**Answer of SET1, SET2 & SET3:**

CREATE DATABASE assignment;

USE assignment;

CREATE TABLE countries(Name CHAR(30), Population BIGINT, Capital VARCHAR(30) );

INSERT INTO countries VALUES('China',1382,'Beijing');

INSERT INTO countries VALUES('India',1326,'Delhi');

INSERT INTO countries VALUES('United States',324,'Washington D.C.');

INSERT INTO countries VALUES('Indonesia',260,'Jakarta');

INSERT INTO countries VALUES('Brazil',209,'Brasilia');

INSERT INTO countries VALUES('Pakistan',193,'Islamabad');

INSERT INTO countries VALUES('Nigeria',187,'Abuja');

INSERT INTO countries VALUES('Bangladesh',163,'Dhaka');

INSERT INTO countries VALUES('Russia',143,'Moscow');

INSERT INTO countries VALUES('Mexico',128,'Mexico City');

INSERT INTO countries VALUES('Japan',126,'Tokyo');

INSERT INTO countries VALUES('Philippines',102,'Manila');

INSERT INTO countries VALUES('Ethiopia',101,'Addis Ababa');

INSERT INTO countries VALUES('Vietnam',94,'Hanoi');

INSERT INTO countries VALUES('Egypt',93,'Ciaro');

INSERT INTO countries VALUES('Germany',81,'Berlin');

INSERT INTO countries VALUES('Iran',80,'Tehran');

INSERT INTO countries VALUES('Turkey',79,'Ankara');

INSERT INTO countries VALUES('Congo',79,'Kinshasa');

INSERT INTO countries VALUES('France',64,'Paris');

INSERT INTO countries VALUES('United Kingdom',65,'London');

INSERT INTO countries VALUES('Italy',60,'Rome');

INSERT INTO countries VALUES('South Afrca',55,'Pretoria');

INSERT INTO countries VALUES('Myanmar',54,'Naypyidaw');

SELECT \* FROM countries;

insert into Countries values ('Kuwait',2989,'Kuwait City');

insert into Countries values ('Peru',8852,'Lima');

update Countries Set name='New Delhi' where name='Delhi';

set SQL\_safe\_Updates=0;

update Countries Set Capital='New Delhi' where name='India';

Alter Table Countries rename to Big\_Countries;

use assignment;

select \* from big\_countries;

create table Product(

Product\_id int primary key auto\_increment,

Product\_Name char(30) Not null,

Description text(50)

);

alter table Product add Supplier\_id int primary key auto\_increment;

alter table product add primary key Auto\_increment (Supplier\_id);

select \* from product;

create table Product(

Product\_id int primary key auto\_increment,

Product\_Name char(30) Not null

Description text(50),

Supplier\_id int,

foreign key(supplier\_id) references Suppliers(Supplier\_id)

);

select \* from Product;

create table Suppliers(

Supplier\_id int primary key auto\_increment,

Supplier\_Name Char(30) Not null,

Location text(50)

);

select \* from suppliers;

create table Stock(

Id int primary key auto\_increment,

Product\_id int,

Balance\_Stock int,

foreign key(Product\_id) references product(Product\_id)

);

select \* from Stock;

insert into Suppliers(Supplier\_Name,Location) values ('Venkatesh','Chittore'),('Ganesh','Chittore'),('Ashwin','Vellore');

insert into product(Product\_Name,Description,Supplier\_id) values ('Realme Mobile','Mobile Phone',1),

('Philips Trimmer','Trimmer',2),

('Asus Gaming Laptop','Laptop',3);

insert into Stock(Product\_id,Balance\_Stock) values (2,1000),(3,550),(1,2500);

alter table Suppliers add primary key (Supplier\_Name);

select \* from emp;

alter table emp add DeptNo int;

update emp set DeptNo=

case

when emp\_no%2=0 then 20

when emp\_no%3=0 then 30

when emp\_no%4=0 then 40

when emp\_no%5=0 then 50

else 10

end;

alter table emp modify column deptno int after hire\_date;

set sql\_safe\_updates=0;

alter table emp rename column emp\_no to Emp\_ID;

create unique index Emp\_ID\_Unique on emp(emp\_id);

alter table Emp Rename Column Emp\_ID to Emp\_No;

create view emp\_sal as select emp\_no,concat(First\_Name,' ',Last\_name) as Employee,Salary from emp order by salary desc;

select \* from emp\_sal;

use assignment;

select \* from employee;

select \* from employee where deptno=10 && Salary>3000.00;

select \* from Students;

select \*, case

when marks between 40 and 50 then 'Third Class'

when marks between 50 and 60 then 'Second Class'

when marks between 60 and 80 then 'First Class'

when marks between 80 and 100 then 'Distinction'

else

'Failed'

end As Grades from Students;

alter table students add Grades Char(30);

set sql\_safe\_updates=0;

update Students Set Grades=

case

when marks between 40 and 50 then 'Third Class'

when marks between 50 and 60 then 'Second Class'

when marks between 60 and 80 then 'First Class'

when marks between 80 and 100 then 'Distinction'

else

'Failed'

end ;

select count(\*) FirstClassStudents from Students where Grades= 'First Class';

set Sql\_safe\_updates=0;

select count(\*) DistinctionStudents from Students where Grades= 'Distinction';

select count(\*) from Station;

select Distinct(City), id from Station where id%2=0 ;

select count(Distinct(City)) as Distinct\_City\_Names from station;

select count(City) As All\_City\_Names from station;

select count(City)- Count(Distinct(city)) as 'Difference Between Total\_Cities And Distinct\_Cities' from Station ;

select \* from station;

select distinct(City) from Station where left(city,1) in ('a','e','i','o','u');

select distinct(City) from Station where left(City,1) in ('a','e','i','o','u') and right(city,1) in ('a','e','i','o','u');

select Distinct(City) from Station where Left(City,1) not in ('a','e','i','o','u');

select distinct(City) from Station where left(City,1) not in ('a','e','i','o','u') and right(city,1) not in ('a','e','i','o','u');

select concat(first\_name,' ',Last\_name) as Emp\_Name, concat(Salary,'$') as 'Salary in $' ,hire\_date,

timestampdiff(month,hire\_date,now()) as Total\_Months from emp

where salary>2000 having Total\_Months<36 order by Salary Desc;

select \* from emp;

select \* from employee;

select deptno,sum(salary) as Total\_Salary from employee group by deptno ;

select \* from city;

select count(Name) as 'CitiesHavingPopulation>100000' from City where population>100000 ;

select district,sum(population) as 'TotalPopuationOfCalifornia' from City where District='California' group by district;

select District as Districts,avg(population) as 'AveragePopulationOfDistricts' from City group by District;

select \* from customers;

select \* from orders;

select O.ordernumber,O.status,O.customernumber,C.customername,O.comments from orders as O join customers as C

using (customernumber)

where O.status='Disputed' ;

select \* from payments;

select \* from customers;

select \* from orders;

select C.CustomerNumber,C.CustomerName,O.Status from Customers as C join Orders as O on C.Customernumber=O.customernumber;

select fname,salary from employee order by salary desc limit 2, 1;

select fname,salary,dense\_rank() over (order by salary desc) as 'Rank' from employee;