**Syllabus:**

**Introduction to Database Management System:**

**An overview of database management system,**

**Database System Vs. File System,**

**Database system concepts and architecture,**

**data models, schema and instances,**

**data independence and data base language and interfaces, Data definitions language, DML, Overall Database Structure.**

* **An overview of database management system**
* **Data** is a collection of a distinct small unit of information. It can be used in a variety of forms like text, numbers, media, bytes, etc. it can be stored in pieces of paper or electronic memory, etc.
* Word 'Data' is originated from the word 'datum' that means 'single piece of information.' It is plural of the word datum. In computing, Data is information that can be translated into a form for efficient movement and processing. Data is interchangeable.
* The **database** is a collection of inter-related data which is used to retrieve, insert and delete the data efficiently. It is also used to organize the data in the form of a table, schema, views, and reports, etc. For example: The college Database organizes the data about the admin, staff, students and faculty etc. Using the database, you can easily retrieve, insert, and delete the information.
* Example of database are: MySQL, Sybase, Oracle, MongoDB, Informix, Postgre SQL, SQL Server, etc.
* **Database Management System**
* Dbms is a software which is used to manage the database. For example: [MySQL](https://www.javatpoint.com/mysql-tutorial), [Oracle](https://www.javatpoint.com/oracle-tutorial), etc. are a very popular commercial database which is used in different applications.
* DBMS provides an interface to perform various operations like database creation, storing data in it, updating data, creating a table in the database and a lot more.
* It provides protection and security to the database. In the case of multiple users, it also maintains data consistency.
* DBMS allows users the following tasks:
* Data Definition: It is used for creation, modification, and removal of definition that defines the organization of data in the database.
* Data Updation: It is used for the insertion, modification, and deletion of the actual data in the database.
* Data Retrieval: It is used to retrieve the data from the database which can be used by applications for various purposes.
* User Administration: It is used for registering and monitoring users, maintain data integrity, enforcing data security, dealing with concurrency control, monitoring performance and recovering information corrupted by unexpected failure.
* Characteristics of DBMS
* It uses a digital repository established on a server to store and manage the information.
* It can provide a clear and logical view of the process that manipulates data.
* DBMS contains automatic backup and recovery procedures.
* It contains ACID properties which maintain data in a healthy state in case of failure.
* It can reduce the complex relationship between data.
* It is used to support manipulation and processing of data.
* It is used to provide security of data.
* It can view the database from different viewpoints according to the requirements of the user.
* Advantages of DBMS
* Controls 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.
* Data sharing: In DBMS, the authorized users of an organization can share the data among multiple users.
* Easily Maintenance: It can be easily maintainable due to the centralized nature of the database system.
* Reduce time: It reduces development time and maintenance need.
* Backup: It provides backup and recovery subsystems which create automatic backup of data from [hardware](https://www.javatpoint.com/hardware) and [software](https://www.javatpoint.com/software) failures and restores the data if required.
* multiple user interface: It provides different types of user interfaces like graphical user interfaces, application program interfaces
* Disadvantages of DBMS
* Cost of Hardware and Software: It requires a high speed of data processor and large memory size to run DBMS software.
* Size: It occupies a large space of disks and large memory to run them efficiently.
* Complexity: Database system creates additional complexity and requirements.
* 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 or database corruption then the data may be lost forever.

# **Application of DBMS:**

### **Railway Reservation System**

### **🡺**In the rail route reservation framework, the information base is needed to store the record or information of ticket appointments, status of train’s appearance, and flight.

### 🡺Additionally, if trains get late, individuals become acquainted with it through the information base update.

### **Library Management System**

🡺There are many books in the library so; it is difficult to store the record of the relative multitude of books in a register or duplicate. Along these lines, the data set administration framework (DBMS) is utilized to keep up all the data identified with the name of the book, issue date, accessibility of the book, and its writer.

### **Banking**

🡺Database the executive’s framework is utilized to store the exchange data of the client in the information base.

### **Education Sector**

🡺Presently, assessments are led online by numerous schools and colleges. They deal with all assessment information through the data set administration framework (DBMS). In spite of that understudy’s enlistments subtleties, grades, courses, expense, participation, results, and so forth all the data is put away in the information base.

### **Credit card exchanges**

🡺The database Management framework is utilized for buying on charge cards and age of month to month proclamations.

### **Social Media Sites**

🡺We all utilization of online media sites to associate with companions and to impart our perspectives to the world. Every day, many people group pursue these online media accounts like Pinterest, Facebook, Twitter, and Google in addition to. By the utilization of the data set administration framework, all the data of clients are put away in the information base and, we become ready to interface with others.

### **Broadcast communications**

🡺Without DBMS any media transmission organization can’t think. The Database the executive’s framework is fundamental for these organizations to store the call subtleties and month to month postpaid bills in the information base.

### **Accounting and Finance**

🡺The information base administration framework is utilized for putting away data about deals, holding and acquisition of monetary instruments, for example, stocks and bonds in a data set.

### **E-Commerce Websites**

🡺These days, web-based shopping has become a major pattern. Nobody needs to visit the shop and burn through their time. Everybody needs to shop through web based shopping sites, (for example, Amazon, Flipkart, Snapdeal) from home. So all the items are sold and added uniquely with the assistance of the information base administration framework (DBMS). Receipt charges, installments, buy data these are finished with the assistance of DBMS.

### **Human Resource Management**

🡺Big firms or organizations have numerous specialists or representatives working under them. They store data about worker’s compensation, assessment, and work with the assistance of an information base administration framework (DBMS).

### **Manufacturing**

🡺Manufacturing organizations make various kinds of items and deal them consistently. To keep the data about their items like bills, acquisition of the item, amount, inventory network the executives, information base administration framework (DBMS) is utilized.

### **Airline Reservation System**

🡺This framework is equivalent to the railroad reservation framework. This framework additionally utilizes an information base administration framework to store the records of flight takeoff, appearance, and defer status.

# **Types of Databases:**



1. **Centralized Database**

* It is the type of database that stores data at a centralized database system. It comforts the users to access the stored data from different locations through several applications. These applications contain the authentication process to let users access data securely. An example of a Centralized database can be Central Library that carries a central database of each library in a college/university.

### Advantages of Centralized Database

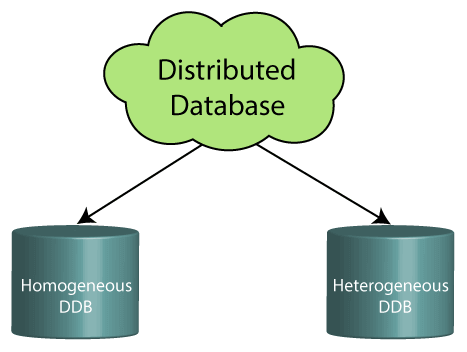
* It has decreased the risk of data management, i.e., manipulation of data will not affect the core data.
* Data consistency is maintained as it manages data in a central repository.
* It provides better data quality, which enables organizations to establish data standards.
* It is less costly because fewer vendors are required to handle the data sets.

### Disadvantages of Centralized Database

* The size of the centralized database is large, which increases the response time for fetching the data.
* It is not easy to update such an extensive database system.
* If any server failure occurs, entire data will be lost, which could be a huge loss.

## Distributed Database

* In distributed systems, data is distributed among different database systems of an organization. These database systems are connected via communication links. Such links help the end-users to access the data easily. **Examples** of the Distributed database are Apache Cassandra, HBase, Ignite, etc.



* **Homogeneous DDB:** Those database systems which execute on the same operating system and use the same application process and carry the same hardware devices.
* **Heterogeneous DDB:** Those database systems which execute on different operating systems under different application procedures, and carries different hardware devices.

### Advantages of Distributed Database

* Modular development is possible in a distributed database, i.e., the system can be expanded by including new computers and connecting them to the distributed system.
* One server failure will not affect the entire data set.

1. **Relational Database**

* This database is based on the relational data model, which stores data in the form of rows(tuple) and columns(attributes), and together forms a table(relation).
* A relational database uses SQL for storing, manipulating, as well as maintaining the data. E.F. Codd invented the database in 1970. Each table in the database carries a key that makes the data unique from others. **Examples** of Relational databases are MySQL, Microsoft SQL Server, Oracle, etc.

### Properties of Relational Database

**A means Atomicity:** This ensures the data operation will complete either with success or with failure. It follows the 'all or nothing' strategy. For example, a transaction will either be committed or will abort.

**C means Consistency:** If we perform any operation over the data, its value before and after the operation should be preserved. For example, the account balance before and after the transaction should be correct, i.e., it should remain conserved.

**I means Isolation:** There can be concurrent users for accessing data at the same time from the database. Thus, isolation between the data should remain isolated. For example, when multiple transactions occur at the same time, one transaction effects should not be visible to the other transactions in the database.

**D means Durability:** It ensures that once it completes the operation and commits the data, data changes should remain permanent.

## NoSQL Database

* Non-SQL/Not Only SQL is a type of database that is used for storing a wide range of data sets.
* It is not a relational database as it stores data not only in tabular form but in several different ways. It came into existence when the demand for building modern applications increased. Thus, NoSQL presented a wide variety of database technologies in response to the demands.

### Advantages of NoSQL Database

* It enables good productivity in the application development as it is not required to store data in a structured format.
* It is a better option for managing and handling large data sets.
* It provides high scalability.
* Users can quickly access data from the database through key-value.

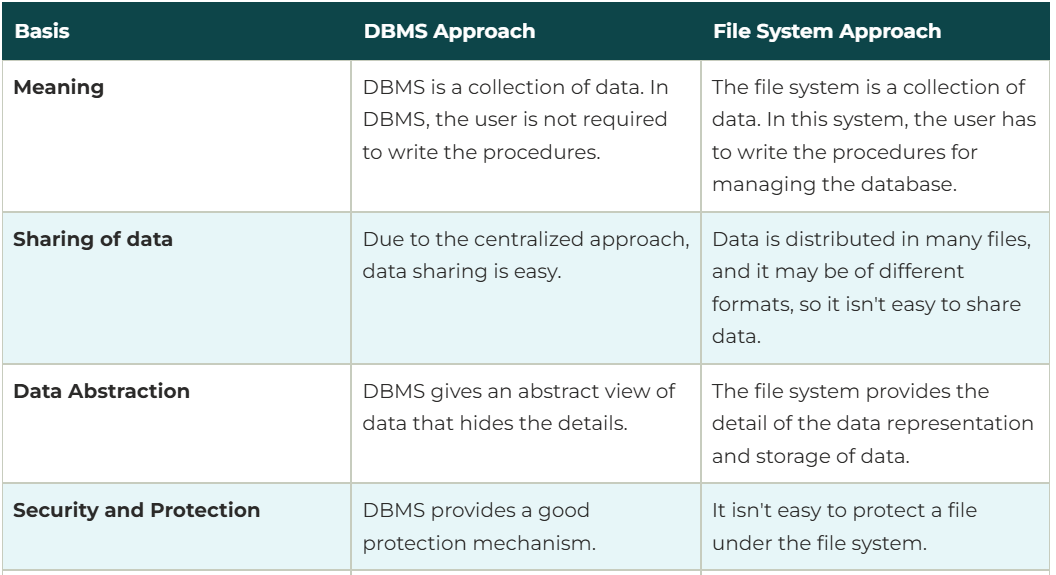
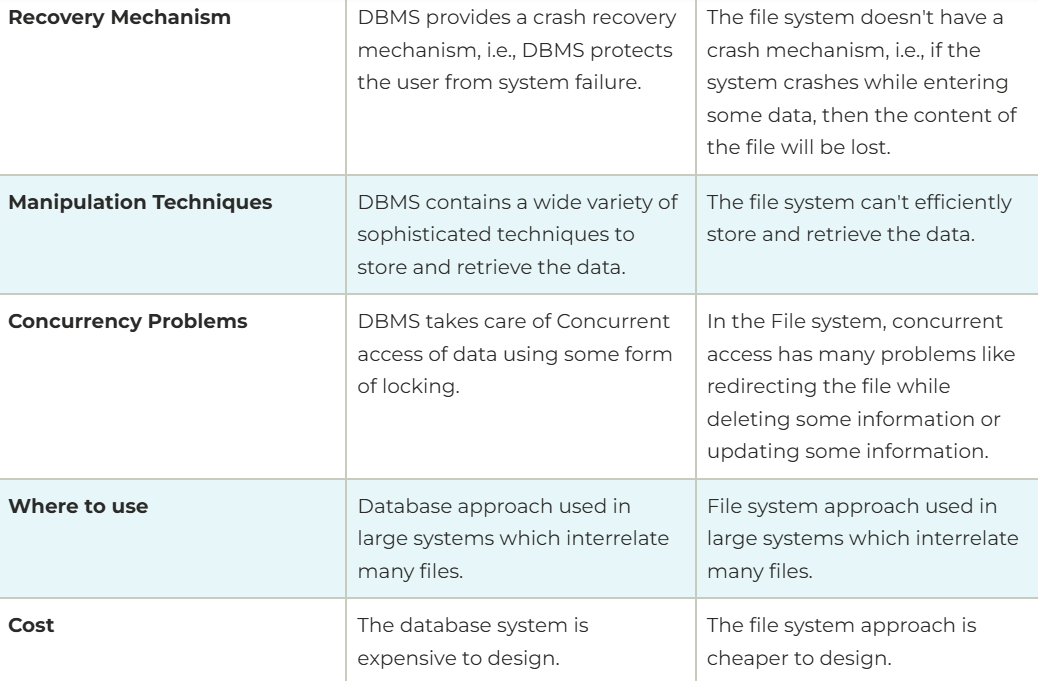
## Cloud Database

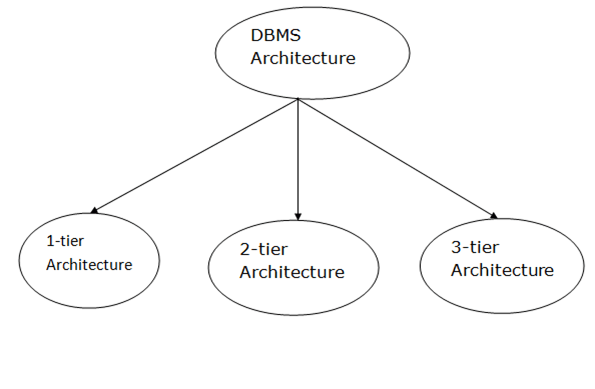
* A type of database where data is stored in a virtual environment and executes over the cloud computing platform. It provides users with various cloud computing services (SaaS, PaaS, IaaS, etc.) for accessing the database.

## Personal Database

* Collecting and storing data on the user's system defines a Personal Database. This database is basically designed for a single user.

### Advantage of Personal Database

* It is simple and easy to handle.
* It occupies less storage space as it is small in size.
* **Database System Vs. File System**
* ****
* ****
* **Architecture of dbms**
* The DBMS design depends upon its architecture. The basic client/server architecture is used to deal with a large number of PCs, web servers, database servers and other components that are connected with networks.
* The client/server architecture consists of many PCs and a workstation which are connected via the network.
* DBMS architecture depends upon how users are connected to the database to get their request done.
* Types of DBMS Architecture

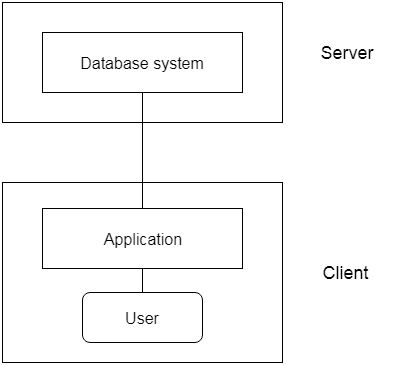


### **1-Tier Architecture**

* In this architecture, the database is directly available to the user. It means the user can directly sit on the DBMS and uses it.
* Any changes done here will directly be done on the database itself. It doesn't provide a handy tool for end users.
* 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**

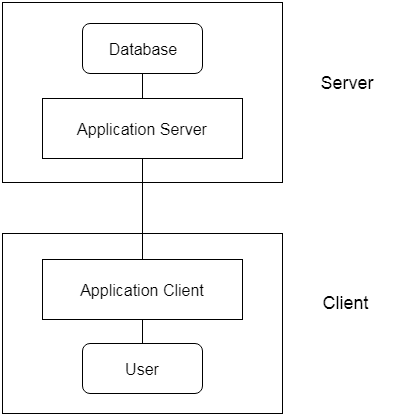
* The 2-Tier architecture is same as basic client-server. In the two-tier architecture, applications on the client end can directly communicate with the database at the server side. For this interaction, API's like: **ODBC**, **JDBC** are used.
* The user interfaces and application programs are run on the client-side.
* The server side is responsible to provide the functionalities like: query processing and transaction management.
* To communicate with the DBMS, client-side application establishes a connection with the server side.



**Fig: 2-tier Architecture**

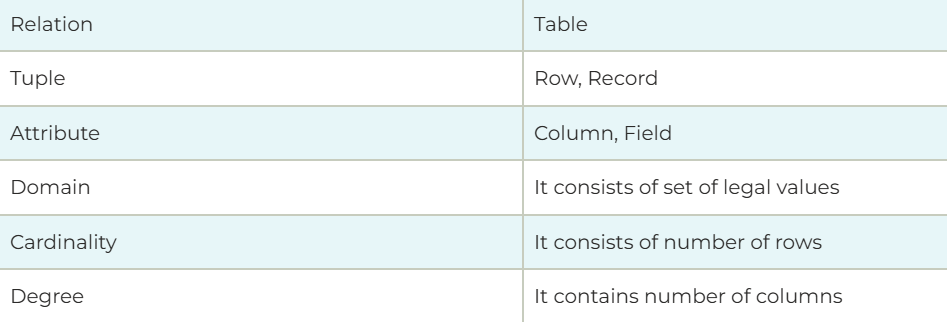
### **3-Tier Architecture**

* The 3-Tier architecture contains another layer between the client and server. In this architecture, client can't directly communicate with the server.
* The application on the client-end interacts with an application server which further communicates with the database system.
* End user has no idea about the existence of the database beyond the application server. The database also has no idea about any other user beyond the application.
* The 3-Tier architecture is used in case of large web application.

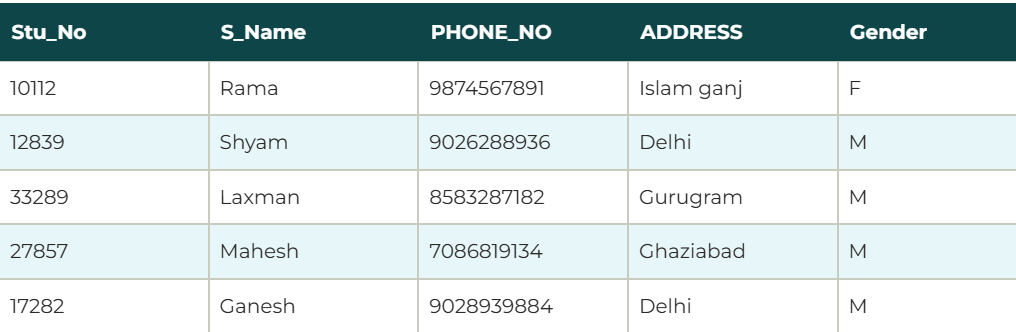


**Fig: 3-tier Architecture**

* **data models**
* Data Model is the modelling of the data description, data semantics, and consistency constraints of the data. It provides the conceptual tools for describing the design of a database at each level of data abstraction.
* Types of data models are-
  1. **Relational Data Model:**
* This type of model designs the data in the form of rows and columns within a table. Thus, a relational model uses tables for representing data and in-between relationships. Tables are also called relations. This model was initially described by Edgar F. Codd, in 1969. The relational data model is the widely used model which is primarily used by commercial data processing applications.



* **Example: STUDENT Relation**



## Properties of Relations

* Each attribute in a relation has only one data value corresponding to it i.e. they do not contain two or more values.
* Name of the relation is distinct from all other relations.
* Each attribute contains a distinct name Attribute domain has no significance
* tuple has no duplicate value Order of tuple can have a different sequence
* It also provides information about metadata.

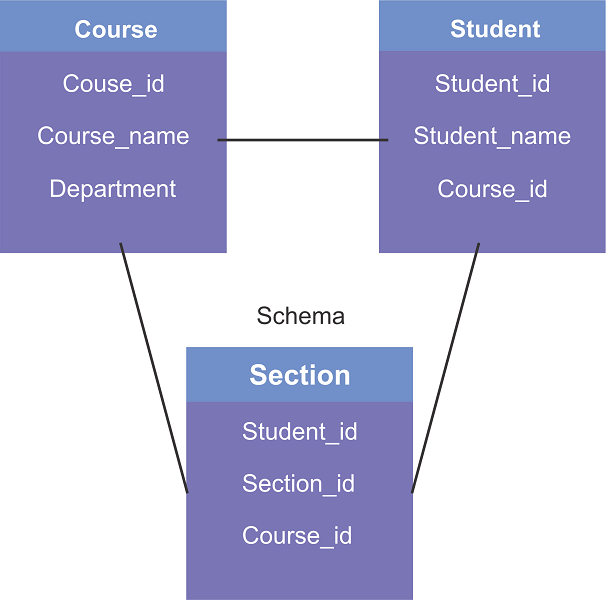
### **Merits of Relational Model:**

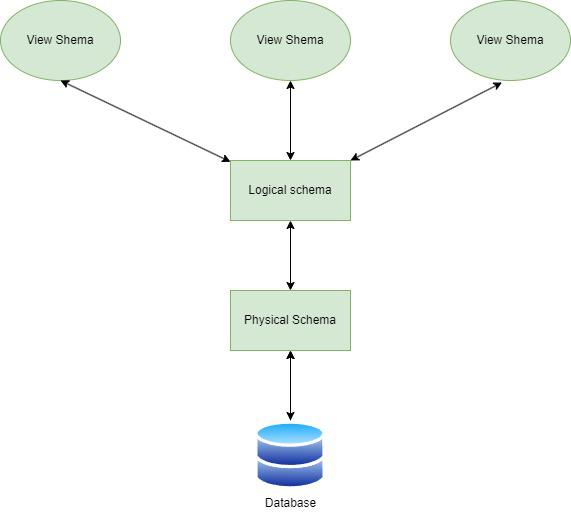
* This provides an abstract view of the data. It abstracts the physical structure from the logical structure of data.
* This model is very easy to design. Tables can use different attributes as per requirements.
* Relational database helps the user to use a query language to query the database.
* It offers more flexibility than other models.
* It is useful for representing most real world objects and the relationships between them. It is very easy to implement a relationship through the use of a composite key, so this model persistence method dominates the market.

### **Demerits of Relational Model:**

* The main disadvantage of relational models is that they do not support binary data **for example:** images, documents, spreadsheets etc.
* The relational model can easily adapt to new hardware so incurs large hardware overhead.
* Relational databases use a simple mapping of logical tables to physical structures.
  1. **Entity-Relationship Data Model:**
* An ER model is the logical representation of data as objects and relationships among them. These objects are known as entities, and relationship is an association among these entities. This model was designed by Peter Chen and published in 1976 papers. It was widely used in database designing. A set of attributes describe the entities. For example, student\_name, student\_id describes the 'student' entity. A set of the same type of entities is known as an 'Entity set', and the set of the same type of relationships is known as 'relationship set'.
  1. **Object-based Data Model:**
* An extension of the ER model with notions of functions, encapsulation, and object identity, as well. This model supports a rich type system that includes structured and collection types. Thus, in 1980s, various database systems following the object-oriented approach were developed. Here, the objects are nothing but the data carrying its properties.
  1. **Semistructured Data Model:**
* This type of data model is different from the other three data models (explained above). The semistructured data model allows the data specifications at places where the individual data items of the same type may have different attributes sets. The Extensible Markup Language, also known as XML, is widely used for representing the semistructured data. Although XML was initially designed for including the markup information to the text document, it gains importance because of its application in the exchange of data.
* **data schema and instances**
* **database schema-**
* DBMS schema means designing the database. For example, if we take the example of the employee table. The employee table contains the following attributes. These attributes are EMP\_ID, EMP\_ADDRESS, EMP\_NAME, EMP\_CONTACT. These are the schema of the employee table.
* The schema defines the logical view of the database. It provides some knowledge about the database and what data needs to go where.
* Schema is further divided into three types. These three are as follows.

1. Logical schema.
2. View schema.
3. Physical schema.





### **1. Physical schema:**

In the physical schema, the database is designed at the physical level. At this level, the schema describes how the data block is stored and how the storage is managed.

### **2. Logical schema:**

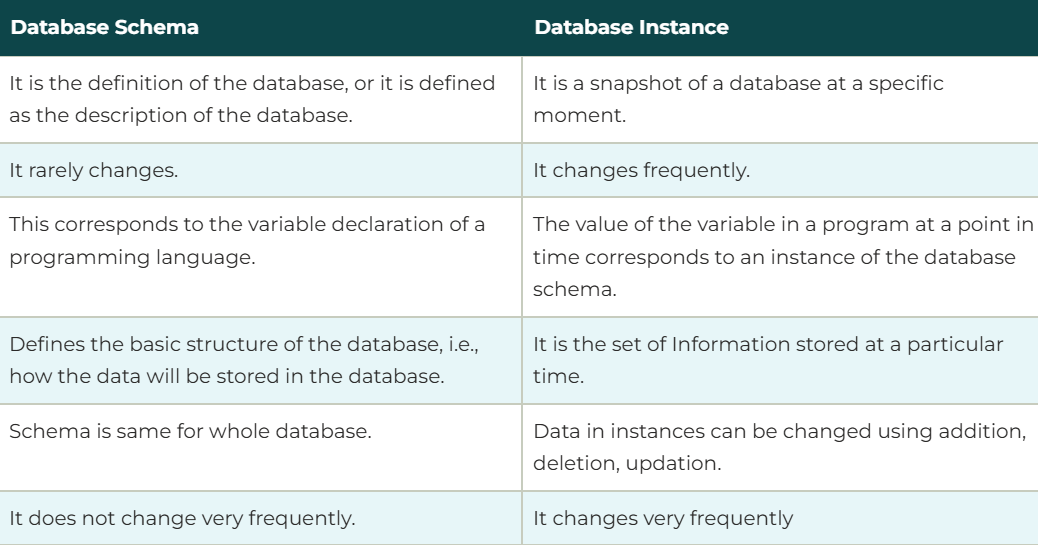
In the logical schema, the database is designed at a logical level. At this level, the programmer and data administrator perform their work. Also, at this level, a certain amount of data is stored in a structured way. But the internal implementation data are hidden in the physical layer for the security proposed.

### **3. View schema:**

In view schema, the database is designed at the view level. This schema describes the user interaction with the database system.

Moreover, Data Definition Language (DDL) statements help to denote the schema of a database. The schema represents the name of the table, the name of attributes, and their types; constraints of the tables are related to the schema. Therefore, if users want to modify the schema, they can write DDL statements.

## Advantages of Database Schema

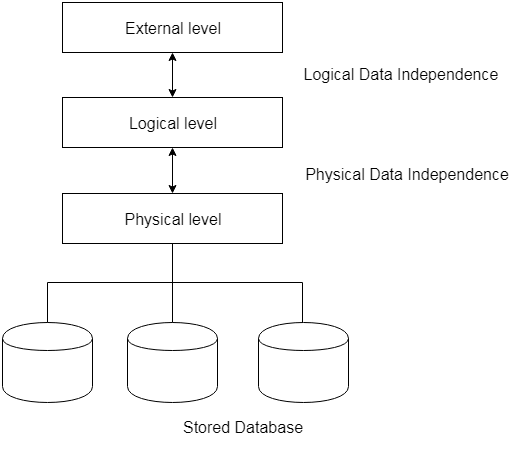
* **Providing Consistency of data:**[Database schema](https://www.geeksforgeeks.org/database-schemas/) ensures the data consistency and prevents the duplicates.
* **Maintaining Scalability:** Well designed database schema helps in maintaining addition of new tables in database along with that it helps in handling large amounts of data in growing tables.
* **Performance Improvement:** Database schema helps in faster data retrieval which is able to reduce operation time on the database tables.
* **Easy Maintenance:** Database schema helps in maintaining the entire database without affecting the rest of the database
* **Security of Data:** Database schema helps in storing the sensitive data and allows only authorized access to the database.
* **DBMS Instance-**
* In DBMS, the data is stored for a particular amount of time and is called an instance of the database. The database schema defines the attributes of the database in the particular DBMS. The value of the particular attribute at a particular moment in time is known as an instance of the DBMS.
* **example:** Let's say we have a single table student in the database; today, the table has 100 records, so today, the instance of the database has 100 records. We are going to add another 100 records to this table by tomorrow, so the instance of the database tomorrow will have 200 records in the table. In short, at a particular moment, the data stored in the database is called the instance; this change over time as and when we add, delete or update data in the database.
* 
* **data independence**
* Data independence can be explained using the three-schema architecture.
* Data independence refers characteristic of being able to modify the schema at one level of the database system without altering the schema at the next higher level.
* There are two types of data independence:

1. Logical Data Independence

* Logical data independence refers characteristic 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 view.
* If we do any changes in the conceptual view of the data, then the user view of the data would not be affected.
* Logical data independence occurs at the user interface level.

2. Physical Data Independence

* Physical data independence can be defined as the capacity to change the internal schema without having to change the conceptual schema.
* If we do any changes in the storage size of the database system server, then the Conceptual structure of the database will not be affected.
* Physical data independence is used to separate conceptual levels from the internal levels.
* Physical data independence occurs at the logical interface level.



## **Advantages of Data Independence**

1. **Application Portability:** Data independence enables deploying applications across different database systems without rewriting code. It promotes reuse across various environments and cuts down time and costs for migration.
2. **Enhanced Flexibility:** Data independence provides the flexibility to change database schemas and physical storage transparently. In addition, it helps with easy adaptability to new requirements and accommodates future growth and changes.
3. **Improved Performance:** Data independence enables physical optimizations, such as indexing, to enhance performance without making changes to applications. Tuning can be carried out seamlessly without disrupting applications, helping achieve scalability targets.
4. **Simplified Maintenance:** Separating physical and logical aspects simplifies database maintenance, minimizing unexpected issues and lowering the need for extensive regression testing.
5. **Increased Security:**Hiding physical details through logical abstraction layers enhances security against attacks. It lessens vulnerabilities related to inference, providing more robust prevention against unauthorized access.
6. **Reduced Coding Time:** Application developers only deal with logical views rather than physical database internals. This abstraction simplifies things for developers, leading to a quicker time to market for enhancements.

* **Disadvantages of Data Independence**

1. **Complexity Overhead:** Additional mapping layers between the physical, logical, and external schemas add complexity. There are more abstraction relationships and translations to manage.
2. **Processing Overhead:** Transforming requests and moving data between underlying physical structures and higher-level logical views requires additional processing. This can lead to a potential performance lag.
3. **Data Duplication:** Physical optimization techniques like denormalization and caching that improve performance can result in data redundancy across mapping schemas. This needs to be managed proactively.
4. **Constraint Management:**Complex referential and relational integrity constraints may need to be configured and enforced separately at logical and physical levels. Constraint mapping can be error-prone.
5. **Limitations in Practice:**Full physical and logical independence is difficult to achieve for complex database applications accessing low-level structures directly for performance gains. Changes often impact multiple levels of reality.

## **Real-World Applications of Data Independence**

1. **Enterprise Resource Planning (ERP) Systems**
2. Enterprise Resource Planning (ERP) systems integrate various business processes and functions into a centralized database. Data independence plays a crucial role in ERP systems by allowing businesses to modify the underlying database schema to accommodate changes in organizational structure or business requirements.
3. **Customer Relationship Management (CRM) Systems**

* CRM systems store and manage customer-related data, including customer profiles, interactions, and sales information. Data independence enables CRM systems to evolve and scale with a growing customer base. For example, if additional customer attributes need to be captured, such as social media profiles or purchase histories, logical data independence allows for the expansion of the CRM database without disrupting the existing functionality.

1. **Data Warehousing**

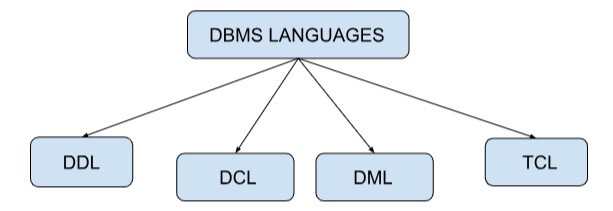
* Data warehousing involves aggregating data from various sources into a central repository for analysis and reporting. Data independence ensures that changes made to the source systems do not impact the reporting and analytical capabilities of the data warehouse.

1. **E-commerce Platforms**

* E-commerce platforms rely heavily on databases to store product catalogs, customer information, and transaction data. Data independence is crucial in this context, as it allows for seamless updates and modifications to the database schema as new products are added or business rules change.
* **data base language and interfaces**

## Database Languages

* Database languages are used to read, store and update the data in the database. Specific languages are used to perform various operations of the database.
* Types of Database Languages are-



### **DDL(Data Definition Language):**

* Data Definition Language(DDL) is used for describing structures or patterns and its relationship in a database.
* It is also used to define the database schema, tables, index, Constraints, etc. It can also be used to store information like the number of tables, names, columns, indexes, etc. The commands only affect the database structure and not the data.
* Advantages are:

1. DDL enables the systematic design and modification of database schema.
2. It ensures uniformity in database structure definitions.
3. With proper precautions, DDL allows users to change the database structure without affecting the data within.

* Disadvantages are:

1. Some DDL operations, such as DROP, can't be rolled back and may lead to data loss.
2. Executing certain DDL statements, especially on large datasets or complex structures, can take a lot of time.
3. Altering a table structure can lead to errors if other objects or applications in the database depend on it.

* The commands used in DDL are:

Create: It is used to create a database or table.

Alter: It is used to make a change in the structure of a database.

Drop: It is used to completely delete a table from the database

Rename: It is used to rename a table.

Truncate: It is used to delete the entities inside the table while holding the structure of the table.

Comment: It is used to comment on the data dictionary.

CREATE TABLE Students (column1 INT,column2 VARCHAR(50),column3 INT);

ALTER TABLE Students ADD column\_name;

DROP Table Table\_name;

TRUNCATE TABLE table\_name;

ALTER TABLE Old\_Table\_Name RENAME TO New\_Table\_Name;

### **DML(Data Manipulation Language):**

* DML is used to manipulate the data present in the table or database. We can easily perform operations such as store, modify, update, and delete on the database.
* Advantages are:
  1. DML statements can modify the data stored in a database.
  2. User can specify what data is needed.
  3. DML tends to have many different flavours and capabilities between database vendors.
  4. It provides efficient human interaction with the system.
* Disadvantages are:

1. DML cannot be used to change the database structure.
2. Tables or columns cannot be created or deleted using dml.

* The commands used in DML are:

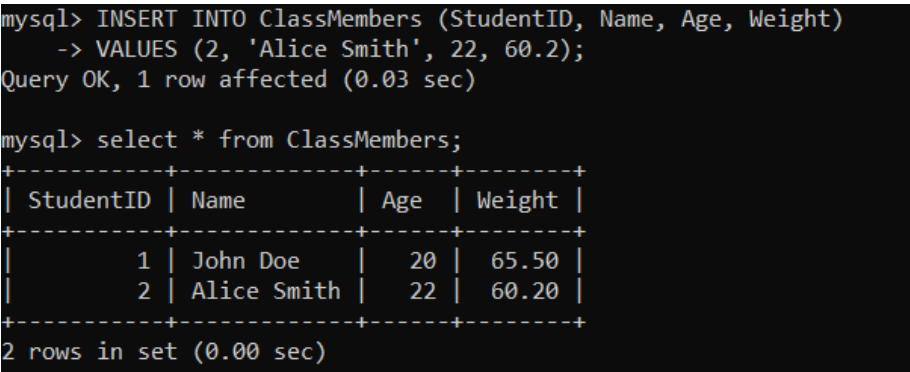
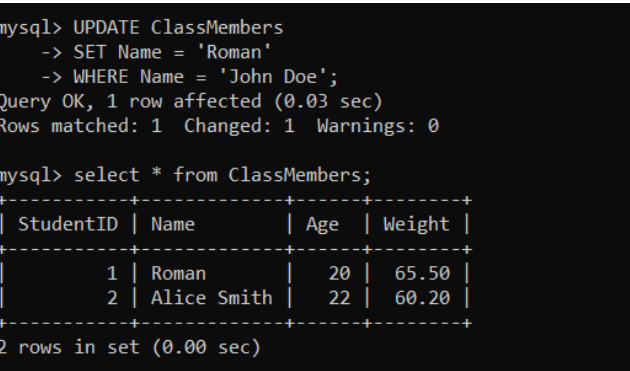
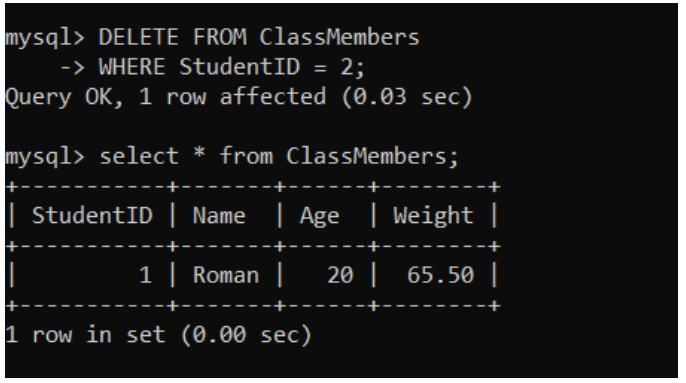
Select: It shows the record of the specific table. Also, it can be used with a WHERE clause to get the particular record.

Insert: It allows users to insert data into the database or tables.

Update: It is used to update or modify the existing data in database tables.

Delete: It is used to delete records from the database tables. Also, it can be used with a WHERE clause to delete a particular row from the table.

Merge: It allows the insert and update(UPSERT) operations.

* 
* 
* 
* 

### **DCL(Data Control Language):**

* DCL works to deal with SQL commands that are used to permit a user to access, modify and work on a database. it is used to access stored data. It gives access, revokes access, and changes the permission to the owner of the database as per the requirement.
* Advantages are:
  + 1. Security:
* DCL commands allow database administrators to control who has access to the data in a database and what actions they can perform on that data. Thus, ensuring the security and integrity of the data stored in the database.
  + 1. Granular control:
* DCL commands allow administrators to grant and revoke specific privileges and permissions, Thus giving differing levels of access (granular control) to a particular user.
  + 1. Flexibility:
* The ability to grant and revoke privileges as per the requirements provides flexibility to database administrators in order to manage access to the database.
* Disadvantages are:
  1. Complexity:
* If there is a large number of users and the database is complex, then granting and revoking privileges can be complex and time-consuming.
  1. Risk of human error:
* Human administrators execute DCL commands and can make mistakes in granting or revoking privileges. Thus, giving unauthorized access to data or imposing unintended restrictions on access.
  1. Lack of audit trail:
* There may be no built-in mechanism to track changes to privileges and permissions over time. Thus, it is extremely difficult to determine who has access to the data and when that access was granted or revoked.
* The commands used in DCL are:

Grant: It is used to give access to security privileges to a specific database user.

Revoke: It is used to revoke the access from the user that is being granted by the grant command.



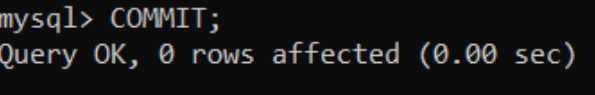
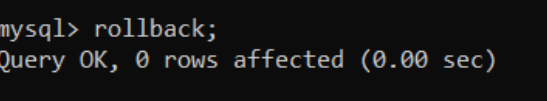


### **TCL(Transaction Control Language):**

* It can be grouped into a logical transaction and is used to run the changes made by the DML command in the database.
* Advantages are-
  + 1. The TCL commands are easy to remember and handy.
    2. Database users can very easily access, edit, and save changes to the database.
    3. The coding required is not very complex. Hence, even new users can use these commands feasibly.
    4. TCL commands ensure the consistency of the data in a database. Hence, if any operation within a transaction fails, the transaction is rolled back.
    5. TCL commands ensure the durability of the data in a database. Hence, when a transaction is committed, its changes are made permanent, even if the system fails or restarts.
* Disadvantages are-
  + 1. For new users, identifying savepoints can be a lot confusing.
    2. The ROLLBACK command is of no use when a user commits using the COMMIT command.
    3. Transactions add overhead to the database system, requiring additional processing and storage resources. Thus, resulting in slower performance and increased resource consumption.
    4. Transactions can add complexity to database systems if the database is large or complex. Hence, it will require careful management and monitoring to ensure the system remains stable and functional.
* The commands used are-

Commit: Transaction on the database is saved using Commit.

Rollback: The database gets restored to the original since the last commit.

* 
* 
* Database interface
* An interface is a program that allows users to input queries into a database without writing the code in the query language.
* An interface can be used to manipulate the database for adding, deleting, updating, or viewing the data.

### Types of Interface are

### **Form−based Interface**

* A form is displayed to each user by the form−based interface. The user fills in the details and submits the form to make a new entry into the database.
* It can also be done when the user only fills in some details and the system will help by retrieving the rest of the details from the database.
* The form−based interface is built for the naive user(inexperienced user) which deals with a limited number of operations. Many DBMS have specification language which helps the programmer define such forms.
* **Example**

Student entering his roll. no, branch in the form to get the grade card.

### **Menu−based User Interface**

* In this interface, the user was provided with a list of options (called a menu) through which the user forms a request. The user doesn’t need to memorize the command and syntax and the query is composed step by step by picking options from a menu.
* Pull−down menu interfaces are mostly used in web−based user interfaces and are often used in browsing interfaces by which the database content can be looked through.
* **Example**

In a shopping website, categories are selected from the menu, brands are selected from the menu of brands, and budget ranges are applied from the menu of budget range.

### **GUI(Graphical User Interface)**

* Users are provided a schema of diagrammatic form by which query can be specified through manipulating the diagram.
* GUI utilizes both menu and form in several cases. Schema Diagram's specific parts are selected using devices used by GUI.
* **Example**

You liked a video on Instagram by tapping with your finger, and the color changes to red. The visual graphic gets changed due to user action.

### **Natural Language Interface**

* A natural language interface contains its unique schema more like the high−level conceptual schema. It also has a directory of important words. It generates a query based on the interpretation of important words in the input by the user and if the interpretation is successful, then it displays the result to the user.
* **Example**

A user googled the fastest car in India, and now the natural language interface will look for the important words i. e. fastest, car, India, and show the result accordingly.

### **Speech Input and Output**

* The users query the interface with speech and get the answer in speech. The input is detected using predefined words and conversions are done into speech to provide the output. Nowadays, it has become the most common type of interface.
* **Example**

OK Google, Siri on Apple, and Alexa is used in the form of speech.

### **Interface for DBA**

* DBA staff are provided commands that can only be used by them only to create an account, grant account authorization, and change a schema, and storage structure reorganization.