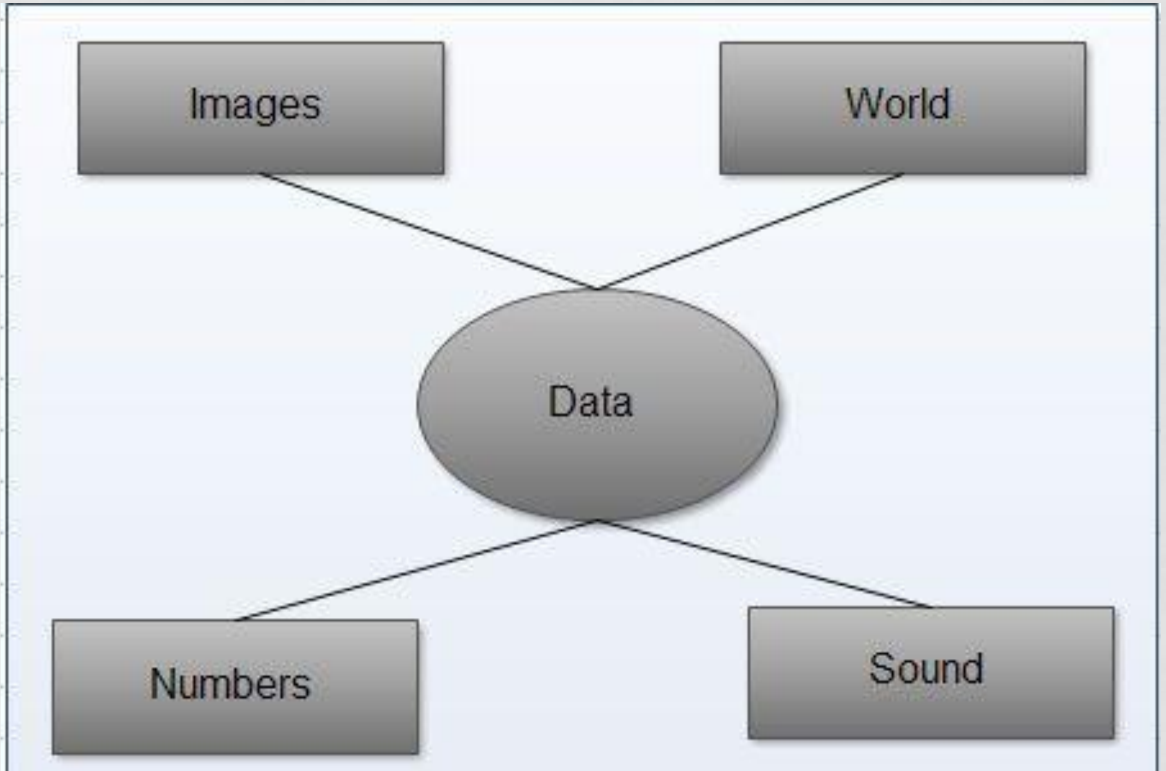
DATABASE MANAGEMENT SYSTEM

UNIT 1

DATA



Different forms of data representation

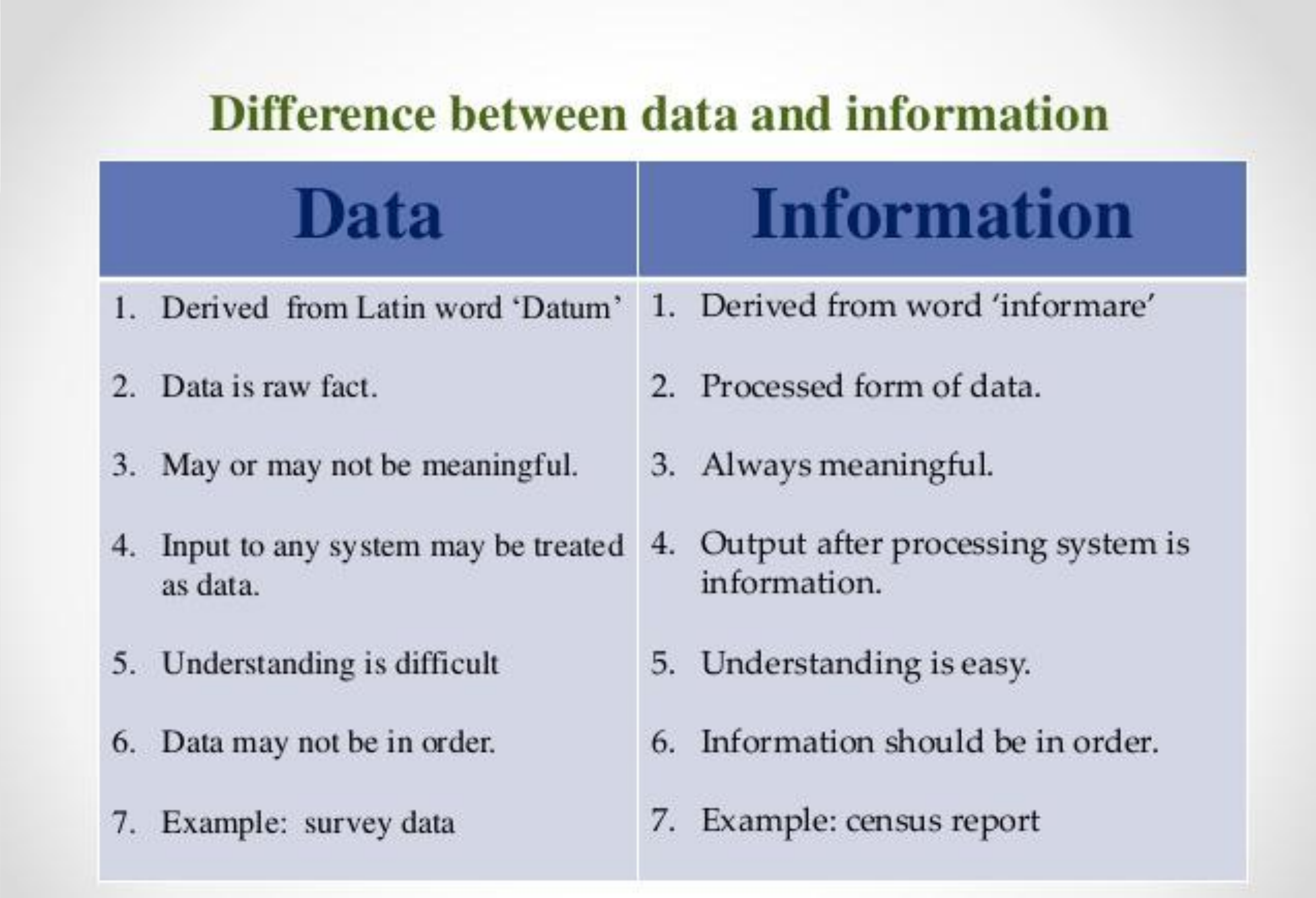
INFORMATION

- Information is organised or classified data, which has some meaningful values for the receiver.
- Information is data that has been converted into more useful or intelligible form.

Characteristics of Information:

- Timely
- Accuracy
- completeness

Compiled by Ms. Prajakta Joshi



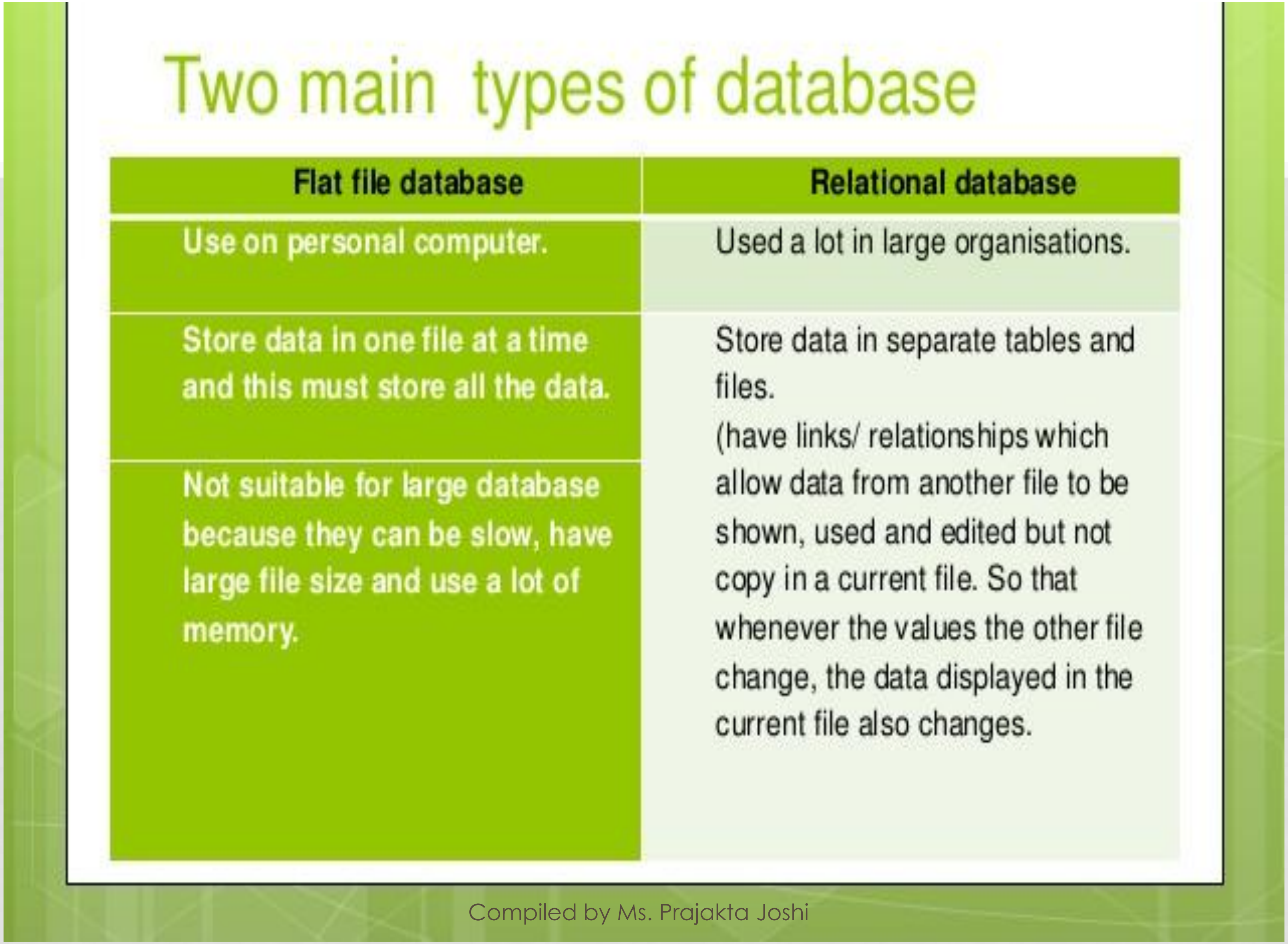
Difference between data and information

Data	Information
1. Derived from Latin word ‘Datum’	1. Derived from word ‘informare’
2. Data is raw fact.	2. Processed form of data.
3. May or may not be meaningful.	3. Always meaningful.
4. Input to any system may be treated as data.	4. Output after processing system is information.
5. Understanding is difficult	5. Understanding is easy.
6. Data may not be in order.	6. Information should be in order.
7. Example: survey data	7. Example: census report

DATABASE

- A database is a collection of related data which represents some aspect of the real world. A database system is designed to be built and populated with data for a certain task

Compiled by Ms. Prajakta Joshi



Two main types of database

Flat file database	Relational database
Use on personal computer.	Used a lot in large organisations.
Store data in one file at a time and this must store all the data.	Store data in separate tables and files. (have links/ relationships which allow data from another file to be shown, used and edited but not copy in a current file. So that whenever the values the other file change, the data displayed in the current file also changes.
Not suitable for large database because they can be slow, have large file size and use a lot of memory.	

WHAT IS DBMS?

- Collection of interrelated data
- Set of programs to access the data
- DBMS contains information about a particular enterprise
- DBMS provides an environment that is both *convenient* and *efficient* to use.

Compiled by Ms. Prajakta Joshi

DBMS

- DBMS is a collection of organized, interrelated data and set of programs to store the data efficiently and access those data in an easy and effective manner

A software package/ system to facilitate the creation and maintenance of a computerized database.

It

- defines (data types, structures, constraints)
- construct (storing data on some storage medium

controlled by DBMS)

- manipulate (querying, update, report generation) databases for various applications.

ADVANTAGES OF DBMS

- Reduction of Redundancies
- Elimination of Inconsistencies
- Shared Data
- Integrity
- Security
- Data Independence

➢Physical Data Independence

➢Logical Data Independence

DISADVANTAGES OF DBMS

- Backup and Recovery Issues
- Complexity
- Performance
- Security
- Problem associated with centralization

NEED OF DBMS

- Creation of a database.
- Retrieval of information from the database.
- Updating the database.
- Managing a database.
- Storing Database

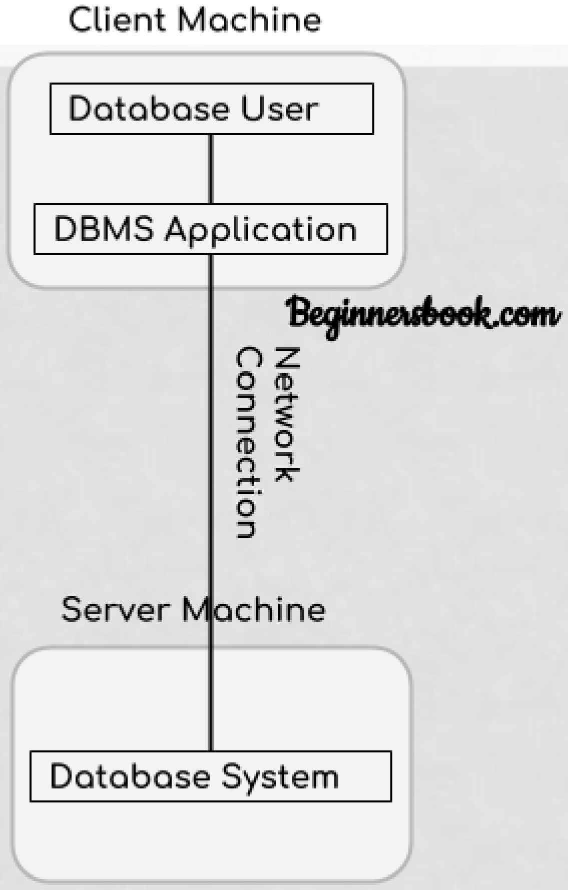
APPLICATIONS OF DBMS

- Home
- Banking
- Reservation System
- Finance
- E-Commerce
- Industry
- Education
- Sales

ARCHITECTURE OF DBMS

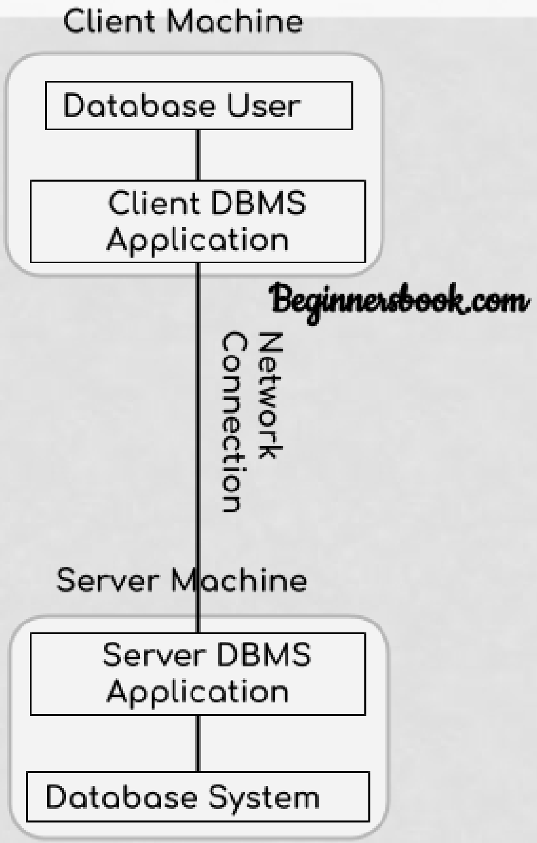
- **1. Single tier architecture**
- In this type of architecture, the database is readily available on the client machine, any request made by client doesn't require a network connection to perform the action on the database.
- For example, lets say you want to fetch the records of employee from the database and the database is available on your computer system, so the request to fetch employee details will be done by your computer and the records will be fetched from the database by your computer as well. This type of system is generally referred as local database system.

TWO TIER ARCHITECTURE



Two-Tier architecture

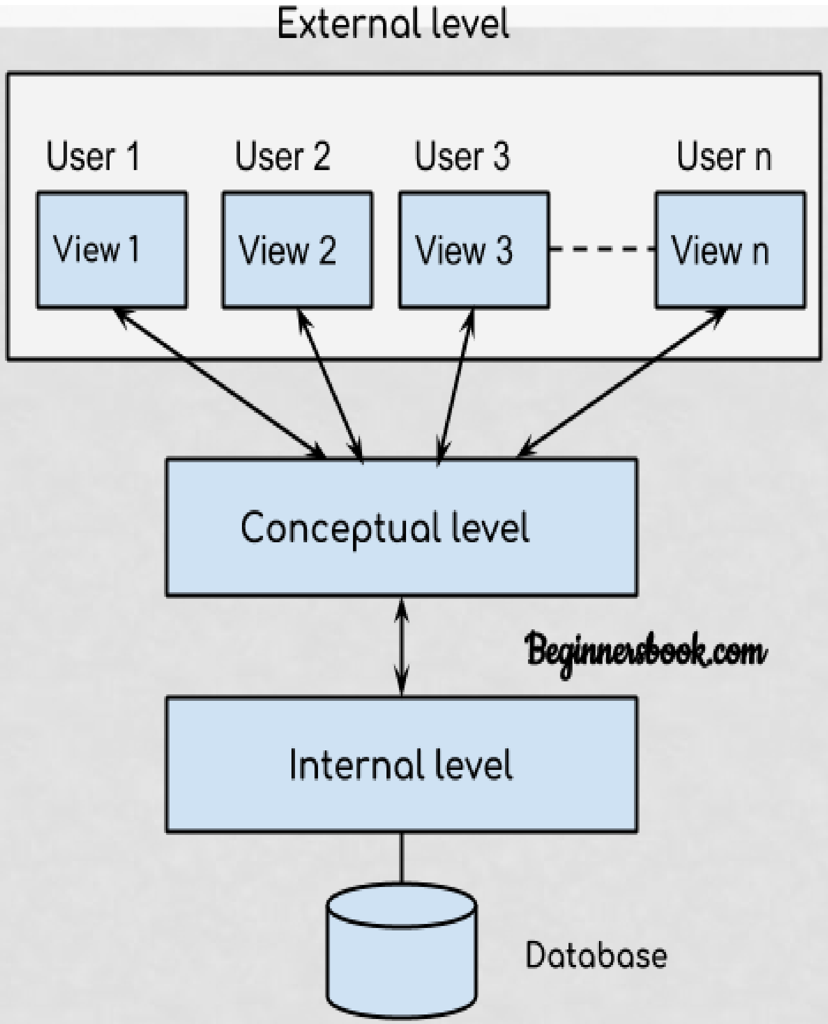
THREE TIER ARCHITECTURE



Three-Tier architecture

Compiled by Ms. Prajakta Joshi

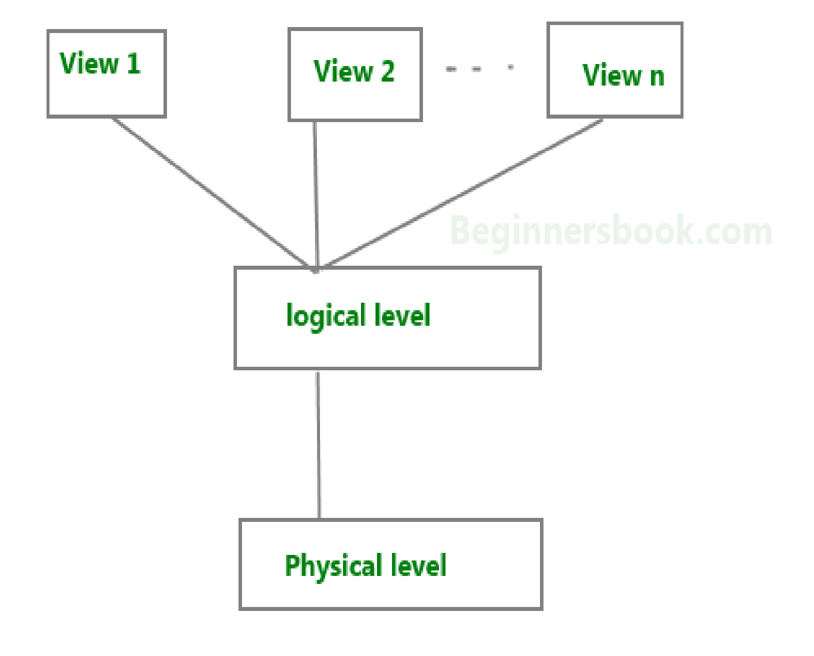
THREE LEVEL ARCHITECTURE OF DATABASE



VIEW OF DATA

- Data abstraction
- Instance and schema

DATA ABSTRACTION IN DBMS



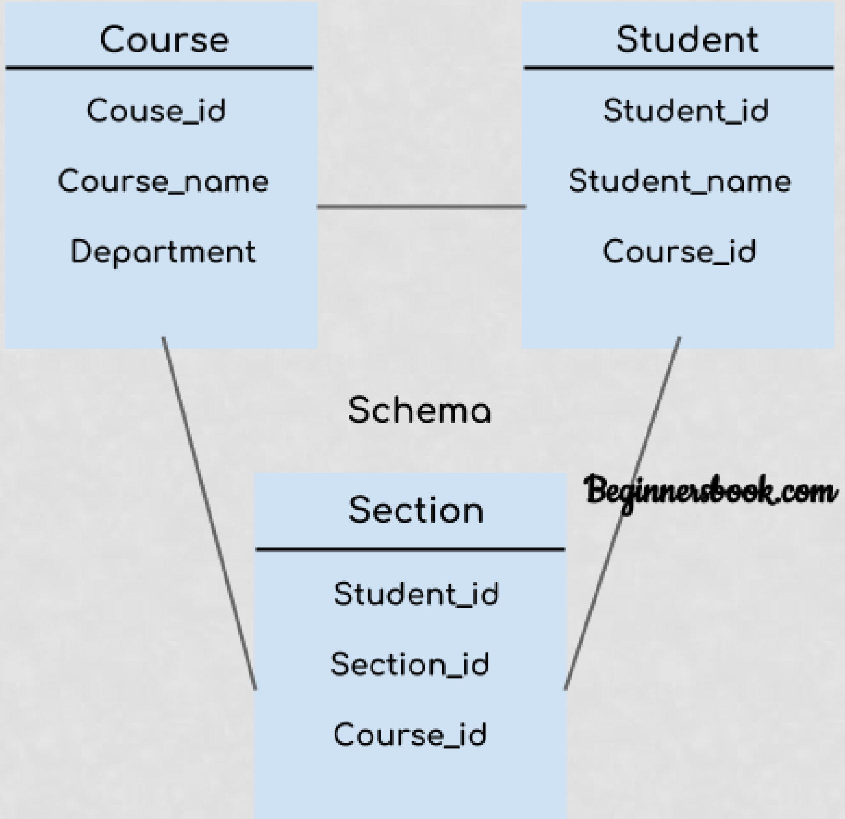
Three Levels of data abstraction

- **Physical level**: This is the lowest level of data abstraction. It describes how data is actually stored in database. You can get the complex data structure details at this level.
- **Logical level**: This is the middle level of 3-level data abstraction architecture. It describes what data is stored in database.
- **View level**: Highest level of data abstraction. This level

INSTANCE AND SCHEMA IN DBMS

- **DBMS Schema**
- **Definition of schema**: Design of a database is called the schema. Schema is of three types: Physical schema, logical schema and view schema.

- Physical SchemaThe design of a database at physical level is called **physical schema**, how the data stored in blocks of



SCHEMAS IN DBMS

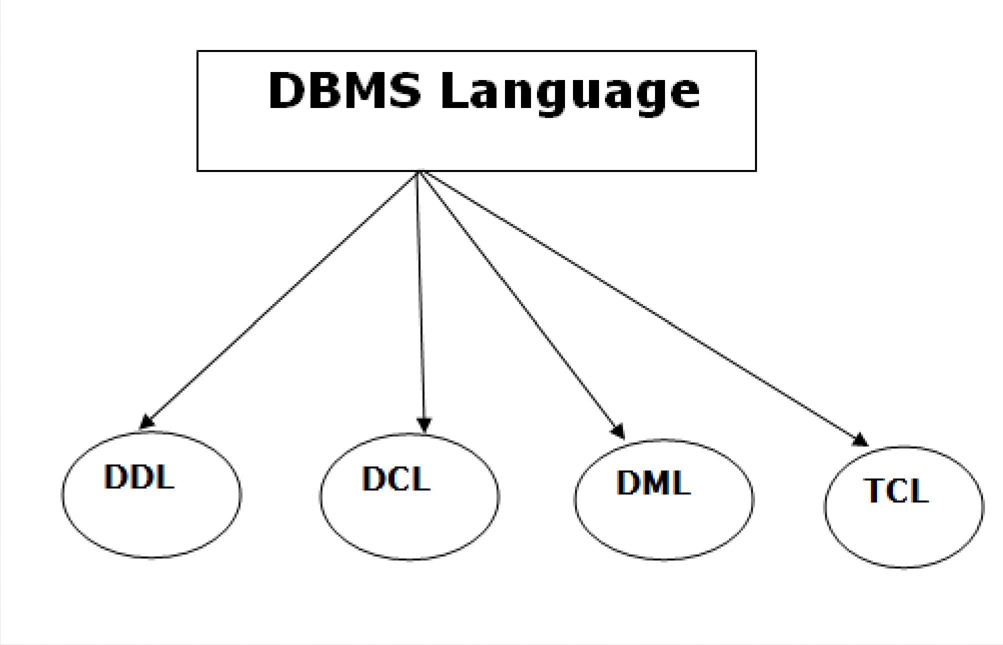
- Logical Schema:
- Design of database at logical level is called **logical schema**, programmers and database administrators work at this level, at this level data can be described as certain types of data records gets stored in data structures, however the internal details such as implementation of data

- View Schema:
- Design of database at view level is called **view schema**. This generally describes end user interaction with database systems.

DBMS INSTANCE

- **Definition of instance**: The data stored in database at a particular moment of time is called instance of database.
- Database schema defines the variable declarations in tables that belong to a particular database; the value of these variables at a moment of time is called the instance of that database.

DATABASE LANGUAGES



DBMS Language

DDL

DCL

DML

TCL

DATA DEFINITION LANGUAGE

- **DDL** stands for **D**ata **D**efinition **L**anguage. It is used to define database structure or pattern.
- It is used to create schema, tables, indexes, constraints, etc. in the database.
- Using the DDL statements, you can create the skeleton of the database.
- Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.
- Here are some tasks that come under DDL:
- **Create:** It is used to create objects in the database.
- **Alter:** It is used to alter the structure of the database.
- **Drop:** It is used to delete objects from the database.
- **Truncate:** It is used to remove all records from a table.
- **Rename:** It is used to rename an object.

DATA MANIPULATION LANGUAGE

- **DML** stands for **D**ata **M**anipulation **L**anguage. It is used for accessing and manipulating data in a database. It handles user requests.
- Here are some tasks that come under DML:
- **Select:** It is used to retrieve data from a database.
- **Insert:** It is used to insert data into a table.
- **Update:** It is used to update existing data within a table.
- **Delete:** It is used to delete all records from a table.
- **Merge:** It performs UPSERT operation, i.e., insert or update operations.
- **Call:** It is used to call a structured query language or a Java subprogram.
- **Explain Plan:** It has the parameter of explaining data.

DATA CONTROL LANGUAGE

- **DCL** stands for **D**ata **C**ontrol **L**anguage. It is used to retrieve the stored or saved data.
- The DCL execution is transactional. It also has rollback parameters.
- (But in Oracle database, the execution of data control language does not have the feature of rolling back.)
- Here are some tasks that come under DCL:
- **Grant:** It is used to give user access privileges to a database.
- **Revoke:** It is used to take back permissions from the user.
- There are the following operations which have the authorization of Revoke:
- CONNECT, INSERT, USAGE, EXECUTE, DELETE, UPDATE and SELECT.

TRANSACTION CONTROL LANGUAGE

- TCL is used to run the changes made by the DML statement. TCL can be grouped into a logical transaction.
- Here are some tasks that come under TCL:
- **Commit:** It is used to save the transaction on the database.
- **Rollback:** It is used to restore the database to original since the last Commit.

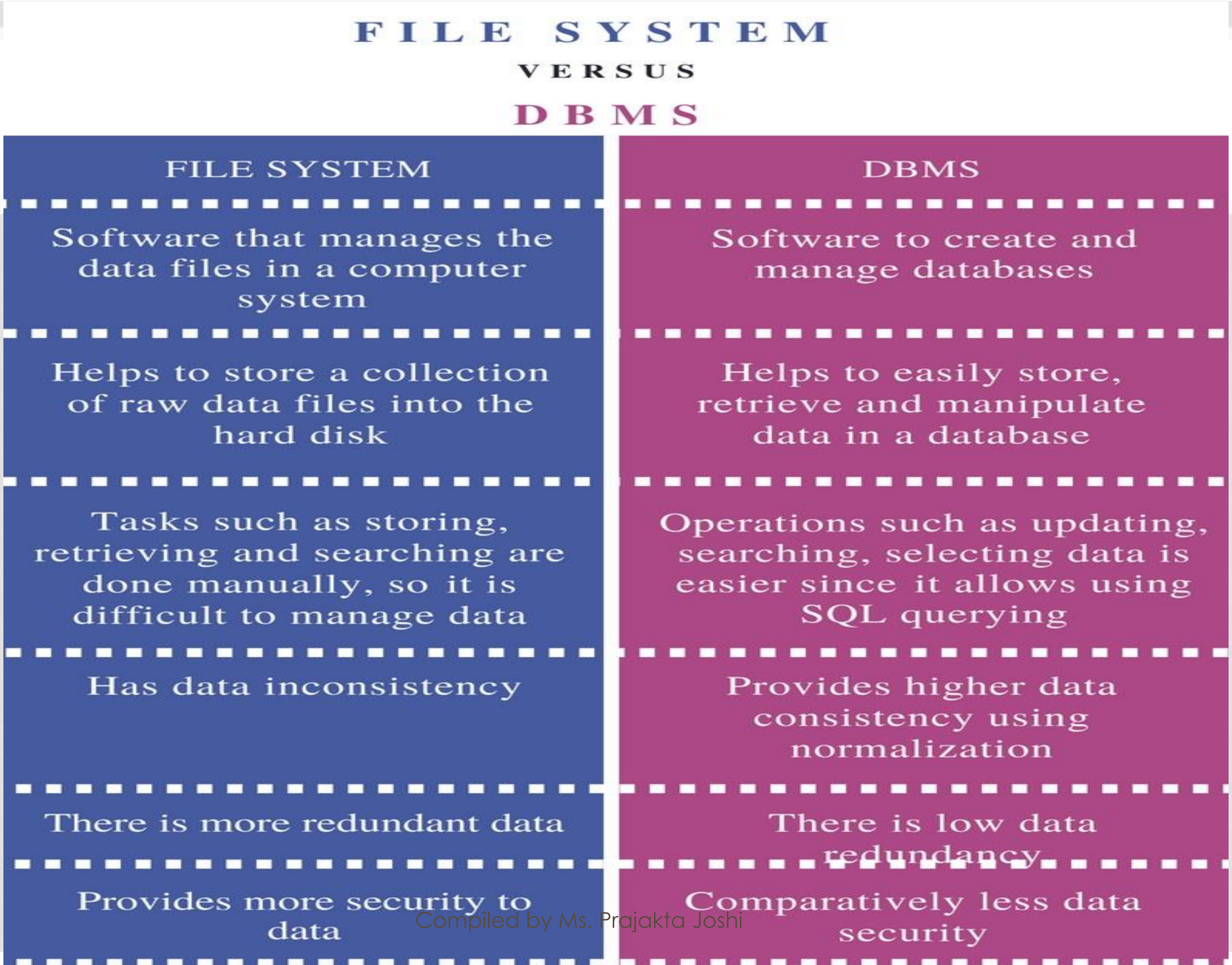
FILE PROCESSING SYSTEM

- **A file processing system helps people keep track of files as they move throughout the various departments of a business.**
- It should be recorded in such a way that
- Should be able to get the data any point in time latter
- Should be able to add details to it whenever required
- Should be able to modify stored information, as needed
- Should also be able to delete them

DISADVANTAGES OF FILE PROCESSING SYSTEM

- Data Redundancy
- Data Inconsistency
- Difficult in Accessing Data
- Integrity Problem
- Data Isolation
- Concurrent Access Problem
- Atomicity Problem

Compiled by Ms. Prajakta Joshi



FILE SYSTEM VERSUS DBMS

FILE SYSTEM	DBMS
Software that manages the data files in a computer system	Software to create and manage databases
Helps to store a collection of raw data files into the hard disk	Helps to easily store, retrieve and manipulate data in a database
Tasks such as storing, retrieving and searching are done manually, so it is difficult to manage data	Operations such as updating, searching, selecting data is easier since it allows using SQL querying
Has data inconsistency	Provides higher data consistency using normalization
There is more redundant data	There is low data redundancy
Provides more security to data	Comparatively less data security

TRANSACTION MANAGEMENT

- A transaction is a logical unit of work that contains one or more SQL statements
- It is a collection of operations that performs a single logical function in a database application
- A transaction is an atomic unit
- Transactions in a database environment have two main purposes:
- Reliability of data
- Concurrent Access of Data

DATA MODELS

- Data models can facilitate interaction among the designer, the application programmer and the end user.
- A well- developed data model can even foster improved understanding of the organization for which the database design is developed.
- Data models are a communication tool.

IMPORTANCE OF DATA MODELS

- **The data model is the blueprint.**
- **The data model is the requirements.**
- **The data model is the specifications.**
- **The data model is reusable.**

THE BASIC BUILDING BLOCKS OF DATA MODELS

- The basic building blocks of all data models are entities, attributes, and relationships.
- An entity is anything, such as a person, place, thing, or event, about which data are to be collected and stored. Entities may be physical objects such as customers or products. But entities may also be abstractions such as flight routes or musical concerts.
- An attribute is a characteristic of an entity. For example, a CUSTOMER entity would be described by attributes such as customer last name, customer first name, customer phone, customer address, and customer credit limit. The attributes are

TYPES OF RELATIONS

- One-to-many (1:M) relationship. A painter paints many different paintings, but each one of them is painted by only one painter. Thus the painter (the "one") is related to the paintings (the "many"). Therefore, database designers label the relationship "PAINTER paints PAINTING" as 1:M. Similarly, a customer (the "one") might generate many invoices, but each invoice (the "many") is generated by only a single customer. The "CUSTOMER generates INVOICE" relationship would also be labeled 1:M.
- Many-to-many (M:N or M:M) relationship. An employee might learn many job skills, and each job skill might be learned by many employees. Database designers label the relationship "EMPLOYEE learns SKILL" as M:N. Similarly, a student can take many classes, and each class can be taken by many students, thus yielding the M:N relationship label for the relationship expressed by "STUDENT