1. Insert data query

INSERT INTO table\_name (column\_name,column\_name) VALUES (1,2)

2. Insert multiple data in one time query

INSERT INTO table\_name (id,name,last\_name,address,age)

VALUES (44,"hamza","numan","lahore","44"),

(45,"numan","ashraf","lahore","43"),

(46,"hamza","numan","lahore","44")

3. Creat the table using query

CREATE TABLE personal( id INT , name VARCHAR(255),

city VARCHAR(255), gender VARCHAR(255) )

4. Constrain (Restriction)

CREATE TABLE kamboh( id INT NOT NULL , name VARCHAR(255) NOT NULL,

city VARCHAR(255) NOT NULL DEFAULT 'gujrnawala', gender VARCHAR(255) ,

age VARCHAR(255) CHECK(age>=18))

5. Select command

SELECT \* FROM table\_name

SELECT id,name,address FROM table\_name

6. Use of AS in query

SELECT id AS ID,name AS "Applicant Name",phone AS Mobile FROM table\_name

7. Where clause

SELECT \* FROM tbl\_name WHERE address=gujranwala

SELECT \* FROM tbl\_name WHERE id!=1

SELECT \* FROM tbl\_name WHERE age<15

SELECT \* FROM tbl\_name WHERE age>15

SELECT \* FROM tbl\_name WHERE age>=15

SELECT \* FROM tbl\_name WHERE age<=15

8. AND & OR & NOT operator

SELECT \* FROM `sql\_practice` WHERE age=28 AND address='gujranwala'

SELECT \* FROM `sql\_practice` WHERE age>15 AND cast='kamboh'

SELECT \* FROM `sql\_practice` WHERE age>=17 OR address='LAHORE'

SELECT \* FROM `sql\_practice` WHERE (age=15 OR age=28) AND cast='KAMBOH' // braket give one answer then combine other condition

SELECT \* FROM `sql\_practice` WHERE NOT age>=17 OR address='LAHORE'

9. IN & NOT IN operator

SELECT \* FROM `sql\_practice` WHERE age IN (18,15,28) // show data age is 18,15 and 28

SELECT \* FROM `sql\_practice` WHERE age NOT IN (18,15,28)

SELECT \* FROM `sql\_practice` WHERE address NOT IN ('gujranwala','lahore')

SELECT \* FROM `sql\_practice` WHERE address IN ('gujranwala','lahore')

10. BETWEEN operator

SELECT \* FROM `sql\_practice` WHERE age BETWEEN 15 AND 22

SELECT \* FROM `sql\_practice` WHERE name BETWEEN 'a' AND 'k' //show data A & K character

11. LIKE operator

SELECT \* FROM `sql\_practice` WHERE name LIKE "a%" // start with a alpha character

SELECT \* FROM `sql\_practice` WHERE name LIKE "%a" // end with a alpha character

SELECT \* FROM `sql\_practice` WHERE name LIKE "%a%" // center with a alpha character

SELECT \* FROM `sql\_practice` WHERE name LIKE "a%" OR name LIKE "S%" // two condition

SELECT \* FROM `sql\_practice` WHERE name NOT LIKE "%a" // reverse condition 1st

SELECT \* FROM `sql\_practice` WHERE name LIKE "a%n" // start with a and end with n alpha character

SELECT \* FROM `sql\_practice` WHERE name LIKE "\_a%" // 2nd character with a alpha character

12. Regular Expression

SELECT \* FROM `sql\_practice` WHERE name REGEXP "^al" //start with al character

SELECT \* FROM `sql\_practice` WHERE name REGEXP "imran$" //end with imran word

SELECT \* FROM `sql\_practice` WHERE name REGEXP "imran|kamran|safia" //search word these squence

SELECT \* FROM `sql\_practice` WHERE name REGEXP "imran$|kamran|^safia" //end with imran start with safia and kamran any position

SELECT \* FROM `sql\_practice` WHERE name REGEXP "[ls]" //search l and s character in any position in all words

SELECT \* FROM `sql\_practice` WHERE name REGEXP "[ls]a" //make combination ls,sa search

SELECT \* FROM `sql\_practice` WHERE name REGEXP "[b-s]" //search b to s alpha character

13. ORDER BY & DISTINCT

SELECT \* FROM `sql\_practice` ORDER BY name ASC // show ascending order

SELECT \* FROM `sql\_practice` ORDER BY name DESC // show descending order

SELECT DISTINCT address FROM `sql\_practice` // if 2 student from grw then only 1 row execute

SELECT DISTINCT age FROM `sql\_practice` ORDER BY age // reduce dublicity and order by ascending order

14. LIMIT & OFFSET

SELECT \* FROM `sql\_practice` LIMIT 3 // show first 3 record

SELECT \* FROM `sql\_practice` WHERE address='lahore' LIMIT 3

SELECT \* FROM `sql\_practice` LIMIT 3,3 // show next 3 records in first 3 offset

15. COUNT,MIN,MAX,AVG & SUM

SELECT COUNT(age) FROM `sql\_practice`

SELECT COUNT(DISTINCT age) FROM `sql\_practice` //reduce duplication

SELECT COUNT(DISTINCT age) AS Count FROM `sql\_practice` //new name a coulmn

SELECT MAX(age) FROM `sql\_practice`

SELECT MAX(age),name,address FROM `sql\_practice` // view how name is achived maximum number

SELECT MIM(age) FROM `sql\_practice`

SELECT AVG(age) FROM `sql\_practice`

SELECT SUM(age) FROM `sql\_practice`

16. Update query

UPDATE `sql\_practice` SET `name`="kami kamboh" WHERE id=1

UPDATE `sql\_practice` SET `age`="5555" , name='jani' WHERE id=5 //multiple column update in one time

UPDATE `sql\_practice` SET `age`="50" WHERE id IN (2,3) // two students record updated in one time

17. COMMIT & ROLLBACK

rollback work same as undo and reverse the transaction until commit command runs

after commit command execute rollback command did not working

18. DELETE

DELETE FROM sql\_practice where cast="jutt"

DELETE FROM sql\_practice where age>59

DELETE FROM sql\_practice where id=2

19. PRIMARY & FORIEGN KEY

create table city(

cid int not null auto\_increment, // primary key

cityname varchar(50) not null,

primary key (cid)

);

create table personal(

id int not null auto\_increment,

name varchar(50) not null,

age varchar(50) not null,

city int not null, // foreign key

gender varchar(50) not null,

primary key (id),

foreign key (city) references city (cid)

);

20. INNER JOIN (match the same value and show result)

SELECT \* FROM personal INNER JOIN city ON personal.city = city.cid

// first table is foreign key table and foreign key column and then other table name and

primary column in equal condition.

SELECT \* FROM personal p INNER JOIN city c ON p.city = c.cid; // use allies and reduce the table name character

SELECT p.id,p.name,p.age,c.cityname FROM personal p INNER JOIN city c ON p.city = c.cid;

// remove the two column that we do not need to show in inner join

SELECT p.id,p.name,p.age,c.cityname FROM personal p INNER JOIN city c ON p.city = c.cid

where c.cityname='lahore' order by p.name; // use where and order command in inner join

21. CROSS JOIN (Make combination of every city in on person)

select \* from personal cross join city;

22. LEFT & RIGHT JOIN

SELECT \* FROM personal p left JOIN city c ON p.city = c.cid

SELECT \* FROM personal p right JOIN city c ON p.city = c.cid

23. Multiple table join

select \* from personal inner join city

on personal.city=city.cid inner join year on personal.city=year.id;

24. GROUP BY

select cityname, count(cityname) from city group by cityname; // count total student who from

one district

25. SUBQUERY

select name from personal where city = (select cid from city where cityname = "lahore");

// this query show two table of data without join. find the name who from lahore

select name from personal where city in (select cid from city where cityname in ("lahore","grw"));

// if find more result who from grw and lahore

26. UNION

SELECT id,name FROM personal union select cid,cityname from city;

// combine two tables

27. IF Clause

select \* , if(age >=33,"Pass","Fail") as result from personal; // find who age below 33 and he or she is fail

28. CAUSE

select \* ,

case

when age =66 then "first"

when age =56 then "second"

when age =33 then "third"

when age =22 then "fail"

else "wrong information"

end as grade from personal

order by age desc; // multiple condition in clause