**MY SQL SYNTAXS**

select cust.\*,ord.\*

from customers cust inner join orders ord

on cust.customer\_id=ord.customer\_id

where cust.country='USA' and amount<500

order by ord.order\_id desc

create table:

create table family(name varchar(10),gender varchar(1),age int,relation varchar(10));

copy same table change table name:

create table customerdata as

select\*

from customers

add table in column:

alter table customerdata add location varchar;

update column data:

update customerdata

set location="canada"

where country="USA"

update customerdata

set location="dubai"

where country="UAE"

update customerdata

set location="london"

where country="UK"

sub\_query:

select \*

from customers

where customer\_id not in (select customer\_id from orders where amount>500)

order by customer\_id desc

select \*

from customers

where customer\_id in (select customer\_id from orders where amount>500)

order by customer\_id desc

select \*

from customers

where customer\_id in (select customer\_id from customerdata where location="london"and age=22)

order by customer\_id desc

select \*

from customers

where age in (select age from Family where age>30);

insert data in table:

insert into Family (name,gender,age,relation)

values("irf","f",31,"sis"),("tab","f",32,"sis"),(),........

family

name gender age relation

zub m 30 husband

zub m 30 husband

sul m 56 dad

zain m 1 son

raz f 50 mom

irf f 31 sis

tab f 32 sis

and or condition:

select \*

from Family

where relation='son' or relation='husband' or relation='mom'and gender='m'

except operator:

select age from customers

except

select country from customerdata

Output

age

22

25

28

31

intersect operator:

select age from customers

intersect

select age from customerdata

Output

age

22

25

28

31

union concept:

select age from customers

union

select age from customerdata

Output

age

22

25

28

31

union all:

select age from customers

union all

select age from customerdata

Output

age

31

22

22

25

28

31

22

22

25

28

insert into Friends(name,gender,qualification,age)

values ('santosh','m','btech','22'),('jaisri','f','btech','22'),('anwar','m','btech','30'),('parveen','f','degree','32'),('Durga','f','degree','22')

Friends

name gender qualification age

santosh m btech 22

jaisri f btech 22

anwar m btech 30

parveen f degree 32

Durga f degree 22

group by concept:

select age,count(gender) total

from Family

where age>20

group by age

select age,count(gender) total

from Family

where age>20

group by age

order by age

select \*

from Friends

where gender in (select gender from Family where age>20);

select customers.\*,orders.\*

from customers inner join orders

on customers .customer\_id=orders.order\_id

where customers.customer\_id!=5 and age<30

order by customers.customer\_id desc

SELECT customers.\*,count(age)

from customers

where country='USA'

group by customer\_id

SELECT customers.\*,count(age)

from customers

where country='USA'or country='UK'or country='UAE'

group by customer\_id

SELECT customers.\*,count(age)

from customers

where not country='USA'and( country='UK'or country='UAE')

group by customer\_id

SELECT customers.\*,count(age)

from customers

where country='USA'and not ( country='UK'or country='UAE')

group by customer\_id

q1)Select all the different values from the Country column in the Customers table.

SELECT DISTINCT Country

FROM Customers;

q2)Select all records from the Customers table, sort the result alphabetically, first by the column Country, then, by the column City.

SELECT \* FROM Customers

order by country,city;

1)Insert a new record in the Customers table.

INSERT INTO

Customers (CustomerName, Address, City, PostalCode,Country)

VALUES('Hekkan Burger','Gateveien 15','Sandnes','4306','Norway');

2)Select all records from the Customers where the PostalCode column is empty.

SELECT \* FROM Customers

WHERE PostalCode IS NULL;

3)Select all records from the Customers where the PostalCode column is NOT empty.

SELECT \* FROM Customers

WHERE postalcode is not null;

4)Update the City column of all records in the Customers table.

UPDATE Customers

SET City = 'Oslo';

5)Set the value of the City columns to 'Oslo', but only the ones where the Country column has the value "Norway".

update Customers

set City = 'Oslo'

where Country = 'Norway';

6)Update the City value and the Country value.

update Customers

set City = 'Oslo',country = 'Norway'

WHERE CustomerID = 32;

7)Delete all the records from the Customers table where the Country value is 'Norway'.

DELETE FROM Customers

WHERE Country = 'Norway';

8)Use the MIN function to select the record with the smallest value of the Price column.

SELECT MIN(price)

FROM Products;

9)Use an SQL function to select the record with the highest value of the Price column.

SELECT MAX(price)

FROM Products;

10)Use the correct function to return the number of records that have the Price value set to 18.

SELECT COUNT(\*)

FROM Products

WHERE Price = 18;

11)Use an SQL function to calculate the average price of all products.

SELECT avg(price)

FROM Products;

12)Use an SQL function to calculate the sum of all the Price column values in the Products able.

SELECT sum(price)

FROM Products;

13)Select all records where the value of the City column starts with the letter "a".

SELECT \* FROM Customers

WHERE City LIKE 'a%';

14)Select all records where the value of the City column contains the letter "a".

SELECT \* FROM Customers

WHERE City LIKE '%a%';

15)Select all records where the value of the City column starts with letter "a" and ends with the letter "b".

SELECT \* FROM Customers

WHERE City LIKE 'a%b';

16)Select all records where the value of the City column does NOT start with the letter "a".

SELECT \* FROM Customers

WHERE City NOT LIKE 'a%';

17)Insert the missing parts in the JOIN clause to join the two tables Orders and Customers, using the CustomerID field in both tables as the relationship between the two tables.

SELECT \*

FROM Orders LEFT JOIN Customers

ON Orders.CustomerID=Customers.CustomerID

18)Choose the correct JOIN clause to select all records from the two tables where there is a match in both tables.

SELECT \*

FROM Orders INNER JOIN Customers

ON Orders.CustomerID=Customers.CustomerID;

19)Choose the correct JOIN clause to select all the records from the Customers table plus all the matches in the Orders table.

SELECT \*

FROM Orders RIGHT JOIN Customers

ON Orders.CustomerID=Customers.CustomerID;

20)List the number of customers in each country.

SELECT COUNT(CustomerID),Country

FROM Customers

GROUP BY Country;

21)List the number of customers in each country, ordered by the country with the most customers first.

SELECT COUNT(CustomerID),Country

FROM Customers

GROUP BY Country

ORDER BY COUNT(CustomerID) DESC;

22)Write the correct SQL statement to delete a database named testDB.

DROP DATABASE testDB;

23)Write the correct SQL statement to create a new table called Persons.

CREATE TABLE Persons ( PersonID int,

LastName varchar(255),

FirstName varchar(255),

Address varchar(255),

City varchar(255) );

24)Write the correct SQL statement to delete a table called Persons.

DROP TABLE Persons;

25)Use the TRUNCATE statement to delete all data inside a table.

TRUNCATE TABLE Persons;

26)Add a column of type DATE called Birthday.

ALTER TABLE Persons

ADD Birthday DATE;

27)Delete the column Birthday from the Persons table.

ALTER TABLE Persons

DROP COLUMN Birthday;

28)Use the BETWEEN operator to select all the records where the value of the Price column is between 10 and 20.

SELECT \* FROM Products

WHERE Price BETWEEN 10 AND 20;

29)Use the BETWEEN operator to select all the records where the value of the Price column is NOT between 10 and 20.

SELECT \* FROM Products

WHERE Price NOT BETWEEN 10 AND 20;

30)Use the BETWEEN operator to select all the records where the value of the ProductName column is alphabetically between 'Geitost' and 'Pavlova'.

SELECT \* FROM Products

WHERE ProductName

BETWEEN 'Geitost' AND 'Pavlova';

31)Use the IN operator to select all the records where Country is either "Norway" or "France".

SELECT \* FROM Customers

WHERE Country IN('Norway', 'France');

32)Use the IN operator to select all the records where Country is NOT "Norway" and NOT "France".

SELECT \* FROM Customers

WHERE Country NOT IN ('Norway', 'France');

33)When displaying the Customers table, make an ALIAS of the PostalCode column, the column should be called Pno instead.

SELECT CustomerName,Address,PostalCode as pno

FROM Customers;

34)When displaying the Customers table, refer to the table as Consumers instead of Customers.

SELECT \*

FROM Customers AS Consumers;

create table nashhu.Friends (name varchar(100),gender varchar(10),qualification varchar(100), age int);

insert into Friends(name,gender,qualification,age) values ('santosh','m','btech','24');

insert into Friends(name,gender,qualification,age) values('jaisri','f','btech','22');

insert into Friends(name,gender,qualification,age) values('anwar','m','btech','30');

insert into Friends(name,gender,qualification,age) values('parveen','f','degree','32');

insert into Friends(name,gender,qualification,age) values('shabana','f','degree','32');

insert into Friends(name,gender,qualification,age) values('durga','f','degree','22');