

Database (DBMS)

* Advantages of DBMS:-

- 1) Control database redundancy :- It can control data redundancy because it stores all the data in one single database file and that recorded data is placed in the database.
- 2) Data sharing :- In DBMS, the authorized users of an organisation can share the data among multiple users.
- 3) Easily maintenance :- It can be easily maintainable due to the centralized nature of the database system.
- 4) Reduce time :- It reduces development recovery subsystems which create automatic backup of data from hardware and software failure to restores the data if required.
- 5.) Multiple User Interface :- It provides diffⁿ types of user interface like graphical user interfaces, application program interfaces.

* Disadvantages of DBMS:-

- 1) Cost of hardware & software :- It requires a high speed of data processor and Large memory size to run DBMS / s/w.
- 2) Size :- It occupies a large space of disks and large memory to run them efficiently.
- 3) Complexity :- Database system creates additional complexity and requirements.
- 4) Higher impact of failure :- Failure is highly impacted the database because in most

of the organization, all the data stored in a single database and if the database is damaged due to electric failure / database corruption then the data may be lost forever.

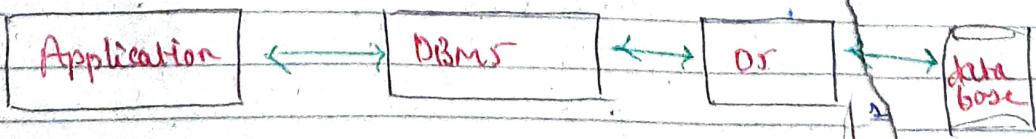
Q2) DB File System and DBMS

Basis	File System	DBMS
Structure	The file system is software that manages & organizes the files in a storage medium within a computer.	DBMS is software for managing the database.
Data Redundancy	Repeated data can be present in a file system.	In DBMS there is no repeated data.
Backup & Recovery	It doesn't provide backup & recovery of data if lost.	It provides backup & recovery if lost.
Query processing	There is no efficient query processing in file system.	Efficient query processing is there.
Consistency	There is less data consistency in file system.	There is more data consistency because of process of normalization.
Complexity	It is less complex as compared to DBMS.	It has more complexity in handling as compared to file.
Security constraints	File systems provide less security in comparison to DBMS.	DBMS has more security mechanisms as compared to file systems.

Cost	It is less expensive than DBMS.	It has a comparatively higher cost than file.
Data independence	There is no data independence.	In DBMS data independence exists.
User access	Only one user can access data at time.	Multiple users can access data at a time.
Data abstraction	It gives details of storage & representation of data.	It hides the internal details of database.
Integrity Constraints	Integrity constraints are difficult to implement.	Integrity constraints are easy to implement.
Example -	Cobol, c++	Oracle, Sql server.

Q.3) What is DBMS ?

- A database management system (DBMS) is a collection of interrelated data and a set of programs to access that data.
- DBMS is used to organize the data in the form of table, schema, view & report etc.
- The primary goal of a DBMS is to provide a way to store and retrieve database information that is both convenient & efficient.
- DBMS is an interface between the application program and the operating system to access & manipulate that database.



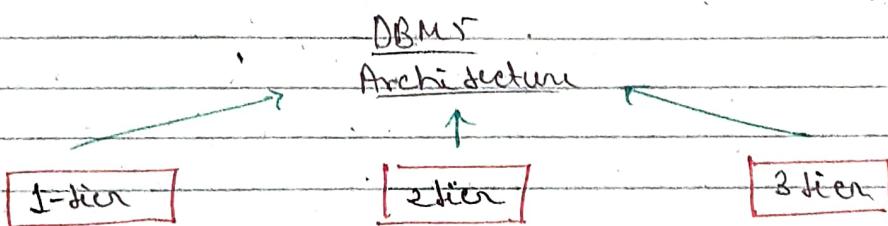
- Database management system is a software which is used to manage the database for eg:- MySQL, Oracle, etc are very popular commercial database.

Q4) Architecture / Application Of DBMS :-

* Application:-

- The DBMS design depends upon its architecture.
- The basic client/server architecture is used to deal with a large no. of PCs, web servers, database servers and other components that are connected with network.
- DBMS architecture depends upon how users are connected to database to get their request done.

Types of DBMS Architecture:-



1-Tier architecture:-

- In this architecture, the database is directly available to the user. It means the user can directly sit on the DBMS and uses it.
- The 1-tier architecture is used for development of the local application, where programmers can directly communicate with the database for the quick response.

2-Tier architecture:-

- The 2-tier architecture is same as basic client-server. In 2-tier applications on the client end can directly communicate with the database at the server side interaction API like - ODBC, JDBC

- The user interface and applications are run on the client side.
- The server side is responsible to provide the functionality query processing & transaction processing

3-Tier architecture:-

- The 3-tier architecture contains another layer between the client & server
- Client can't directly communicate with the server.
- The application on the client-end interact with an application server which further communicates with database systems.
- The 3-tier architecture is used in case of large web application.

Q.5) Data Independence

- The ability to modify schema definition in one level without affecting a schema definition in the next higher level.
- Data independence is one of main advantages in DBMS and it is of two types
 - 1) logical
 - 2) physical

1) Physical data Independence:- physical data independence can be defined as capacity to change internal schema without having a to change the conceptual schema

- If we do any changes in the storage size of the DBMS server, then the conceptual structure of the database will not be affected.
- It is the ability to modify the physical schema without causing application programs to be rewritten.
- Physical data independence is used to separate conceptual levels from the internal levels.
- It occurs at logical interface levels.

2) Logical data Independence:-

- It is the ability to modify the conceptual schema without causing application program to be rewritten.
- Logical data independence refers characteristics of being able to change the conceptual schema without having to change the external schema.
- Logical data independence is used to separate the external level from the conceptual.
- If we do any change in conceptual view of data, then the user view of data would not be affected.
- Logical data independence occurs at user interface level.

Q.6) What is Data dictionary

- A data dictionary contains "metadata"
- A data dictionary is an integral part of a database. It holds the information about the database and the data stored in data dictionary is called metadata.
- Metadata is data that provides information about other data, but not the content of data. It helps to

organize, find & understand data.

- Data dictionary contains information such as what is the database, who is allowed to access it, where is the database physically stored etc.
- Data dictionary stores details like:-
 - Name of the table
 - Name of fields of tables
 - No. of rows of tables
 - Data types, field size, etc.
 - Constraints like primary key, FK, Not null, etc
 - Access privileges on tables.

e.g:-

Employee

Emp-ID	Name	DOB	Phone no.	Salary
1	Ram	10/19/95	88888888	100000
2	Raju	10/15/92	99999999	40000

- 1) Data dictionary does not store actual data of database.
It is used to store the real data description of data.
- 2) It gives well structured & clear information about the database.
- 3) Normal users do not interact with the data dictionary.
It is handled by (DBA) database administrator.
- 4) It is also called "Data repository" or "System catalog".
- 5) Data dictionary is updated when we perform DDL statements like - create, alter, drop, delete.

Active Data dictionary

Passive data dictionary

- Active data dictionary is created & managed by DBMS → It is created manually or any dedicated software
- Automatically updated in database structure whenever DDL statement is performed → It is not managed by DBMS, so if the database is modified the data dictionary is not automatically updated
- integrated data dictionary → This needs careful handling.

Q.)

E-R Model

Entity relationship model

The entity-relationship (E-R) data model is based on concept of real world that consist of a collection of basic objects called entities. and relationship among these objects

* Entity :- An entity in an ER model is a real world object having properties called attributes.

Entities are described by a set of attributes.

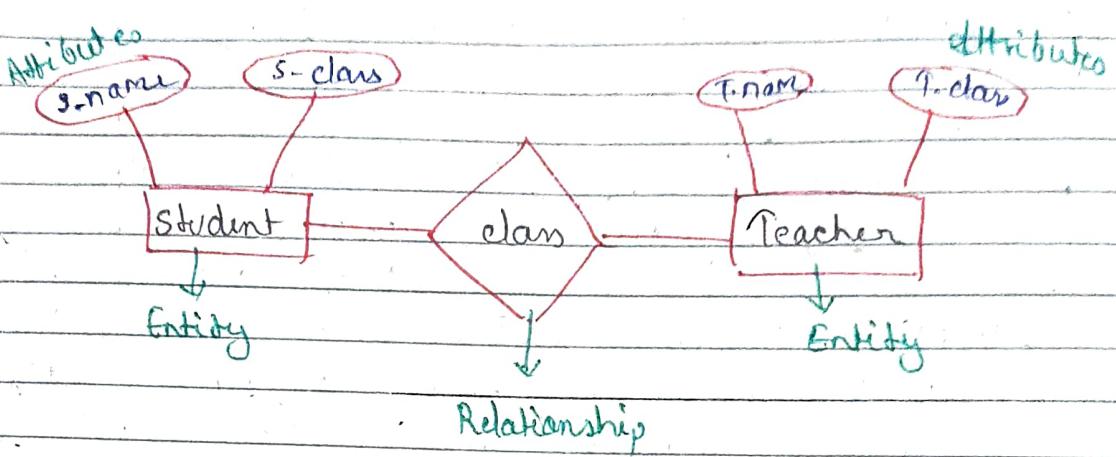
* Attributes - Every attribute is defined by its set of values called domains

e.g. - In a school database, a student is considered as an entity

→ student has various attributes like Name, age

* Relationship :- a logical association among several entities

e.g. we add 'Teacher' entity with attribute 'Teach T-class'. So student & teacher entity can be related with relation class.



* Entity Sets :- An entity is an object of real world so set of entities of the same type
 e.g. - a person is an enterprise is an entity.

(a) Define :-

- 1.) Schema :- Schema is the logical representation of the database. Schema is of two types :-
 • physical schema - describes the database design at physical level.
 • logical schema - describes the database design at logical level.
- 2.) Instance :- At any instance of the content of the database is called the instance of database.