NAME: Anjali Samala

BATCH: 2022-7607

ENROLMENT NUMBER: EBEON0722625522

LOCATION: Chennai

SQL ALL COMMANDS:

SQL(structured query language):

Structured Query Language(SQL) as we all know is the database language by the use of which we can perform certain operations on the existing database and also we can use this language to create a database. SQL uses certain commands like Create, Drop, Insert, etc. to carry out the required tasks.

These SQL commands are mainly categorized into four categories as:

1. DDL – Data Definition Language
2. DQl – Data Query Language/ DRL Data Retrieval Language
3. DML – Data Manipulation Language
4. DCL – Data Control Language

Though many resources claim there to be another category of SQL clauses TCL – Transaction Control Language. So we will see in detail about TCL as well.

1.DDL (Data Definition Language):

DDL or Data Definition Language actually consists of the SQL commands that can be used to define the database schema. It simply deals with descriptions of the database schema and is used to create and modify the structure of database objects in the database. DDL is a set of SQL commands used to create, modify, and delete database structures but not data. These commands are normally not used by a general user, who should be accessing the database via an application.

List of DDL commands:

- **CREATE:** This command is used to create the database or its objects (like table, index, function, views, store procedure, and triggers).
- **DROP:** This command is used to delete objects from the database.
- **ALTER:** This is used to alter the structure of the database.
- **TRUNCATE:** This is used to remove all records from a table, including all spaces allocated for the records are removed.
- **COMMENT:** This is used to add comments to the data dictionary.
- **RENAME:** This is used to rename an object existing in the database.

2.DQL (Data Query Language): Data Retrieval language.

DQL statements are used for performing queries on the data within schema objects. The purpose of the DQL Command is to get some schema relation based on the query passed to it. We can define DQL as follows it is a component of SQL statement that allows getting data from the database and imposing order upon it. It includes the SELECT statement. This command allows getting the data out of the database to perform operations with it. When a SELECT is fired against a table or tables the result is compiled into a

further temporary table, which is displayed or perhaps received by the program i.e. a front-end.

List of DQL:

- **SELECT:** It is used to retrieve data from the database.

3.DML(Data Manipulation Language):

The SQL commands that deals with the manipulation of data present in the database belong to DML or Data Manipulation Language and this includes most of the SQL statements. It is the component of the SQL statement that controls access to data and to the database. Basically, DCL statements are grouped with DML statements.

List of DML commands:

- **INSERT** : It is used to insert data into a table.
- **UPDATE:** It is used to update existing data within a table.
- **DELETE** : It is used to delete records from a database table.
- **LOCK:** Table control concurrency.
- **CALL:** Call a PL/SQL or JAVA subprogram.
- **EXPLAIN PLAN**: It describes the access path to data.

4.DCL (Data Control Language):

DCL includes commands such as GRANT and REVOKE which mainly deal with the rights, permissions, and other controls of the database system.

List of DCL commands:

- **GRANT:** This command gives users access privileges to the database.
- **REVOKE:** This command withdraws the user's access privileges given by using the GRANT command.

Though many resources claim there to be another category of SQL clauses TCL – Transaction Control Language. So we will see in detail about TCL as well. TCL commands deal with the transaction within the database.

List of TCL commands:

- **COMMIT:** Commits a Transaction.
- **ROLLBACK:** Rollbacks a transaction in case of any error occurs.
- **SAVEPOINT:**Sets a savepoint within a transaction.
- **SET TRANSACTION:** Specify characteristics for the transaction

1.DDL: Data Definition language

1.Create:

Syntax: create table table name(col1 datatype, col2 datatype2);

Ex: create table stu(SN int, First_ name varchar(30), Last_ name varchar(40), age int,phone_number);

Output:

mysql> create database teach1;

Query OK, 1 row affected (0.01 sec)

mysql> use teach1;

Database changed

```
mysql> create table animals(SN int,Name varchar(30),color varchar(20));

Query OK, 0 rows affected (0.06 sec)
```

2.Alter:

Syntax: alter table table name add col_name datatype1;update table name set col_name=’value’ where condition

EX: alter table **black** add salary int(20);

Update **black** set salary=’20000’ where SN=’1’;

Output:

```
mysql> select * from animals;

+------+----------+---------------+

| SN   | Name     | color         |

+------+----------+---------------+

|    1 | cat      | white         |

|    2 | buffallo | black         |

|    3 | dog      | balckandwhite |

+------+----------+---------------+

3 rows in set (0.00 sec)

mysql> alter table animals add food int(20);update animals set food='grass' where SN='2';

ERROR 1060 (42S21): Duplicate column name 'food'

Query OK, 0 rows affected (0.00 sec)

Rows matched: 1  Changed: 0  Warnings: 0


mysql> alter table animals add food varchar(20);update animals set food='biscate' where SN='3';

ERROR 1060 (42S21): Duplicate column name 'food'

Query OK, 1 row affected (0.02 sec)
```

Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from animals;

+------+----------+---------------+---------+

| SN | Name | color | food |

+------+----------+---------------+---------+

| 1 | cat | white | milk |

| 2 | buffallo | black | grass |

| 3 | dog | balckandwhite | biscate |

+------+----------+---------------+---------+

3 rows in set (0.00 sec)

3.DROP:

Syntax: drop database database name;

EX: drop database teach1;

OUTPUT:

mysql> use teach1;

Database changed

mysql> select * from animals;

+------+----------+-------+-------+

| SN | Name | color | food |

+------+----------+-------+-------+

| 1 | cat | white | milk |

| 2 | buffallo | black | grass |

+------+----------+-------+-------+

2 rows in set (0.03 sec)

*** mysql> drop database teach1;**

Query OK, 1 row affected (0.07 sec)

mysql> select * from animals;

ERROR 1046 (3D000): No database selected

4.TRUNCATE:

Syntax: truncate table table name;

EX: truncate table per2;

OUTPUT:

mysql> use teach4;

Database changed

mysql> create table per2(SN int,First_name varchar(20),Last_name varchar(15),age int);

Query OK, 0 rows affected (0.04 sec)

mysql> insert into per2(SN,First_name,Last_name,age)values(1,'anjali','samala',21),(2,'swapna','samala',35),(3,'ajay','samala',20),(4,'laxmi','kanukula',40);

Query OK, 4 rows affected (0.03 sec)

Records: 4 Duplicates: 0 Warnings: 0

```
mysql> select * from per2;
+------+------------+-----------+------+
| SN   | First_name | Last_name | age  |
+------+------------+-----------+------+
|    1 | anjali     | samala    |   21 |
|    2 | swapna     | samala    |   35 |
|    3 | ajay       | samala    |   20 |
|    4 | laxmi      | kanukula  |   40 |
+------+------------+-----------+------+
4 rows in set (0.00 sec)

mysql> truncate table per2;
Query OK, 0 rows affected (0.03 sec)

mysql> select * from per2;
Empty set (0.02 sec)
```

2.DQL: Data Query language

1.SELECT:

Syntax: select * from table name;

EX: select * from animals;

Output:

```
mysql> select * from animals;
+------+----------+---------------+
| SN   | Name     | color         |
+------+----------+---------------+
|    1 | cat      | white         |
|    2 | buffallo | black         |
|    3 | dog      | balckandwhite |
+------+----------+---------------+
3 rows in set (0.00 sec)
```

3.DML:Data manipulation language

1.Insert:

Syntax: insert into table name(col1,col2,col3,col4)values(vale1,value2,value3,value4);have to put same

Note: if the value is number you have to put same

But if value is name(ex: 'Anjali')then you have to write that in **single codes.**

EX: insert into **black**

(SN, First_ name, Last_ name, age, phone_ number)

values

(1,'anjali','samala',21,9704974315)

Output:

```
mysql> insert into animals(SN,Name,color)values(1,'cat','white');

Query OK, 1 row affected (0.03 sec)


mysql> insert into animals(SN,Name,color)values(2,'buffallo','black');

Query OK, 1 row affected (0.03 sec)


mysql> insert into animals(SN,Name,color)values(3,'dog','balckandwhite');

Query OK, 1 row affected (0.02 sec)
```

2.Update:

Syntax: update table name set col name=’value’

Where col name=’value’;

EX: update **black** set salary=’50000’ where SN=’1’;

OUTPUT:

```
mysql> select * from animals;

+------+----------+---------------+---------+

| SN   | Name     | color         | food    |

+------+----------+---------------+---------+

|    1 | cat      | white         | milk    |

|    2 | buffallo | black         | grass   |

|    3 | dog      | balckandwhite | biscate |

+------+----------+---------------+---------+

3 rows in set (0.00 sec)
```

3.Delete:

Syntax: delete from table name where

Col name=’value’;

EX: delete from **black** where First_name= ’Suchithra’;

Delete from animals where SN=3;

Output:

```
mysql> select * from animals;
+------+----------+---------------+---------+
| SN   | Name     | color         | food    |
+------+----------+---------------+---------+
|    1 | cat      | white         | milk    |
|    2 | buffallo | black         | grass   |
|    3 | dog      | balckandwhite | biscate |
+------+----------+---------------+---------+
3 rows in set (0.00 sec)
mysql> delete from animals where SN='3';
Query OK, 1 row affected (0.02 sec)

mysql> select * from animals;
+------+----------+-------+-------+
| SN   | Name     | color | food  |
+------+----------+-------+-------+
|    1 | cat      | white | milk  |
|    2 | buffallo | black | grass |
+------+----------+-------+-------+
```

2 rows in set (0.00 sec)

5.TCL: transaction control language

OUTPUT:

mysql> select * from letters1;

+-------+------------+------+

| name | email | age |

+-------+------------+------+

| radha | anjali1233 | 21 |

| aruna | aruna345 | 30 |

| kavya | kavya567 | 34 |

+-------+------------+------+

3 rows in set (0.00 sec)

mysql> start transaction;

Query OK, 0 rows affected (0.00 sec)

mysql> select * from letters1;

+-------+------------+------+

| name | email | age |

+-------+------------+------+

| radha | anjali1233 | 21 |

| aruna | aruna345 | 30 |

| kavya | kavya567 | 34 |

+-------+------------+------+

3 rows in set (0.00 sec)

mysql> update letters1 set age='27' where name='kavya';

Query OK, 1 row affected (0.00 sec)

Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from letters1;

+-------+----------+------+

| name | email | age |

+-------+----------+------+

| radha | radha234 | 21 |

| aruna | aruna345 | 30 |

| kavya | kavya567 | 27 |

+-------+----------+------+

3 rows in set (0.00 sec)

mysql> sacepoint insertion1;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'sacepoint insertion1' at line 1

mysql> savepoint insertion1;

Query OK, 0 rows affected (0.00 sec)

mysql> rollback to insertion;

Query OK, 0 rows affected (0.00 sec)

mysql> select * from letters1;

+-------+----------+------+

| name | email | age |

+-------+----------+------+

| radha | radha234 | 21 |

| aruna | aruna345 | 30 |

| kavya | kavya567 | 34 |

+-------+----------+------+

3 rows in set (0.00 sec)

mysql> select * from letters1;

+-------+----------+------+

| name | email | age |

+-------+----------+------+

| radha | radha234 | 21 |

| aruna | aruna345 | 26 |

| kavya | kavya567 | 34 |

+-------+----------+------+

3 rows in set (0.00 sec)

mysql> svaepoint updation;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'svaepoint updation' at line 1

mysql> savepoint updation;

Query OK, 0 rows affected (0.00 sec)

mysql> rollback to updation;

Query OK, 0 rows affected (0.00 sec)

mysql> select * from letters1;

+-------+----------+------+

| name | email | age |

+-------+----------+------+

| radha | radha234 | 21 |

| aruna | aruna345 | 26 |

| kavya | kavya567 | 34 |

+-------+----------+------+

3 rows in set (0.00 sec)

*SQL select Distinct:

Syntax: select distinct col_name,col_name

From table_name;

EX: select distinct age

From emp1;

Output:

mysql> create database teach2;

Query OK, 1 row affected (0.03 sec)

mysql> use teach2;

Database changed

mysql> create table emp1(SN int,Name varchar(20),age int,phone_number varchar(25),salary varchar(26));

Query OK, 0 rows affected (0.04 sec)

mysql> insert into
emp1(SN,Name,age,phone_number,salary)values(1,'anjali',21,9704974315,20000),(2,'srilatha',22,628280097,25000),(3,'su
chi',23,9346796446,30000),(4,'mounika',21,7032546792,50000),(5,'navya',22,8466850434,30000);

Query OK, 5 rows affected (0.03 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> select * from emp1;

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

```
+------+----------+------+-------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
|   5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+-------------+--------+
5 rows in set (0.00 sec)

mysql> select distinct age emp1;
ERROR 1054 (42S22): Unknown column 'age' in 'field list'
mysql> select distinct age from emp1;
+------+
| age  |
+------+
|   21 |
|   22 |
|   23 |
+------+
3 rows in set (0.00 sec)
mysql> select distinct salary from emp1;
+--------+
| salary |
+--------+
| 20000  |
| 25000  |
| 30000  |
```

| 50000 |

+--------+

4 rows in set (0.00 sec)

*SQL select Count :

Syntax: select count(col_name) from table_name;

EX: select count(salary) from emp1;

Output:

mysql> select count(SN) from emp1;

+-----------+

| count(SN) |

+-----------+

| 5 |

+-----------+

1 row in set (0.00 sec)

mysql> select count(salary) from emp1;

+---------------+

| count(salary) |

+---------------+

| 5 |

+---------------+

1 row in set (0.00 sec)

KEYS:

1.PRIMARY KEY:

1. Syntax: CREATE TABLE students
2. (
3. S_Id int NOT NULL,
4. LastName varchar (255) NOT NULL,
5. FirstName varchar (255),
6. Address varchar (255),
7. City varchar (255),
8. PRIMARY KEY (S_Id)
9.)

OUTPUT:

```
mysql> create table stud1(S_id int not null,LastName varchar(200) not null,FirstName varchar(200),Address
varchar(200),City varchar(200),primary key(S_id));

Query OK, 0 rows affected (0.03 sec)



mysql> select * from stud1;

Empty set (0.01 sec)



mysql> inser into
stud1(S_id,LastName,FirstName,Address,City)vlues(101,'samala','anjali','hyderabad','warangal'),(102,'porika','srilatha','mul
ugu','warangal'),(103,'bhukya','suchi','hanumakonda','parkal');

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server
version for the right syntax to use near 'inser into stud1(S_id,LastName,FirstName,Address,City)vlues(101,'samala','anjali'
at line 1

mysql> insert into
stud1(S_id,LastName,FirstName,Address,City)values(101,'samala','anjali','hyderabad','warangal'),(102,'porika','srilatha','mu
lugu','warangal'),(103,'bhukya','suchi','hanumakonda','parkal');

Query OK, 3 rows affected (0.01 sec)

Records: 3  Duplicates: 0  Warnings: 0



mysql> select * from stud1;

+------+----------+-----------+-------------+----------+
```

| S_id | LastName | FirstName | Address | City |

+------+----------+-----------+-------------+----------+

| 101 | samala | anjali | hyderabad | warangal |

| 102 | porika | srilatha | mulugu | warangal |

| 103 | bhukya | suchi | hanumakonda | parkal |

+------+----------+-----------+-------------+----------+

3 rows in set (0.00 sec)

mysql> alter table stud1 add primary key(S_id);

ERROR 1068 (42000): Multiple primary key defined

JOIN :

Syntax: select col name1,col name2,col name3 from table name1 t,table name2 t where s.same col name=p.col name;

OUTPUT:

mysql> create database taech8;

Query OK, 1 row affected (0.03 sec)

mysql> use taech8;

Database changed

mysql> create table staff(ID int,staff_name varchar(20),staff_age int,staff_address varchar(30),salary varchar(15));

Query OK, 0 rows affected (0.05 sec)

mysql> insert into staff(ID,staff_name,staff_age,staff_address,salary)values(1,'chandu',21,'warangal',20000),(2,'mani',22,'parkal',30000),(3,'ajay',20,'mulugu',20000),(4,'sandy',21,'gudepu',35000);

Query OK, 4 rows affected (0.01 sec)

Records: 4 Duplicates: 0 Warnings: 0

```
mysql> select * from staff;
+------+------------+-----------+---------------+--------+
| ID   | staff_name | staff_age | staff_address | salary |
+------+------------+-----------+---------------+--------+
|    1 | chandu     |        21 | warangal      | 20000  |
|    2 | mani       |        22 | parkal        | 30000  |
|    3 | ajay       |        20 | mulugu        | 20000  |
|    4 | sandy      |        21 | gudepu        | 35000  |
+------+------------+-----------+---------------+--------+
4 rows in set (0.00 sec)

mysql> create table pay(payment_ID int,date varchar(20),staff_ID int,amount varchar(20));
Query OK, 0 rows affected (0.04 sec)

mysql> insert into
pay(payment_ID,date,staff_ID,amount)values(101,21/5/21,1,2000),(102,12/10/22,2,1000),(103,17/10/19,3,1500);
Query OK, 3 rows affected (0.03 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> select * from pay;
+------------+----------------------+----------+--------+
| payment_ID | date                 | staff_ID | amount |
+------------+----------------------+----------+--------+
|        101 | 0.200000000000000000 |        1 | 2000   |
|        102 | 0.054545454545454545 |        2 | 1000   |
|        103 | 0.089473684210526315 |        3 | 1500   |
+------------+----------------------+----------+--------+
```

3 rows in set (0.00 sec)

```
mysql> select * from staff;
+------+------------+-----------+---------------+--------+
| ID   | staff_name | staff_age | staff_address | salary |
+------+------------+-----------+---------------+--------+
|    1 | chandu     |        21 | warangal      |  20000 |
|    2 | mani       |        22 | parkal        |  30000 |
|    3 | ajay       |        20 | mulugu        |  20000 |
|    4 | sandy      |        21 | gudepu        |  35000 |
+------+------------+-----------+---------------+--------+
4 rows in set (0.00 sec)

mysql> select staff_ID,staff_name,amount from staff s,pay p where s.ID=p.staff_ID;
+----------+------------+--------+
| staff_ID | staff_name | amount |
+----------+------------+--------+
|        1 | chandu     |   2000 |
|        2 | mani       |   1000 |
|        3 | ajay       |   1500 |
+----------+------------+--------+
3 rows in set (0.02 sec)
```

RIGHT JOIN:

Syntax: SELECT table1.column1, table2.column2....
FROM table1
Right JOIN table2
ON table1.column_field = table2.column_field;

EX: select ID,staff_name,amount,date from staff right join pay

On staff.ID=pay.staff_ID;

OUTPUT:

```
mysql> use taech8;

Database changed

mysql> select * from staff;

+------+------------+-----------+---------------+--------+

| ID   | staff_name | staff_age | staff_address | salary |

+------+------------+-----------+---------------+--------+

|    1 | chandu     |       21 | warangal      | 20000  |

|    2 | mani       |       22 | parkal        | 30000  |

|    3 | ajay       |       20 | mulugu        | 20000  |

|    4 | sandy      |       21 | gudepu        | 35000  |

+------+------------+-----------+---------------+--------+

4 rows in set (0.00 sec)


mysql> select * from pay;

+------------+---------------------+----------+--------+

| payment_ID | date                | staff_ID | amount |

+------------+---------------------+----------+--------+

|       101 | 0.200000000000000000 |       1 | 2000   |

|       102 | 0.054545454545454545 |       2 | 1000   |

|       103 | 0.089473684210526315 |       3 | 1500   |

+------------+---------------------+----------+--------+

3 rows in set (0.00 sec)
```

```
mysql> select ID,staff_name,amount,date from staff right join pay on staff.ID=pay.staff_ID;
+------+------------+--------+---------------------+
| ID   | staff_name | amount | date                |
+------+------------+--------+---------------------+
|    1 | chandu     | 2000   | 0.200000000000000000 |
|    2 | mani       | 1000   | 0.054545454545454545 |
|    3 | ajay       | 1500   | 0.089473684210526315 |
+------+------------+--------+---------------------+
3 rows in set (0.02 sec)
```

LEFT JOIN:

Syntax: SELECT table1.column1, table2.column2....

FROM table1

Left JOIN table2
ON table1.column_field = table2.column_field;

EX: select ID,staff_name,amount,date from staff left join pay

On staff.ID=pay.staff_ID;

OUTPUT:

```
mysql> use taech8;
Database changed
mysql> select * from staff;
+------+------------+-----------+---------------+--------+
| ID   | staff_name | staff_age | staff_address | salary |
+------+------------+-----------+---------------+--------+
|    1 | chandu     |        21 | warangal      |  20000 |
```

```
|   2 | mani     |       22 | parkal       | 30000  |
|   3 | ajay     |       20 | mulugu       | 20000  |
|   4 | sandy    |       21 | gudepu       | 35000  |
+------+----------+----------+--------------+--------+
4 rows in set (0.00 sec)

mysql> select * from pay;
+------------+---------------------+----------+--------+
| payment_ID | date                | staff_ID | amount |
+------------+---------------------+----------+--------+
|        101 | 0.200000000000000000 |        1 | 2000   |
|        102 | 0.054545454545454545 |        2 | 1000   |
|        103 | 0.089473684210526315 |        3 | 1500   |
+------------+---------------------+----------+--------+
3 rows in set (0.00 sec)

mysql> select ID,staff_name,amount,date from staff left join pay on staff.ID=pay.staff_ID;
+------+----------+--------+---------------------+
| ID   | staff_name | amount | date                |
+------+----------+--------+---------------------+
|   1 | chandu    | 2000   | 0.200000000000000000 |
|   2 | mani      | 1000   | 0.054545454545454545 |
|   3 | ajay      | 1500   | 0.089473684210526315 |
|   4 | sandy     | NULL   | NULL                |
+------+----------+--------+---------------------+
4 rows in set (0.00 sec)
```

SELECT RANDAOM:

Syntax: select column from table order by rand() limit 1;

EX: select age from emp1 order by rand() limit 1;

OUTPUT:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
|   5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+----------+
5 rows in set (0.00 sec)
mysql> select age from emp1 order by rand() limit 1;
+------+
| age  |
+------+
|   22 |
+------+
```

1 row in set (0.00 sec)

mysql> select age from emp1 order by rand() limit 3;

+------+

| age |

+------+

| 22 |

| 22 |

| 21 |

+------+

3 rows in set (0.00 sec)

mysql> select age from emp1 order by rand() limit 5;

+------+

| age |

+------+

| 22 |

| 21 |

| 23 |

| 22 |

| 21 |

+------+

5 rows in set (0.00 sec)

mysql> select age from emp1 order by rand() limit 4;

+------+

| age |

```
+------+
|  21 |
|  23 |
|  22 |
|  22 |
+------+
4 rows in set (0.00 sec)

mysql> select age from emp1 order by rand();
+------+
| age  |
+------+
|  23 |
|  22 |
|  21 |
|  21 |
|  22 |
+------+
5 rows in set (0.00 sec)

mysql> select salary from emp1 order by rand();
+--------+
| salary |
+--------+
| 50000  |
| 30000  |
| 20000  |
```

| 25000 |

| 30000 |

+--------+

5 rows in set (0.00 sec)

SELECT IN:

Syntax: select * from table name where col name IN(value,value2,value3);

EX: select * from emp1 where salary in(20000,25000,30000);

OUTPUT:

mysql> select * from emp1;

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

+------+----------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

| 2 | srilatha | 22 | 628280097 | 25000 |

| 3 | suchi | 23 | 9346796446 | 30000 |

| 4 | mounika | 21 | 7032546792 | 50000 |

| 5 | navya | 22 | 8466850434 | 30000 |

+------+----------+------+--------------+--------+

5 rows in set (0.00 sec)

mysql> select * from emp1 where salary in(20000,25000,30000);

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

+------+----------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

```
|   2 | srilatha |   22 | 628280097    | 25000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
|   5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
4 rows in set (0.00 sec)

mysql> select * from emp1 where salary in(20000,25000);
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
+------+----------+------+--------------+--------+
2 rows in set (0.00 sec)

mysql> select * from emp1 where salary in(20000,25000,50000);
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
+------+----------+------+--------------+--------+
3 rows in set (0.00 sec)

mysql> select * from emp1 where salary in(20000);
+------+--------+------+--------------+--------+
```

| SN | Name | age | phone_number | salary |

+------+--------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

+------+--------+------+--------------+--------+

1 row in set (0.00 sec)

SELECT UNIQUE:

there is no difference between DISTINCT and UNIQUE.

Syntax: select unique column_name from table name;

EX: select unique Name from emp1;

OUTPUT:

mysql> select * from emp1;

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

+------+----------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

| 2 | srilatha | 22 | 628280097 | 25000 |

| 3 | suchi | 23 | 9346796446 | 30000 |

| 4 | mounika | 21 | 7032546792 | 50000 |

| 5 | navya | 22 | 8466850434 | 30000 |

+------+----------+------+--------------+--------+

5 rows in set (0.00 sec)

mysql> select unique Name from emp1;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'unique Name from emp1' at line 1

mysql> select unique SN from emp1;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'unique SN from emp1' at line 1

mysql> select unique salary from emp1;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'unique salary from emp1' at line 1

mysql> select * from emp1 unique salary;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'unique salary' at line 1

SELECT SUM:

Syntax: select sum(salary) as"total salary" from table name where salary>2000;

EX: select sum(salary) as "total salary" from emp1 where salary>15000;

OUTPUT:

mysql> select * from emp1;

```
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   |  20000 |
|    2 | srilatha |   22 | 628280097    |  25000 |
|    3 | suchi    |   23 | 9346796446   |  30000 |
|    4 | mounika  |   21 | 7032546792   |  50000 |
|    5 | navya    |   22 | 8466850434   |  30000 |
+------+----------+------+--------------+--------+
```

5 rows in set (0.00 sec)

```
mysql> select sum(salary) as "total salary" from emp1 where salary >15000;
+--------------+
| total salary |
+--------------+
|       155000 |
+--------------+
1 row in set (0.02 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <40000;
+--------------+
| total salary |
+--------------+
|       105000 |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <50000;
+--------------+
| total salary |
+--------------+
|       105000 |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <20000;
+--------------+
```

```
| total salary |
+--------------+
|         NULL |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <25000;
+--------------+
| total salary |
+--------------+
|        20000 |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(Name) as "total Name" from emp1 where Name <4;
+------------+
| total Name |
+------------+
|          0 |
+------------+
1 row in set, 10 warnings (0.00 sec)

mysql> select sum(age) as "total age" from emp1 where age <3;
+-----------+
| total age |
+-----------+
|      NULL |
```

+-----------+

1 row in set (0.00 sec)

WHERE AND DISTINCT IN SUM:

Syntax: select sum(distinct col name) as “total col name” from table name

Where col name <value;

EX: select sum(distinct salary) as “total salary” from emp1 where salary <20000;

OUTPUT:

mysql> use teach2;

Database changed

mysql> select * from emp1;

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

+------+----------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

| 2 | srilatha | 22 | 628280097 | 25000 |

| 3 | suchi | 23 | 9346796446 | 30000 |

| 4 | mounika | 21 | 7032546792 | 50000 |

| 5 | navya | 22 | 8466850434 | 30000 |

+------+----------+------+--------------+--------+

5 rows in set (0.00 sec)

mysql> select sum(distinct age) as "total age" from emp1 where age <25;

+-----------+

| total age |

+-----------+

| 66 |

+----------+

1 row in set (0.02 sec)

mysql> select sum(distinct salary) as "total salary" from emp1 where salary <45000;

+--------------+

| total salary |

+--------------+

| 75000 |

+--------------+

1 row in set (0.00 sec)

GROUP BY IN SELECT SUM:

Syntax: select col name1 sum(col name2) as “total col name2” from table name group by col name1;

EX: select age sum(salary) as “total salary” from emp1 group by age;

OUTPUT:

mysql> select Name,sum(salary) as "total salary" from emp1 group by Name;

+----------+--------------+

| Name | total salary |

+----------+--------------+

| anjali | 20000 |

| srilatha | 25000 |

| suchi | 30000 |

| mounika | 50000 |

| navya | 30000 |

+----------+--------------+

5 rows in set (0.02 sec)

mysql> select age,sum(salary) as "total salary" from emp1 group by age;

+------+--------------+

| age | total salary |

+------+--------------+

| 21 | 70000 |

| 22 | 55000 |

| 23 | 30000 |

+------+--------------+

3 rows in set (0.00 sec)

SELECT NULL:

Syntax: select col name1,col name2,col name3 from table name

Where col name3 is null;

EX: select Name,enroll_number from batch where enroll_number is null;

OUTPUT:

mysql> select * from batch;

+------+---------+-------------+---------------+

| SN | Name | Roll_number | enroll_number |

+------+---------+-------------+---------------+

| 1 | akhila | 239 | 2022 |

| 2 | navya | 234 | 2021 |

| 3 | rajitha | 231 | 2000 |

| 4 | sandhya | 230 | 2018 |

+------+---------+-------------+---------------+

4 rows in set (0.00 sec)

mysql> select Name,enroll_number from batch where enroll_number is null;

Empty set (0.00 sec)

mysql> select Name,SN from batch where SN is null;

Empty set (0.00 sec)

SELECT DATE:

Syntax: select * from table name where your date<= ‘value’;

EX: select * from data where date<=’12/10/21’;

OUTPUT:

mysql> select * from data;

```
+------+--------+------+-----------+-------+----------+
| SN   | Name   | CGPA | Grade     | marks | date     |
+------+--------+------+-----------+-------+----------+
|    1 | anusha |  9.6 | excellent | -30   | 20/10/22 |
|    2 | ammu   |  8.5 | good      | -40   | 12/10/22 |
|    3 | anjali |  6.5 | average   | NULL  | 12/9/22  |
+------+--------+------+-----------+-------+----------+
```

3 rows in set (0.00 sec)

mysql> select * from data where date>='12/10/21';

```
+------+--------+------+-----------+-------+----------+
| SN   | Name   | CGPA | Grade     | marks | date     |
+------+--------+------+-----------+-------+----------+
```

```
|   1 | anusha |  9.6 | excellent | -30   | 20/10/22 |

|   2 | ammu   |  8.5 | good      | -40   | 12/10/22 |

|   3 | anjali |  6.5 | average   | NULL  | 12/9/22  |

+------+--------+------+-----------+-------+----------+

3 rows in set (0.00 sec)


mysql> select * from data where date<='12/10/21';

Empty set (0.00 sec)


mysql> select * from data where date<='12/10/22';

+------+------+------+-------+-------+----------+

| SN   | Name | CGPA | Grade | marks | date     |

+------+------+------+-------+-------+----------+

|   2 | ammu |  8.5 | good  | -40   | 12/10/22 |

+------+------+------+-------+-------+----------+

1 row in set (0.00 sec)
```

WHERE CONDITION:

Syntax: select sum(col name1) as “total col name1” from col name2 where col name1<><=>==value;

EX: select sum(salary) as "total salary" from emp1 where salary >15000;

OUTPUT:

```
mysql> select * from emp1;

+------+----------+------+--------------+--------+

| SN   | Name     | age  | phone_number | salary |

+------+----------+------+--------------+--------+
```

```
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
|   5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary >15000;
+--------------+
| total salary |
+--------------+
|       155000 |
+--------------+
1 row in set (0.02 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <40000;
+--------------+
| total salary |
+--------------+
|       105000 |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <50000;
+--------------+
| total salary |
```

```
+--------------+
|       105000 |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <20000;
+--------------+
| total salary |
+--------------+
|         NULL |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(salary) as "total salary" from emp1 where salary <25000;
+--------------+
| total salary |
+--------------+
|        20000 |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(Name) as "total Name" from emp1 where Name <4;
+------------+
| total Name |
+------------+
|          0 |
+------------+
```

1 row in set, 10 warnings (0.00 sec)

mysql> select sum(age) as "total age" from emp1 where age <3;

+-----------+

| total age |

+-----------+

| NULL |

+-----------+

AND:

Syntax: select * from table name where col name=”value” and col name=”value”;

EX: select * from stud where city=”warangal” and Name=”anju”;

OUTPUT:

mysql> select * from stud;

+------+-------+------------+---------------+

| SN | Name | city | enroll_number |

+------+-------+------------+---------------+

| 1 | anju | warangal | 2022 |

| 2 | srii | hanamkonda | 2021 |

| 3 | suchi | mulugu | 2020 |

+------+-------+------------+---------------+

3 rows in set (0.00 sec)

mysql> select * from stud where city="warangal" and Name="mulugu";

Empty set (0.00 sec)

```
mysql> select * from stud where city="warangal" and Name="warangal";
Empty set (0.00 sec)

mysql> select * from stud where city="warangal" and Name="anju";
+------+------+----------+---------------+
| SN   | Name | city     | enroll_number |
+------+------+----------+---------------+
|    1 | anju | warangal | 2022          |
+------+------+----------+---------------+
1 row in set (0.00 sec)

mysql> select * from stud where city="warangal" and Name="srii";
Empty set (0.00 sec)

mysql> select * from stud where city="hanamkonda" and Name="srii";
+------+------+------------+---------------+
| SN   | Name | city       | enroll_number |
+------+------+------------+---------------+
|    2 | srii | hanamkonda | 2021          |
+------+------+------------+---------------+
1 row in set (0.00 sec)
```

OR:

Syntax: select * from table name where condion1 or condition2;

EX: select * from stud where city="warangal" or Name="anju";

OUTPUT:

```
mysql> select * from stud;
+------+-------+------------+---------------+
| SN   | Name  | city       | enroll_number |
+------+-------+------------+---------------+
|    1 | anju  | warangal   | 2022          |
|    2 | srii  | hanamkonda | 2021          |
|    3 | suchi | mulugu     | 2020          |
+------+-------+------------+---------------+
3 rows in set (0.00 sec)

mysql> select * from stud where city="warangal" or Name="anju";
+------+------+----------+---------------+
| SN   | Name | city     | enroll_number |
+------+------+----------+---------------+
|    1 | anju | warangal | 2022          |
+------+------+----------+---------------+
1 row in set (0.00 sec)

mysql> select * from stud where city="warangal" or Name="srii";
+------+------+------------+---------------+
| SN   | Name | city       | enroll_number |
+------+------+------------+---------------+
|    1 | anju | warangal   | 2022          |
|    2 | srii | hanamkonda | 2021          |
+------+------+------------+---------------+
2 rows in set (0.00 sec)
```

AS:

Syntax: select col name as new name,avg(col name) as new name from table name;

EX: select Name as Full_name, avg(city) as address from stud;

OUTPUT:

```
mysql> select * from stud;
+------+-------+------------+---------------+
| SN   | Name  | city       | enroll_number |
+------+-------+------------+---------------+
|    1 | anju  | warangal   | 2022          |
|    2 | srii  | hanamkonda | 2021          |
|    3 | suchi | mulugu     | 2020          |
+------+-------+------------+---------------+
3 rows in set (0.00 sec)

mysql> select Name as Full_name, avg(city) as address from std;
ERROR 1146 (42S02): Table 'teach5.std' doesn't exist
mysql> select Name as Full_name, avg(city) as address from stud;
+-----------+---------+
| Full_name | address |
+-----------+---------+
| anju      |       0 |
+-----------+---------+
1 row in set, 3 warnings (0.00 sec)
```

```
mysql> select Name as Full_name, avg(SN) as number from stud;
+-----------+--------+
| Full_name | number |
+-----------+--------+
| anju      | 2.0000 |
+-----------+--------+
1 row in set (0.00 sec)
```

##GROUP BY:

Syntax: select sum(col name1),col name2 from

Table_name group by col name2;

EX: select sum(salary),Name from

Emp1 group by Name;

Output:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
```

| 2 | srilatha | 22 | 628280097 | 25000 |

| 3 | suchi | 23 | 9346796446 | 30000 |

| 4 | mounika | 21 | 7032546792 | 50000 |

| 5 | navya | 22 | 8466850434 | 30000 |

+------+----------+------+--------------+--------+

5 rows in set (0.00 sec)

mysql> select sum(salary),Name from emp1 group by Name;

+-------------+----------+

| sum(salary) | Name |

+-------------+----------+

| 20000 | anjali |

| 25000 | srilatha |

| 30000 | suchi |

| 50000 | mounika |

| 30000 | navya |

+-------------+----------+

5 rows in set (0.00 sec)

mysql> select sum(age),SN from emp1 group by SN;

+----------+------+

| sum(age) | SN |

+----------+------+

| 21 | 1 |

| 22 | 2 |

| 23 | 3 |

| 21 | 4 |

| 22 | 5 |

+----------+------+

5 rows in set (0.00 sec)

mysql> select sum(age),salary from emp1 group by salary;

+----------+--------+

| sum(age) | salary |

+----------+--------+

| 21 | 20000 |

| 22 | 25000 |

| 45 | 30000 |

| 21 | 50000 |

+----------+--------+

4 rows in set (0.00 sec)

##HAVING:

Syntax: select sum(col name1),col name2 from

Table_name group by col name2 having sum(col name1)>5000;

EX: select sum(salary),SN from

Emp1 group by SN having sum(salary)<50000;

Output:

mysql> select sum(salary),age from emp1 group by age having sum(salary)<40000;

+-------------+------+

| sum(salary) | age |

+-------------+------+

| 30000 | 23 |

```
+-------------+------+

1 row in set (0.00 sec)


mysql> select sum(salary),age from emp1 group by age having sum(salary)<50000;

+-------------+------+

| sum(salary) | age  |

+-------------+------+

|       30000 |   23 |

+-------------+------+

1 row in set (0.00 sec)


mysql> select sum(salary),SN from emp1 group by SN having sum(salary)<50000;

+-------------+------+

| sum(salary) | SN   |

+-------------+------+

|       20000 |    1 |

|       25000 |    2 |

|       30000 |    3 |

|       30000 |    5 |

+-------------+------+

4 rows in set (0.00 sec)


mysql> select sum(salary),SN from emp1 group by SN having sum(salary)=50000;

+-------------+------+

| sum(salary) | SN   |

+-------------+------+

|       50000 |    4 |
```

+-------------+------+

1 row in set (0.00 sec)

LIMIT:

Syntax: select * from table name limit 3;

EX: select * from emp1 limit 3;

Output:

mysql> use teach2;

Database changed

mysql> select * from emp1;

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

+------+----------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

| 2 | srilatha | 22 | 628280097 | 25000 |

| 3 | suchi | 23 | 9346796446 | 30000 |

| 4 | mounika | 21 | 7032546792 | 50000 |

| 5 | navya | 22 | 8466850434 | 30000 |

+------+----------+------+--------------+--------+

5 rows in set (0.00 sec)

mysql> select * from emp1 limit 3;

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

+------+----------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

| 2 | srilatha | 22 | 628280097 | 25000 |

```
|   3 | suchi    |   23 | 9346796446   | 30000  |
+------+----------+------+--------------+--------+
3 rows in set (0.00 sec)
mysql> select * from emp1 limit 1;
+------+--------+------+--------------+--------+
| SN   | Name   | age  | phone_number | salary |
+------+--------+------+--------------+--------+
|   1 | anjali |   21 | 9704974315   | 20000  |
+------+--------+------+--------------+--------+
1 row in set (0.00 sec)

mysql> select * from emp1 limit 0;
Empty set (0.00 sec)

mysql> select * from emp1 limit 4;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
+------+----------+------+--------------+--------+
4 rows in set (0.00 sec)
```

* COL VALUE CHANGES:

Syntax: update table name set col name=’new value’ where condition;

EX: update data set Name=’Anjali’ where SN=’3’;

OUTPUT:

```
mysql> select * from data;

+------+----------+------+-----------+-------+

| SN   | Name     | CGPA | Grade     | marks |

+------+----------+------+-----------+-------+

|    1 | anusha   |  9.6 | excellent | -30   |

|    2 | ammu     |  8.5 | good      | -40   |

|    3 | akshitha |  6.5 | average   | NULL  |

+------+----------+------+-----------+-------+

3 rows in set (0.00 sec)


mysql> update data set Name='anjali' where SN='3';

Query OK, 1 row affected (0.03 sec)

Rows matched: 1  Changed: 1  Warnings: 0


mysql> select * from data;

+------+--------+------+-----------+-------+

| SN   | Name   | CGPA | Grade     | marks |

+------+--------+------+-----------+-------+

|    1 | anusha |  9.6 | excellent | -30   |

|    2 | ammu   |  8.5 | good      | -40   |

|    3 | anjali |  6.5 | average   | NULL  |

+------+--------+------+-----------+-------+

3 rows in set (0.00 sec)
```

ORDER BY:

arrange the data ascending or descending order.

But if we are not gave any order then it automatically in built ascending order.

Syntax to sort the records in ascending order:

SELECT ColumnName1,...,ColumnNameN FROM TableName ORDER BY ColumnName ASC;

EX: select * from emp1 order by salary asc;

select * from emp1 order by age asc;

OUTPUT:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
1. |    2 | srilatha |   22 | 628280097    | 25000  |
2. |    3 | suchi    |   23 | 9346796446   | 30000  |
3. |    4 | mounika  |   21 | 7032546792   | 50000  |
4. |    5 | navya    |   22 | 8466850434   | 30000  |
5. +------+----------+------+--------------+--------+
6. 5 rows in set (0.00 sec)
7. select * from emp1 order by age asc;
8.
9. +------+----------+------+--------------+--------+
10. | SN   | Name     | age  | phone_number | salary |
11. +------+----------+------+--------------+--------+
12. |    1 | anjali   |   21 | 9704974315   | 20000  |
13. |    4 | mounika  |   21 | 7032546792   | 50000  |
14. |    2 | srilatha |   22 | 628280097    | 25000  |
15. |    5 | navya    |   22 | 8466850434   | 30000  |
16. |    3 | suchi    |   23 | 9346796446   | 30000  |
17. +------+----------+------+--------------+--------+
```

```
5 rows in set (0.01 sec)
mysql> select * from emp1 order by salary asc;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)
```

Syntax to sort the records in descending order:

SELECT ColumnName1,...,ColumnNameN FROM TableName ORDER BY ColumnNameDESC;

EX: select * from emp1 order by age desc;

Output:

```
mysql> select * from emp1 order by age desc;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
|    1 | anjali   |   21 | 9704974315   | 20000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)

mysql> select * from emp1 order by salary desc;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    4 | mounika  |   21 | 7032546792   | 50000  |
```

```
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    1 | anjali   |   21 | 9704974315   | 20000  |
+------+----------+------+--------------+--------+
5 rows in set (0.02 sec)
```

Syntax to sort the records in ascending order without using ASC keyword:

Syntax: SELECT ColumnName1,...,ColumnNameN FROM TableName ORDER BY ColumnName;

EX: select * from emp1 order by salary;

OUTPUT:

```
mysql> select * from emp1 order by salary ;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)
```

Order by Random :

1. Syntax: SELECT column FROM table ORDER BY RAND () LIMIT 1;
   EX: EX: select * from emp1 order by rand() limit 1;

2. OUTPUT:

```
3. mysql> select * from emp1;
4. +------+----------+------+--------------+--------+
```

```
| SN  | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |  21 | 9704974315   | 20000  |
|   2 | srilatha |  22 | 628280097    | 25000  |
|   3 | suchi    |  23 | 9346796446   | 30000  |
|   4 | mounika  |  21 | 7032546792   | 50000  |
|   5 | navya    |  22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.02 sec)

mysql> select * from emp1 order by rand() limit1;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'limit1' at line 1
mysql> select * from emp1 order by rand() limit 1;
+------+--------+------+--------------+--------+
| SN  | Name   | age  | phone_number | salary |
+------+--------+------+--------------+--------+
|   1 | anjali |  21 | 9704974315   | 20000  |
+------+--------+------+--------------+--------+
1 row in set (0.03 sec)

mysql> select * from emp1 order by rand() limit 1;
+------+--------+------+--------------+--------+
| SN  | Name   | age  | phone_number | salary |
+------+--------+------+--------------+--------+
|   1 | anjali |  21 | 9704974315   | 20000  |
+------+--------+------+--------------+--------+
1 row in set (0.00 sec)

mysql> select * from emp1 order by rand() limit 5;
+------+----------+------+--------------+--------+
| SN  | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   5 | navya    |  22 | 8466850434   | 30000  |
|   2 | srilatha |  22 | 628280097    | 25000  |
|   1 | anjali   |  21 | 9704974315   | 20000  |
|   3 | suchi    |  23 | 9346796446   | 30000  |
|   4 | mounika  |  21 | 7032546792   | 50000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)
```

```
mysql> select * from emp1 order by rand() limit 2;
+------+---------+------+--------------+--------+
| SN   | Name    | age  | phone_number | salary |
+------+---------+------+--------------+--------+
|    4 | mounika |   21 | 7032546792   | 50000  |
|    1 | anjali  |   21 | 9704974315   | 20000  |
+------+---------+------+--------------+--------+
2 rows in set (0.00 sec)

mysql> select * from emp1 order by rand() limit 3;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    2 | srilatha |   22 | 628280097    | 25000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
+------+----------+------+--------------+--------+
3 rows in set (0.00 sec)

mysql> select * from emp1 order by rand() limit 0;
Empty set (0.00 sec)

mysql> select * from emp1 order by rand() ;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    4 | mounika  |   21 | 7032546792   | 50000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
|    1 | anjali   |   21 | 9704974315   | 20000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)
```

Order by multiple cols:

Syntax: select * from table name order by col name1,col name2;

EX: select * from emp1 order by Name;

OUTPUT:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
|   5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)

mysql> select * from emp1 order by Name;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
|   5 | navya    |   22 | 8466850434   | 30000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)

mysql> select * from emp1 order by age,SN;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|   1 | anjali   |   21 | 9704974315   | 20000  |
|   4 | mounika  |   21 | 7032546792   | 50000  |
|   2 | srilatha |   22 | 628280097    | 25000  |
|   5 | navya    |   22 | 8466850434   | 30000  |
|   3 | suchi    |   23 | 9346796446   | 30000  |
+------+----------+------+--------------+--------+
```

LEAST:

The LEAST is a SQL numeric function which shows the least value from the specified inputs in Structured Query Language.

Syntax of LEAST Function:

SELECT LEAST(Number1, Number2, Number3, Number4, ..., NumberN) AS Alias_Name;

EX: SELECT LEAST(2,4,6,7,10) AS least_of_numbers;

2.select age,salary,least(age,salary)as least_of_age_salary from emp1 where salary>10000;

Output:

```
mysql> select least(4,5,1,0.2,6,9,10)as least_of_numbers;
+------------------+
| least_of_numbers |
+------------------+
|              0.2 |
+------------------+
1 row in set (0.02 sec)

mysql> select least(null)as least_of_null;
ERROR 1582 (42000): Incorrect parameter count in the call to native function 'least'.
If we gave null then it shows error.
mysql> select least(4,5,1,-7,-9,9,10)as least_of_numbers;
+------------------+
| least_of_numbers |
+------------------+
|               -9 |
+------------------+
1 row in set (0.02 sec)
mysql> select age,salary,least(age,salary)as least_of_age_salary from emp1 where salary>10000;
+------+--------+----------------------+
| age  | salary | least_of_age_salary |
+------+--------+----------------------+
|   21 | 20000  | 20000                |
|   22 | 25000  | 22                   |
```

```
|  23 | 30000  | 23                  |
|  21 | 50000  | 21                  |
|  22 | 30000  | 22                  |
+------+--------+---------------------+
5 rows in set (0.00 sec)

mysql> select age,salary,least(age,salary)as least_of_salary from emp1 where salary>10000;
+------+--------+-----------------+
| age  | salary | least_of_salary |
+------+--------+-----------------+
|  21 | 20000  | 20000           |
|  22 | 25000  | 22              |
|  23 | 30000  | 23              |
|  21 | 50000  | 21              |
|  22 | 30000  | 22              |
+------+--------+-----------------+
5 rows in set (0.00 sec)

mysql> select age,salary,least(age,salary)as least_of_salary from emp1 where salary<50000;
+------+--------+-----------------+
| age  | salary | least_of_salary |
+------+--------+-----------------+
|  21 | 20000  | 20000           |
|  22 | 25000  | 22              |
|  23 | 30000  | 23              |
|  22 | 30000  | 22              |
+------+--------+-----------------+
4 rows in set (0.00 sec)

mysql> select salary,SN,least(salary,SN)as least_of_salary from emp1 where salary<50000;
+--------+------+-----------------+
| salary | SN   | least_of_salary |
+--------+------+-----------------+
| 20000  |    1 | 1               |
| 25000  |    2 | 2               |
| 30000  |    3 | 3               |
| 30000  |    5 | 30000           |
+--------+------+-----------------+
4 rows in set (0.00 sec)
```

```
mysql> select salary,SN,least(salary,SN)as least_of_salary from emp1 where salary;
+--------+------+-----------------+
| salary | SN   | least_of_salary |
+--------+------+-----------------+
| 20000  |    1 | 1               |
| 25000  |    2 | 2               |
| 30000  |    3 | 3               |
| 50000  |    4 | 4               |
| 30000  |    5 | 30000           |
+--------+------+-----------------+
5 rows in set (0.00 sec)
```

COMMENTS:

Syntax: select col name1,/* col name2,*/ col name3 from table name;

EX: select SN,/*Name,*/city,enroll_number from stud;

OUTPUT:

```
mysql> select * from stud;
+------+-------+------------+---------------+
| SN   | Name  | city       | enroll_number |
+------+-------+------------+---------------+
|    1 | anju  | warangal   | 2022          |
|    2 | srii  | hanamkonda | 2021          |
|    3 | suchi | mulugu     | 2020          |
+------+-------+------------+---------------+
3 rows in set (0.01 sec)


mysql> select SN,/*Name,*/ city,enroll_number from stud;
+------+------------+---------------+
| SN   | city       | enroll_number |
+------+------------+---------------+
```

```
|  1 | warangal   | 2022         |
|  2 | hanamkonda | 2021         |
|  3 | mulugu     | 2020         |
+------+------------+---------------+
3 rows in set (0.00 sec)
```

MULTILINE:

Syntax: /* select all the columns of all records in the table name1:*/

Select * from table name2;

EX: /* select all columns of all records in the emp1:*/

Select * from stud;

OUTPUT:

```
mysql> select * from stud;
+------+-------+------------+---------------+
| SN   | Name  | city       | enroll_number |
+------+-------+------------+---------------+
|  1 | anju  | warangal   | 2022         |
|  2 | srii  | hanamkonda | 2021         |
|  3 | suchi | mulugu     | 2020         |
+------+-------+------------+---------------+
3 rows in set (0.00 sec)

mysql> /* select all the columns of all records in the Name:*/ select * from stud;
+------+-------+------------+---------------+
| SN   | Name  | city       | enroll_number |
+------+-------+------------+---------------+
|  1 | anju  | warangal   | 2022         |
|  2 | srii  | hanamkonda | 2021         |
|  3 | suchi | mulugu     | 2020         |
+------+-------+------------+---------------+
3 rows in set (0.00 sec)

mysql> /* select all the columns of all records in the SN:*/ select * from stud;
```

```
+------+-------+------------+---------------+
| SN   | Name  | city       | enroll_number |
+------+-------+------------+---------------+
|    1 | anju  | warangal   | 2022          |
|    2 | srii  | hanamkonda | 2021          |
|    3 | suchi | mulugu     | 2020          |
+------+-------+------------+---------------+
3 rows in set (0.00 sec)
```

FUNCTIONS:

SQL functions are **divided** into two parts:

1. Aggregate Functions
2. Scalar Functions

SQL Aggregate Functions

SQL **Aggregate** functions return a single value which is calculated from the values.

- **AVG()**: It returns the average value of the column.
- **COUNT()**: It returns the number of rows in the table.
- **FIRST()**: It returns the first value of the column.
- **LAST()**: It returns the last value
- **MAX()**: It returns the largest value of the column.
- **MIN()**: It returns the smallest value of the column.
- **SUM()**: It returns the sum of rows of the table.

SQL Scalar functions

- **UCASE()**: It converts the database field to uppercase.
- **LCASE()**: It converts a field to lowercase.
- **MID()**: It extracts characters from the text field.
- **LEN()**: It returns the length of a text field.
- **ROUND()**: It rounds a numeric field to the number of decimals.
- **NOW()**: It returns the current date and time.
- **FORMAT()**: It formats how a field is to be displayed.

1. avg():

Syntax: select avg(col name) from table name;

EX: select avg(salary) from emp1;

OUTPUT:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)


mysql> select avg(salary) from emp1;
+-------------+
| avg(salary) |
+-------------+
|       31000 |
+-------------+
```

1 row in set (0.00 sec)

mysql> select avg(age) from emp1;

```
+----------+
| avg(age) |
+----------+
|  21.8000 |
+----------+
```

1 row in set (0.00 sec)

2. COUNT:

3. Syntax: select count(col name) from table name;
4. EX: select count(salary) from emp1;

Output:

5. mysql> select count(salary) from emp1;
6. +---------------+
7. | count(salary) |
8. +---------------+
9. | 5 |
10. +---------------+
11. 1 row in set (0.00 sec)
12. mysql> select count(salary) from emp1 where salary=30000;
13. +---------------+
14. | count(salary) |
15. +---------------+
16. | 2 |
17. +---------------+
18. 1 row in set (0.00 sec)
19. **Count and distinct:**
20. mysql> select count(distinct salary) from emp1;
21. +------------------------+

```
| count(distinct salary) |
+------------------------+
|                      4 |
+------------------------+
1 row in set (0.01 sec)

mysql> select count(distinct age) from emp1;
+---------------------+
| count(distinct age) |
+---------------------+
|                   3 |
+---------------------+
1 row in set (0.00 sec)
```

3.MAX AND MIN:

1. Syntax: select max(col name) from table name;
2. Syntax: select min(col name) from table name;

3. OUTPUT:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)
mysql> select max(age) from emp1;
+----------+
| max(age) |
+----------+
|       23 |
+----------+
1 row in set (0.00 sec)
```

```
mysql> select max(salary) from emp1;
+--------------+
| max(salary) |
+--------------+
| 50000       |
+--------------+
1 row in set (0.00 sec)

mysql> select min(salary) from emp1;
+--------------+
| min(salary) |
+--------------+
| 20000       |
+--------------+
1 row in set (0.00 sec)
```

4.SUM:

3. Syntax: select sum(col name) from table name;
4. EX: select sum(salary) from emp1;

OUTPUT:

```
mysql> select sum(salary) from emp1;
+--------------+
| sum(salary) |
+--------------+
|      155000 |
+--------------+
1 row in set (0.00 sec)

mysql> select sum(age) from emp1;
+-----------+
| sum(age) |
+-----------+
|      109 |
+-----------+
1 row in set (0.00 sec)
```

5.UPPER CASE AND LOWER CASE:

Syntax: select u case(col name) from table name;

Select l case(col name) from table name;

EX: select u case(Name) from emp1;

Select l case(Name) from emp2;

OUTPUT:

```
mysql> select ucase(Name) from emp1;

+-------------+

| ucase(Name) |

+-------------+

| ANJALI      |

| SRILATHA    |

| SUCHI       |

| MOUNIKA     |

| NAVYA       |

+-------------+

5 rows in set (0.02 sec)


mysql> select lcase(salary) from emp1;

+---------------+

| lcase(salary) |

+---------------+
```

```
| 20000         |
| 25000         |
| 30000         |
| 50000         |
| 30000         |
+---------------+
5 rows in set (0.02 sec)

mysql> select lcase(Name) from emp1;
+--------------+
| lcase(Name) |
+--------------+
| anjali      |
| srilatha    |
| suchi       |
| mounika     |
| navya       |
+--------------+
5 rows in set (0.00 sec)
```

6.MID:

Syntax: select mid(col name,value,value) from table name;

EX: select mid(Name,2,2) from emp1;

OUTPUT:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)
mysql> select mid(Name,2,2) from emp1;
+---------------+
| mid(Name,2,2) |
+---------------+
| nj            |
| ri            |
| uc            |
| ou            |
| av            |
+---------------+
5 rows in set (0.02 sec)
```

```
mysql> select mid(Name,3,2) from emp1;
+---------------+
| mid(Name,3,2) |
+---------------+
| ja            |
| il            |
| ch            |
| un            |
| vy            |
+---------------+
5 rows in set (0.00 sec)

mysql> select mid(SN,3,2) from emp1;
+-------------+
| mid(SN,3,2) |
+-------------+
|             |
|             |
|             |
|             |
|             |
+-------------+
5 rows in set (0.00 sec)
```

```
mysql> select mid(age,3,2) from emp1;
+--------------+
| mid(age,3,2) |
+--------------+
|              |
|              |
|              |
|              |
|              |
+--------------+
5 rows in set (0.00 sec)
```

7.ROUND:

Syntax: select round(col name) from table name;

EX: select round(cgpa) from data;

OUTPUT:

```
mysql> use teach6;
Database changed
mysql> create table data(SN int,Name varchar(30),CGPA float,Grade varchar(20));
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> insert into data(SN,Name,CGPA,Grade)values(1,'anusha',9.6,'excellent'),(2,'ammu',8.5,'good'),(3,'akshitha',6.5,'average');

Query OK, 3 rows affected (0.01 sec)

Records: 3  Duplicates: 0  Warnings: 0


mysql> select * from data;

+------+----------+------+-----------+

| SN   | Name     | CGPA | Grade     |

+------+----------+------+-----------+

|    1 | anusha   |  9.6 | excellent |

|    2 | ammu     |  8.5 | good      |

|    3 | akshitha |  6.5 | average   |

+------+----------+------+-----------+

3 rows in set (0.00 sec)


mysql> select round(CGPA) from data;

+-------------+

| round(CGPA) |

+-------------+

|          10 |

|           8 |

|           6 |

+-------------+

3 rows in set (0.01 sec)
```

DEGREE:

Syntax: select col name,degrees(col name) as degree value_of_col name From table name;

EX: select age, degrees(age) as degree value_of_age from table name;

OUTPUT:

```
mysql> select * from emp1;
+------+----------+------+--------------+--------+
| SN   | Name     | age  | phone_number | salary |
+------+----------+------+--------------+--------+
|    1 | anjali   |   21 | 9704974315   | 20000  |
|    2 | srilatha |   22 | 628280097    | 25000  |
|    3 | suchi    |   23 | 9346796446   | 30000  |
|    4 | mounika  |   21 | 7032546792   | 50000  |
|    5 | navya    |   22 | 8466850434   | 30000  |
+------+----------+------+--------------+--------+
5 rows in set (0.00 sec)
mysql> select age,degrees(age) as degreevalue_of_age from emp1;
+------+--------------------+
| age  | degreevalue_of_age |
+------+--------------------+
|   21 | 1203.2113697747288 |
|   22 |  1260.507149287811 |
```

| 23 | 1317.8029288008934 |

| 21 | 1203.2113697747288 |

| 22 | 1260.507149287811 |

+------+--------------------+

5 rows in set (0.02 sec)

FLOOR:

Syntax: select col name, floor(col name) as floor value_of_col name

From table name;

EX: select age, floor(age) as floor value_of_age from emp1;

OUTPUT:

mysql> select * from emp1;

+------+----------+------+--------------+--------+

| SN | Name | age | phone_number | salary |

+------+----------+------+--------------+--------+

| 1 | anjali | 21 | 9704974315 | 20000 |

| 2 | srilatha | 22 | 628280097 | 25000 |

| 3 | suchi | 23 | 9346796446 | 30000 |

| 4 | mounika | 21 | 7032546792 | 50000 |

| 5 | navya | 22 | 8466850434 | 30000 |

+------+----------+------+--------------+--------+

```
mysql> select age,floor(age) as floorvalue_of_age from emp1;
+------+------------------+
| age  | floorvalue_of_age |
+------+------------------+
|  21 |            21 |
|  22 |            22 |
|  23 |            23 |
|  21 |            21 |
|  22 |            22 |
+------+------------------+
5 rows in set (0.01 sec)

mysql> select salary,floor(salary) as floorvalue_of_salary from emp1;
+--------+----------------------+
| salary | floorvalue_of_salary |
+--------+----------------------+
| 20000  |            20000 |
| 25000  |            25000 |
| 30000  |            30000 |
| 50000  |            50000 |
| 30000  |            30000 |
+--------+----------------------+
5 rows in set (0.00 sec)
```

ABSOLUTE:

Syntax: select col name,abs(col name) as absolute_col name from

Table name

EX: select marks,abs(marks) as absolute_marks from data;

OUTPUT:

```
mysql> select * from data;
+------+----------+------+-----------+-------+
| SN   | Name     | CGPA | Grade     | marks |
+------+----------+------+-----------+-------+
|    1 | anusha   |  9.6 | excellent | -30   |
|    2 | ammu     |  8.5 | good      | -40   |
|    3 | akshitha |  6.5 | average   | NULL  |
+------+----------+------+-----------+-------+
3 rows in set (0.00 sec)


mysql> select marks,abs(marks) as absolute_marks from data;
+-------+----------------+
| marks | absolute_marks |
+-------+----------------+
| -30   |             30 |
| -40   |             40 |
| NULL  |           NULL |
```

+-------+----------------+

3 rows in set (0.00 sec)

REPEAT:

Syntax: select col name,repeat(col name,value) as repeat_value col name from table name;

EX: select Name,repeat(Name,3) as absolute_4Name from data;

OUTPUT:

mysql> select * from data;

+------+----------+------+-----------+-------+

| SN | Name | CGPA | Grade | marks |

+------+----------+------+-----------+-------+

| 1 | anusha | 9.6 | excellent | -30 |

| 2 | ammu | 8.5 | good | -40 |

| 3 | akshitha | 6.5 | average | NULL |

+------+----------+------+-----------+-------+

3 rows in set (0.00 sec)

mysql> select Name,repeat(Name,3) as absolute_4Name from data;

+----------+--------------------------+

| Name | absolute_4Name |

+----------+--------------------------+

| anusha | anushaanushaanusha |

| ammu | ammuammuammu |

| akshitha | akshithaakshithaakshitha |

+----------+--------------------------+

3 rows in set (0.00 sec)

MOD:

Syntax: select col name, mod(col name,value) as division_of_col name_by value from table name;

EX: select CGPA,mod(CGPA,50) as division_of_CGPA_by 50

From data;

OUTPUT:

mysql> select * from data;

+------+----------+------+-----------+--------+

| SN | Name | CGPA | Grade | marks |

+------+----------+------+-----------+--------+

| 1 | anusha | 9.6 | excellent | -30 |

| 2 | ammu | 8.5 | good | -40 |

| 3 | akshitha | 6.5 | average | NULL |

+------+----------+------+-----------+--------+

3 rows in set (0.00 sec)

mysql> select * from data;

+------+----------+------+-----------+--------+

| SN | Name | CGPA | Grade | marks |

```
+------+----------+------+-----------+-------+

|   1 | anusha   |  9.6 | excellent | -30   |

|   2 | ammu     |  8.5 | good      | -40   |

|   3 | akshitha |  6.5 | average   | NULL  |

+------+----------+------+-----------+-------+

3 rows in set (0.00 sec)
```

```
mysql> select CGPA, mod(CGPA,50) as division_of_CGPA_by50 from data;
+------+-----------------------+
| CGPA | division_of_CGPA_by50 |
+------+-----------------------+
|  9.6 |     9.600000381469727 |
|  8.5 |                   8.5 |
|  6.5 |                   6.5 |
+------+-----------------------+
3 rows in set (0.00 sec)
```

SUBSTR:

Syntax: select col name substr(col name,value1,value2) as substr_value1_value2 from table name;

EX: select Name substr(Name,2,2) as substr_2_2 from data;

OUTPUT:

```
mysql> select * from data;

+------+--------+------+-----------+-------+

| SN   | Name   | CGPA | Grade     | marks |

+------+--------+------+-----------+-------+

|   1 | anusha |  9.6 | excellent | -30   |

|   2 | ammu   |  8.5 | good      | -40   |
```

```
|    3 | anjali |  6.5 | average   | NULL  |
+------+--------+------+-----------+-------+
mysql> select Name,substr(Name,2,2) as substr_2_2 from data;
+--------+------------+
| Name   | substr_2_2 |
+--------+------------+
| anusha | nu         |
| ammu   | mm         |
| anjali | nj         |
+--------+------------+
3 rows in set (0.00 sec)
mysql> select Name,substr(Name,-2,2) as substr_-2_2 from data;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '-2_2 from data' at line 1
mysql> select Name,substr(Name,-3,2) as substr_-3_2 from data;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '-3_2 from data' at line 1
mysql> select Name,substr(Name,3,2) as substr_3_2 from data;
+--------+------------+
| Name   | substr_3_2 |
+--------+------------+
| anusha | us         |
| ammu   | mu         |
| anjali | ja         |
+--------+------------+
```

3 rows in set (0.00 sec)

LN FUNCTION:

Syntax: select col name, ln(col name) as ln_value_of_col name from table name;

EX: select SN,ln(col name) as ln_value_of_SN from data;

Output:

```
mysql> select * from data;
+------+--------+------+-----------+-------+
| SN   | Name   | CGPA | Grade     | marks |
+------+--------+------+-----------+-------+
|    1 | anusha |  9.6 | excellent | -30   |
|    2 | ammu   |  8.5 | good      | -40   |
|    3 | anjali |  6.5 | average   | NULL  |
+------+--------+------+-----------+-------+
3 rows in set (0.00 sec)
mysql> select SN ,ln(SN) as ln_value_of_SN from data;
+------+--------------------+
| SN   | ln_value_of_SN     |
+------+--------------------+
|    1 |                  0 |
|    2 | 0.6931471805599453 |
|    3 | 1.0986122886681098 |
```

+------+--------------------+

3 rows in set (0.02 sec)

mysql> select CGPA ,ln(CGPA) as ln_value_of_CGPA from data;

+------+--------------------+

| CGPA | ln_value_of_CGPA |

+------+--------------------+

| 9.6 | 2.2617631382102195 |

| 8.5 | 2.1400661634962708 |

| 6.5 | 1.8718021769015913 |

+------+--------------------+

3 rows in set (0.00 sec)

POSITON:

Syntax: select col name,position('value' in col name) as position_col name from table;

EX: select Name, position('a' in Name)as position_Name from data;

OUTPUT:

mysql> select * from data;

+------+--------+------+----------+-------+

| SN | Name | CGPA | Grade | marks |

+------+--------+------+----------+-------+

```
|    1 | anusha |  9.6 | excellent | -30   |
|    2 | ammu   |  8.5 | good      | -40   |
|    3 | anjali |  6.5 | average   | NULL  |
+------+--------+------+-----------+-------+
3 rows in set (0.00 sec)

mysql> select Name,position('a'in Name) as position_a from data;
+--------+------------+
| Name   | position_a |
+--------+------------+
| anusha |          1 |
| ammu   |          1 |
| anjali |          1 |
+--------+------------+
3 rows in set (0.02 sec)


mysql> select Grade,position('e'in Grade) as position_e from data;
+-----------+------------+
| Grade     | position_e |
+-----------+------------+
| excellent |          1 |
| good      |          0 |
| average   |          3 |
+-----------+------------+
3 rows in set (0.00 sec)
```