

Database

- The database approach separates data and program structure
- If the new data structures are added or existing structures are modified then the application programs are unaffected, provided they do not directly depend upon what has been modified.
 - If we add a new field or record or create a new file, existing applications are unaffected.
 - If we remove a field from a file that an application program uses, then that application program is affected by this change and must be modified accordingly.

Some Definitions

- **entity**
- **attribute**
- **relationship**

Logically Related data???

- The database represents the entities, the attributes, and the logical relationships between the entities. That is, the database holds data that is logically related.

Database Management System

- **DEFN: The Database Management System (DBMS) is a software system** that enables users to define, create, maintain, and control access to the database.
- The DBMS is the software that interacts using DDL and DML with the users' application programs and the database.

Languages for DBMS

- DDL

- DML

DBMS Functions

- Provides data Independence
- Concurrency Control
- Provides Recovery services
- Provides a clear view of the process that manipulates data.

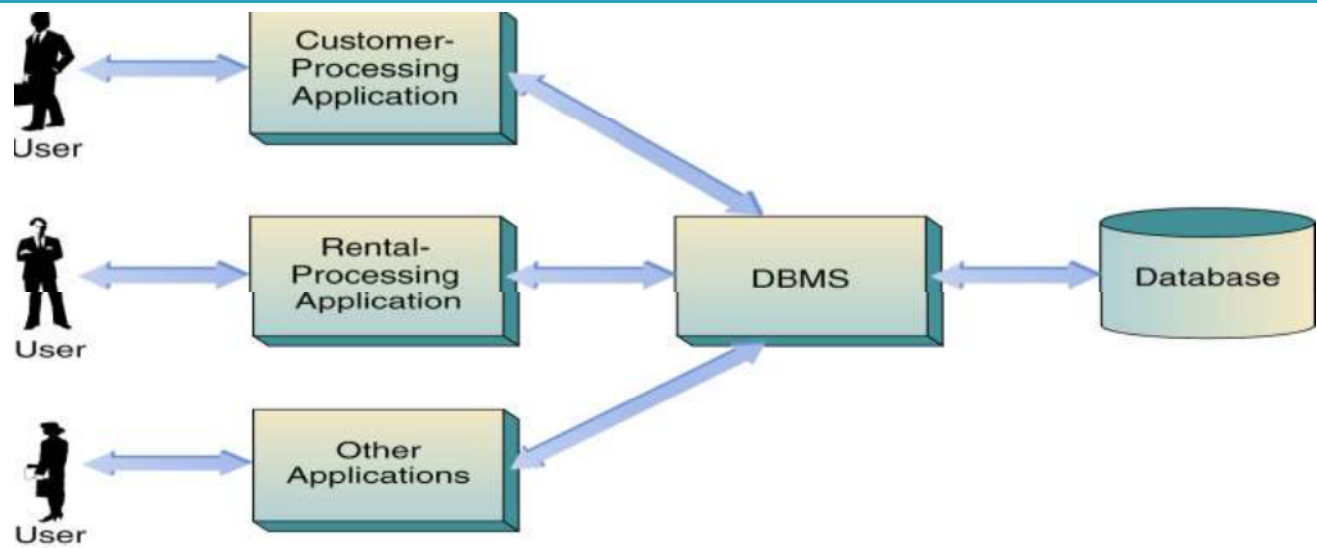
DBMS Functions cont.

- It provides access to the database.
 - ▣ A security system
 - ▣ An integrity system
 - ▣ A concurrency system
 - ▣ A recovery
 - ▣ A user-accessible catalogue

DBMS

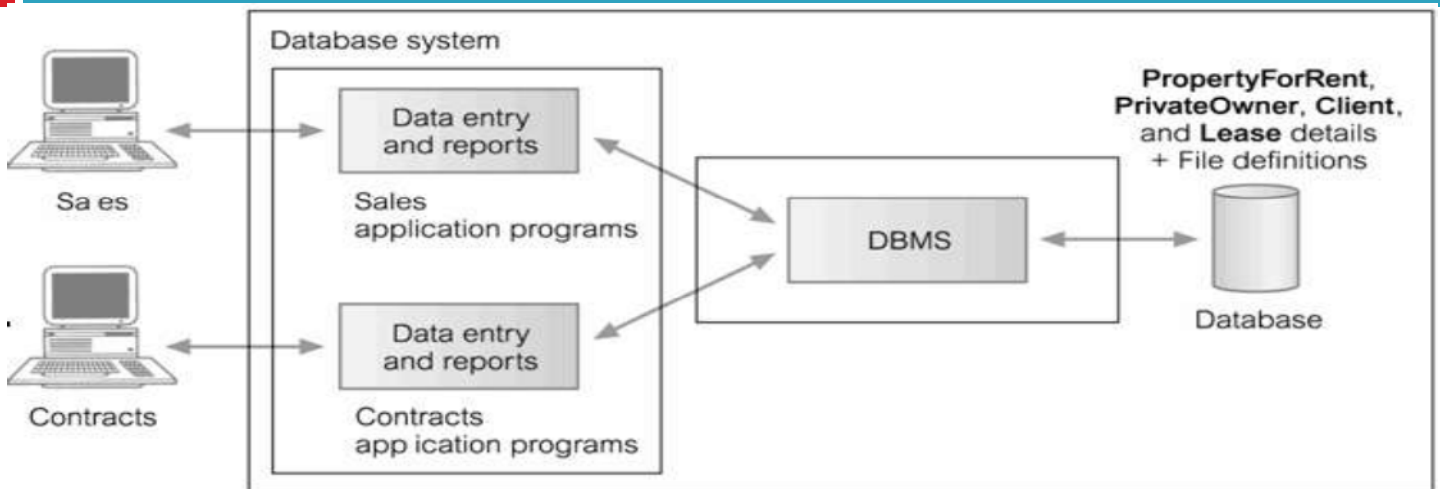
- Supports efficient access to very large amounts of data.
- Supports concurrent access to very large amounts of data.
 - Example: bank and its ATM machines (With Transparency)

DBMS



Application Program

- **DEFN: The Application Program is a computer** program that interacts with the database by issuing an appropriate request (typically an **SQL statement**) to the DBMS.
- Users interact with the database through a number of **application programs that are used** to create and maintain the database and to generate information. The application programs may be written in some programming language.



PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)

PrivateOwner (ownerNo, fName, lName, address, telNo)

Client (clientNo, fName, lName, address, telNo, prefType, maxRent)

Lease (leaseNo, propertyNo, clientNo, paymentMethod, deposit, paid, rentStart, rentFinish)

DBS Environment



- Machine – Bridge – Human

Hardware

- Needed for DBMS and the applications to run on.
 - ▣ Can range from a single personal computer, to a single mainframe, to a network of computers.
 - ▣ Depends on the organization's requirements and the DBMS used.

Software

- Made up of DBMS software and the application programs, together with operating system, including network software if the DBMS is being used over a network (CL)
- Application programs may be written using a SQL embedded in programming layer

Data

- From the end-
important component of the DBMS environment.
- Bridge between the machine and human components.
- Operational data and the metadata
- The structure of the database is called the **schema**.

Procedures

- Procedures may consist of instructions on how to:
 - ▣ log on to the DBMS;
 - ▣ use a particular DBMS facility or application program;
 - ▣ start and stop the DBMS;
 - ▣ change the structure of a table.

People

- There are four distinct types of people:
 - ▣ Data and database administrators,
 - ▣ Database designers,
 - ▣ Application developers,
 - ▣ End-users.

Data and Database Administrators

- They deal with the management and control of a DBMS and its data.
- The Data Administrator (DA) is responsible for the management of the data resource including
 - ▣ database planning,
 - ▣ development and maintenance of standards,
 - ▣ policies and procedures,
 - ▣ and conceptual/logical database design.
- This person consults with and advises senior managers so that the database development will support corporate objectives.

DBA

- The Database Administrator (DBA) is responsible for the
 - ▣ physical realization of the database,
 - ▣ including physical database design and implementation,
 - ▣ security and integrity control, maintenance of the operational system,
 - ▣ ensuring satisfactory performance of the application for users.
- Some organizations have one person performing both roles.

Database Designers

- Large database -- two types of designer
- Logical database designer:
 - identifies the data (the entities and attributes), the relationships, and the constraints on the data
- Physical database designer:
 - decides how the logical database design is to be physically realized.

Application Programmers

- People who implement the application programs that provide the required functionality for the end-users.
- Usually application developers work from a specification produced by systems analysts.
- Each program contains statements that request the DBMS to perform some operations on the database including retrieving data, inserting, updating, and deleting data.

End Users

- End-
has been designed and implemented, and is being maintained to serve their information needs.
- End-users can be classified according to the way they use the database system:
- **Naïve users**
- **Sophisticated users**

Advantages of DBMSs



Disadvantages of DBMSs

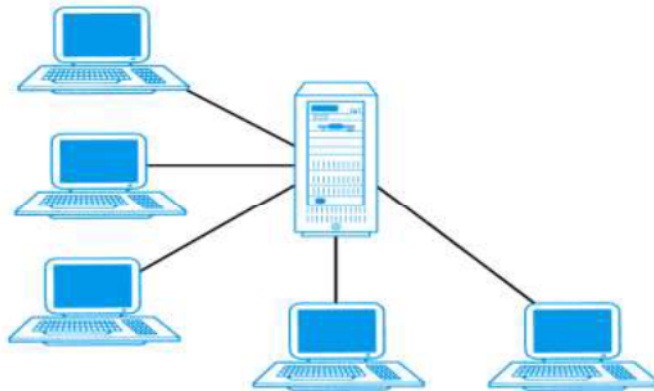


Multi-User DBMS Architectures

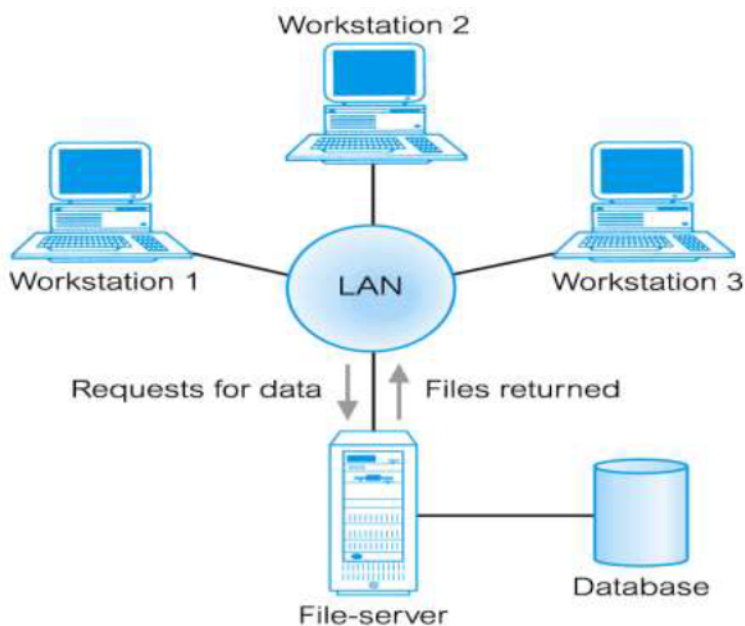
- ❑ **Teleprocessing**
- ❑ **File-server**
- ❑ **Client-server**

Teleprocessing

- ❑ **Traditional architecture.**
- ❑ **Single mainframe with a number of terminals attached.**
- ❑ **Trend is now towards downsizing.**



File-Server Architecture



connected to several workstations

Database resides on file-server

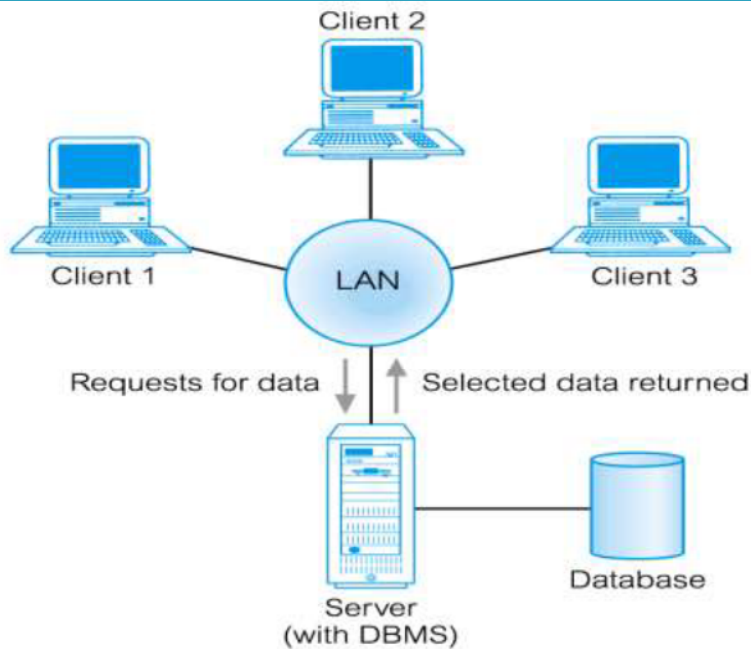
DBMS and applications run on each workstation

Significant network traffic.

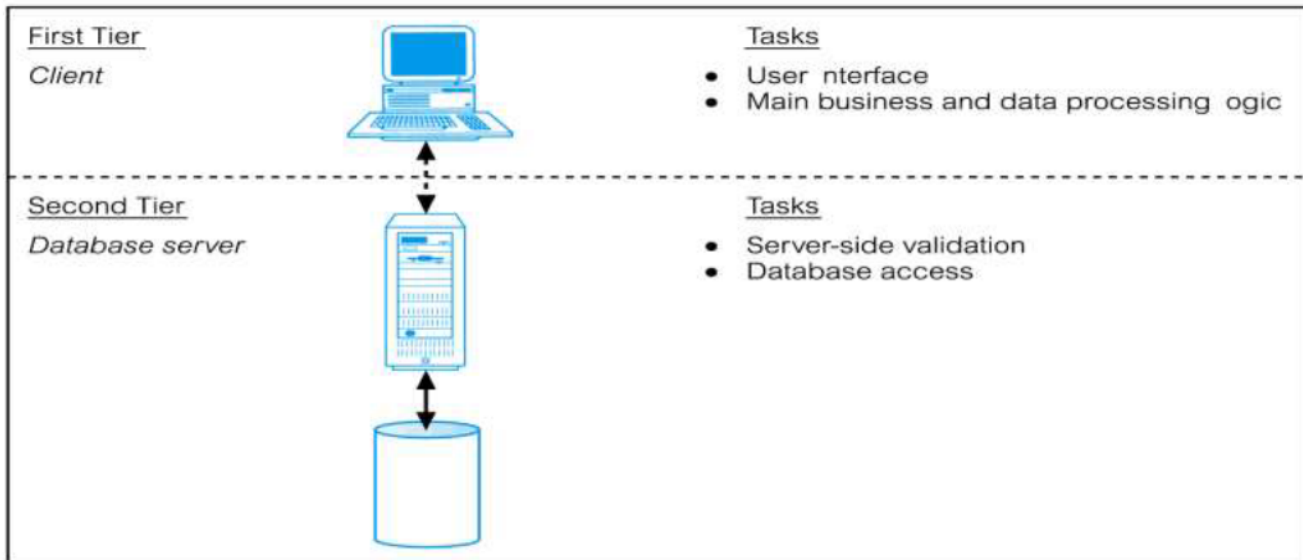
Traditional Two-Tier Client-Server

- ❑ **Client (tier 1) manages user interface and runs applications.**
- ❑ **Server (tier 2) holds database and DBMS.**
- ❑ **Advantages include:**
 - ❑ wider access to existing databases;
 - ❑ increased performance;
 - ❑ possible reduction in hardware costs;
 - ❑ reduction in communication costs;
 - ❑ increased consistency.

Traditional Two-Tier Client-Server



Traditional Two-Tier Client-Server



Three-Tier Client-Server

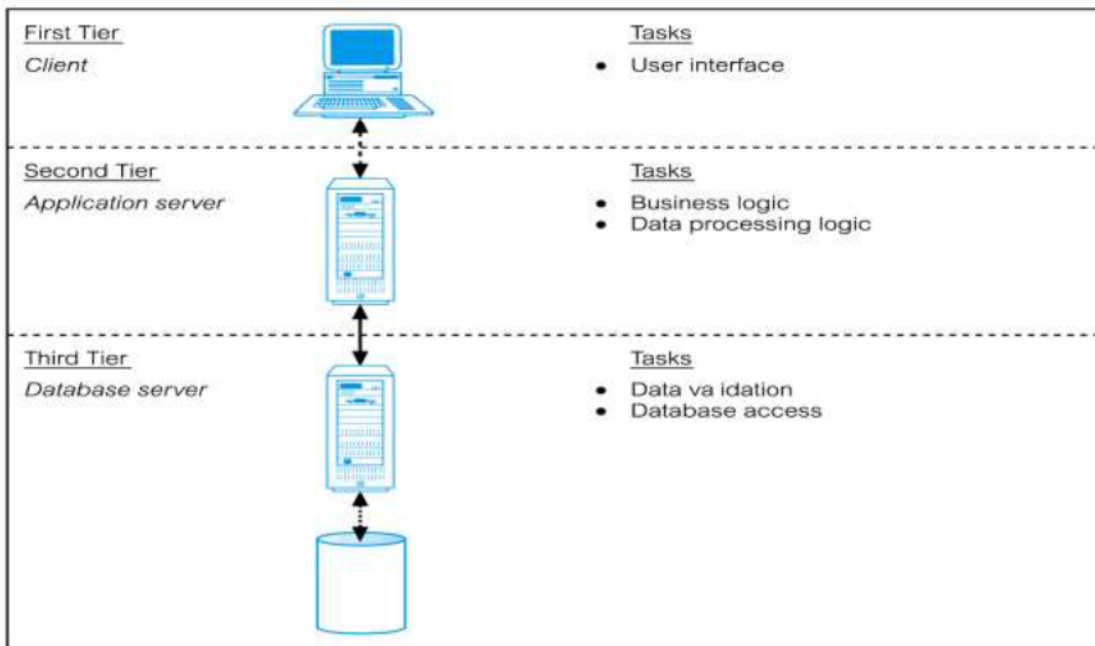
- ❑ **Client side presented a problem**
Significant client side administration overhead.
- ❑ **By 1995, three layers proposed, each potentially running on a different platform.**

Three-Tier Client-Server

□ Advantages:

- ▣ Client end requiring less expensive hardware.
- ▣ Application maintenance centralized.
- ▣ Easier to modify or replace one tier without affecting others.
- ▣ Separating business logic from database.
- ▣ Maps quite naturally to Web environment.

Three-Tier Client-Server



Database Models

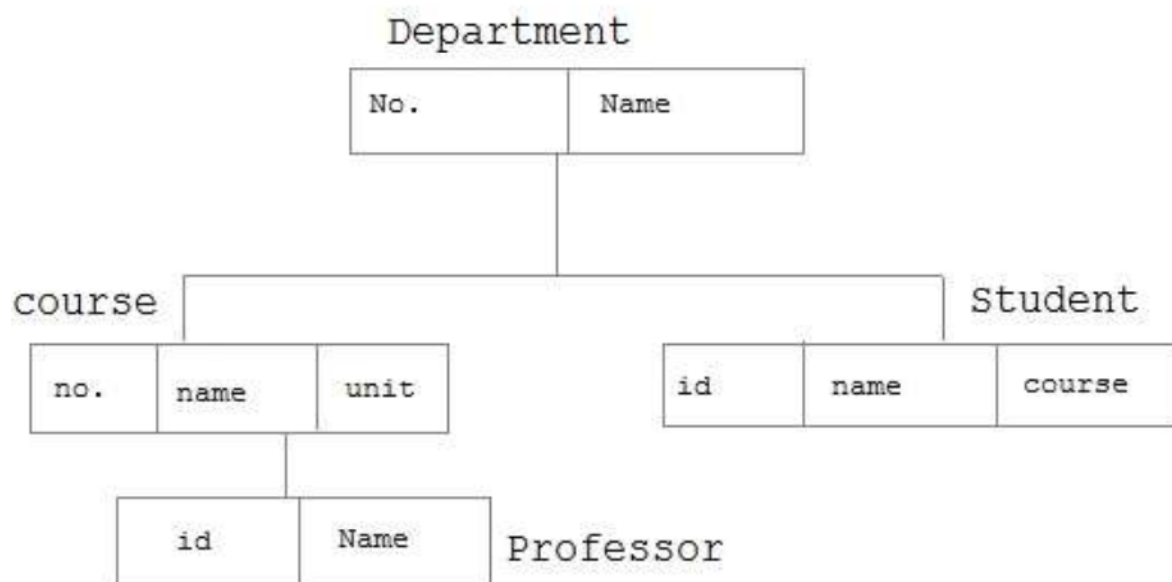
Logical design of data

- Hierarchical Model
- Network Model
- Relational Model

Hierarchical Model

- Entity has only one parent and can have several children .
- At the top of hierarchy there is only one entity called Root.

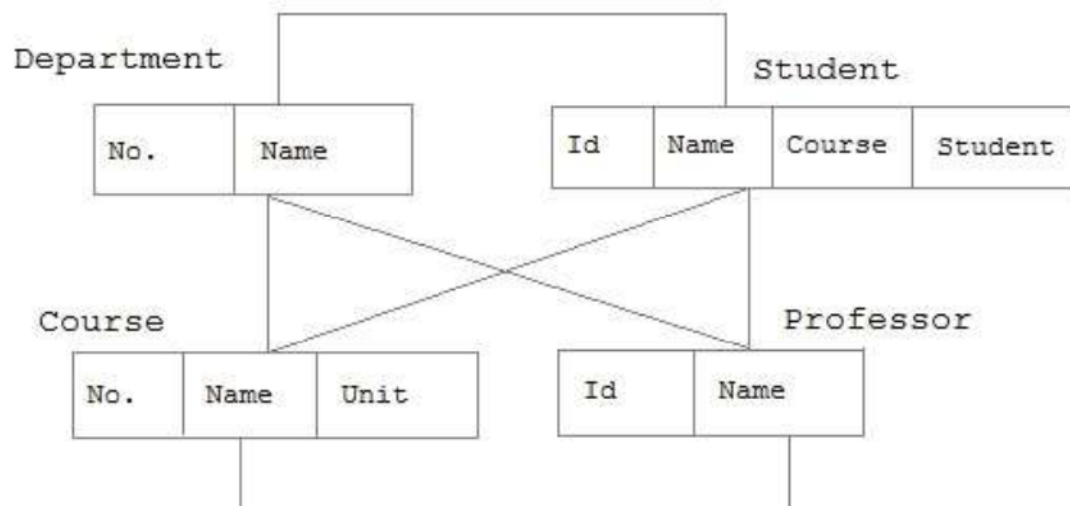
Hierarchical Model



Network Model

- In the network model entities can be accessed through several path

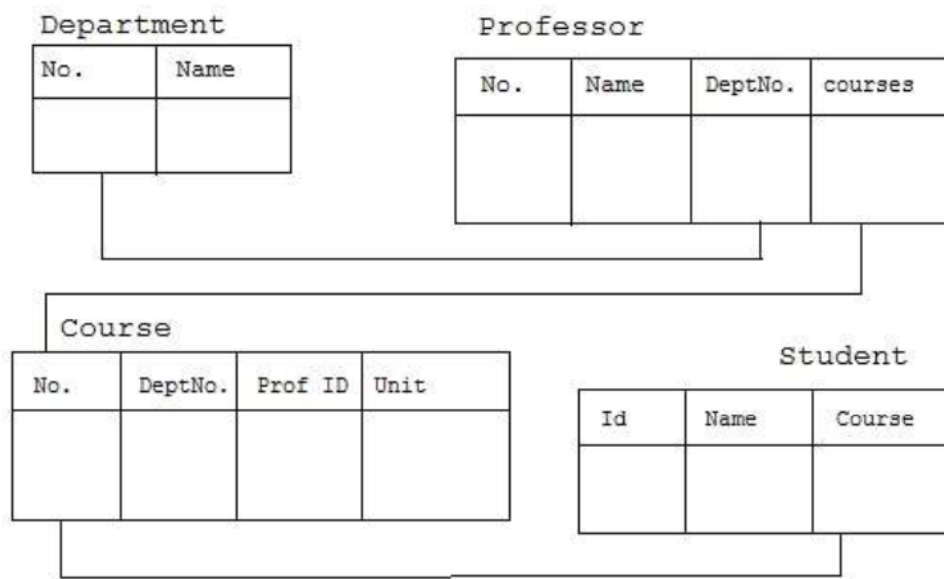
Network Model



Relational Model

- In this model, data is organized in two-dimensional tables called relations related to each other.

Relational Model



RDBMS Concepts

- A Relational Database management System (RDBMS) based on relational model introduced by E.F Codd. In relational model, data is represented in terms of tuples (rows).
- RDBMS is used to manage Relational database, a collection of organized set of tables from which data can be accessed easily.

- Table: a collection of data elements organized in terms of rows and columns.

The diagram illustrates a table with four columns: ID, Name, Age, and Salary. The first four rows of data are shown. Annotations include: a blue line across the top header row labeled 'Fields'; a green bracket on the left side of the data rows labeled 'Column'; a red bracket on the right side of the data rows labeled 'Records'; a blue arrow pointing from the 'Fields' label to the 'Name' column; and a green arrow pointing from the 'Column' label to the 'Name' column.

ID	Name	Age	Salary
1	Adam	34	13000
2	Alex	28	15000
3	Stuart	20	18000
4	Ross	42	19020


Database Keys

- Keys are very important part of Relational database. They are used to establish and identify relation between tables. They also ensure that each record within a table can be uniquely identified by combination of one or more fields within a table.

Primary Key

- is most appropriate to become main key of the table. It is a key that uniquely identify each record in a table.

Primary Key



s_id	S_name	age	course	address

Composite Key

- Key that consist of two or more attributes that uniquely identify an entity occurrence is called Composite key.

Composite Key

