Name – Sayak Sen

Enrollment No – 2023CSB047

Assignment Sheet – 1

Assignment 1

Creation of the tables:

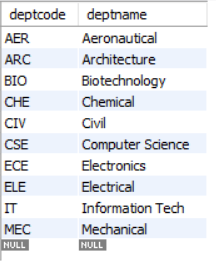
1.Depts Table

create table depts(

    deptcode varchar(3) primary key,

    deptname varchar(30) not null

);



2.Students Table

CREATE TABLE students (

    rollno INT PRIMARY KEY,

    name varchar(30),

    bdate date check (bdate < '1997-01-01'),

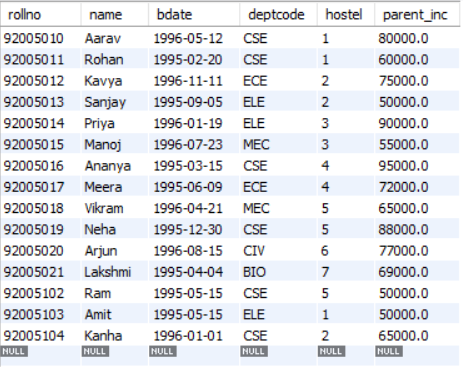
    deptcode varchar(3),

    hostel int check (hostel < 10),

    parent\_inc decimal(8,1),

    FOREIGN KEY (deptcode) REFERENCES depts(deptcode) ON DELETE CASCADE

)



2.Faculty Table

create table faculty(

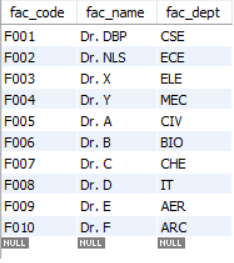
    fac\_code varchar(8) primary key,

    fac\_name varchar(30) not null,

    fac\_dept varchar(3),

    foreign key (fac\_dept) references depts(deptcode)

);



4.Course Offered Table

create table crs\_offrd(

    crs\_code varchar(5) primary key,

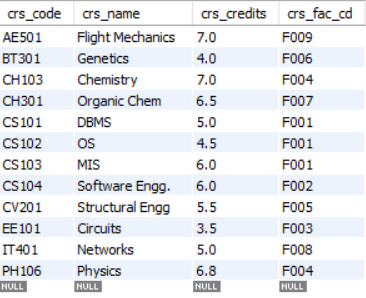
    crs\_name varchar(35) not null,

    crs\_credits decimal(2,1),

    crs\_fac\_cd varchar(8),

    foreign key (crs\_fac\_cd) references faculty(fac\_code)

);



5.Course Registered Table

create table crs\_regd(

    crs\_rollno int,

    crs\_cd varchar(5),

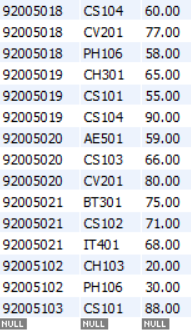
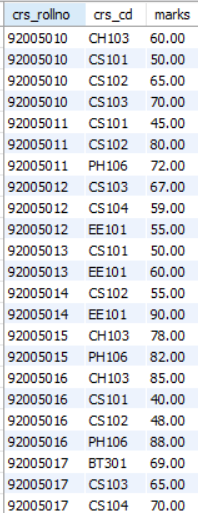
    marks decimal(5,2),

    primary key(crs\_rollno, crs\_cd),

    foreign key (crs\_rollno) references students(rollno),

    foreign key (crs\_cd) references crs\_offrd(crs\_code)

);



Queries

**1.Inserting Queries that violate the constraints**

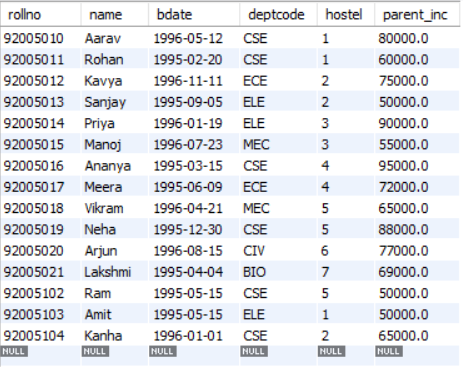
insert into students values(92005010,'John Doe','2000-05-15','CSE',1,50000);



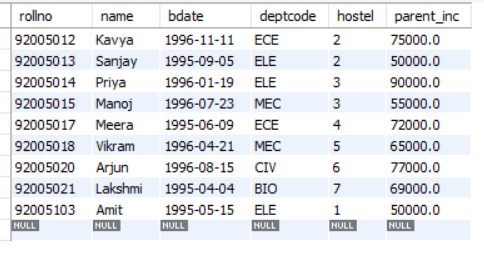
**2. Delete records from dept where deptcode='CSE'.**

delete from students where deptcode = 'CSE';

Before deleting :



After Deleting :



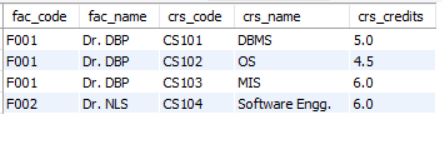
After deleting i re-inserted the values for future queries that require CSE students.

**3.Find out the courses offered by the faculty dbp and nls.**

select faculty.fac\_code, faculty.fac\_name, crs\_offrd.crs\_code, crs\_offrd.crs\_name, crs\_offrd.crs\_credits

from faculty JOIN

crs\_offrd ON faculty.fac\_code = crs\_offrd.crs\_fac\_cd where faculty.fac\_name = 'Dr. NLS' or faculty.fac\_name = 'Dr. DBP';

****

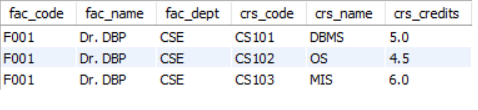
**4.Find out the courses with full details offered by dbp.**

select faculty.fac\_code, faculty.fac\_name, faculty.fac\_dept, crs\_offrd.crs\_code, crs\_offrd.crs\_name, crs\_offrd.crs\_credits

from faculty

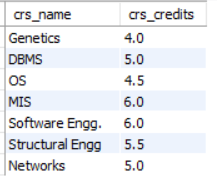
JOIN

crs\_offrd ON faculty.fac\_code = crs\_offrd.crs\_fac\_cd where faculty.fac\_name = 'Dr. DBP';

****

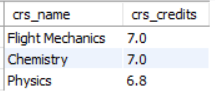
**5.Get the courses the credits of which lies between 4.0 and 6.0.**

select crs\_name,crs\_credits from crs\_offrd where crs\_credits between 4.0 and 6.0;

****

**6.Get the courses the credits of which are > 6.5.**

select crs\_name , crs\_credits from crs\_offrd where crs\_credits>6.5;

****

Assignment 2

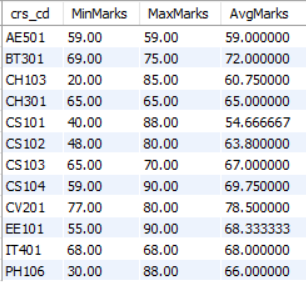
**1.Count the number of students in CSE dept.**

select count(\*) from students where deptcode = 'CSE';

****

**2.Determine the minimum, maximum and average marks of each courses.**

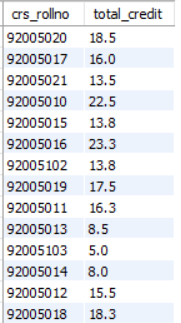
select crs\_cd,min(marks) as MinMarks,max(marks) as MaxMarks,avg(marks) as AvgMarks from crs\_regd group by crs\_cd;

****

**3.Determine the total credits of the courses registered by a student.**

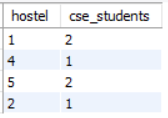
select crs\_rollno,sum(crs\_credits) as total\_credit from crs\_offrd

join crs\_regd on crs\_offrd.crs\_code = crs\_regd.crs\_cd group by crs\_rollno;

****

**4.Count the number of students in each hostel whose department is CSE.**

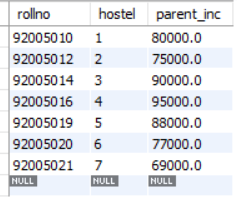
select hostel , count(\*) as cse\_students from students where deptcode = 'CSE' group by hostel;

****

**5.Display the hostel,rollno,parent\_inc of the student**

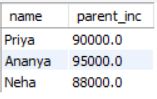
**who has the max(parent\_inc) in a hostel.**

select rollno,hostel,parent\_inc from students as s where parent\_inc = (select max(parent\_inc) from students where hostel = s.hostel);

****

**6.Display the name and parental income of each student greater than the parental income of some rollno 92005010.**

select name , parent\_inc from students where parent\_inc >(select parent\_inc from students where rollno = 92005010);

****

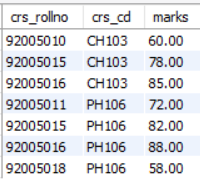
**7.Find out marks of students who have marks more than**

**rollno 92005102 for course CH103 and PH106.**

select \* from crs\_regd where (crs\_cd = 'CH103' and marks > (select marks from crs\_regd where crs\_rollno = 92005102 and crs\_cd = 'CH103'))

UNION

select \* from crs\_regd where (crs\_cd = 'PH106' and marks > (select marks from crs\_regd where crs\_rollno = 92005102 and crs\_cd = 'PH106'));

****

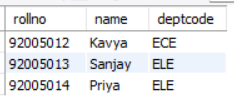
Assignment 3

**1.List students (rollno,name,deptcode) registered for course EE101.**

select rollno,name,deptcode from students

join crs\_regd on students.rollno = crs\_regd.crs\_rollno

where crs\_cd = 'EE101';

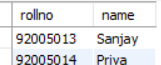
****

**2.List students (rollno,name) in ELE dept registered for course EE101.**

select rollno,name from students

join crs\_regd on students.rollno = crs\_regd.crs\_rollno

where deptcode = 'ELE' and crs\_cd = 'EE101';

****

**3.List students (rollno,name) in ELE dept not registered for course EE101.**

select rollno,name from students

join crs\_regd on students.rollno = crs\_regd.crs\_rollno

where deptcode = 'ELE' and rollno not in (select crs\_rollno from crs\_regd where crs\_cd = 'EE101');

****

**4.List the names of the students who have registered for both the courses 'DBMS' and 'OS'.**

select name from students

join crs\_regd on crs\_regd.crs\_rollno = students.rollno

join crs\_offrd on crs\_regd.crs\_cd = crs\_offrd.crs\_code

where crs\_name in ('DBMS','OS')

group by rollno

having count(DISTINCT crs\_name) = 2;

****

**5.Find the names of the faculty members who have offered either 'MIS' or 'Software Engg.'**

select fac\_name from faculty

join crs\_offrd on faculty.fac\_code = crs\_offrd.crs\_fac\_cd

where crs\_name = 'MIS'

UNION

select fac\_name from faculty

join crs\_offrd on faculty.fac\_code = crs\_offrd.crs\_fac\_cd

where crs\_name = 'Software Engg.';

****

**6.Find the names of the faculty members who have offered 'MIS' but not offered 'Software Engg.'**

select fac\_name from faculty

join crs\_offrd on faculty.fac\_code = crs\_offrd.crs\_fac\_cd

where crs\_name = 'MIS'

EXCEPT

select fac\_name from faculty

join crs\_offrd on faculty.fac\_code = crs\_offrd.crs\_fac\_cd

where crs\_name = 'Software Engg.';

****

**7.Find out the students in each hostel who are not registered for any course.**

select name,hostel from students

where rollno not in (select crs\_rollno from crs\_regd);

****

**8.Select the students who are in ELE dept or who have registered for course CS101.**

select rollno,name,deptcode from students

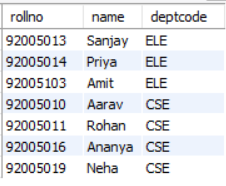
where deptcode = 'ELE'

UNION

select rollno,name,deptcode from students

join crs\_regd on students.rollno = crs\_regd.crs\_rollno

where crs\_cd = 'CS101';

****

**9.Display the students who have registered to all the courses.**

select rollno,name,deptcode from students

join crs\_regd on students.rollno = crs\_regd.crs\_rollno

group by rollno having count(crs\_cd) = (select count(\*) from crs\_offrd);

****

**10.Give Grace Marks 5 in subject ‘DBMS’to the students who have scored less than 50 in that subject.**

update crs\_regd

set marks = marks + 5

where crs\_cd = (select crs\_code from crs\_offrd where crs\_name = 'DBMS') and marks < 50;

****