MySQL

Data :- Raw facts and figures which are useful to an organization. We cannot take decisions on the basis of data.

Information:- Well processed data is called information. We can take decisions on the basis of information Field: Set of characters that represents specific data element.

Record: Collection of fields is called a record. A record can have fields of different data types.

File: Collection of similar types of records is called a file.

Table: Collection of rows and columns that contains useful data/information is called a table. A table generally refers to the passive entity which is kept in secondary storage device.

Relation: Relation (collection of rows and columns) generally refers to an active entity on which we can perform various operations.

Database: Collection of logically related data along with its description is termed as database. OR Computerized record keeping system is called database

Tuple: A row in a relation is called a tuple.

Attribute: A column in a relation is called an attribute. It is also termed as field or data item.

Degree: Number of attributes in a relation is called degree of a relation. Cardinality: Number of tuples in a relation is called cardinality of a relation.

Primary Key: Primary key is a key that can uniquely identifies the records/tuples in a relation. This key can never be duplicated and NULL.

Foreign Key: Foreign Key is a key that is defined as a primary key in some other relation. This key is used to enforce **referential integrity** in RDBMS.

Candidate Key: Set of all attributes which can serve as a primary key in a relation.

Alternate Key: All the candidate keys other than the primary keys of a relation are alternate keys for a relation.

Relational Algebra: Relation algebra is a set operation such as select, project, union, & Cartesian product etc.

Select operation :shows set of set of rows depending upon certain condition. Mathematically it is denoted as e.g σ rollno >10(student) -means show those rows of student table whose roll no.'s are >10.

Project Operation : shows **set of columns** as result which are specified. Mathematically it is denoted as π . e.g.π rollno,name (student) - means show only rollno and name column only.

Union Operation : Two relation are said to be union compatible if their degree and column are same. e.g

Relation A

X	Y	Z
X1	Y1	Z1
X2	Y2	Z2

Relation B

X	Y	Z
X1	Y1	Z1
X3	Y3	Z3

Resultant Relation A U B=

X	Y	Z
X1	Y1	Z1
X2	Y2	Z2
X3	Y3	Z3

Cartesian product: Cartesian Product of two relation A and B gives resultant relation whose **no. of column are sum of degrees of two relation and no. of rows are product of cardinality of two relations**.

Relation A

X	Y	Z
X1	Y1	Z1
X2	Y2	Z2

Relation B

U	V
U1	V1
U2	V2

Resultant Relation A X B=

X	Y	Z	U	V
X1	Y1	Z1	U1	V1
X1	Y1	Z1	U2	V2
X2	Y2	Z2	U1	V1
X2	Y2	Z2	U2	V2

Structured Query Language (SQL) is a non procedural language that is used to create, manipulate and process the databases(relations).

Characteristics of SQL
1. It is very easy to learn and use.
2. Large volume of databases can be handled quite easily.
3. It **is non procedural language**. It means that we do not need to specify the procedures to accomplish a task but just to give a command to perform the activity.
4. SQL can be linked to most of other high level languages that makes it first choice for the database programmers.

Processing Capabilities of SQL The following are the processing capabilities of SQL
1. **Data Definition Language (DDL**)- DDL contains commands that are used to create the tables, databases, indexes, views, sequences and synonyms etc. e.g:Create table, create view, create index, alter table etc.
2. **Data Manipulation Language (DML)** -DML contains command that can be used to manipulate the data base objects and to query the databases for information retrieval. e.g Select, Insert, Delete, Update etc.
3. **Data Control Language**(**DCL)**: This language is used for controlling the access to the data. Various commands like GRANT, REVOKE etc are available in DCL.
4. **Transaction Control Language (TCL)** -TCL include commands to control the transactions in a data base system. The commonly used commands in TCL are COMMIT, ROLLBACK etc.

Data types of SQL- Following are the most common data types of SQL.
1) NUMBER / INTEGER
2) CHAR
3) VARCHAR / VARCHAR2
4) DATE
5) LONG
6) RAW/LONG RAW
7) DECIMAL

SQL COMMANDS
DDL- Create, Alter, Drop

Create

Creating a Database-To create a database in RDBMS, create command is used.
Syntax,
create database database-name;
Example
create database Test;

CREATE TABLE Command: Create table command is used to create a table in SQL.
Syntax : CREATE TABLE tablename(column_name data_type(size), column_name2 data_type(size)….);
e.g. Create table student (rollno integer(2), name char(20), dob date);

Constraints: Constraints are the conditions that can be enforced on the attributes of a relation. The constraints come in play whenever we try to insert, delete or update a record in a relation.
They are **used to ensure integrity of a relation**, hence named as integrity constraints.
1. NOT NULL
2. UNIQUE
3. PRIMARY KEY
4. FOREIGN KEY

5. CHECK
6. DEFAULT

i. **Not Null constraint** : It ensures that the column cannot contain a NULL value.

ii. **Unique constraint** : A candidate key is a combination of one or more columns, the value of which uniquely identifies each row of a table.

iii. Primary Key : It ensures two things : (i) Unique identification of each row in the table.
(ii) No column that is part of the Primary Key constraint can contain a NULL value.

iv. **Foreign Key** : The foreign key designates a column or combination of columns as a foreign key and establishes its relationship with a primary key in different table.

Example:

```
Create table Fee (RollNo integer(2) Foreign key (Rollno) references Student (Rollno),
                  Name char(20) Not null,
                  Amount integer(4),
                  Fee_Date date);
```

v. **Check Constraint** : Sometimes we may require that values in some of the columns of our table are to be within a certain range or they must satisfy certain conditions.

```
Example: Create table Employee (EmpNo integer(4) Primary Key,
                                Name char(20) Not Null,
                                Salary integer(6,2) check (salary > 0),
                                DeptNo integer(3)
                                );
```

DDL-Alter command is used for alteration of table structures. Various uses of alter command, such as,

- to add a column to existing table
- to rename any existing column
- to change datatype of any column or to modify its size.
- alter is also used to drop a column.

To Add Column to existing Table

Using alter command we can add a column to an existing table.

```
Syntax,
alter table table-name add(column-name datatype);
e.g.
alter table Student add(address char);
```

To Add Multiple Column to existing Table-

we can even add multiple columns to an existing table.

Syntax,

alter table table-name add(column-name1 datatype1, column-name2 datatype2, column-name3 datatype3);

Example:

alter table Student add(father-name varchar(60), mother-name varchar(60), dob date);

To Add column with Default Value

alter command can add a new column to an existing table with default values.

Syntax:-

alter table table-name add(column-name1 datatype1 default data);

e.g

alter table Student add(dob date default '1-Jan-99');

To Modify an existing Column

alter command is used to modify data type of an existing column .

Syntax:-

alter table table-name modify(column-name datatype);

e.g.

alter table Student modify(address varchar(30));

To Rename a column

Using alter command you can rename an existing column.

Syntax:-

alter table table-name change old-column-name new_ column-name;

e.g.

alter table Student change address Location;

The above command will rename address column to Location.

To Drop a Column

alter command is also used to drop columns also. Syntax:-

alter table table-name drop(column-name)

e.g.

alter table Student drop column (address);

DDL - Drop command

This command completely removes a table from database. This will also destroy the table structure. Syntax,

drop table table-name

Example

drop table Student;

To drop a database,

drop database Test;

RENAME command is used to rename a table. Syntax,

rename table old-table-name to new-table-name

Example

rename table Student to Student-record;

DML Commands

SELECT - Used for making queries

INSERT - Used for adding new row or record into table

UPDATE- used for modification in existing data in a table

DELETE – used for deletion of records.

INSERT Statement To insert a new tuple(row or record) into a table is to use the insert statement

(i) To insert records into specific columns

Syntax:

insert into table_name(column_name1, column_name2…)values (value1,value2….);

e.g. INSERT INTO student (rollno,name)VALUES(101,'Rohan');

(ii) **insert records in all the columns**

insert into table_name values(value1,value2……);

e.g.INSERT INTO student (VALUES(101,'Rohan','XI',400,'Jammu');

Update command –it is used to update a row of a table. syntax,

UPDATE table-name set column-name = value where condition;
e.g.
UPDATE Student set s_name='Abhi',age=17 where s_id=103;

Delete command

It is used to delete data(record) from a table.It can also be used with condition to delete a particular row. (i) syntax:- **to Delete all Records from a Table**

DELETE from table-name;

Example

DELETE from Student;

(ii) syntax: **to Delete a particular Record from a Table**

DELETE from Student where s_id=103;

SELECT command

Select query is used to retrieve data from a tables. It is the most used SQL query. We can retrieve complete tables, or partial by mentioning conditions using WHERE clause.

(I)Syntax : display specific columns

SELECT column-name1, column-name2, column-name3, column-nameN from table-name;
Example
SELECT s_id, s_name, age from Student;

(ii) to Select all Records from Table- A special character asterisk * is used to address all the data(belonging to all columns) in a query. SELECT statement uses * character to retrieve all records from a table.

Example: SELECT * from student;

To Perform Simple Calculations using Select Query

SELECT eid, name, salary+3000 from Employee;

WHERE clause

Where clause is used to specify condition while retriving data from table. Where clause is used mostly with Select, Update and Delete query. If condititon specified by where clause is true then only the result from table is returned.

Syntax

SELECT column-name1, column-name2, column-name3, column-nameN

from table-name
WHERE [condition];

AND & OR operator- AND and OR operators are used with Where clause to make more precise conditions for fetching data from database by combining more than one condition together.

AND operator- It is used to set multiple conditions with Where clause. we use AND to combine two or more than two conditions, records satisfying all the condition will be in the result

EXAMPLE

TO return records where salary is less than 10000 and age greater than 25.

SELECT * from Emp WHERE salary < 10000 AND age > 25;

OR operator- OR operator is also used to combine multiple conditions with Where clause. In this , atleast one condition from the conditions specified must be satisfied by any record to be in the result.

Example

To return records where either salary is greater than 10000 or age greater than 25.

SELECT * from Emp WHERE salary > 10000 OR age > 25;

Like clause

Like clause is used as condition in SQL query. Like clause compares data with an expression using wildcard operators. It is used to find similar data from the table.

Wildcard operators - There are two wildcard operators that are used in like clause.

(i) Percent sign % : represents zero, one or more than one character.

(ii) Underscore sign _ : represents only one character.

Example of LIKE clause

To display all records where s_name starts with character 'A'.

SELECT * from Student where s_name like 'A%';

Example

To display all records from Student table where s_name contain 'd' as second character.

SELECT * from Student where s_name like '_d%';

Order By Clause- is used with Select statement for arranging retrieved data in sorted order. The Order by clause by default sort data in ascending order. To sort data in descending order DESC keyword is used with Order by clause.

Syntax :

SELECT column-list|* from table-name order by asc|desc;

To display all records in ascending order of the salary.

SELECT * from Emp order by salary;

To display all records in descending order of the salary.

SELECT * from Emp order by salary DESC;

Group By Clause- it is used to group the results of a SELECT query based on one or more columns. It is also used with SQL functions to group the result from one or more tables.

Syntax: .

SELECT column_name, aggregate_function(column_name)
FROM table_name
WHERE condition

GROUP BY column_name;

To find name and age of employees grouped by their salaries
Example
SELECT name, age
from Emp
group by salary;

Group by in a Statement with WHERE clause
select name, max(salary)
from Emp
where age > 25
group by salary;

Group By clause will always come at the end

HAVING Clause
It is used to give more precise condition for a statement. It is used to mention condition in Group based SQL functions, just like WHERE clause.
Syntax:
select column_name, function(column_name)
FROM table_name
WHERE column_name condition
GROUP BY column_name
HAVING function(column_name) condition;

Consider the following Sale table.

Oid	order_name	previous_balance	customer

To find the customer whose previous_balance sum is more than 3000.
SELECT *
from sale
group by customer
having sum(previous_balance) > 3000;

Distinct keyword- it is used with Select statement to retrieve unique values from the table. Distinct removes all the duplicate records while retrieving from database.

Syntax :

SELECT distinct column-name from table-name;
Example
To display only the unique salary from Emp table
select distinct salary from Emp;

Aggregrate Functions-These functions return a single value after calculating from a group of values.

frequently used Aggregrate functions.

Avg(), Sum(), max(), min(), count(column_name),count(distinct)

count(column name)- Count returns the number of rows present in the table either based on some condition or without condition.

COUNT(distinct)
SELECT COUNT(distinct salary) from emp;

SQL Alias- Alias is used to give an alias name to a table or a column.

(i) Syntax :**Alias name to table**

SELECT column-name
from table-name table_alias-name;
Example
SELECT * from Employee_detail ed;

(ii) Syntax:Alias name to column

SELECT
column-name "alias-name "
from table-name;
SELECT customer_id "cid" from Emp;

SQL View

A view in SQL is a logical subset of data from one or more tables. View is used to restrict data access.

Syntax:

CREATE view view_name AS
SELECT column_name(s)
FROM table_name
WHERE condition

Example
The data fetched from select statement will be stored in another object called sale_view where customer is Alex

CREATE view sale_view as select * from Sale where customer = 'Alex';

Displaying a View

Syntax : of displaying a view is similar to fetching data from table using Select statement.

SELECT * from sale_view;

Update a View Update command for view is same as for tables.

Syntax:,

UPDATE view-name
set value
WHERE condition;

If we update a view it also updates base table data automatically.

To Drop the View

Drop view Viewname;

Example

Drop view sale_view;

---

1. What is a relation? What is the difference between a tuple and an attribute?

Ans A relation is table having atomic values, unique rows and unordered rows and columns. A row in a relation is known as **tuple** whereas a column of a table is known as an **attribute.**

2. What is primary key in a table?

Ans A **Primary Key** is a set of one or more attributes that can be uniquely identify tuples within the relation.

3. What is data redundancy? What are the problems associated with it?

Ans Duplication of data is data redundancy. It leads to the problems like wastage of space and data inconsistency.

4. Define the following terms: (i) Degree (ii) Cardinality.

Ans (i) **Degree:** The numbers of attributes (columns) in a relation determine the degree
(ii) **Cardinality**: The number of tuples (rows) in a relation is called the cardinality

5. What are views? How are they useful?

Ans A view is a virtual table that does not really exist in its own right but it instead derived from one and more underlying base table(s). The view is kind of table whose contents are taken upon other tables depending upon a given query condition. No stored file is created to store contents of a view rather its definition is stored only.

The usefulness of views lies in the fact that they provide an excellent way to give people access to some but not all of the information in a table.

6. Differentiate between **Candidate Key** and **Primary Key** in context of RBDMS.

Ans **Candidate Key**. A candidate key is the one that is capable of becoming primary key. i.e., a field or attribute that has unique value for each row in the relation.

Primary Key is a designed attribute or a group of attributes whose values can uniquely identify the tuples in the relation.

7. Differentiate between Candidate key and Alternate key in context of RDBMS.

Ans Candidate Key. A candidate key is the one that is capable of becoming primary key i.e., a field or attribute that has unique value for each row in the relation.

A candidate key that is not a primary key is called an Alternate key.

8. Differentiate between primary key and alternate key.

Ans Primary Key. It is the set of one or more attributes that can uniquely identify tuples within a relation.

Alternate Key. It is a candidate key which is not primary key.

9. **What are candidate keys in a table? Give a suitable example of candidate keys**

Ans A candidate key is the one that is capable of becoming primary key i., a field or attribute that has unique value for each row in the relation.

Example Table: ITEM

Ino	Item	Quantity
101	Pen	560
102	Pencil	340
104	CD	540
10	DVD	200
110	Floppy	400

{ Item-Candidate Keys}

10. Differentiate between Data Definition language(DDL) and Data Manipulation language(DML).

Ans (i) The SQL DDL provides commands for defining relation schemas, deleting relationship, creating indexes and modifying schemas.

The SQL DML includes a query language to insert, delete and modify tuples in the database.

(ii) Data Manipulation Language (DML) is used to put values and manipulate them in tables and Data Definition language (DDL) is used to create tables and other database objects.

11. What is the different between WHERE and HAVING clause?

Ans The HAVING clause places conditions on groups in contrast to WHERE clause, which places conditions on individual rows.

12. Write the SQL statement to create EMPLOYEE relation which contains EMPNO, Name, Skill, PayRate.

Ans
```
CREATE TABLE Employee
(       EmpNo           CHAR(4)     NOT NULL PRIMARY KEY,
        Name        CHAR(20)   NOT NULL,
        Skill       CHAR(1),
        PayRate     DECIMAL(8,2));
```

13. Insert a record with suitable data in the table EMP, having system date as the Hiredate.

Ans Date ( ) function gives the system date.

```
INSERT INTO Emp
VALUES (3008, 18, "XAVIER", "Manager", Date( ), 3250, NULL);
```

14. Illustrate Cartesian product operation between the two tables/relations using a suitable example.

Ans The two table GABS1 and GABS are as follows:

GAB 1

ROLL NO	NAME
1	ABC
2	GABS

GAB 2

MARKS	SROLL NO	AGE
90	1	19
92	3	17

The certesian product of above two tables is as follows:

Cartesian Product

RollNo	Name	Marks	SRollNo	Age
1	ABC	90	1	19
1	ABC	92	3	17
2	GABS	90	1	19
2	GABS	92	3	17

15. What is the purpose of key in a table? Give an example of key in a table.

Ans A key is used to identify a tuple uniquely with in the relation. The value of key is unique. No rows in the relation can have same value.

e.g. In an Employee relation EmpCode is a key using EmpCode one can obtain the information of a particular employee.