UNIT - 1
INTRODUCTION

PURPOSE OF DATABASE SYSTEM

Drawbacks of file system

In the early days, database applications were built directly on top of file systems

■ Drawbacks of using file systems to store data:

- Data redundancy and inconsistency

    Multiple file formats, duplication of information in different files

- Difficulty in accessing data

    Need to write a new program to carry out each new task

- Data isolation — multiple files and formats

- Integrity problems

    Integrity constraints (e.g. account balance > 0) become "buried" in program code rather than being stated explicitly

    Hard to add new constraints or change existing ones

- Atomicity of updates

    Failures may leave database in an inconsistent state with partial updates carried out

    Example: Transfer of funds from one account to another should either complete or not happen at all

- Concurrent access by multiple users

    Concurrent accessed needed for performance

    Uncontrolled concurrent accesses can lead to inconsistencies – Example: Two people reading a balance and updating it at the same time

- Security problems 4 Hard to provide user access to some, but not all, data

■ Database systems offer solutions to all the above problems

Purpose of database system

The purpose of a database system is to store, organize, and retrieve data in a safe and efficient way:

- Store data: Database systems store large amounts of data in a structured way.
- Organize data: Database systems use predefined schemas and data models to organize data.
- Retrieve data: Database systems provide a way to access data on demand.
- Maintain data integrity: Database systems use rules and constraints to ensure data accuracy and reliability.
- Protect data: Database systems use security measures to protect sensitive data from unauthorized access.

Database systems are used in many applications, including:

- Banking

- Airlines
- Universities
- Sales
- Online retailers
- Manufacturing
- Human resources

Why do we use DBMS?
- To avoid data redundancy and inconsistency
    - Multiple file formats, duplication of information in different files
- To avoid difficulty in accessing data
    - Need to write a new program to carry out each new task
- To deal with data isolation — multiple files and formats
- To deal with integrity problems
    - Integrity constraints (e.g. account balance > 0) become part of program code
    - Easy to add new constraints or change existing ones
- Atomicity of updates
    - Failures may leave database in an inconsistent state with partial updates carried out
    - E.g. transfer of funds from one account to another should either complete or not happen at all
- Concurrent access by multiple users
    - Concurrent accessed needed for performance
    - Uncontrolled concurrent accesses can lead to inconsistencies
    - E.g. two people reading a balance and updating it at the same time
- Security problems