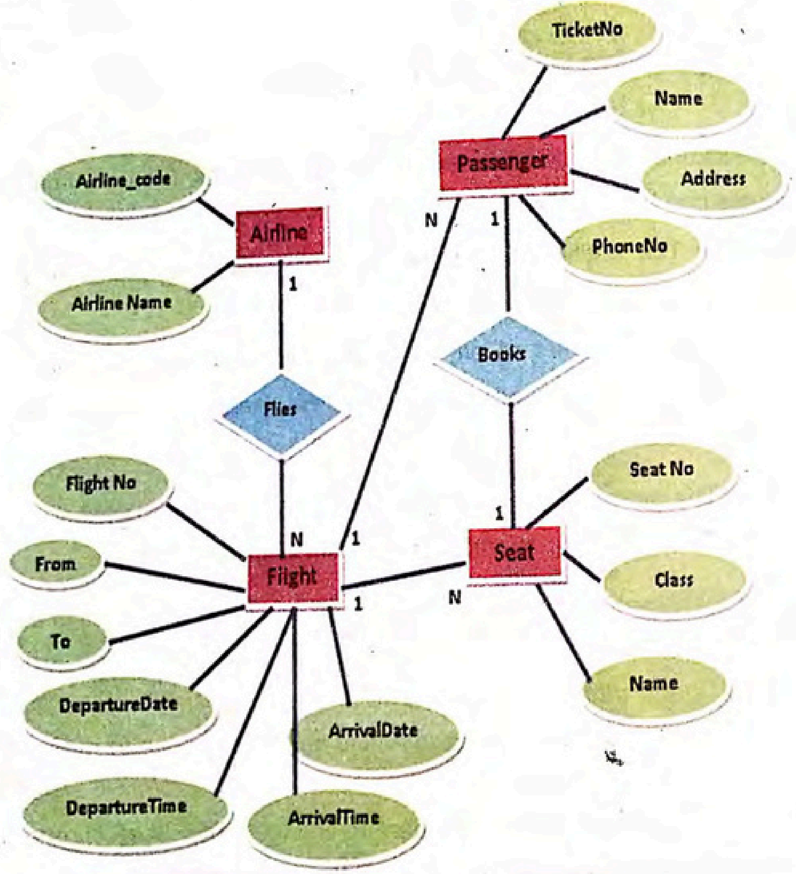


Figure 7.20 E-R Diagram for Ticket Booking System

## 7.5 RELATIONAL SCHEMA

A relational database schema is the tables, columns and relationships that make up a relational database. The schema describes how real world entities are modeled in the database. An ER Model is intended as description of real world entities but it represents the conceptual level of the database design and must be translated into a logical level of database design.

A relational database schema helps users to organize and understand the structure of a database. This is particularly useful when designing a new database, modifying an existing database to support more functionality, or building integration between databases.

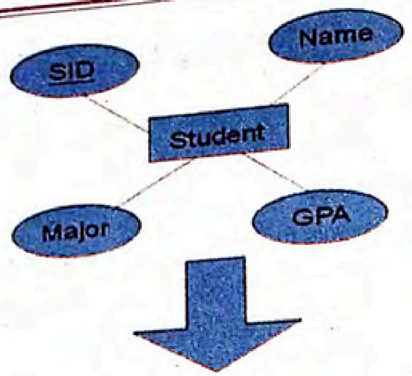
### 7.5.1 Transformation of E-R Model into Relational Schema

#### a. Transforming Entities to Relational Schema

Each entity set is replaced by a table or a relation. Each table has a name. The name used is the entity name. Each table has a number of rows and columns. Each row corresponds to an entity instance. Each column corresponds to an attribute.

#### b. Transforming Attributes to Relational Schema

Each attribute of the entity type is replaced by a column (field) in the table. Suppose we have an Entity Type Student that has attributes, SID, Name, Major and GPA. All the attributes of Student entity will become columns in the Student table. Figure 7.18 shows this transformation.



| SID | Name | Major | GPA |
|---|---|---|---|
| 1234 | Ali | CS | 2.8 |
| 5678 | Asad | EE | 3.6 |

Figure 7.21 Transformation of E-R Model into Relational Schema

### c. Transforming Relationships to Relational Schema

Each relationship in an E-R diagram must also be represented in relational schema. The representation depends upon the nature of relationship. In some cases, a relationship is represented by making the primary key of one relation, a foreign key of another relation. In some cases, a separate relation is created to represent a relationship.

## 7.5.2 Normalization

Normalization is the process of organizing data in relational database in order to minimize duplication of information (data) and to safeguard the database against certain anomalies. The basic purpose of normalization is to divide large table into smaller and well formed tables/relations and remove the inconsistencies or anomalies.

### Normal Form

The degree of normalization is termed as Normal Form. For example, First Normal Form (1st NF), Second Normal Form (2nd NF), Third Normal Form (3rd NF) etc.

### First Normal Form

The relation is considered to be in First Normal Form (1st NF) if the intersection of each row and column contains only one value. In 1st NF we remove the repeating groups. Consider the following example.

**Employee Table**

| Proj_no | Proj_name | Emp_no | Emp_name | Job_class | Charge_per_hour | Hours |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | Jameel Khan | Elect. Engineer | 1000 | 23 |
| | | 101 | Faisal Naeem | DB Designer | 900 | 19 |
| | | 105 | Bilal Ahmad | DB Designer | 900 | 35 |
| 18 | MIS | 114 | Zubair Ahmad | Programmer | 750 | 12.6 |
| | | 118 | M. Waseem | System Analyst | 800 | 45.3 |
| | | 104 | Shaukat Ali | App Designer | 600 | 32.4 |

Figure 7.22

The Employee table contains repeating groups. To normalize the above table, all repeating groups must be eliminated.

**Employee Table**

| Proj_no | Proj_name | Emp_no | Emp_name | Job_class | Charge_per_hour | Hours |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | Jameel Khan | Elect. Engineer | 1000 | 23 |
| 15 | Evergreen | 101 | Faisal Naeem | DB Designer | 900 | 19 |
| 15 | Evergreen | 105 | Bilal Ahmad | DB Designer | 900 | 35 |
| 18 | MIS | 114 | Zubair Ahmad | Programmer | 750 | 12.6 |
| 18 | MIS | 118 | M. Waseem | System Analyst | 800 | 45.3 |
| 18 | MIS | 104 | Shaukat Ali | App Designer | 600 | 32.4 |

Figure 7.23

Attribute Proj_no is not an adequate key and does not uniquely identifies all the records. To maintain a proper primary key, the new key must be composed of a combination of Proj_no and Emp_no.

## 2nd Normal Form

A relation will be in 2nd NF if it is in the first normal form and all non key attributes must be fully functional dependent on the whole primary key (No partial dependencies). Consider the table 7.24

**Employee Table**

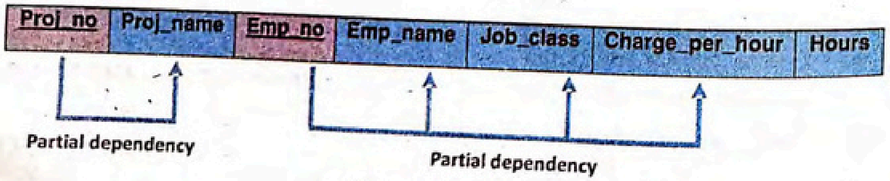


Figure 7.24

Attribute Proj_name is dependent on Proj_no, while Emp_name, Job_class and Charge_per_hour are dependent on Emp_no. Dependencies based on only part of a composite primary key are called partial dependencies.

**Project Table**

| Proj_no | Proj_name |
|---|---|

Figure 7.25

**Employee Table**

| Emp_no | Emp_name | Job_class | Charge_per_hour |
|---|---|---|---|

Figure 7.26

**Hours Table**

| Proj_no | Emp_no | Hours |
|---|---|---|

Figure 7.27

Because the number of hours spent on each project by each employee is dependent on both Proj_no and Emp_no, we place these hours in the Hours table hours as in Figure 7.27.

## 3rd Normal Form

The relation should be in the 1st and 2nd normal forms and all transitive dependencies must be removed. A transitive dependency is a dependency of one non-key attribute on another non-key attribute. Consider Table in Figure 7.28.

**Employee Table**

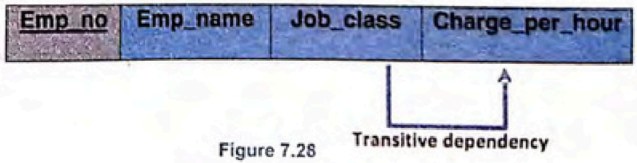


Figure 7.28

Both Job_class and Charge_per_hour are nonprime attributes and the later one is dependent on the former. So by removing transitive dependency the following new tables will be created.

### Employee Table

| Emp_no | Emp_name | Job_class |
|---|---|---|

Figure 7.29

### Job Table

| Job_class | Chg_per_hour |
|---|---|

Figure 7.30

# SUMMARY

- Data is a collection of facts, figures, numbers or ideas that can be organized and processed.
- When facts, figures or numbers (data) are processed and converted into meaningful form that can be used for decision making or any other useful activity, it is called Information.
- File Management system also known as Conventional file system or simply file system is a method of storing and organizing collection of data in the form of files on the secondary storage devices.
- Database is a shared collection of logically related data (and a description of this data i.e. metadata), designed to meet the information needs of multiple users in an organization.
- A database management system (DBMS) is a set of programs that allow users to create a database, edit and update data in database files, store and retrieve data from those database files.
- A database model is a set of rules or specifications which state that how data can be stored, organized and manipulated in a database system.
- Hierarchal Database Model is a type of model in which data is organized into a tree-like structure.
- A network database is similar to a hierarchical database except that each child can have more than one parent record.
- Relational Database Model uses a collection of tables to represent both data and the relationship among those data. Each table has multiple columns and each column has a unique name.
- Object-Oriented database Model is a database model in which information is represented in the form of objects as used in object-oriented programming.
- Object relational database model add new object storage capabilities to the relational systems at the core of modern information systems.
- SQL (Structured Query Language) is the standard language for relational database systems.

- Data definition language (DDL) is a database language that defines the structure in which data are stored.
- Data manipulation language (DML) is á language that enables users to access or manipulate data.
- Data Control Language is a database language used to control access to the data in a database.
- Field/Attribute/Column is a property or characteristic of an entity that is of interest to the organization.
- Record/Tuple/Row is a collection of related fields treated as a single unit.
- Key is a unique identifier which is required to track and analyze data effectively.
- Data Modeling is a logical representation of data in an organization.
- Entity Relationship model is expressed in terms of entities, the relationships (or associations) among those entities, and the attribute (or properties) of both the entities and their relationships in the business environment.
- An entity is a person, place, object, event or concept in user environment about which the organization wishes to maintain data.
- Foreign key is generally a primary key from one table that appears as a field in another where the first table has relationship to the second.
- Secondary key is an attribute or combination of attributes that is not a primary key and can have duplicate data.
- Cardinality refers to the maximum number of times an instance in one entity can be associated with instances in the related entity.
- Modality refers to the minimum number of times an instance in one entity can be associated with an instance in the related entity.
- Relational database schema is the tables, columns and relationships that make up a relational database.
- Normalization is the process of organizing data in relational database in order to minimize duplication of information (data) and to safeguard the database against certain anomalies.

# EXERCISE

## Q1. Select the best choice for the following MCQs.

i. Hierarchical DBMS organizes data elements into ____________.

A. Segments B. Data compartments.

C. Data units. D. Objects

ii. ____________ is a database model in which information is represented in the form of objects.

A. Network Database Model B. Relational Database Model

C. Object-Oriented Database Model D. Hierarchal Database Model

iii. Who is the person who has central control over data and programs in a database system?

A. DBA B. Designer

C. System Analyst D. Programmer

iv. In relational terminology, an attribute is also called____________.

A. a record B. an entity

C. a field D. a table

v. A row in a table is also known as ____________.

A. Column B. Relation

C. Tuple D. Field

vi. ____________ is a set of rules or specifications which state that how data can be stored, organized and manipulated in a database system.

A. Database model B. Database design

C. Database architecture D. Database structure

vii. Which of the following database models uses a collection of tables to represent both data and the relationship among those data?

A. Network Database Model B. Relational Database Model

C. Object-Oriented Database Model D. Hierarchal Database Model

viii. ______________ is a database language that defines the structure in which data are stored.

A. DNL
B. DCL
C. DDL
D. DML

ix. A field, or a combination of fields used to identify particular row in a relation is called______________.

A. Secondary key
B. Foreign key
C. Primary key
D. Alternate key.

x. A combination of two or more columns used to identify particular row in a relation is a ______________.

A. Secondary key
B. Composite key
C. Foreign key
D. Primary key

## Q2. Give short answers to the following questions.

i. Define the terms database and DBMS.

ii. Give three examples of data and information.

iii. Write three disadvantages of file management system.

iv. Give any four advantages of database system.

v. Differentiate between DDL and DML.

vi. What is data model?

vii. Differentiate between cardinality and modality.

viii. What is primary key and foreign key?

ix. What is Relation?

x. What is the role of DBA?

## Q3. Give detailed answers of the following questions.

i. Explain different types of database models.

ii. What is SQL? Explain its types.

iii. Explain various steps of planning a database.

iv. What is a relation? Explain the degree of relationships with examples.

v. What is ERD? Draw an ERD for the following systems.

vi. Hospital management system

vii. Airline booking system

viii. Describe different steps of Transformation of E-R Model into Relational Schema.

ix. What is Normalization? Explain the following Normal Forms 1NF, 2NF, 3NF.