



DATABASE AND DATABASE MANAGEMENT SYSTEM

By Kajol Ramtel

INTRODUCTION TO DATA

Data is raw facts, figures, symbols, or observations that have no meaning in themselves. Raw data is of no use until and unless it is processed to get some useful information. Examples of data include the temperature readings from a thermometer, the text of a document, the pixels in a digital image, or the sales figures for a company.

INTRODUCTION TO DATA

Characteristics of data:-

- ☐ Accurate
- ☐ Completeness
- ☐ Consistent
- ☐ Relevance
- ☐ Timeliness
- ☐ Validity
- ☐ Precision
- ☐ Consolidation

INTRODUCTION

- A database is the collection of related data.
- A database is an organized collection of structured information, or data, typically stored electronically in a computer system.
- A database is designed, built and populated with data for specific purpose.

CHARACTERISTICS OF DATABASE

- Shared
- Persistence
- Integrity
- Security
- Concurrency control
- Non- redundancy
- Consistency
- Independency
- Scalability

FILE SYSTEM VS DATABASE SYSTEM

File System

- File systems organize data as files and directories in a hierarchical structure. Each file contains data, and directories provide a way to organize and group related files.
- Redundant data can be present in the file system
- File systems do not inherently enforce data integrity constraints. Users and applications are responsible for ensuring data consistency and accuracy.

Database System

- Databases organize data into tables, rows, and columns, following a relational model. Data is stored in a structured format, and relationships between data entities are defined using keys and foreign keys.
- There is no redundant data in the database system.
- Databases enforce data integrity constraints, such as primary key constraints, foreign key constraints, and data type constraints, to maintain data accuracy, consistency, and validity.

FILE SYSTEM VS DATABASE SYSTEM

File System

- Data retrieval in a file system typically involves navigating directories and opening files using file paths.
- File systems may lack built-in concurrency control mechanisms, leading to potential issues such as data corruption or lost updates when multiple users or applications access the same files simultaneously.

Database System

- Databases support structured query languages (e.g., SQL) for retrieving data based on specified criteria.
- Databases provide concurrency control mechanisms to ensure that multiple transactions can access and modify data concurrently without conflicting with each other. Techniques such as locking and transaction isolation levels help maintain data consistency and integrity.

FILE SYSTEM VS DATABASE SYSTEM

File System

- Data is widely spread , so there is poor security
- It is less complex system
- It takes up much more space and memory is wasted.
- It does not provide any consistency constraints.
- It is less expensive system

Database System

- There are many ways to provide data security in database system.
- It is a very complex system
- It stores data efficiently as a result it takes up less space and memory is also not wasted.
- It provides different consistency constraints to provide data integrity in the system.
- It is expensive storage system

DATABASE MANAGEMENT SYSTEM

- A Database Management System (DBMS) is a software system that provides an interface for users and applications to interact with a database. It facilitates the creation, management, organization, retrieval, modification, and deletion of data in a database.
- Examples of DBMS are:-
 - ORACLE, Sybase, Microsoft SQL server, MySQL, MongoDB.



FUNCTIONS OF DBMS

- Data storage
- Data retrieval
- Data manipulation
- Data definition
- Data security
- Concurrency control
- Data integrity
- Data backup and recovery

DATA DEFINITION LANGUAGE (DDL)

- It is a subset of SQL (Structured Query Language) used for defining the structure of a database and its objects. DDL commands allow users to create, modify, and delete database objects such as tables, indexes, views, and constraints.
- Examples:-
 - Create table, create database, drop, alter, truncate etc
- DDL commands are typically used by database administrators and developers to create and manage the structure of the database.

DATA MANIPULATION LANGUAGE (DML)

- DML stands for Data Manipulation Language, and it is a subset of SQL (Structured Query Language) used for manipulating data stored in a database. DML commands allow users to perform operations such as inserting, updating, deleting, and retrieving data from database tables.
- Examples:-
 - Insert , update, delete, select .
- DML commands are commonly used by application developers, database administrators, and users to interact with the data stored in the database. They are essential for managing and manipulating data to ensure that it remains accurate, consistent, and up-to-date.

DATABASE MODELS

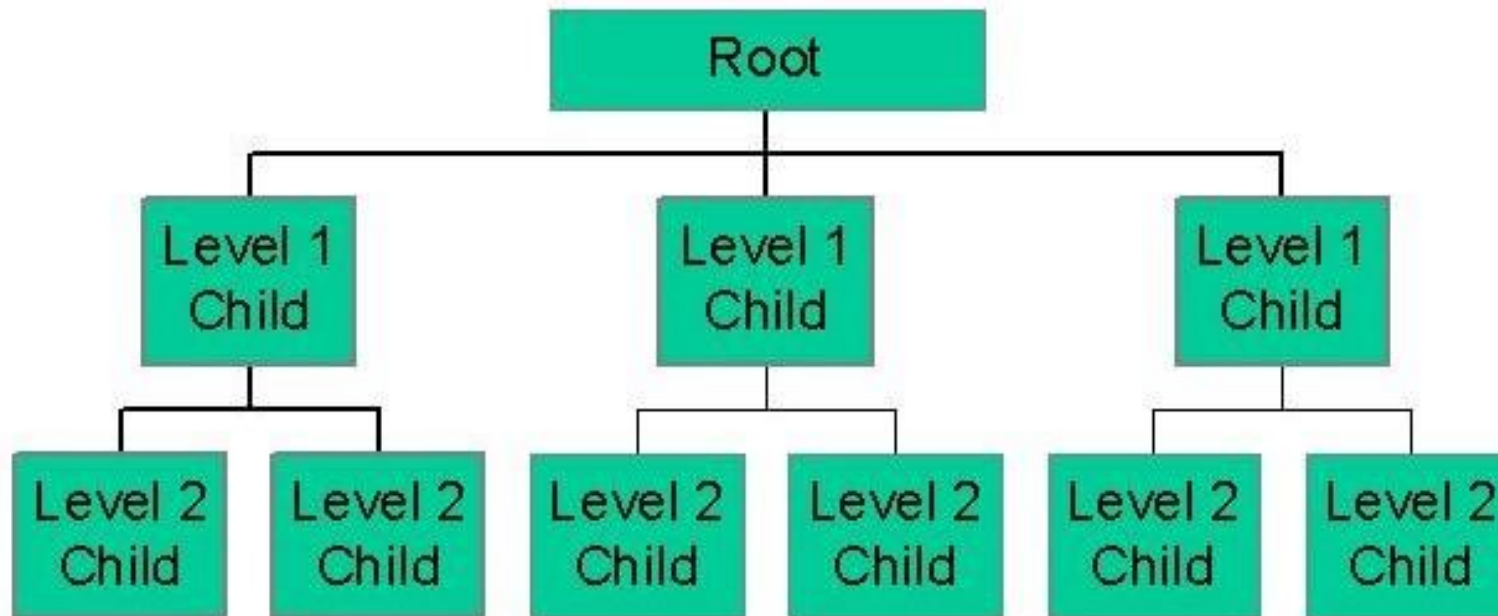
- A database model is the conceptual representation of how data is organized, structured and stored in the database.
- There are different database models that are used to design and develop the database of the organization.
 - a. Hierarchical Model
 - b. Network Model
 - c. Relational Model
 - d. Object oriented model

HIERARCHICAL MODEL

- Data elements are linked in the form of an inverted tree structure with the root at the top and the branches formed below. The single root data element is the parent elements, each of which in turn has one or more other elements. There is a parent child relationship among the data elements of a hierarchical database. There may be many child elements under a parent element, but there can be only one parent element for any child element. The branches in the tree are not connected. It is used in many database applications because the data elements of many applications can be neatly organized in the form of hierarchical tree structure.

HIERARCHICAL MODEL

Hierarchical database model



ADVANTAGES OF HIERARCHICAL MODEL

- Easiest model
- Secure model because no one can modify the child without consulting the parent.
- Searching is fast and easy.
- It is efficient in handling one to many relationships

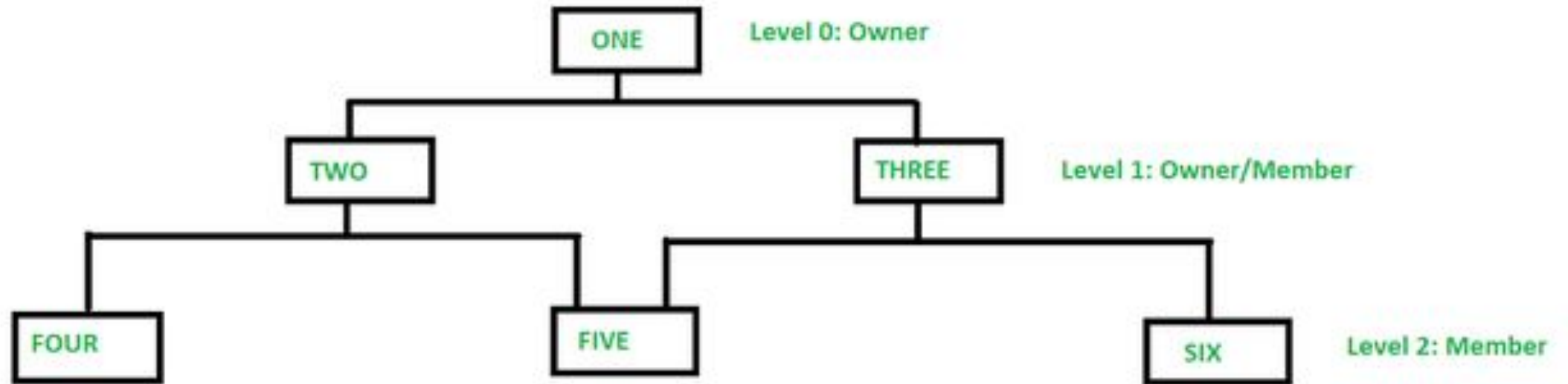
DISADVANTAGES OF HIERARCHICAL MODEL

- Old fashioned and outdated
- Modification and addition of child without consulting its parent is impossible.
- Cannot handle many to many relationships
- Increase redundancy
- It does not support flexible data access, because data can be accessed only by following the path down the trees structure.

NETWORK DATA MODEL

- It is an extension of hierarchical database model. In this model, the data elements of a database are organized in the form of parent-child relationships and all types of relationships among the data elements must be determined when the database is first designed. A child data element must be determined when the database is first designed. Network database model organized data more like a graph and can have **more than one parent** node. It means that it can have M: N relations i.e., many-to-many which allows a record to have more than one parent segment. The network model is a database model conceived as a flexible way of representing objects and their relationships.
- Here, a relationship is called a set, and each set is made up of at least 2 types of record which are given below:
 - An owner record that is the same as of parent in the hierarchical model.
 - A member record that is the same as of child in the hierarchical model.

NETWORK DATA MODEL



ADVANTAGES OF NETWORK DATA MODEL

- Simple and easy like hierarchical model
- Capable of handling multiple types of relationships which can help in modeling real-life applications.
- More flexible than hierarchical model because it accepts many to many relationships as well.
- Network does not allow a member to exist without an owner which leads to the concept of Data integrity.

DISADVANTAGES OF NETWORK DATA MODEL

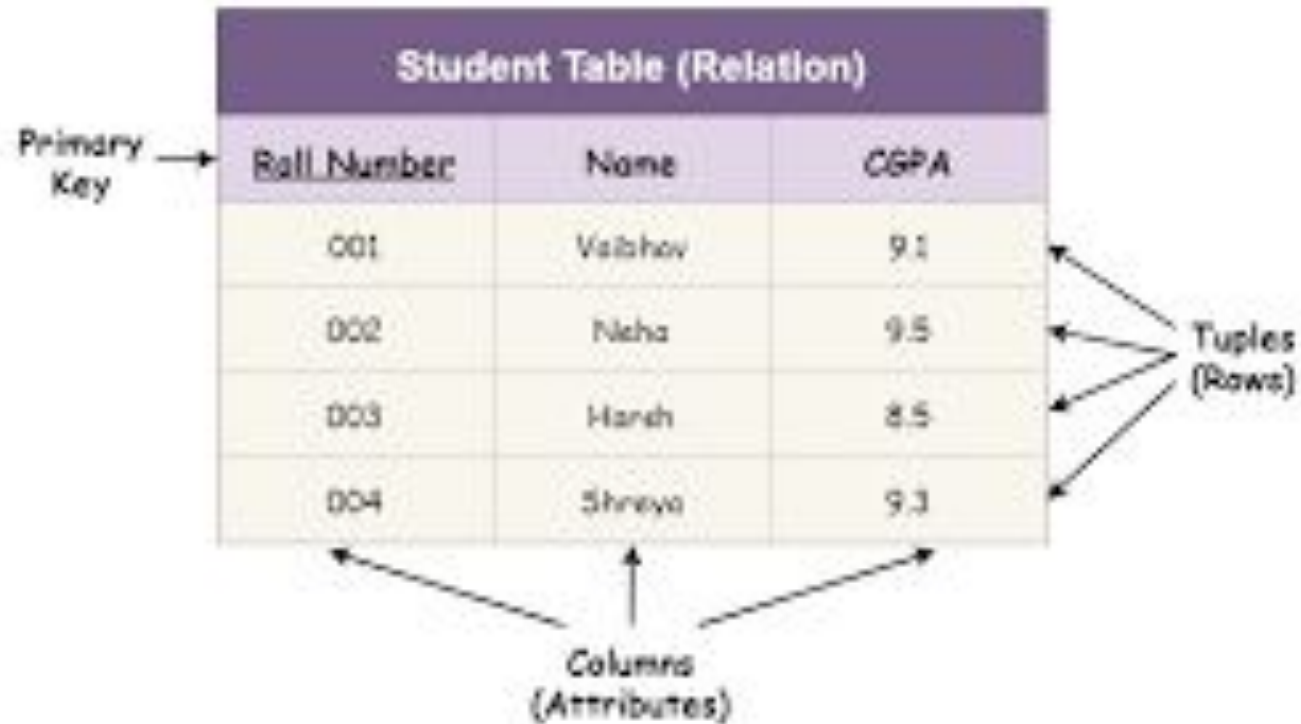
- Database is very complex in nature as all the records are maintained by the use of pointers. Points also increases the overhead of storage.
- Needs long program to handle this relationship.
- Lack of structural independence.
- This model does not have any scope of automated query optimization.

RELATIONAL DATA MODEL

- The relational model uses a collection of tables to represent both data and the relationships among those data. Each table has multiple columns, and each column has a unique name. Each table of the database is store in a separate file. Tables are also known as relations. Each columns represent a data field and each rows represent a data record. A record is known as a tuple.
- The data in one table is related to the data in another table with a common field.
- Each table contains records of a particular type.
- The relational data model is the most widely used data model, and a vast majority of current database systems are based on the relational model.

RELATIONAL DATA MODEL

Relational Model in DBMS



ADVANTAGES OF RELATIONAL DATA MODEL

- Less data redundancy
- Normalization of data is possible in this model
- Quick database processing is possible
- Flexibility
- Secure

DISADVANTAGES OF RELATIONAL DATA MODEL

- Relational Database Model is not very good for large databases.
- Sometimes, it becomes difficult to find the relation between tables.
- Because of the complex structure, the response time for queries is high.
- Too many rules makes it not user-friendly

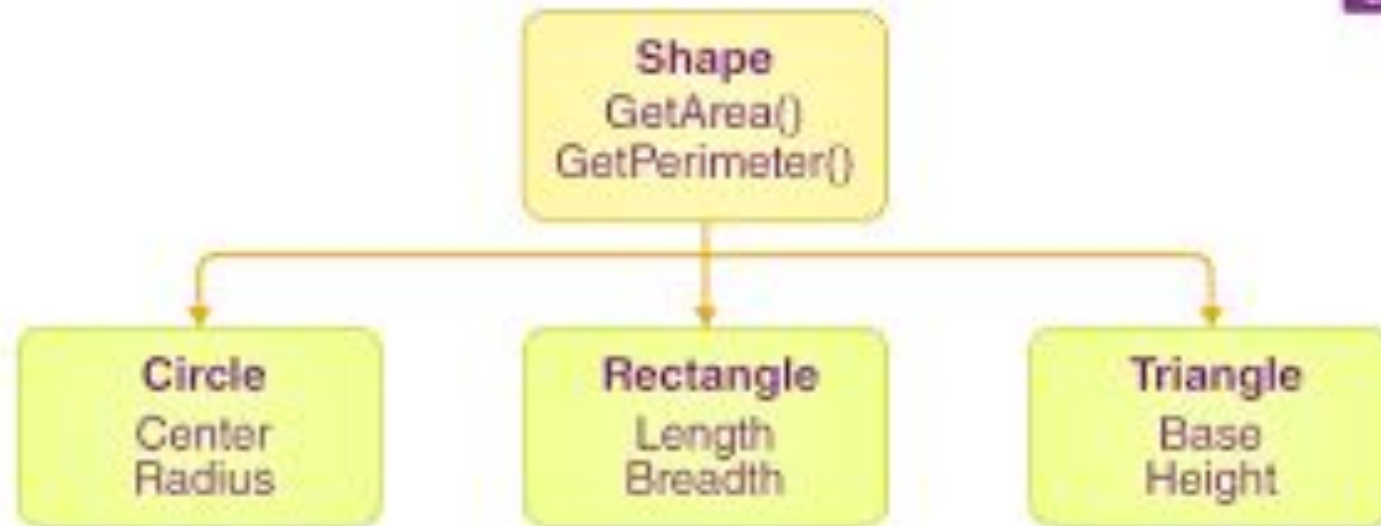
DISADVANTAGES OF RELATIONAL DATA MODEL

- Relational Database Model is not very good for large databases.
- Sometimes, it becomes difficult to find the relation between tables.
- Because of the complex structure, the response time for queries is high.
- Too many rules makes it not user-friendly

OBJECT ORIENTED DATABASE MODEL

In Object Oriented data (attributes) and behaviors (methods) are encapsulated in a single structure which is referred as object in this data model. In this, real world problems are represented as objects with different attributes. All objects have multiple relationships between them. Basically, it is combination of Object Oriented programming and Relational Database Model.

OBJECT ORIENTED DATABASE MODEL



COMPONENTS OF OBJECT ORIENTED DATABASE MODEL

- Objects:-

- An object is an abstraction of a real world entity or we can say it is an instance of class. Objects encapsulates data and code into a single unit which provide data abstraction by hiding the implementation details from the user.

- Attributes

- An attribute describes the properties of object. For example: In Circle class the object is CIRCLE and attributes are radius and center.

- Methods

- Method represents the behavior of an object. Basically, it represents the real-world action. For example: Calculating the area in the Shape class is done by method getArea().

- Class

- A class is a collection of similar objects with shared structure i.e. attributes and behavior i.e. methods. An object is an instance of class. For example: Shape, Circle, rectangle and triangle in above figure.

COMPONENTS OF OBJECT ORIENTED DATABASE MODEL

- Inheritance

- By using inheritance, new class can inherit the attributes and methods of the old class i.e. base class. For example: as classes Circle , rectangle and triangle are inherited from the base class Shape.

ADVANTAGES OF OODBMS

- Codes can be reused due to inheritance.
- Easily understandable.
- Cost of maintenance can be reduced due to reusability of attributes and functions because of inheritance.

DISADVANTAGES OF OODBMS

- It is not properly developed so not accepted by users easily.

DATABASE ARCHITECTURE

- DBMS architecture describes the structure and how the users are connected to a specific database system. Database architecture describes how a database management system (DBMS) will be integrated with your application. When designing a database architecture, you will make decisions that will change how your applications are created.
- A DBMS architecture allows dividing the database system into individual components that can be independently modified, changed, replaced, and altered. It also helps to understand the components of a database.
- A Database Architecture is a representation of DBMS design. It helps to design, develop, implement, and maintain the database management system.
- A Database stores critical information and helps access data quickly and securely. Therefore, selecting the correct Architecture of DBMS helps in easy and efficient data management.

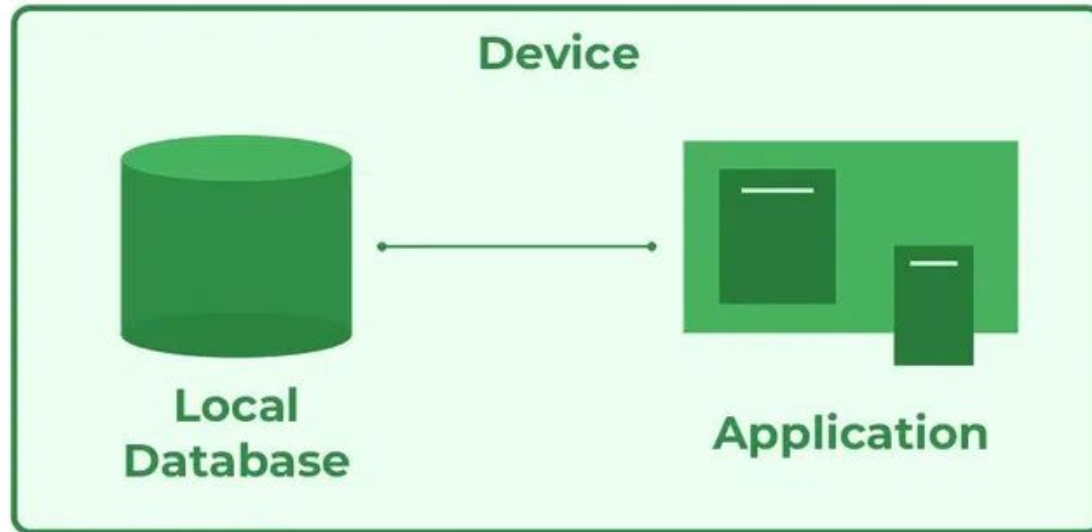
DATABASE ARCHITECTURE

- There are mainly three types of DBMS architecture:
 - One Tier Architecture (Single Tier Architecture)
 - Two Tier Architecture
 - Three Tier Architecture

ONE TIER ARCHITECTURE

- 1 Tier Architecture in DBMS is the simplest architecture of Database in which the client, server, and Database all reside on the same machine.
- In 1-Tier Architecture the database is directly available to the user, the user can directly sit on the DBMS and use it that is, the client, server, and Database are all present on the same machine.
- A simple one tier architecture example would be anytime you install a Database in your system and access it to practice SQL queries.
- For Example: to learn SQL we set up an SQL server and the database on the local system. This enables us to directly interact with the relational database and execute operations. The industry won't use this architecture they logically go for 2-tier and 3-tier Architecture.

ONE TIER ARCHITECTURE

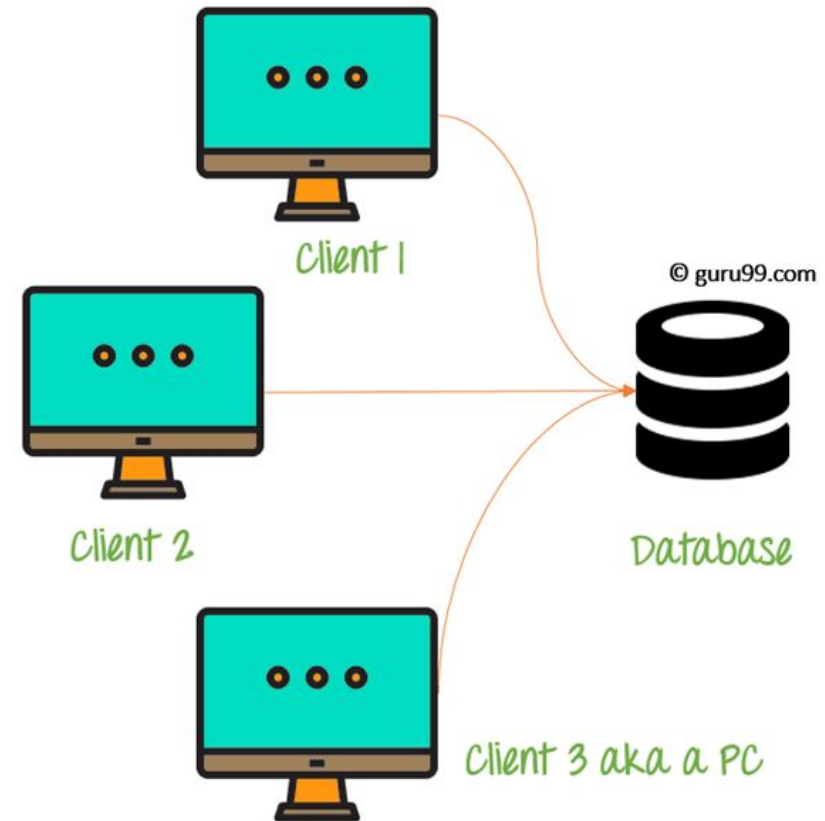


Single Tier Architecture

TWO TIER ARCHITECTURE

- A 2 Tier Architecture in DBMS is a Database architecture where the presentation layer runs on a client (PC, Mobile, Tablet, etc.), and data is stored on a server called the second tier. Two tier architecture provides added security to the DBMS as it is not exposed to the end-user directly. It also provides direct and faster communication.
- It is similar to a basic client-server model. The application at the client end directly communicates with the database on the server side. APIs like ODBC and JDBC are used for this interaction. The server side is responsible for providing query processing and transaction management functionalities. On the client side, the user interfaces and application programs are run. The application on the client side establishes a connection with the server side to communicate with the DBMS.
- An advantage of this type is that maintenance and understanding are easier, and compatible with existing systems. However, this model gives poor performance when there are a large number of users.

TWO TIER ARCHITECTURE



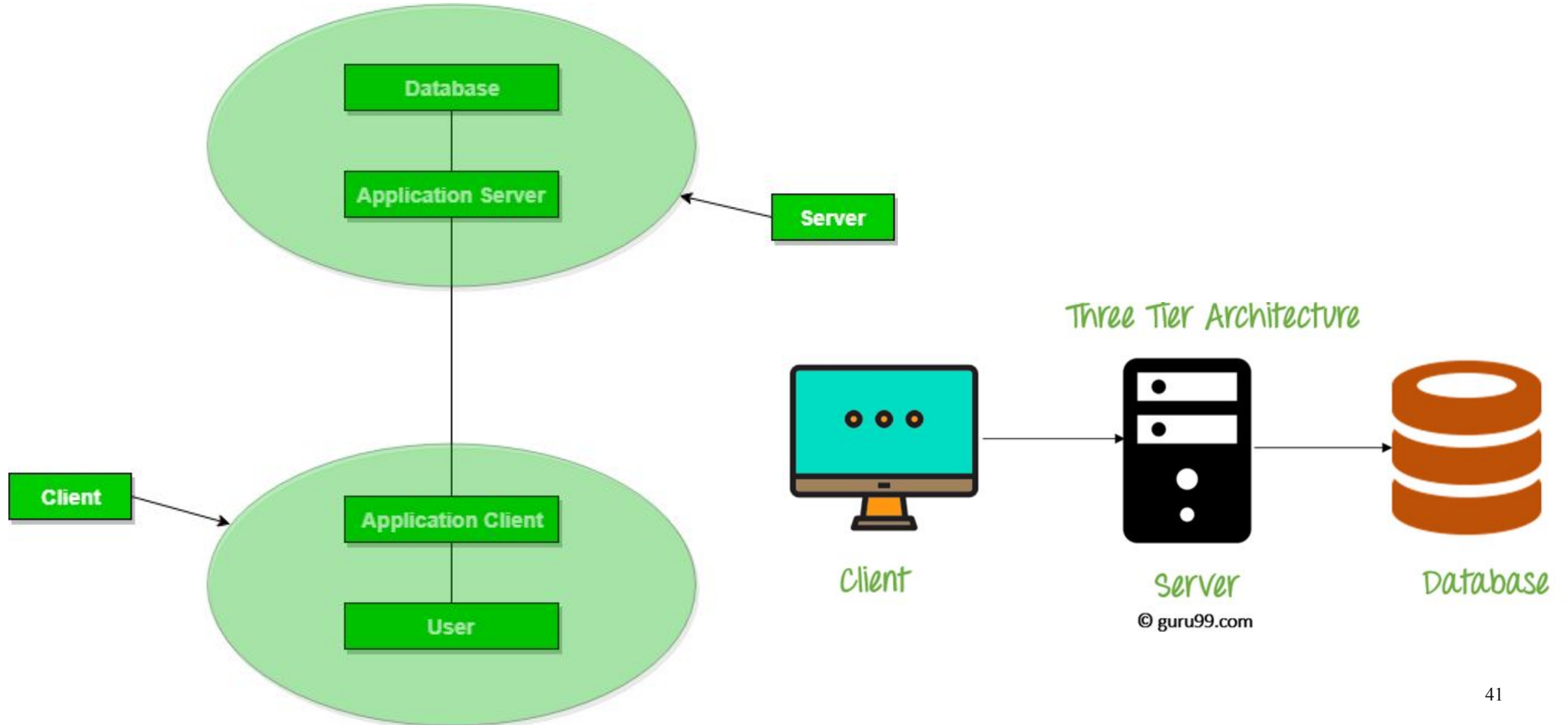
THREE TIER ARCHITECTURE

- A 3 Tier Architecture in DBMS is the most popular client server architecture in DBMS in which the development and maintenance of functional processes, logic, data access, data storage, and user interface is done independently as separate modules. Three Tier architecture contains a presentation layer, an application layer, and a database server.
- 3-Tier database Architecture design is an extension of the 2-tier client-server architecture. It has the following layers:
 - Presentation layer (your PC, Tablet, Mobile, etc.)
 - Application layer (server)
 - Database Server

THREE TIER ARCHITECTURE

- There is another layer between the client and the server. The client does not directly communicate with the server. Instead, it interacts with an application server which further communicates with the database system and then the query processing and transaction management takes place. This intermediate layer acts as a medium for the exchange of partially processed data between the server and the client. This type of architecture is used in the case of large web applications.

THREE TIER ARCHITECTURE



DATABASE APPLICATIONS



ASSIGNMENT

- What is database architecture? Explain database architecture with advantages and disadvantages.
- Explain about different database applications.