What is Database Management System(DBMS)?

- A software which manages and facilitates access to the data
- It offers an organized way to store, retrieve, and manage data efficiently.

Database Models:

1. Hierarchical Model
2. Network Model
3. Relational Model:

- Data is organized into tables(relations) consisting of rows and columns
- Columns -> Also called attributes or fields
- Rows -> Also called as tuples or records
- Most widely used

Database Technologies:

1. Files

- Storing and retrieving files is tedious and less efficient
- Poor security, difficulty in sharing

2. Databases

- Offers a faster and more efficient way to store and manipulate data.
- Enables data sharing and provides security
- It follows ACID properties.

ACID properties: RnD*

1. Atomicity:
    - A transaction is treated as a single unit; either all steps are completed or none are.
    - Ensures no partial changes are occurring.
2. Consistency:
    - Ensures the database remains in a valid state before and after the transaction.
3. Isolation:
    - Transactions are executed independently without any interference, ensuring intermediate states are not visible to others.
4. Durability:
    - Once a transaction is commited, its changes are permanent even in case of system failure.

- SQL data types

- The type of data a column can hold in db table

1. String Datatypes:
    - VARCHAR(SIZE)- a variable-length string. Maximum size needs to be specified.
      EX. -> Username VARCHAR(50)
      It can store upto 50 characters

    - CHAR(SIZE) - A fixed length string. Pads with spaces if the input is shorter than specified size.
      EX. -> Code CHAR(10)
      It always stores exactly 10 characters.
    - TEXT
    - BINARY(SIZE)
    - BLOB (Binary Large Object)(Ex, images, audios)
    - MEDIUMTEXT*
    - LONGTEXT

2. Numeric Datatypes

- BIT(SIZE): stores the binary data. BIT(1) can store a value of 0 or 1.
- INT(size): Integer numbers.The size species the display width.
  Ex. Balance INT(10);
- BIGINT: for storing larger integer values
- FLOAT(size,d): Approximate numeric values. Size is the total number of digits, d is the number of digits after decimal
  Ex. Value FLOAT(5,2)
- DOUBLE(size, d): Similar to FLOAT but it stores the larger values
- DECIMAL*
- BOOLEAN*

3. Date and Time Datatypes:

- DATE: Store the date in `YYYY-MM-DD` format.
  Ex. EventDate DATE;
- DATETIME: Stores the date and time in `YYYY-MM-DD HH:MM:SS` format
- TIME: stores time in `HH:MM:SS` format

Types of SQL Commands:

1. DDL (Data Definition Language)
2. DQL (Data Query Language)
3. DML (Data Manipulation Language)
4. DCL (Data Control Language)
5. TCL (Transaction Control Language)

1. DDL (Data Definition Language)

CREATE, ALTER, DROP, TRUNCATE

These are the Commands which are used to define or modify database structures.

- CREATE: Creates a table or database
  Ex. CREATE table Students(ID INT, Name VARCHAR(50));
- ALTER: To modify the existing table
  Ex. ALTER table Students ADD Age INT;
- TRUNCATE: Deletes all the records from table but keeps the table structure.
  Ex. Truncate TABLE Students;
- DROP: Deletes a table or database
  Ex. DROP TABLE Students;

2. DQL (Data Query Language)

- SELECT: Retrieves data from a database.

Ex. SELECT * from patient;

3. DML (Data Manipulation Language)

To manipulate data

- INSERT: Adds a new record
  Ex. INSERT INTO Employees(Name, Age) VALUES (‘Krishna’ , 30);
- UPDATE: Modifies the existing data
  Ex. UPDATE Employees SET Age = 31 WHERE Name = ‘Krishna’ ;
- DELETE: Removes records
  Ex. DELETE FROM EMPLOYEES WHERE AGE < 25;

4. DCL (Data Control Language)

Commands to control access

GRANT *

REVOKE*

5. TCL (Transaction Control Language)*
- COMMIT
- ROLLBACK
- SAVEPOINT

CONSTRAINTS:

1. NOT NULL

   It ensures a column cannot have a null value.

   Ex.
   ```
   CREATE TABLE Employees(
   ID INT NOT NULL,
   Name VARCHAR(50) NOT NULL
   );
   ```

2. Primary Key

   It uniquely identifies the rows in a table.
   ```
   CREATE TABLE Departments (
   DeptID INT PRIMARY KEY,
   DeptName VARCHAR(50)
   );
   ```

3. Foreign Key

   It references the primary key in another table

   Ex.
   ```
   CREATE TABLE Employees(
        EmpId INT PRIMARY KEY,
        DeptID INT,
        FOREIGN KEY (DeptID) REFERENCES Departments(DeptID)
   );
   ```

4. CHECK

   To enforce a condition on column values

```
Ex.
Create Table Products(
ProductID INT PRIMARY KEY,
Price DECIMAL(10,2) CHECK (Price > 0)
);
```

5. DEFAULT

- Assigns a default value to a column

```
Ex.
CREATE TABEL Orders(
OrderID Int Primary Key,
OrderDate DATE DEFAULT CURRENT_DATE
);
```

Operators:

Arithmetic Operators: +, -, *,/

Ex. SELECT Salary*12 AS AnnualSalary FROM Emppployees;

Relational Operators: =,!=,>,<

Ex. SELECT * from Employees WHERE Salary > 5000;

Logical Operators: AND, OR, NOT

Ex. Select * from Employees WHERE AGE > 25 AND Salary > 5000;

1. Creating our first database

```
CREATE DATABASE Flynaut;
DROP DATABASE flynaut; // To delete database
```

2. Go inside the flynaut database

```
Use flynaut;
```

3. Creating the table

```
CREATE TABLE EMP (
EMPNO INT(4) NOT NULL PRIMARY KEY,
ENAME VARCHAR(20),
JOB VARCHAR(20),
MGR INT(4),
HIREDATE DATE,
SAL DECIMAL(7,2),
COMM DECIMAL(7,2),
DEPTNO INT(4)
);
```

4. SHOW TABLES; -> TO CHECK TABLE IS CREATED OR NOT
5. DESC EMP; -> TO SEE THE DESCRIPTION OF TABLE
6. Inserting values into table

```
INSERT INTO EMP VALUES
        ( 7369, 'SMITH', 'CLERK', 7902,
           STR_TO_DATE('17-DEC-1980','%d-%b-%Y'),800,NULL, 20);
INSERT INTO EMP VALUES
        (7369, 'SMITH',  'CLERK',     7902,
        STR_TO_DATE('17-DEC-1980', '%d-%b-%Y'),  800, NULL, 20);
INSERT INTO EMP VALUES
        (7499, 'ALLEN',  'SALESMAN',  7698,
        STR_TO_DATE('20-FEB-1981', '%d-%b-%Y'), 1600,  300, 30);
INSERT INTO EMP VALUES
        (7521, 'WARD',   'SALESMAN',  7698,
        STR_TO_DATE('22-FEB-1981', '%d-%b-%Y'), 1250,  500, 30);
INSERT INTO EMP VALUES
        (7566, 'JONES',  'MANAGER',   7839,
        STR_TO_DATE('2-APR-1981', '%d-%b-%Y'),  2975, NULL, 20);
INSERT INTO EMP VALUES
        (7654, 'MARTIN', 'SALESMAN',  7698,
        STR_TO_DATE('28-SEP-1981', '%d-%b-%Y'), 1250, 1400, 30);
INSERT INTO EMP VALUES
        (7698, 'BLAKE',  'MANAGER',   7839,
        STR_TO_DATE('1-MAY-1981', '%d-%b-%Y'),  2850, NULL, 30);
```

```
INSERT INTO EMP VALUES
        (7782, 'CLARK',  'MANAGER',   7839,
        STR_TO_DATE('9-JUN-1981', '%d-%b-%Y'),  2450, NULL, 10);
INSERT INTO EMP VALUES
        (7788, 'SCOTT',  'ANALYST',   7566,
        STR_TO_DATE('09-DEC-1982', '%d-%b-%Y'), 3000, NULL, 20);
INSERT INTO EMP VALUES
        (7839, 'KING',   'PRESIDENT', NULL,
        STR_TO_DATE('17-NOV-1981', '%d-%b-%Y'), 5000, NULL, 10);
INSERT INTO EMP VALUES
        (7844, 'TURNER', 'SALESMAN',  7698,
        STR_TO_DATE('8-SEP-1981', '%d-%b-%Y'),  1500,    0, 30);
INSERT INTO EMP VALUES
        (7876, 'ADAMS',  'CLERK',     7788,
        STR_TO_DATE('12-JAN-1983', '%d-%b-%Y'), 1100, NULL, 20);
INSERT INTO EMP VALUES
        (7900, 'JAMES',  'CLERK',     7698,
        STR_TO_DATE('3-DEC-1981', '%d-%b-%Y'),   950, NULL, 30);
INSERT INTO EMP VALUES
        (7902, 'FORD',   'ANALYST',   7566,
        STR_TO_DATE('3-DEC-1981', '%d-%b-%Y'),  3000, NULL, 20);
INSERT INTO EMP VALUES
        (7934, 'MILLER', 'CLERK',     7782,
        STR_TO_DATE('23-JAN-1982', '%d-%b-%Y'), 1300, NULL, 10);
```

```
mysql> SELECT * FROM EMP;
+-------+--------+-----------+------+------------+---------+---------+--------+
| EMPNO | ENAME  | JOB       | MGR  | HIREDATE   | SAL     | COMM    | DEPTNO |
+-------+--------+-----------+------+------------+---------+---------+--------+
|  7369 | SMITH  | CLERK     | 7902 | 1980-12-17 |  800.00 |    NULL |     20 |
|  7499 | ALLEN  | SALESMAN  | 7698 | 1981-02-20 | 1600.00 |  300.00 |     30 |
|  7521 | WARD   | SALESMAN  | 7698 | 1981-02-22 | 1250.00 |  500.00 |     30 |
|  7566 | JONES  | MANAGER   | 7839 | 1981-04-02 | 2975.00 |    NULL |     20 |
|  7654 | MARTIN | SALESMAN  | 7698 | 1981-09-28 | 1250.00 | 1400.00 |     30 |
|  7698 | BLAKE  | MANAGER   | 7839 | 1981-05-01 | 2850.00 |    NULL |     30 |
|  7782 | CLARK  | MANAGER   | 7839 | 1981-06-09 | 2450.00 |    NULL |     10 |
|  7788 | SCOTT  | ANALYST   | 7566 | 1982-12-09 | 3000.00 |    NULL |     20 |
|  7839 | KING   | PRESIDENT | NULL | 1981-11-17 | 5000.00 |    NULL |     10 |
|  7844 | TURNER | SALESMAN  | 7698 | 1981-09-08 | 1500.00 |    0.00 |     30 |
|  7876 | ADAMS  | CLERK     | 7788 | 1983-01-12 | 1100.00 |    NULL |     20 |
|  7900 | JAMES  | CLERK     | 7698 | 1981-12-03 |  950.00 |    NULL |     30 |
|  7902 | FORD   | ANALYST   | 7566 | 1981-12-03 | 3000.00 |    NULL |     20 |
|  7934 | MILLER | CLERK     | 7782 | 1982-01-23 | 1300.00 |    NULL |     10 |
+-------+--------+-----------+------+------------+---------+---------+--------+
14 rows in set (0.00 sec)

mysql> SELECT ENAME,SAL FROM EMP;
+--------+---------+
| ENAME  | SAL     |
+--------+---------+
| SMITH  |  800.00 |
| ALLEN  | 1600.00 |
| WARD   | 1250.00 |
| JONES  | 2975.00 |
| MARTIN | 1250.00 |
| BLAKE  | 2850.00 |
| CLARK  | 2450.00 |
| SCOTT  | 3000.00 |
| KING   | 5000.00 |
| TURNER | 1500.00 |
| ADAMS  | 1100.00 |
| JAMES  |  950.00 |
| FORD   | 3000.00 |
| MILLER | 1300.00 |
+--------+---------+
```

Where Clause:

Filters records based on the condition

1. Retrieve employees with the job title as 'Manager'.

➔ SELECT EName, Job FROM Emp where Job = 'Manager';

2. Select the employee names, job titles and salaries for those who is working in department no. 30.

➔ SELECT ename, job, sal from emp where deptno = 30;

Assignment:
Q - Select the employees number, name and salary from emp table.

Q - Select the employees names and salaries for those whose job is 'SALESMAN'.

Q - Select the employees names, job titles and salaries fo rthose who is working in department no. 30.

Q - Select the employee names and job titles where the job is either manager or clerk.

Select the employee names and commission where the commission is not null.

Select the employee names and job titles where the job is 'MANAGER'

Select the employee names and salaries where the salary is greater than 1500.

Select the employee names and hire dates for employees hired between April 1, 1981, and May 1, 1981.

Select the employee names and salaries where commission is not null (i.e., employees who receive a commission).

Select the employee names where the name starts with the letter 'S'.

Select the employee names and job titles where the job is either 'MANAGER' or 'CLERK'.

Select the employee names where the salary is not equal to 5000.

Select the employee names and hire dates where the hire date is greater than or equal to December 1, 1981.

Select the employee names and salaries where the salary is between 1000 and 3000.

Select the employee names, job titles, and salaries where the job is 'SALESMAN' and the salary is greater than 1200.