# Medha Madhusudhan

#### 1BM19CS230

### CSE-4A

### **Program 4 - Student Database**

Consider the following database for student enrollment for course:

STUDENT(snum: integer, sname: string, major: string, lvl: string, age: integer)

CLASS(<u>cname</u>: string, meets at: time, room: string, fid: integer)

ENROLLED(<u>snum</u>: integer, <u>cname</u>: string)

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

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level(lvl) is a two character code with 4 different values (example: Junior: JR etc)

Write the following queries in SQL. No duplicates should be printed in any of the answers.

- i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by
- Find the names of all classes that either meet in room R128 or have five or more Students enrolled.
- iii. Find the names of all students who are enrolled in two classes that meet at the same time.
- iv. Find the names of faculty members who teach in every room in which some class is taught.
- v. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.
- vi. Find the names of students who are not enrolled in any class.

vii. For each age value that appears in Students, find the level value that appears most often.

```
create database studentfaculty;
use studentfaculty;
create table Student(
       snum int not null,
       sname varchar(10) not null,
       major varchar(2) not null,
       lvl varchar(2) not null,
       age int not null,
       primary key(snum)
);
create table Faculty(
       fid int not null,
       fname varchar(10) not null,
       deptid int not null,
       primary key(fid)
);
create table Class(
       cname varchar(10) not null,
       meetsat time not null,
```

```
room varchar(10) not null,
       fid int not null,
       primary key(cname),
       foreign key(fid) references Faculty(fid)
);
create table Enrolled(
       snum int not null,
       cname varchar(10) not null,
       primary key(snum,cname),
       foreign key(snum) references Student(snum),
       foreign key(cname) references Class(cname)
);
insert into Student(snum,sname,major,lvl,age)
       values (1,'Jhon','CS','Sr',19),
              (2,'Smith','CS','Jr',20),
              (3,'Jacob','CV','Sr',20),
              (4,'Tom','CS','Jr',20),
              (5,'Rahul','CS','Jr',20),
              (6,'Rita','CS','Sr',21);
insert into Faculty(fid,fname,deptid)
       values (11, 'Harish', 1000),
              (12, 'MV', 1000),
```

```
(13, 'Mira', 1001),
               (14, 'Shiva', 1002),
               (15,'Nupur',1000);
insert into Class(cname,meetsat,room,fid)
       values ('Class1','10:15:16','R1',14),
               ('Class10','10:15:16','R128',14),
               ('Class2','10:15:20','R2',12),
               ('Class3','10:15:25','R3',11),
               ('Class4','20:15:20','R4',14),
               ('Class5','20:15:20','R3',15),
               ('Class6','13:20:20','R2',14),
               ('Class7','10:10:10','R3',14);
insert into Enrolled(snum,cname)
       values (1,'Class1'),
               (2,'Class1'),
               (3,'Class3'),
               (4,'Class3'),
               (5,'Class4');
insert into Enrolled(snum,cname)
       values (1, 'Class5'),
               (2,'Class5'),
               (3,'Class5'),
```

```
(4,'Class5'),
(5,'Class5');
```

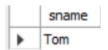
### -- find names of all juniors who are enrolled in a class taught by 'Harish'

select s.sname from Student s,Enrolled e

where s.snum = e.snum

and s.lvl = 'Jr'

and e.cname in (select cname from Class where fid = (select fid from Faculty where fname = 'Harish'));



#### -- find the name of all classes that either meet in room128 or have >=5 students

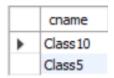
select c.cname from Class c

where c.room = 'R128'

union

select distinct e.cname from Enrolled e

where 5 <= (select count(snum) from Enrolled where cname = e.cname);

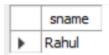


# -- find the names of all students who are enrolled in two classes that meet at the same time

select distinct s.sname from Student s,Class c,Enrolled e

where s.snum = e.snum and c.cname = e.cname

and exists (select 'X' from Class c1,Enrolled e1 where c1.cname = e1.cname and c1.meetsat = c.meetsat and e1.snum = e.snum and c1.cname != e.cname);



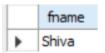
### -- find the names of faculty who teach in every room in which class is taught

select f.fname from Faculty f,Class c

where f.fid = c.fid

group by f.fid

having count(f.fid) = (select count(distinct room) from Class);



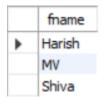
# -- find the names of faculty members for whom combined enrollment of courses that they teach is less than five

select distinct f.fname

from Class c, Faculty f

where c.fid = f.fid

and 5 > (select count(snum) from enrolled where cname in (select cname from Class where Class.fid = c.fid));



### -- find the names of students who are not enrolled in any class

select s.sname from Student s

where not exists (select 'X' from Enrolled where snum = s.snum);



## -- for each age value that appears in Students, find the level that appears the most

select s.age,s.lvl from Student s

where s.lvl = 'Jr'

group by s.age

having count(s.lvl) > (select count(lvl) from Student where lvl = 'Sr' and age = s.age)

union

select s.age,s.lvl from Student s

where s.lvl = 'Sr'

group by s.age

having count(s.lvl) > (select count(lvl) from Student where lvl ='Jr' and age = s.age);

	age	lvl
•	20	Jr
	19	Sr
	21	Sr