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)
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 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.

--

```
create database student_faculty;
use student_faculty;

create table student(
snum INT,
 sname VARCHAR(10),
 major VARCHAR(2),
 lvl 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),
metts_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, 'John', 'CS', 'Sr', 19);
INSERT INTO STUDENT VALUES(2, 'Smith', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(3 , 'Jacob', 'CV', 'Sr', 20);
INSERT INTO STUDENT VALUES(4, 'Tom ', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(5, 'Rahul', 'CS', 'Jr', 20);
INSERT INTO STUDENT VALUES(6, 'Rita', 'CS', 'Sr', 21);
```

```
commit;

select * from student;

INSERT INTO FACULTY VALUES(11, 'Harish', 1000);
INSERT INTO FACULTY VALUES(12, 'MV', 1000);
INSERT INTO FACULTY VALUES(13 , 'Mira', 1001);
INSERT INTO FACULTY VALUES(14, 'Shiva', 1002);
INSERT INTO FACULTY VALUES(15, 'Nupur', 1000);
commit;

select * from faculty;

insert into class values('class1', '12/11/15 10:15:16', 'R1', 14);
insert into class values('class10', '12/11/15 10:15:16', 'R128', 14);
insert into class values('class2', '12/11/15 10:15:20', 'R2', 12);
insert into class values('class3', '12/11/15 10:15:25', 'R3', 11);
insert into class values('class4', '12/11/15 20:15:20', 'R4', 14);
insert into class values('class5', '12/11/15 20:15:20', 'R3', 15);
 insert into class values('class6', '12/11/15 13:20:20', 'R2', 14);
insert into class values('class7', '12/11/15 10:10:10', 'R3', 14);
commit;

select * from class;

insert into enrolled values(1, 'class1');
insert into enrolled values(2, 'class1');
insert into enrolled values(3, 'class3');
insert into enrolled values(4, 'class3');
insert into enrolled values(5, 'class4');
```

insert into enrolled values(1, 'class5');

insert into enrolled values(2, 'class5');

insert into enrolled values(3, 'class5');

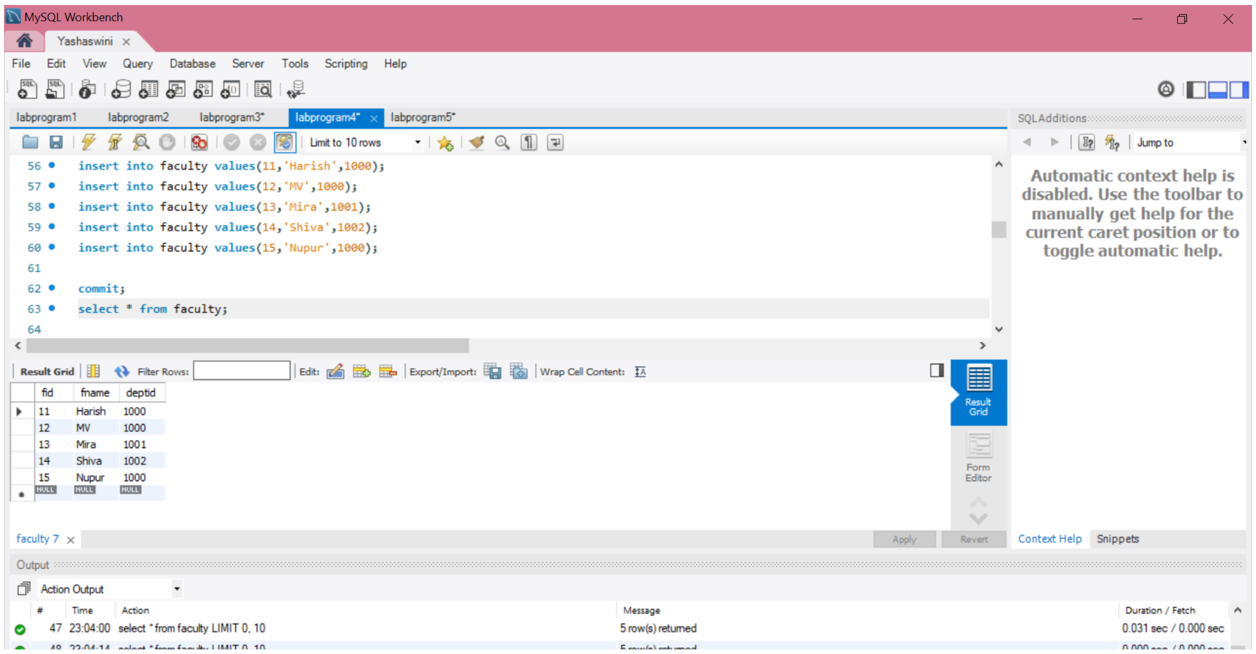
insert into enrolled values(4, 'class5');

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

commit;

select * from enrolled;

STUDENT TABLE



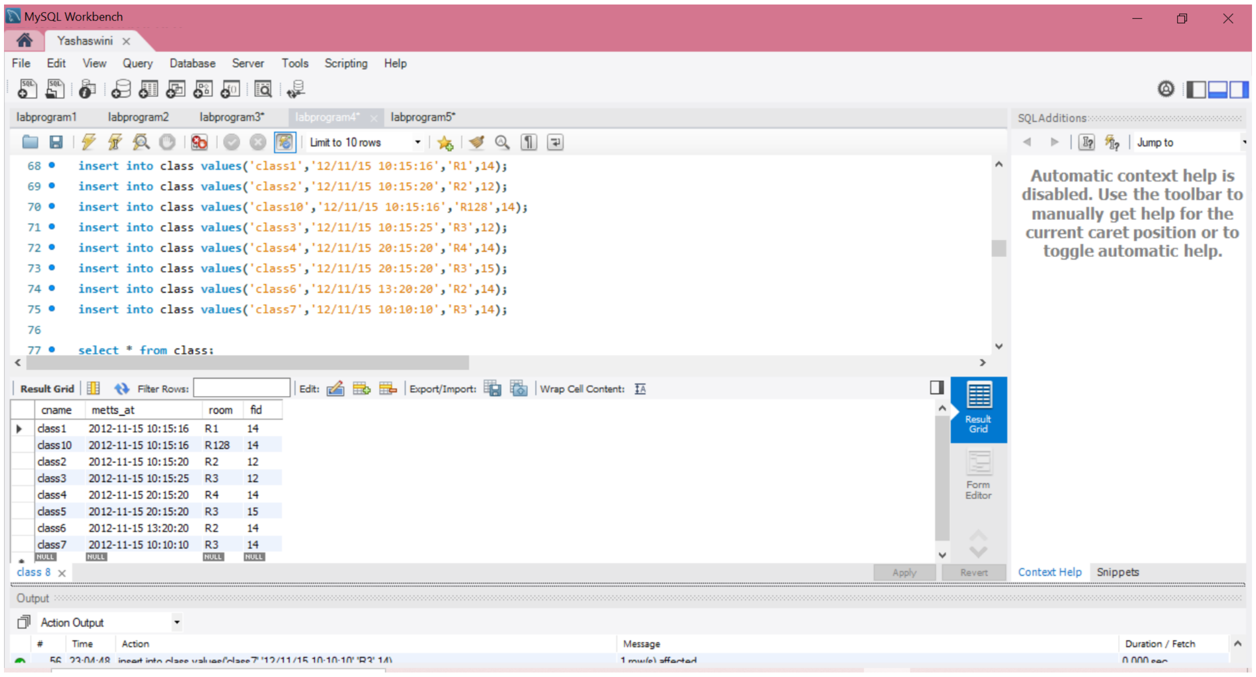
```
insert into faculty values(11,'Harish',1000);
insert into faculty values(12,'MV',1000);
insert into faculty values(13,'Mira',1001);
insert into faculty values(14,'Shiva',1002);
insert into faculty values(15,'Nupur',1000);

commit;
select * from faculty;
```

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

```
#   Time      Action                             Message            Duration / Fetch
47  23:04:00  select * from faculty LIMIT 0, 10  5 row(s) returned  0.031 sec / 0.000 sec
```

CLASS TABLE



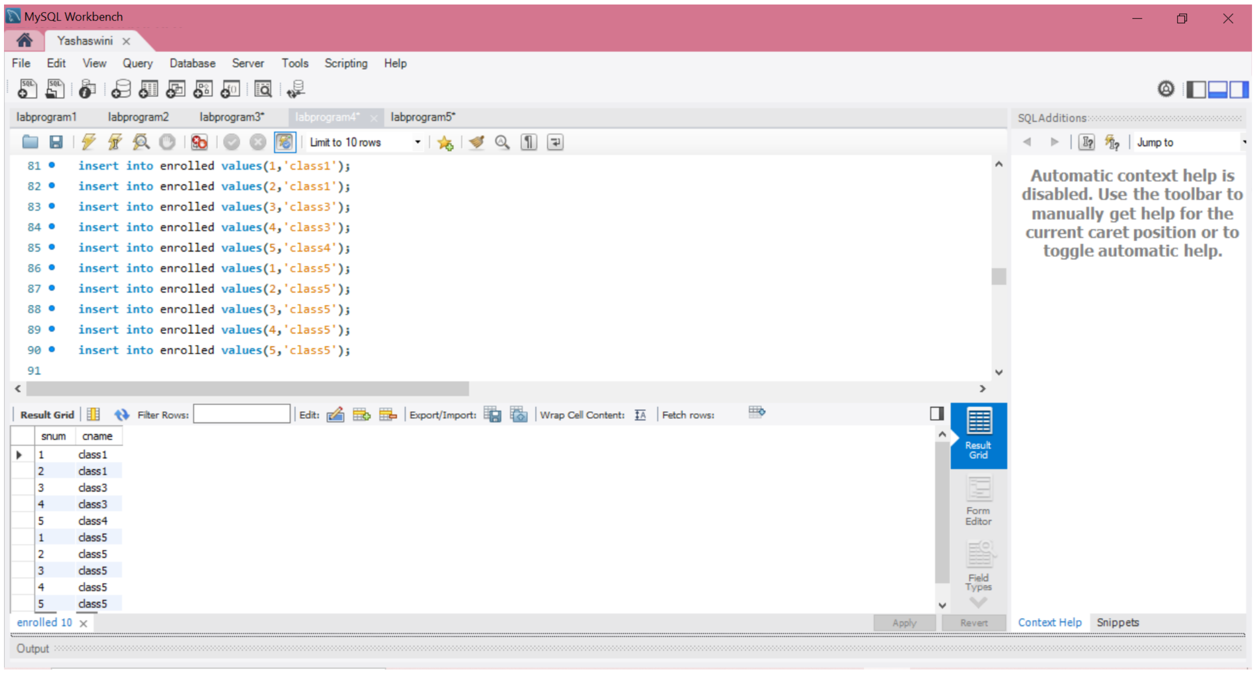
```
insert into class values('class1','12/11/15 10:15:16','R1',14);
insert into class values('class2','12/11/15 10:15:20','R2',12);
insert into class values('class10','12/11/15 10:15:16','R128',14);
insert into class values('class3','12/11/15 10:15:25','R3',12);
insert into class values('class4','12/11/15 20:15:20','R4',14);
insert into class values('class5','12/11/15 20:15:20','R3',15);
insert into class values('class6','12/11/15 13:20:20','R2',14);
insert into class values('class7','12/11/15 10:10:10','R3',14);

select * from class;
```

cname	metts_at	room	fid
class1	2012-11-15 10:15:16	R1	14
class10	2012-11-15 10:15:16	R128	14
class2	2012-11-15 10:15:20	R2	12
class3	2012-11-15 10:15:25	R3	12
class4	2012-11-15 20:15:20	R4	14
class5	2012-11-15 20:15:20	R3	15
class6	2012-11-15 13:20:20	R2	14
class7	2012-11-15 10:10:10	R3	14
NULL	NULL	NULL	NULL

```
#   Time      Action                                                   Message              Duration / Fetch
56  23:04:48  insert into class values('class7','12/11/15 10:10:10','R3',14)  1 row(s) affected  0.000 sec
```

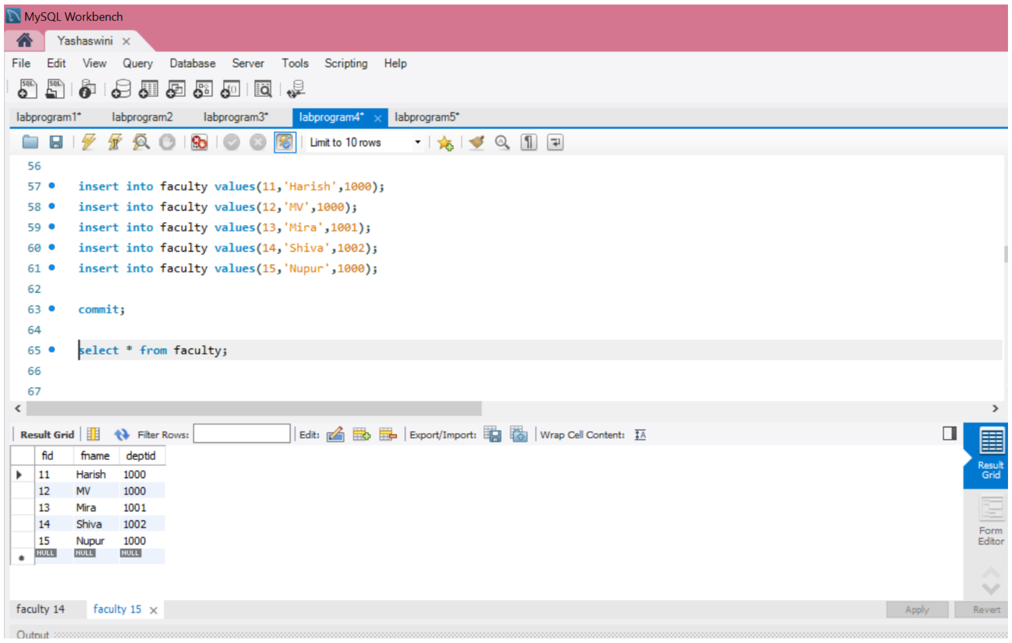
ENROLLED TABLE



```
insert into enrolled values(1,'class1');
insert into enrolled values(2,'class1');
insert into enrolled values(3,'class3');
insert into enrolled values(4,'class3');
insert into enrolled values(5,'class4');
insert into enrolled values(1,'class5');
insert into enrolled values(2,'class5');
insert into enrolled values(3,'class5');
insert into enrolled values(4,'class5');
insert into enrolled values(5,'class5');
```

snum	cname
1	class1
2	class1
3	class3
4	class3
5	class4
1	class5
2	class5
3	class5
4	class5
5	class5

FACULTY TABLE



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

commit;

select * from faculty;
```

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

MySQL Workbench
Yashaswini
File Edit View Query Database Server Tools Scripting Help
labprogram1* labprogram2 labprogram3* labprogram4* labprogram5*
Limit to 10 rows
Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:
Result Grid
Form Editor
faculty 14 faculty 15
Apply Revert
Output

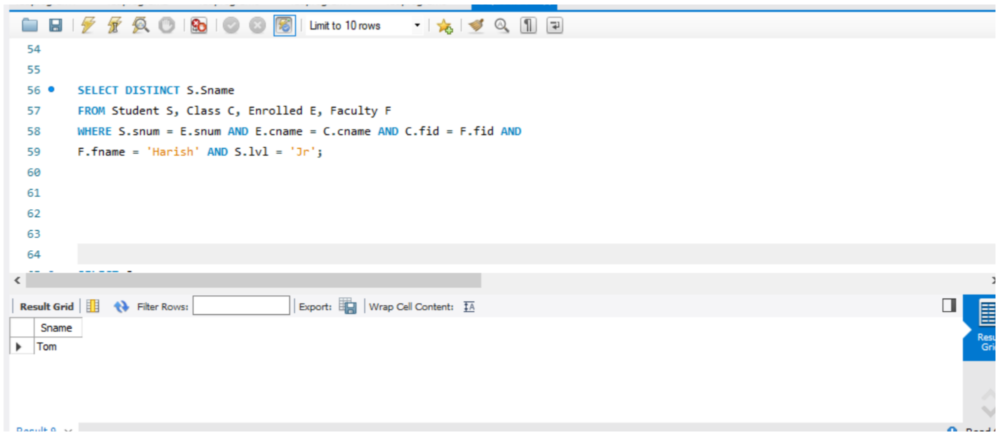
i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by

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 = 'Harish' AND S.lvl = 'Jr';

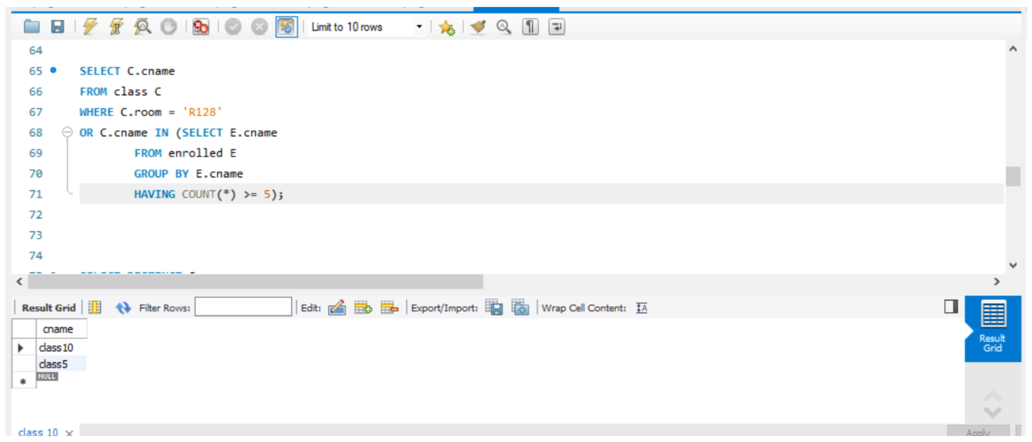


```
SELECT DISTINCT S.Sname
FROM Student S, Class C, Enrolled E, Faculty F
WHERE S.snum = E.snum AND E.cname = C.cname AND C.fid = F.fid AND
F.fname = 'Harish' AND S.lvl = 'Jr';
```

```
Sname
Tom
```

ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.

```
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(*) >= 5);
```



```
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(*) >= 5);
```

```
cname
class10
class5
NULL
```

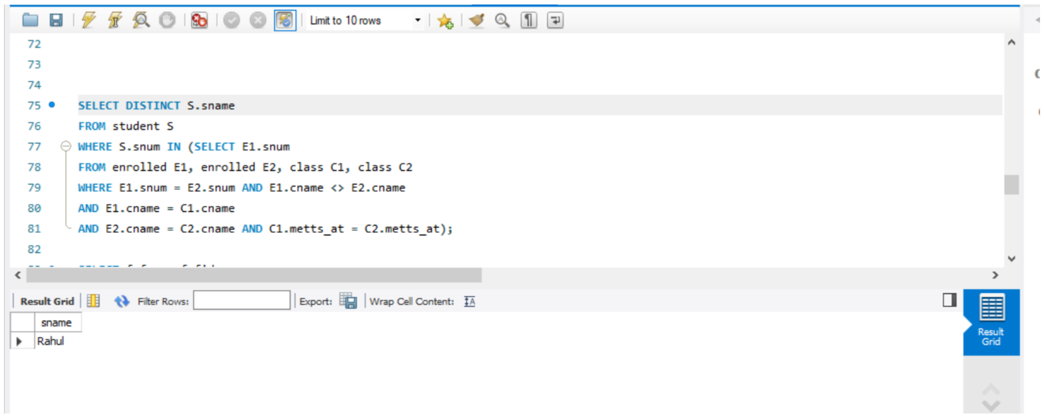
iii. Find the names of all students who are enrolled in two classes that meet at the same time.

```
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.metts_at = C2.metts_at);



```
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.metts_at = C2.metts_at);
```

```
sname
Rahul
```

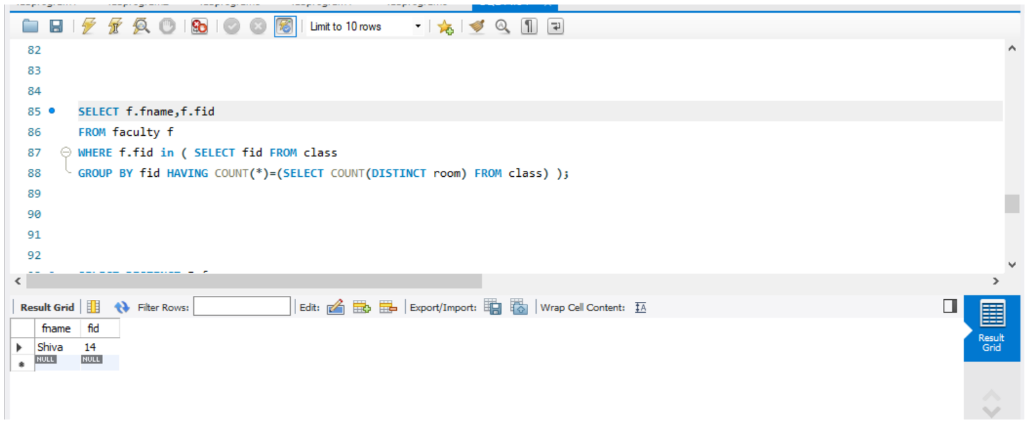
iv. Find the names of faculty members who teach in every room in which some class is taught.

SELECT f.fname,f.fid

FROM faculty f

WHERE f.fid in (SELECT fid FROM class

GROUP BY fid HAVING COUNT(*)=(SELECT COUNT(DISTINCT room) FROM class));



```
SELECT f.fname,f.fid
FROM faculty f
WHERE f.fid in ( SELECT fid FROM class
GROUP BY fid HAVING COUNT(*)=(SELECT COUNT(DISTINCT room) FROM class) );
```

```
fname   fid
Shiva   14
NULL    NULL
```

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

SELECT DISTINCT F.fname

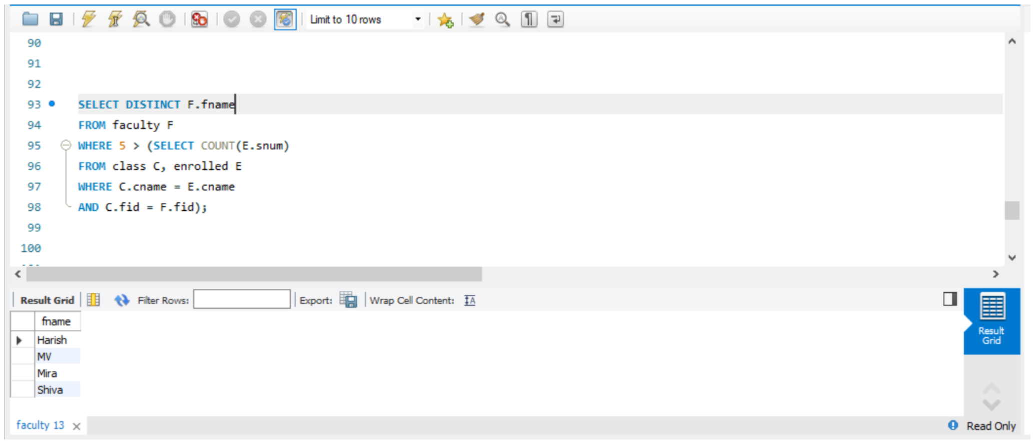
FROM faculty F

WHERE 5 > (SELECT COUNT(E.snum)

FROM class C, enrolled E

WHERE C.cname = E.cname

AND C.fid = F.fid);



```
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
```

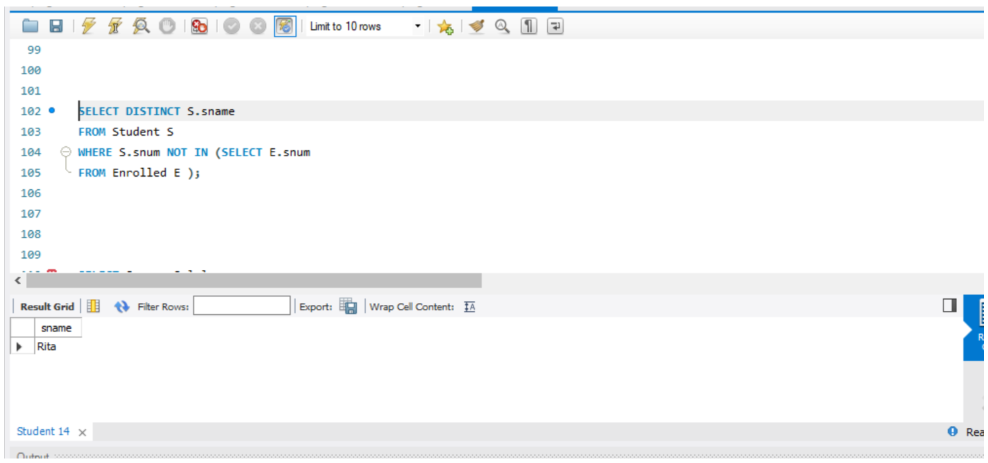
vi. Find the names of students who are not enrolled in any class.

SELECT DISTINCT S.sname

FROM Student S

WHERE S.snum NOT IN (SELECT E.snum

FROM Enrolled E);

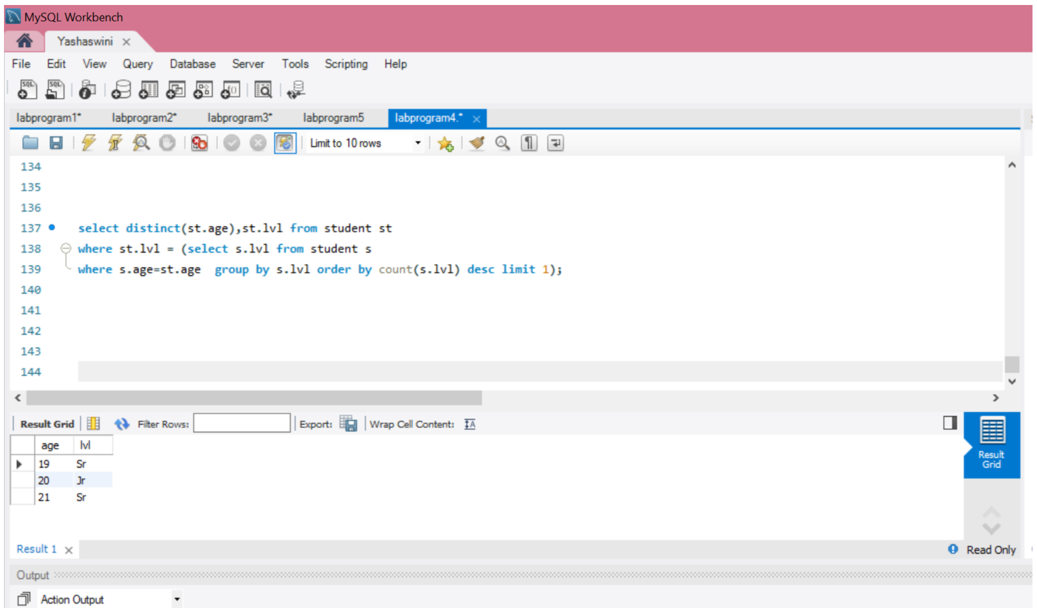


```
SELECT DISTINCT S.sname
FROM Student S
WHERE S.snum NOT IN (SELECT E.snum
FROM Enrolled E );
```

sname
Rita

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 distinct(st.age),st.lvl from student st
where st.lvl = (select s.lvl from student s
where s.age=st.age  group by s.lvl order by count(s.lvl) desc limit 1);
```



```
select distinct(st.age),st.lvl from student st
where st.lvl = (select s.lvl from student s
where s.age=st.age  group by s.lvl order by count(s.lvl) desc limit 1);
```

age	lvl
19	Sr
20	Jr
21	Sr