

Database

To solve the problems of file based database system database is introduced.

- Collection of inter-related data.
- ⇒ Solve redundancy, inconsistency, concurrency, security problem.
- ⇒ Database can be integrated with application (web, Mob)

File Based System

Data

Redundancy - having

multiple copy of same data - Memory wastage.

Data Inconsistency

multiple copy of some data do not match with each other.

Difficult to Access

Security

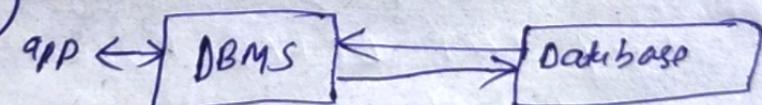
Concurrency

Example

College (same data)

Hostel	Mess	Classes	Registration
Redundant - (Sachin)	(Sachin)	(Sachin)	(Sachin)
Inconsistent - (Roll- 123)	(Roll-123)	(Roll-456)	(Roll-678)

DBMS (Database)



- works as a interface for fast process of data. delete, update, insert, post.
- DBMS is basically a software that manage database for various faster action on data (stored in database).
- DBMS - MySQL
PgSQL
SQLite] - SQL (To deal with DBMS we need SQL)

- DBMS puts constraints over limitation on data.
- Marks of student - Int (✓)
→ (String) (X)
- Recovery and backup.

- MySQL is type of RDBMS (manage Relational database)
- Relational database stores data in the form of table.

Entity - Movies

Name	Releas	Actor	ID
XY2	2008	S	1
UBL	2010	L	2

- every table represents an entity
- and each entity ~~has~~ is real world object which has some property
- column represents property of entity

Name	Roll
XY2	1
UBL	2

- row represents individual entity

→ Entity - STUDENT
 {
 Name | Roll } property
 XY2 | 1
 UBL | 2

{
 Entity } → 2180 ↗ 910 (entity) 2180

Entity 2180 is a relation & 910 is a row
 Entity 2180 contains data - 910

Column and row are called in 2180

In Entity 2180 contains data - 910

SQl commands -

1. **SHOW DATABASES;**

// To show all the database.

2. **USE <database name>;**

Ex - USE studentRecord

// To select a particular database to work on.

3. **SHOW TABLES;**

// To show all the database tables in one database.

4. **CREATE DATABASE <database name>;**

// To create the database.

5. **SHOW TABLES;**

// To show all the tables in one database.

6. **CREATE TABLE <tablename>**

(
 <column 1> Datatype

 <column 2> Datatype
);

jj;

Exam - CREATE TABLE Student

(
 Name VARCHAR(15)

 Roll NUMERIC

 DOB DATE
);

7. `DESC <tablename>`

// To show structure and description of table.

8. `INSERT INTO <tablename>`

`(col1, col2, ...)` VALUES (`Value1`, `Value2`, ...);

// To insert the values into the table.

9. `SELECT * FROM <tablename>;`

// will show the whole table data (column data)

10. `SELECT <columnname> FROM <tablename>;`

// To show the specific column data.

Putting some constraints on column

At the time of table creation we can put some constraints on column-

`(Name NOT NULL, Gender ENUM('Male', 'Female')
, Salary INT DEFAULT 20000)`

ENUM provides some value to users and users have to choose from these values only.

11. `SELECT * FROM <tablename>`

`WHERE (Name = 'Sachin' OR salary > 20000)`

`WHERE (Name = 'Sachin' AND salary < 20000)`

// where is a clause to filter the data from table by giving specific conditions.

12. `INSERT INTO <tablename> (col1, col2, col3, ...) VALUES (v1, v2, v3), (v1', v2', v3'), (v1, v2, v3);`

// inserting multiple rows in a table.

13. 'LIKE' keyword is used as a substring.

`Name LIKE "%-a%"` will return all column value which 'a' in last position.

`Name LIKE "%a%"` will return all column value that contains 'a'.

`SELECT * FROM <tablename> WHERE Name LIKE "%-a%"`

14. // Getting data in sorted form.

`SELECT * FROM <tablename> ORDER BY <colname>`

Ex:-

`SELECT * FROM teachers ORDER BY salary LIMIT 2`

// will give data in ascending order.

`SELECT * FROM teachers ORDER BY name DESC`

// will filter data in descending order.

15-11 To ~~update~~ the data in table `EMP`

`DELETE (. $100, $100, $100)`

`DELETE FROM <tablename> WHERE ();`

16. `DELETE FROM <tablename>;`

// To delete all the rows from table.

17. `UPDATE <tablename> SET col1 = val1, col2 = v2,
WHERE (condition);`

// To update the value in table.

18. To change the structure of table out
add new column or delete new column for these
task - `ALTER TABLE`.

19. To change the column name and datatype

~~`ALTER TABLE <tablename> <old name>`~~

`ALTER TABLE <tablename> CHANGE`

`<old name> <New Name> Datatype (new constraints)`

20. To add a new column in table

`ALTER TABLE <tablename> ADD <colname> (constraints);`

To add a new column in table after a specific col

`ALTER TABLE <tname> ADD <New Name> Datatype
(constraints) AFTER <cname>;`

21. Deleting a column from table -

ALTER TABLE <tname> DROP <colname>;

22. SELECT DOB ^{column n} AS "Date of Birth" FROM <tname>
// for Readability

23. CONCAT

SELECT CONCAT(Firstname, ' ', Lastname) AS Name
FROM users;

24. SELECT DISTINCT <colname> FROM <tname>;

// To show the distinct value in a column -

25. SELECT COUNT(*) FROM <tname>;

// To know the number of rows our data in column

26. SELECT COUNT(*) AS "Male Users" FROM <tname>
WHERE Gender = "Male";

27. SELECT SUM(<colname>) FROM <tname> WHERE -----

28. SELECT AVG(<col>) FROM <tname> WHERE clause

29. SELECT MAX(<col>) FROM <tname> WHERE clause

30. SELECT MIN(<colname>) FROM <tname>