

**18MCA31 – DATABASE MANAGEMENT SYSTEM**

**1. Define: DBMS. List and explain the characteristics of (10) DBMS.**

- A database management system (DBMS) is a collection of programs that enables users to create and maintain a database.
- Defining: involves specifying the data types, structures, and constraints of the data to be stored in the database.
- Constructing: the process of storing the data on some storage medium that is controlled by the DBMS.
- Manipulating: includes functions such as querying the database to retrieve specific data, updating the database to reflect changes in the mini-world, and generating reports from the data
- Sharing: allows multiple users and programs to access the database simultaneously.

**2. List and explain the advantages of using DBMS approach. (10)**

- i. Controlling Redundancy: redundancy in storing the same data multiple times leads to several problems
- ii. Restricting Unauthorized Access: A DBMS should provide a security and authorization subsystem; most users will not be authorized to access all information in the database
- iii. Providing Persistent Storage for Program Objects: Databases can be used to provide persistent storage for program objects and data structures
- iv. Providing Backup and Recovery: responsible for recovery; the database is restored to the state it was in before the transaction started executing

v. Providing Storage Structures and Search Techniques for Efficient Query Processing:

- the DBMS must provide specialized data structures and search techniques to speed up disk search for the desired records – index
- The DBMS often has a buffering or caching module that maintains parts of the database in main memory buffers
- The query processing and optimization module of the DBMS is responsible for choosing an efficient query execution plan for each query based on the existing storage structures.

vi. Providing Multiple User Interfaces:

- a DBMS should provide a variety of user interfaces
- query languages for casual users
- programming language interfaces for application programmers
- forms and command codes for parametric users, and
- menu-driven interfaces and natural language interfaces for standalone users

vii. Representing Complex Relationships among Data: A DBMS must have the capability to represent a variety of complex relationships among the data, to define new relationships as they arise, and to retrieve and update related data easily and efficiently

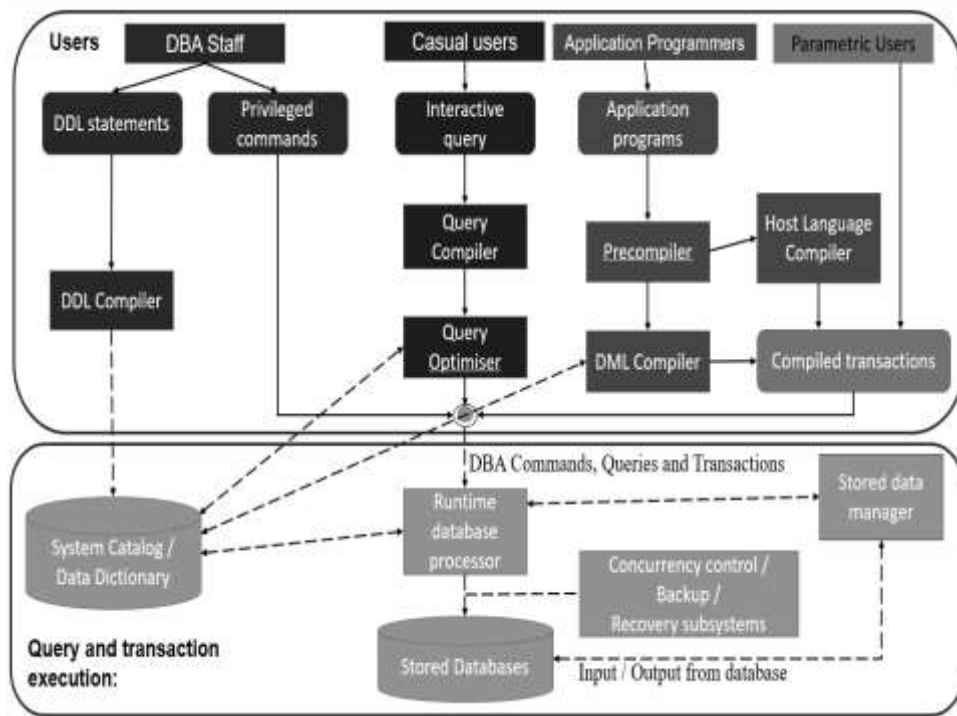
viii. Enforcing Integrity Constraints: The simplest type of integrity constraint involves specifying a data type for each data item

- referential integrity constraint: specifying that a record in one file must be related to records in other files
- Primary key constraint: uniqueness on data item values

ix. Permitting inferencing and Actions Using Rules: A trigger is a form of a rule activated by updates to the table, which results in performing some additional operations to some other tables, sending messages, and so on.

- Stored procedures

**3. With neat diagram, explain the database system (10) environment architecture.**



Users	Component	Functionality
DBA Staff	DDL Statement	used by the DBA for defining the database and tuning it
	DDL Compiler	processes schema definitions and stores descriptions of the schemas in the DBMS catalog
	Privileged commands	The commands which are exclusively used by the DBA
Casual Users	Interactive query	The interface by which the casual users and persons with occasional need for information from the database interact
	Query compiler	Queries are parsed and validated for correctness of the query syntax
	Query optimizer	Responsible for optimising the query and its execution
Application programmers	Pre-compiler	Extracts DML commands from an application program written in a host programming language
	DML compiler	Compilation of pre-compiled DML commands into object code for database access
Parametric Users	Host language compiler	The rest of the program is sent to the host language compiler.
	Compiled transactions	Canned transactions are executed repeatedly by parametric users, who simply supply the parameters to the transactions

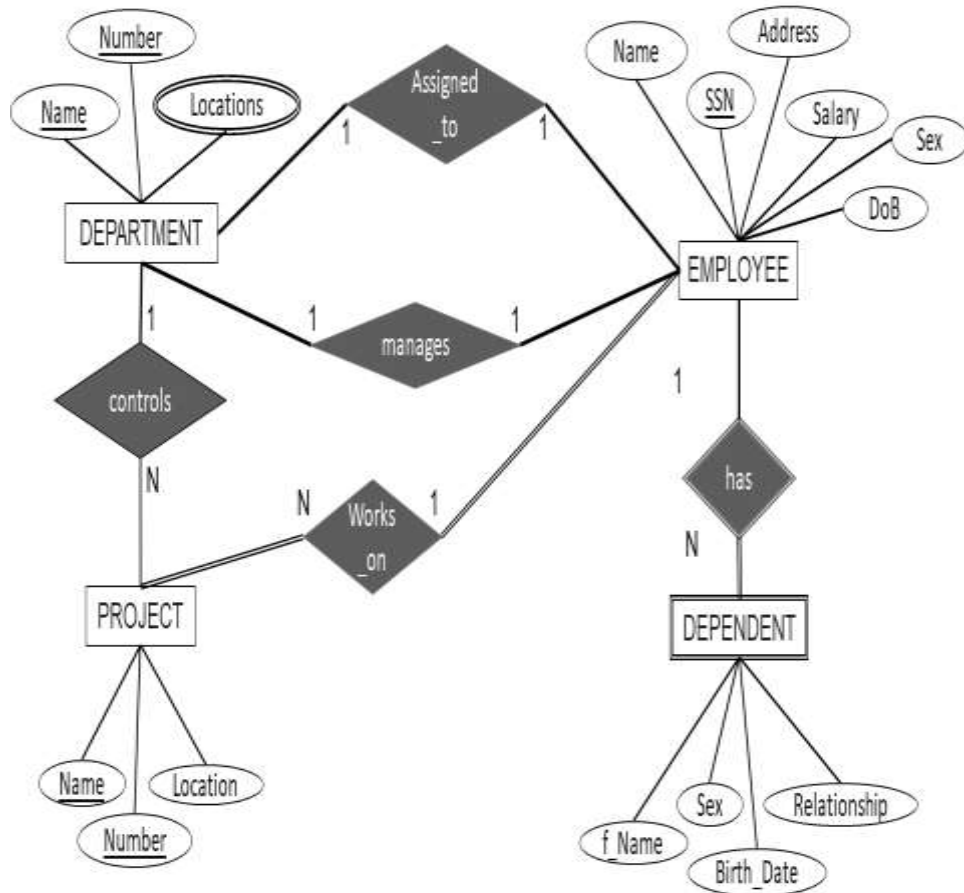
18MCA31 – DATABASE MANAGEMENT SYSTEM

**4. Explain the actors on the scene in DBMS. (10)**

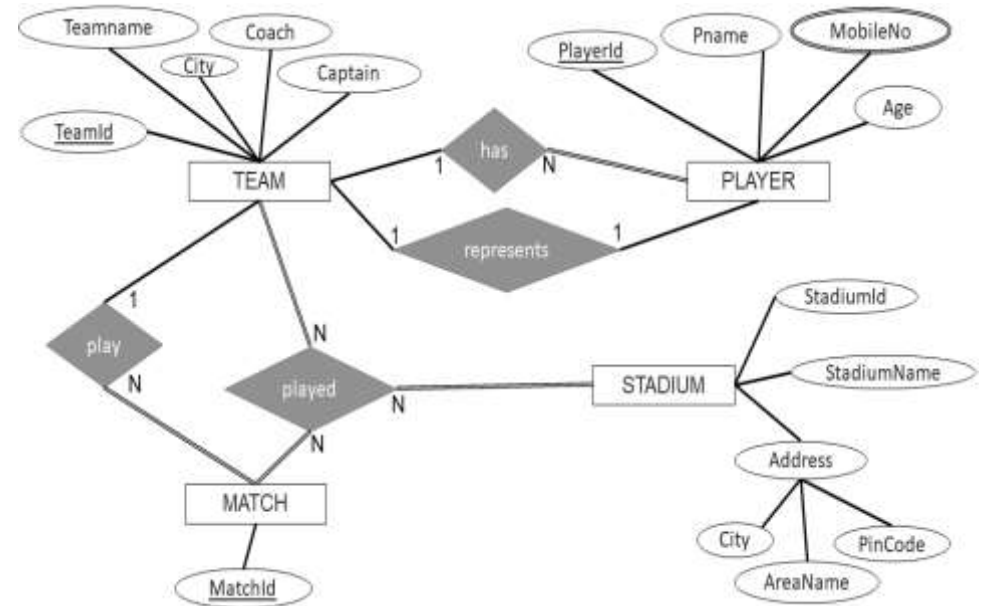
- i. Database Administrators: responsible for authorizing access to the database
- ii. Database Designers: responsible for identifying the data to be stored in the database and for choosing appropriate structures to represent and store this data.
- iii. End Users: the people whose jobs require access to the database for querying, updating, and generating reports
  - several categories of end users:
    - Casual end users: occasionally access the database (middle or higher-level managers)
    - Naive or parametric end users: constantly querying and updating the database (bank tellers, reservation agents)
    - Sophisticated end users: include engineers, scientists, business analysts, and others who thoroughly familiarize themselves with the facilities of the DBMS in order to implement their own applications to meet their complex requirements.
    - Standalone users: maintain personal databases by using ready-made program packages that provide easy-to-use menu-based or graphics-based interfaces.
- iv. System Analysts and Application Programmers
  - System analysts: determine the requirements of end users, especially naive and parametric end users, and develop specifications for standard canned transactions that meet these requirements.
  - Application programmers: implement these specifications as programs; then they test, debug, document, and maintain these canned transactions.

**5. Design an ER diagram for the following scenario: (10)**

The ABC COMPANY PVT LTD., database keeps track of a company's employees, departments, and projects. The company is organized into departments. Each department has a unique name, a unique number, and a particular employee who manages the department. We keep track of the start date when that employee began managing the department. A department may have several locations. A department controls a number of projects, each of which has a unique name, a unique number, and a single location. We store each employee's name, Social Security number, address, salary, sex (gender), and birth date. An employee is assigned to one department, but may work on several projects, which are not necessarily controlled by the same department. We keep track of the current number of hours per week that an employee works on each project. We also keep track of the direct supervisor of each employee (who is another employee). We want to keep track of the dependents of each employee for insurance purposes. We keep each dependent's first name, sex, birth date, and relationship to the employee.



6. In a cricket tournament, there are many teams are contesting (10) each having a Teamid, Team\_Name, City, a coach. Each team is uniquely identified by using Teamid. A team can have many Players and a captain. Each player is uniquely identified by Playerid, having a Name, and multiple phone numbers, age. A player represents only one team. Each stadium is identified using Stadiumid, having a stadium\_name, Address (involves city,area\_name,pincode). A team can play many matches. Each match is identified uniquely by using Matchid.



**7. List and explain the characteristics of relational data model. (10)**

Characteristics	Relational Model
Ordering of tuples in a relation	A relation is not sensitive to the ordering or tuples
Ordering of values within a tuple and an alternative definition of a relation	Ordering of values corresponds to the order of attributes and it is important
Values and NULLs in the tuples	Different meaning: <ul style="list-style-type: none"> <li>• Value unknown;</li> <li>• Value exists but is not available</li> <li>• or attribute does not apply to this tuple (also known as value undefined)</li> </ul>
Interpretation (Meaning) of a relation	Represents facts about both entities and relationships uniformly as relations

- The conditions for a foreign key, specify a referential integrity constraint between the two relation schemas  $R_1$  and  $R_2$
- There are three basic operations that can change the states of relations in the database: Insert, Delete, and Update (or Modify)
- The Insert Operation
  - Insert can violate any of the Four types of constraints / violations:
    - i. Domain constraints can be violated if an attribute value is given that does not appear in the corresponding domain or is not of the appropriate data type
    - ii. Key constraints can be violated if a key value in the new tuple  $t$  already exists in another tuple in the relation  $r(R)$
    - iii. Entity integrity can be violated if any part of the primary key of the new tuple  $t$  is NULL
    - iv. Referential integrity can be violated if the value of any foreign key in  $t$  refers to a tuple that does not exist in the referenced relation
- The Delete Operation
  - can violate only referential integrity
  - occurs if the tuple being deleted is referenced by foreign keys from other tuples in the database
  - To specify deletion, a condition on the attributes of the relation selects the tuple (or tuples) to be deleted
  - Several options are available if a deletion operation causes a violation.
    - i. restrict: reject the deletion.
    - ii. Cascade: to attempt to cascade (or propagate) the deletion by deleting tuples that reference the tuple that is being deleted

**8. Discuss the entity integrity and referential integrity constraints in detail. (10)**

- A relational database schema  $S$  is a set of relation schemas  $S = \{R_1, R_2, \dots, R_m\}$  and a set of integrity constraints (IC)
- The entity integrity constraint states that no primary key (PK) value can be NULL.
- The referential integrity constraint is specified between two relations and is used to maintain the consistency among tuples in the two relations

18MCA31 – DATABASE MANAGEMENT SYSTEM

- 
- iii. set null or set default: modify the referencing attribute values that cause the violation each such value is either set to NULL (except primary key attribute) or changed to reference another default valid tuple
- The update operation:
    - used to change the values of one or more attributes in a tuple (or tuples) of some relation R
    - It is necessary to specify a condition on the attributes of the relation to select the tuple (or tuples) to be modified
-