**PROGRAM 4.**

**STUDENT FACULTY DATABASE**

**Consider the following database for student enrolment for course:**

**STUDENT (snum: integer, sname: string, major: string, level: string, age: integer)**

**CLASS (name: string, meets at: time, room: string, fid: integer)**

**ENROLLED (snum: integer, cname: string)**

**FACULTY (fid: integer, fname: string, deptid: integer)**

use Supriya;

create table student(

snum INT,

sname VARCHAR(10),

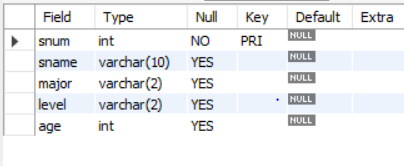
major VARCHAR(2),

level VARCHAR(2),

age INT,

primary key(snum));

desc student;



CREATE TABLE faculty(

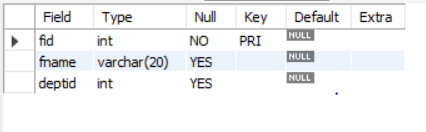
fid INT,

fname VARCHAR(20),

deptid INT,

PRIMARY KEY(fid));

desc faculty;



CREATE TABLE class(

cname VARCHAR(20),

meets\_at TIMESTAMP,

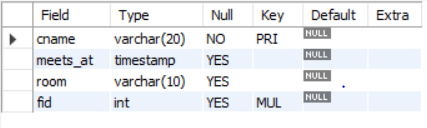
room VARCHAR(10),

fid INT,

PRIMARY KEY(cname),

FOREIGN KEY(fid) REFERENCES faculty(fid));

Desc class;



CREATE TABLE enrolled(

snum INT,

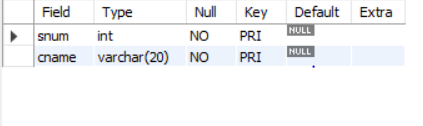
cname VARCHAR(20),

PRIMARY KEY(snum,cname),

FOREIGN KEY(snum) REFERENCES student(snum),

FOREIGN KEY(cname) REFERENCES class(cname));

desc enrolled;



insert into student values(1,'Ram','CS','JR',10);

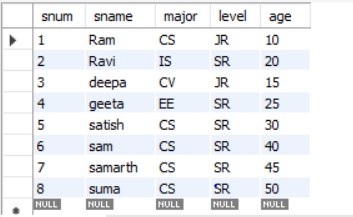
insert into student values(2,'Ravi','IS','SR',20);

insert into student values(3,'deepa','CV','JR',15);

insert into student values(4,'geeta','EE','SR',25);

insert into student values(5,'satish','CS','SR',30);

select \* from student;



insert into faculty values(1,'Seeta',10);

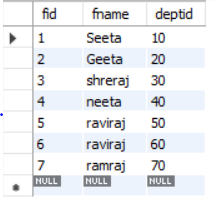
insert into faculty values(2,'Geeta',20);

insert into faculty values(3,'shreraj',30);

insert into faculty values(4,'neeta',40);

insert into faculty values(5,'raviraj',50);

select \* from faculty;



insert into class values('class1','12/11/15 10:15:16',1,1);

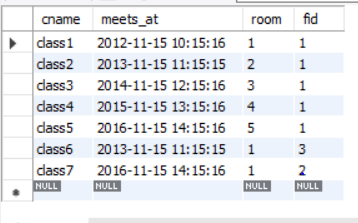
insert into class values('class2','13/11/15 11:15:15',2,2);

insert into class values('class3','14/11/15 12:15:16',3,3);

insert into class values('class4','15/11/15 13:15:16',4,4);

insert into class values('class5','16/11/15 14:15:16',5,5);

select \* from class;



insert into enrolled values(1,'class1');

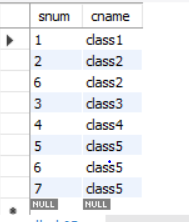
insert into enrolled values(2,'class2');

insert into enrolled values(3,'class3');

insert into enrolled values(4,'class4');

insert into enrolled values(5,'class5');

select \* from enrolled;



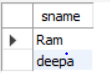
**i)Find the names of all Juniors (level= Jr) who are enrolled in a class taught by Seeta**

use Supriya;

select S.sname from student S where exists

(select S1.snum from student S1 where S.level='JR')AND EXISTS

(select F.fid from faculty F,enrolled E,class C where F.fname='Seeta' AND E.snum=S.snum AND E.cname = C.cname AND C.fid = F.fid );



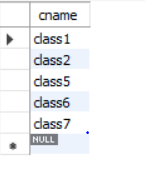
**ii. Find the names of all classes that either meet in room ‘ 1’ or have 2 or more Students enrolled.**

select c.cname

from class c

where c.room='1'

or c.cname in(select e.cname from enrolled e group by e.cname having count(\*)>=2);



**iii)Find the names of all students who are enrolled in two classes that meet at the same time.**

select distinct s.sname

from student s

where s.snum in(select e1.snum

               from enrolled e1,enrolled e2,class c1, class c2

   where e1.snum=e2.snum and e1.cname<>e2.cname

   and e1.cname=c1.cname

               and e2.cname=c2.cname

               and c1.meets\_at=c2.meets\_at);

**iv)Find the names of faculty members who teach in every room in which some class is taught.**

select f.fname

from faculty f

where f.fid in (select fid from class

group by fid having count(\*)=(select count(distinct room) from class));



**v)Find the names of faculty members for whom the combined enrolment of the courses that they teach is less than five.**

SELECT DISTINCT F.fname

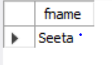
FROM faculty F

WHERE 5<(SELECT COUNT(E.snum)

FROM class C,enrolled E

WHERE C.cname = E.cname

AND C.fid = F.fid);



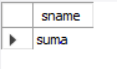
**vi)Find the names of students who are not enrolled in any class**

SELECT DISTINCT S.sname

FROM student S

WHERE S.snum NOT IN (SELECT E.snum

FROM enrolled E );



**vii. For each age value that appears in Students, find the level value that appears most often. For example, if there are more FR level students aged 18 than SR, JR, or SO students aged 18, you should print the pair (18, FR)**

SELECT S.age, S.level

FROM student S

GROUP BY S.age, S.level

HAVING S.level IN(SELECT S1.level FROM student S1

WHERE S1.age = S.age

GROUP BY S1.level,S1.age

HAVING COUNT(\*)>= ALL(SELECT COUNT(\*)

FROM student S2

WHERE s1.age = S2.age

GROUP BY S2.level, S2.age));

