PROGRAM 4: STUDENT FACULTY DATABASE

Consider the following database for student enrollment for course :

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

CLASS(cname: string, meetsat: time, room: string, fid: integer)

ENROLLED(snum: integer, cname: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 "name"
ii. 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. 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).

SQL> select * from student;

SNUM	SNAME	MA	LV	AGE
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

SQL> select * from faculty;

FID	FNAME	DEPTID
11	Harish	1000
12	MV	1000
13	Mira	1001
14	Shiva	1002
15	Nupur	1000

SQL> select * from class;

CNAME	METTS_A	ROOM	FID
Class1	12/11/15 10:15:16.00000	R1	14
Class10	12/11/15 10:15:16.00000	R128	14
Class2	12/11/15 10:15:20.000000	R2	12
Class3	12/11/15 10:15:25.000000	R3	11
Class4	12/11/15 20:15:20.000000	R4	14
Class5	12/11/15 20:15:20.000000	R3	15
Class6	12/11/15 13:20:20.000000	R2	14
Class7	12/11/15 10:10:10.000000	R3	14

SQL> select * from enrolled;

SNUM	CNAME
1	class1
2	class1
3	class3
4	class3
5	class4

Program:-

```
create database studentfaculty4;
use studentfaculty4;

create table STUDENT(
snum int,
sname varchar(60),
major varchar(50),
lvl varchar(50),
age int,
primary key(snum)
);

create table CLASS(
cname varchar(60),
meetsat timestamp,
room varchar(60),
fid int,
primary key (cname)
);

create table enrolled(
snum int,
cname varchar(60),
primary key(snum,cname),
foreign key(snum) references STUDENT(snum)
on update cascade on delete cascade,
foreign key(cname) references CLASS(cname)
```

```
on update cascade on delete cascade

);


create table FACULTY(

fid int,

fname varchar(60),

deptid int,

primary key(fid)

);


insert into STUDENT values (1,'Jhon','CS','Jr',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);

update STUDENT set lvl='Sr' where sname='Jhon';

insert into CLASS values ('Class1',"12/11/15 10:15:16.00000",'R1',14);

select * from CLASS;

delete from CLASS where cname='Class1';

select * from CLASS;

insert into CLASS values ('Class1',"15/11/12 10:15:16.00000",'R1',14);

select * from CLASS;

insert into CLASS values ('Class10',"15/11/12 10:15:16.00000",'R128',14), ('Class2',"15/11/12
10:15:20.00000",'R2',12),

('Class3',"15/11/12 10:15:25.00000",'R3',11), ('Class4',"15/11/12 10:15:20.00000",'R4',14),
('Class5',"15/11/12 10:15:20.00000",'R3',15),

('Class6',"15/11/12 13:20:20.00000",'R2',14), ('Class7',"15/11/12 10:10:10.00000",'R3',14);


insert into ENROLLED values (1,'Class1'),(2,'Class1'),(3,'Class3'),(4,'Class3'),(5,'Class4');


insert into FACULTY values
(11,'Harish',1000),(12,'MV',1000),(13,'Mira',1001),(14,'Shiva',1002),(15,'Nupur',1000);
```

select * from STUDENT;

snum	sname	major	lvl	age
1	Jhon	CS	Jr	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
NULL	NULL	NULL	NULL	NULL

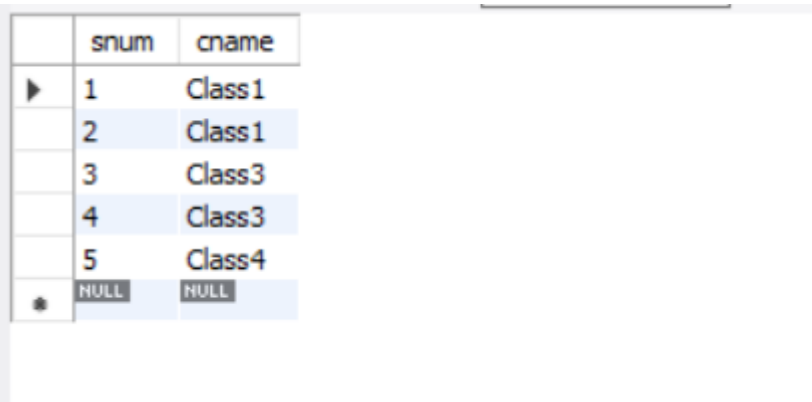
select * from CLASS;

cname	meetsat	room	fid
Class1	2015-11-12 10:15:16	R1	14
Class10	2015-11-12 10:15:16	R128	14
Class2	2015-11-12 10:15:20	R2	12
Class3	2015-11-12 10:15:25	R3	11
Class4	2015-11-12 10:15:20	R4	14
Class5	2015-11-12 10:15:20	R3	15
Class6	2015-11-12 13:20:20	R2	14
Class7	2015-11-12 10:10:10	R3	14
NULL	NULL	NULL	NULL

select * from FACULTY;

fid	fname	deptid
11	Harish	1000
12	MV	1000
13	Mira	1001
14	Shiva	1002
15	Nupur	1000
NULL	NULL	NULL

select * from ENROLLED;



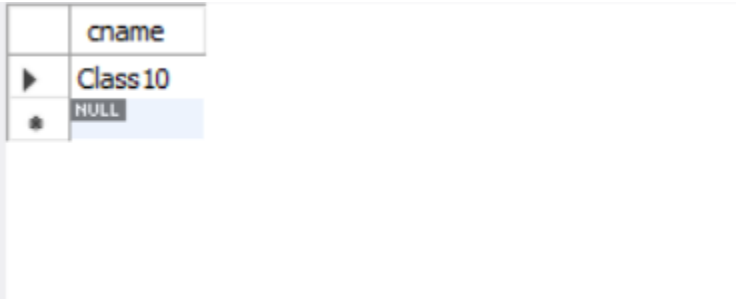
	snum	cname
▶	1	Class1
	2	Class1
	3	Class3
	4	Class3
	5	Class4
*	NULL	NULL

SELECT DISTINCT s.sname FROM STUDENT s, CLASS c, ENROLLED e, FACULTY f WHERE s.snum = e.snum AND

e.cname = c.cname AND c.fid = f.fid AND f.fname = 'Shiva' AND s.lvl = 'Jr';

	sname
▶	Jhon
	Smith
	Rahul

SELECT c.cname FROM CLASS c WHERE c.room = 'R128' OR c.cname IN (SELECT e.cname FROM ENROLLED e GROUP BY e.cname HAVING COUNT(e.snum)>4);

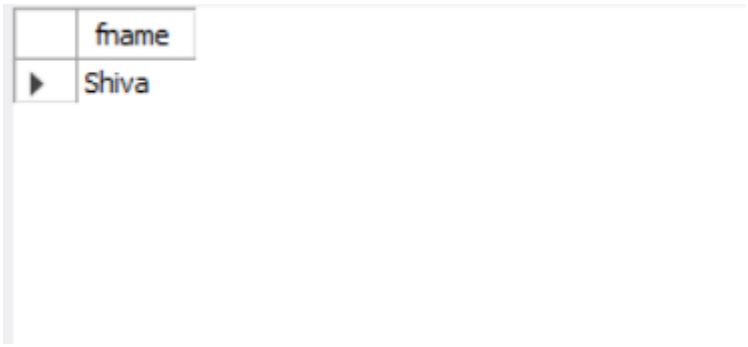


	cname
▶	Class10
*	NULL

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.meetsat = c2.meetsat);

	sname

```
create table ref1 as(
select count(distinct(CLASS.room))as c from CLASS where CLASS.cname in
(select distinct(ENROLLED.cname) from ENROLLED)
);
select FACULTY.fname from FACULTY where FACULTY.fid in (select (CLASS.fid)
from CLASS where class.room in( select distinct(class.room) from class
where CLASS.cname in (select distinct(ENROLLED.cname) from ENROLLED))
group by CLASS.fid having count(*)=(select ref1.c from ref1));
```



	fname
▶	Shiva

```
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);
```

	fname
▶	Harish
	MV
	Mira
	Shiva
	Nupur

SELECT DISTINCT s.sname FROM STUDENT s WHERE s.snum NOT IN(SELECT e.snum FROM ENROLLED e);

	sname
▶	Rita

SELECT s.age, s.lvl FROM STUDENT s GROUP BY s.age, s.lvl HAVING s.lvl IN (SELECT s1.lvl FROM STUDENT s1 WHERE s1.age=s.age GROUP BY s1.age, s1.lvl HAVING COUNT(*) >= ALL (SELECT COUNT(*) FROM STUDENT s2 WHERE s1.age=s2.age GROUP BY s2.lvl, s2.age)) ORDER BY s.age;

	age	lvl
▶	19	Jr
	20	Jr
	21	Sr