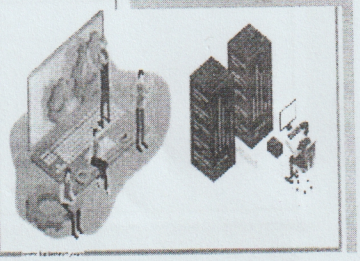
Database system

Theoretical lectures

The first lecture

Assistant teacher Shaymaa Abdulkarem



Introduction to Database Concepts

Data and Information

▪**Data :**

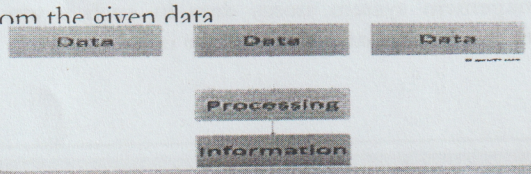
It is raw, unorganized facts that need to be processed. Data can be something simple and seemingly random and useless until it is organized.

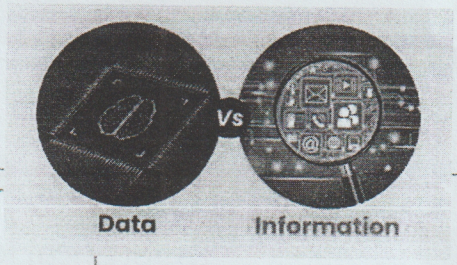
Example: Each student's test score is one piece of data.

▪**Information:**

When data is processed, organized, structured or presented in a given context so as to make it useful, it is called information.

Example: The average score of a class or of the entire school is information that can be derived from the given data.





Data

- Data is collection of facts
- Data is unorganized.
- Data does not depend on information
- Data isn't sufficient for decision making

Information

- Information puts facts into context
- Information is organized
- Information depends on data
- Information is sufficient for decision making

What is a Database ?

A Database is a collection of related data and data is a collection of facts and figures that can be processed to produce information.

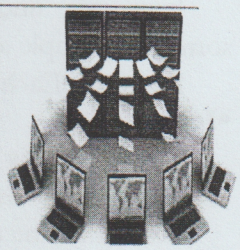
- Mostly data represents recordable facts for examples:

✓Bank needs to store the information relating to customer accounts

✓Hospital needs to keep data about patients and the medication dispensed,

✓University needs to maintain record of its students,

✓Internet sales business needs toTraffic log for future forensic analysis.



A database management system stores data in such a way that it becomes easier to retrieve, manipulate, and produce information.

University Database Example

Application program examples"

✓Add new students, instructors, and courses

✓Register students for courses, and generate class rosters

✓Assign grades to students, compute grade point averages (GPA) and generate transcripts

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

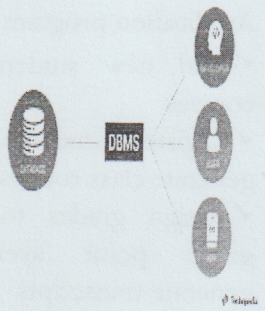
Types of Database

- **Databases can be classified according to:-**
  - ✓Number of users
  - ✓Database location(s)
  - ✓Expected type and extent of use
- **Single-user database supports only one user at a time**
  - ✓Desktop database - single-user, runs on PC
- **Multiuser database supports multiple users at the same time**
  - ✓Workgroup database supports a small number
  - ✓Enterprise database supports a large number

Database Management System (DBMS)

A DBMS is a collection of programs provided by the vendor, which enable the data to be accessed, filtered and generally processed efficiently, in order to yield useful information for the user.

- Database Management System applications like **MySql,** Oracle, etc. are immensely used in businesses to model and manage business objects within corporate databases.
- They provide lots of advantages to enable organizations to keep their business records secured, consistent and relevant.



Database Management System (DBMS)

A modern DBMS has the following characteristics:

1. Real-world entity

A modern DBMS is more realistic and uses real-world entities to design its architecture. It uses the behavior and attributes too. For example, a school database may use students as an entity and their age as an attribute.

2. Relation-based tables

DBMS allows entities and relations among them to form tables. A user can understand the architecture of a database just by looking at the table names.

3. Isolation of data and application

A database system is entirely different than its data. A database is an active entity, whereas data is said to be passive, on which the database works and organizes. DBMS also stores metadata, which is data about data, to ease its own process.

4. Less redundancy

DBMS follows the rules of normalization, which splits a relation when any of its attributes is having redundancy in values. Normalization is a mathematically rich and scientific process that reduces data redundancy.

5. Consistency

Consistency is a state where every relation in a database remains consistent. There exist methods and techniques, which can detect attempt of leaving database in inconsistent state. A DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems.

6. Multiuser and Concurrent Access

DBMS supports multi-user environment and allows them to access and manipulate data in parallel. Though there are restrictions on transactions when users attempt to handle the same data item, but users are always unaware of them.

7. Multiple views

DBMS offers multiple views for different users. A user who is in the Sales department will have a different view of database than a person working in the Production department. This feature enables the users to have a concentrate view of the database according to their requirements.

8. Security

Features like multiple views offer security to some extent where users are unable to access data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage.

Advantages of a DBMS

1- Database Development:

It allows organizations to place control of database development in the hands of database administrators (DBAs) and other specialists.

2-Data independence:

Application programs should be as independent as possible from details of data representation and storage.

3-Efficient data access:

A DBMS utilizes a variety of sophisticated techniques to store and retrieve data efficiently. It allows different user application programs to easily access the same database.

4-Crash recovery:

The DBMS protects users from the effects of system failures

5- Data integrity and security:

If data is always accessed through the DBMS, the DBMS can enforce the integrity constraints on the data. For example, before inserting salary information for an employee, the DBMS can check that the department budget is not exceeded.

6- Data administration and Concurrent access:

When several users share the data(more than one user access the database at the same time), DBMS schedules concurrent accesses to the data in such a manner that users can think of the data as being accessed by only one user at a time.

Objectives

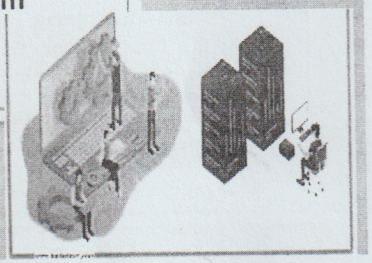
- What a database is.
- What the various types of databases are.
- Types of Database
- The differences between data and information .
- Why they are valuable assets for decision making.
- The importance of database design.

Database system

Theoretical lectures

The second lecture

Assistant teacher Shaymaa Abdulkarem



مكتبة الأمين

DBMS – Architecture

The design of a DBMS depends on its architecture. It can be centralized or decentralized or hierarchical. Also, it can be seen as either single tier or multi-tier. An n-tier architecture divides the whole system into related but independent n modules, which can be independently modified, altered, changed, or replaced.

A) 1-tier architecture

The DBMS is the only entity where the user directly sits on the DBMS and uses it. Any changes done here will directly be done on the DBMS itself. It does not provide handy tools for end-users. Database designers and programmers normally prefer to use single-tier architecture.

B) 2-tier architecture

The DBMS can be accessed. Programmers use 2-tier architecture where they access the DBMS by means of an application. Here the application tier is entirely independent of the database in terms of operation, design, and programming.

C) 3-tier Architecture

A 3-tier architecture separates its tiers from each other based on the complexity of the users and how they use the data present in the database. It is the most widely used architecture to design a DBMS.

DBMS – Architecture

1. Database (Data) Tier

At this tier, the database resides along with its query processing languages. We also have the relations that define the data and their constraints at this level.

2. Application (Middle) Tier

At this tier reside the application server and the programs that access the database. For a user, this application tier presents an abstracted view of the database. End-users are unaware of any existence of the database beyond the application. At the other end, the database tier is not aware of any other user beyond the application tier. Hence, the application layer sits in the middle and acts as a mediator between the end-user and the database.

3. User (Presentation) Tier

End-users operate on this tier, and they know nothing about any existence of the database beyond this layer. At this layer, multiple views of the database can be provided by the application. All views are generated by applications that reside in the application tier.

Multiple-tier database architecture is highly modifiable, as almost all its components are independent and can be changed independently.

Relational Model (RM)

- The relational model (RM) for database management is an approach to managing data using a structure and language consistent with first-order predicate logic, first described in 1969 by Edgar F. Codd .
- In the relational model of a database, all data is represented in terms of tuples, grouped into relations.
- A database organized in terms of the relational model is a relational database.
- The purpose of the relational model is to provide a declarative method for specifying data and queries: users directly state what information the database contains and what information they want from it, and let the database management system software take care of describing data structures for storing the data and retrieval procedures for answering queries.
- Relational data model is the primary data model, which is used widely around the world for data storage and processing .
- This model is simple and it has all the properties and capabilities required to process data with storage efficiency.

Relational Model (RM) Concepts

Tables :- In relational data model, relations are saved in the format of Tables. This format stores the relation among entities. A table has rows and columns, where rows represents records and columns represent the attributes.

Tuple :- A single row of a table, which contains a single record for that relation is called a tuple.

Relation instance :- A finite set of tuples in the relational database system represents relation instance. Relation instances do not have duplicate tuples.

Relation schema :- A relation schema describes the relation name (table name), attributes, and their names.

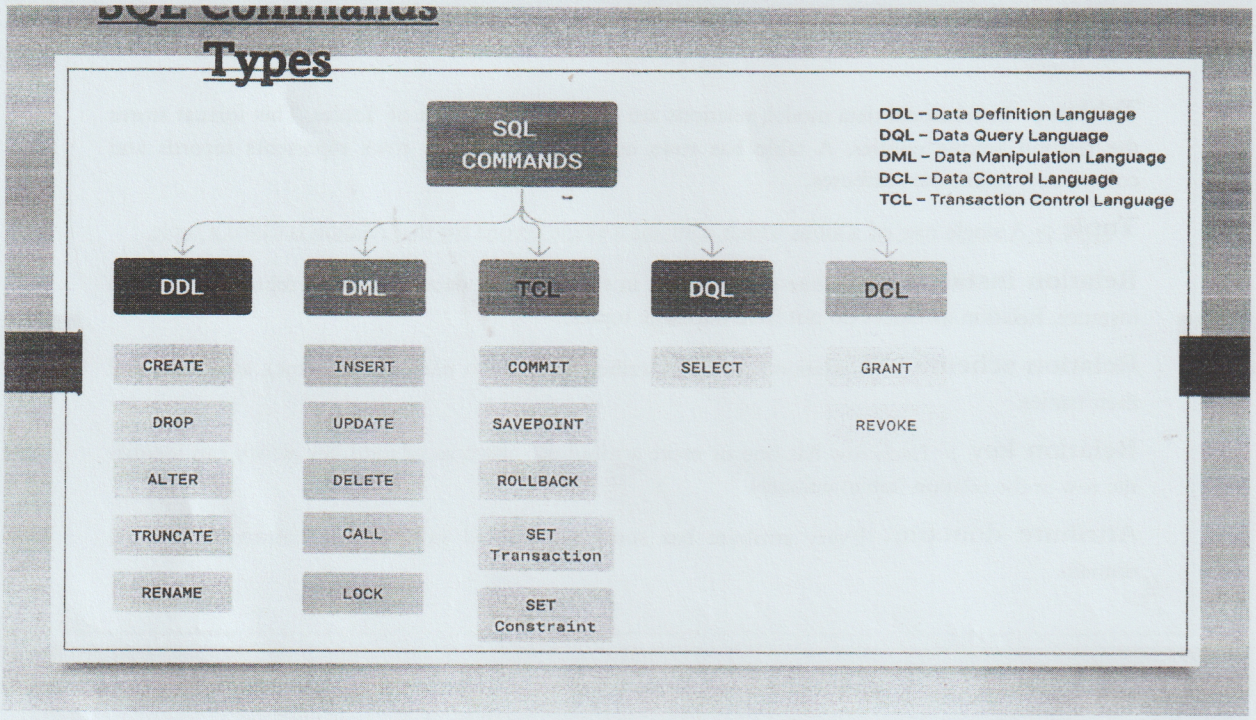
Relation key :- Each row has one or more attributes, known as relation key, which can identify the row in the relation (table) uniquely.

Attribute domain:- Every attribute has some pre-defined value scope, known as attribute domain.

SQL Commands Types

The standard SQL commands to interact with relational databases are CREATE, SELECT, INSERT, UPDATE, DELETE and DROP. These commands can be classified into groups based on their nature:

1. **DDL -Data Definition Language**
2. **DML -Data Manipulation Language**
3. **DCL -Data Control Language**
4. **TCL Transaction Control Language**



SQL Commands Types

1. **DDL -Data Definition Language**

Command	Description
CREATE	Creates a new table, a view of a table, or other object in database
ALTER	Modifies an existing database object, such as a table.
DROP	Deletes an entire table, a view of a table or other object in the database
RENAME	allows you to rename an object
TRUNCATE	used to delete all rows from a table, effectively emptying it.

2. DML -Data Manipulation Language

Command	Description
SELECT	Retrieves certain records from one or more tables
INSERT	Creates a record
UPDATE	Modifies records
DELETE	Deletes records

3. DCL -Data Control Language

Command	Description
GRANT	Gives a privilege to user
REVOKE	Takes back privileges granted from user

4. TCL -Transaction Control Language

Command	Description
COMMIT	Saves the changes made within a transaction.
ROLLBACK	ROLLBACK: Undoes the changes made within a transaction.
SAVEPOINT	SAVEPOINT: Sets a save point within a transaction.