create database if not exists temp\_db;

use temp\_db;

# create a new table in the temp\_db

create table if not exists students (

Roll\_No bigint primary key ,

`name` varchar(20),

age int,

gender char(1),

hobbies varchar(50)

);

use temp\_db;

alter table students

add column `standard` int after age;

#add column

use temp\_db;

alter table students

add column(

`Native` varchar(15) ,

`Address` varchar(30)

);

#alter table

use temp\_db;

alter table students

modify gender varchar(20);

# rename table

alter table students

modify hometown varchar(20);

# rename table name

alter table students

rename to student\_data;

use temp\_db;

alter table student\_data

rename to students;

use temp\_db;

select age , gender from students;

select \* from students;

insert into students

values('100', 'Shiva', '24', '85', 'male', 'playing with bhakthas', 'shiva-lokha', 'Kailash');

insert into students

values('103','Shakthi','26','88','female','playing with bhakthas','devi-lokha','shakthi-peeth');

insert into students

values('105','Krishna','22','87','male','playing with bhakthas','krishna-lokha','Vrindavan');

#Condition on the table

select \* from students;

select `name` , age from students where gender = 'male';

SELECT `name` FROM students Where gender = 'male' and hobbies = 'playing with bhakthas' ;

#operators in mysql

select `name` from students where gender <> 'Female' and Roll\_No = 100;

#WILDCARD CHARACTERS "LIKE" keyword

use temp\_db;

select standard from students where `name` like 'S\_\_\_\_';

select 'name' from students where 'hobbies' like '% with %';

select \* from students;

select \* from students where `name` not in ('Krishna');

# Between

select \* from students;

select \* from students where age between 22 and 24;

select \* from students where gender = 'male' order by age desc;

use temp\_db;

insert into students ( Roll\_No, `name`, age, standard, gender, hobbies, hometown, Address)

values( '109', 'Ganapathy', '20', '5', 'male', 'Activating Root Chakra', 'Ganapathy-lokha', 'Sidhi Vinayak');

insert into students ( Roll\_No, `name`, age, standard, gender, hobbies, hometown, Address)

values ( '110', 'Skandha', '20', '5', 'male', 'Activating Swadisthana', 'Skandha-lokha', 'Pazhani');

select \* from students where `name` is null;

set sql\_safe\_updates = 0;

update students set hobbies = 'Changing lives of bhakthas' where `name` = 'Shakthi';

select \* from students;

use temp\_db;

create table if not exists teachers(

ID int,

Teacher\_Name varchar(20),

students\_handling varchar(20) ,

salary int

);

alter table teachers

modify students\_handling int;

insert into teachers (ID, Teacher\_Name, students\_handling, salary)

values ('100', 'Raghavendra','100', '120000');

select \* from teachers;

alter table teachers

add constraint fk\_number foreign key (students\_handling) references students(`name`);

# ALIAS

select \* from students;

select sum(age) as Total\_Age from students;

select sum(age) as Total\_Age , hobbies from students

group by hobbies;

#FUNCTIONS

use temp\_db;

select count(\*) as 'number of rows' from students;

select count(\*) - count(age) from students;

select \* from students where age is not null;

select sum(age) from students;

select sum(age) from students where gender = 'female';

select min(age) , max(age) from students;

select avg(age) from students;

#GroupBy

select avg(age), gender from students group by gender;

select sum(age), gender from students group by gender;

select avg(age) , `name`, gender from students where gender = 'male' group by `name`

order by avg(age) desc;

#DISTINCT

select `name`,gender from students;

select count(distinct `name`) from students;

#WHERE

select age,gender,avg(age) from students

where gender = 'male'

group by age

order by avg(age) asc;

#HAVING

select age,gender,avg(age)

from students

group by gender,age

having avg(age) >20;

select \* from students;

alter table students

add b\_date date;

set sql\_safe\_updates = 0;

update students

set b\_date = case

when Roll\_No = '104' then '1966-06-12'

when Roll\_No = '105' then '1970-05-17'

else '1990-02-25'

end;

select \* from students;

select gender,`name`,extract(year from b\_date) as Year\_Of\_Birth from students;

use temp\_db;

select \* from students;

#NESTED SUBQUERIES

select distinct `name`, age, hobbies from students

where age in(

select age from students where age > 20);

select \* from students;

select `name` , gender, b\_date from students where `name` in

(select distinct `name` from students where `name` is not null);

select \* from students;

select \* from teachers;

use temp\_db;

select \* from teachers;

alter table teachers;

insert into teachers

values(100,'Devi','Skandha',150000),

(101, 'Shiva', 'Ganapathy',150000);

set sql\_safe\_updates = 0;

delete from teachers where ID = 100;

select \* from teachers;

select \* from students as s left join teachers as t on s.name= t.students\_handling;

select s.Roll\_No,s.name, s.standard, t.Teacher\_Name,t.students\_handling from students as s left join teachers as t on s.name = t.students\_handling

where Teacher\_Name is null;

#INLINE SUBQUERY

select min(avg\_age) , max(avg\_age)

from (

select gender, avg(age) as avg\_age from students group by gender

) as age\_team ;

use temp\_db;

select \* from students;

#INLINE SUBQUERY

select age

from (

select gender,age from students

) as gender\_age;

select min(avg\_age\_gender.avg\_age),max(avg\_age\_gender.avg\_age)

from(

select avg(age) as avg\_age , gender from students group by gender

) as avg\_age\_gender ;

SELECT

MIN(age), MAX(age), gender

FROM

students

GROUP BY gender;

SELECT

`name`, gender, age

FROM

students

WHERE

age > '20';

#Common Table expression CTE

with students\_cte as (

select name, age, gender from students left join teachers on students.name = teachers.students\_handling

group by age,`name`,gender )

select \* from students\_cte;

#Correlated SubQueries

select \* from students

where

age > (select avg(age) from students where age = students.age ) ;