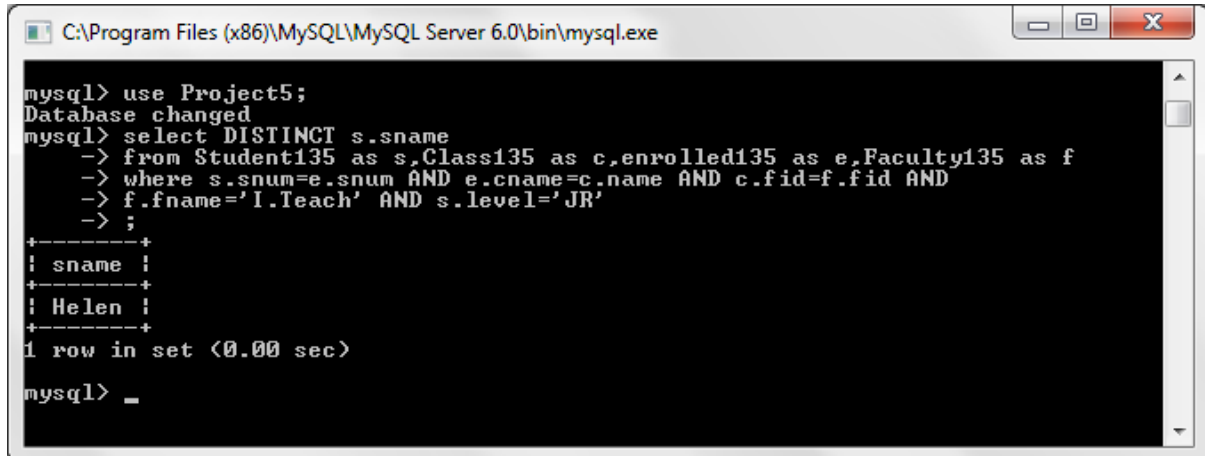


1) Find the names of the juniors (level=JR) who are enrolled in class taught by I.Teach.

```
SELECT DISTINCT S.Sname
FROM Student135 as S, Class135 as C, Enrolled135 as E, Faculty135 as F
WHERE S.snum = E.snum AND E.cname = C.name AND C.fid = F.fid AND
F.fname = 'I.Teach' AND S.level = 'JR';
```

A screenshot of a MySQL command window titled "C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe". The window has a black background with white text. The user has entered the command "use Project5;" and the prompt "Database changed" appears. Then, the user enters a complex SELECT query with multiple table aliases and conditions. The query is displayed in a multi-line format with arrows indicating continuation. The result shows a single row with the name "Helen".

```
mysql> use Project5;
Database changed
mysql> select DISTINCT s.sname
-> from Student135 as s,Class135 as c,enrolled135 as e,Faculty135 as f
-> where s.snum=e.snum AND e.cname=c.name AND c.fid=f.fid AND
-> f.fname='I.Teach' AND s.level='JR'
-> ;
+-----+
| sname |
+-----+
| Helen |
+-----+
1 row in set (0.00 sec)

mysql> _
```

2) Find the age of the oldest student who is either a History major or enrolled in c course taught by I.Teach.

```
SELECT MAX(S.age)
FROM Student135 as S
WHERE (S.major = 'History')
OR S.snum IN (SELECT E.snum
FROM Class135 as C, Enrolled135 as E, Faculty135 as F
WHERE E.cname = C.name AND C.fid = F.fid
AND F.fname = 'I.Teach' )
```

```
C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe
1 row in set (0.00 sec)
mysql> select MAX(s.age)
-> from Student135 as s
-> where (s.major='History')
-> OR s.snum IN (select e.snum from Class135 as c,enrolled135 as e
-> ,Faculty135 as f
-> where e.cname=c.name AND c.fid=f.fid
-> AND f.fname='I.Teach')
-> ;
+-----+
| MAX(s.age) |
+-----+
|          25 |
+-----+
1 row in set (0.00 sec)
mysql>
```

3) Find the names of all classes that either meet in room BA1080 or have two or more students enrolled.

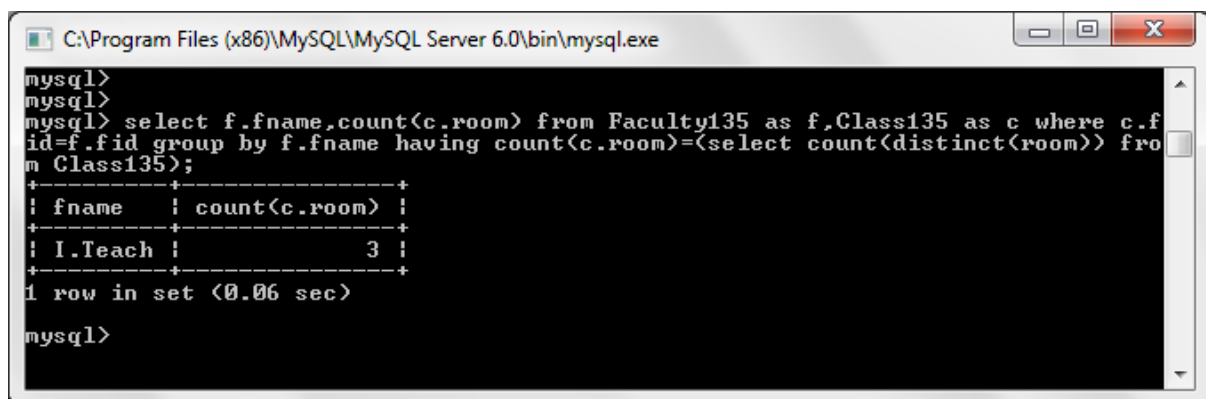
```
SELECT C.name
FROM Class135 as C
WHERE C.room = 'BA1080'
OR C.name IN (SELECT E.cname
FROM Enrolled135 as E
GROUP BY E.cname
HAVING COUNT (*) >= 2)
```

```
C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe
mysql> select c.name from Class135 as c
-> where c.room='BA1080'
-> OR c.name IN (select e.cname
-> from enrolled135 as e
-> Group by e.cname
-> having count(*) >= 2)
-> ;
+-----+
| name |
+-----+
| CSC343 |
| CSC443 |
| ECE201 |
| ECE300 |
+-----+
4 rows in set (0.01 sec)
mysql> _
```

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

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

```
select f.fname,count(c.room)
from Faculty135 as f,Class135 as c
where c.fid=f.fid
group by f.fname
having count(c.room)=(select count(distinct(room)) from Class135);
```



The screenshot shows a MySQL command window titled "C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe". The command prompt shows the following SQL query being executed:

```
mysql>
mysql>
mysql> select f.fname,count(c.room) from Faculty135 as f,Class135 as c where c.f
id=f.fid group by f.fname having count(c.room)=(select count(distinct(room)) fro
m Class135);
```

The output of the query is displayed as a table:

fname	count(c.room)
I.Teach	3

Below the table, the output indicates "1 row in set (0.06 sec)". The command prompt then shows "mysql>" again.

6) 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 Faculty135 as F
WHERE 5 > (SELECT COUNT (E.snum)
FROM Class135 as C, Enrolled135 as E
WHERE C.name = E.cname
AND C.fid = F.fid)
```

```
C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe

line 6
mysql> select DISTINCT f.fname
-> from Faculty135 as f
-> where 5> (select count(e.snum)
-> from Class135 as c,enrolled135 as e
-> where c.name=e.cname
-> AND c.fid=f.fid)
-> ;

+-----+
| fname |
+-----+
| S.Jackson |
| M.Shanks |
| I.Teach |
+-----+
3 rows in set (0.00 sec)
```

7) For each level, print the level and the average age of students for that level.

```
SELECT S.level, AVG(S.age)
FROM Student135 as S
GROUP BY S.level;
```

```
C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe

mysql>
mysql> select s.level,AVG(s.age)
-> from Student135 as s
-> group by s.level
-> ;

+-----+-----+
| level | AVG(s.age) |
+-----+-----+
| GR    | 28.0000    |
| JR    | 19.0000    |
| SR    | 22.0000    |
+-----+-----+
3 rows in set (0.02 sec)

mysql> _
```

8) For all levels except JR, print the level and the average age of students for that level.

```
. SELECT S.level, AVG(S.age)
FROM Student135 as S
WHERE S.level <> 'JR'
GROUP BY S.level
```

```
C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe
3 rows in set (0.02 sec)
mysql> select s.level,AVG(s.age)
-> from Student135 as s
-> where s.level <> 'JR'
-> group by s.level
-> ;
+-----+-----+
| level | AVG(s.age) |
+-----+-----+
| GR    | 28.0000    |
| SR    | 22.0000    |
+-----+-----+
2 rows in set (0.00 sec)
mysql>
```

9) For each faculty member that has taught classes only in room BA1080, print the faculty members name and the total number of classes she or he has taught.

Select f.fname,count(room)

From Faculty135 as f,Class135 as c

Where f.fid=c.fid AND c.room='BA1080'

Group by f.fname;

```
C:\Program Files (x86)\MySQL\MySQL Server 6.0\bin\mysql.exe
mysql> select f.fname,count(room) from Faculty135 as f,Class135 as c where f.fid
=c.fid AND c.room='BA1080' group by f.fname;
+-----+-----+
| fname      | count(room) |
+-----+-----+
| I.Teach    | 1           |
| S.Jackson  | 1           |
+-----+-----+
2 rows in set (0.00 sec)
mysql>
```

10) Find the names of students enrolled in the maximum number of classes.