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.

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

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 enrolment 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).

```
CREATE TABLE `class` (
  `name` varchar(30) NOT NULL,
  `meets_at` time NOT NULL,
  `room` varchar(30) NOT NULL,
```

```
  `fid` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- --------------------------------------------------------

--
-- Table structure for table `enrolled`
--

CREATE TABLE `enrolled` (
  `snum` int(11) NOT NULL,
  `cname` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- --------------------------------------------------------

--
-- Table structure for table `faculty`
--

CREATE TABLE `faculty` (
  `fid` int(11) NOT NULL,
  `fname` varchar(30) NOT NULL,
  `deptid` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- --------------------------------------------------------

--
-- Table structure for table `student`
```

```
--

CREATE TABLE `student` (
  `snum` int(11) NOT NULL,
  `sname` varchar(30) NOT NULL,
  `major` varchar(30) NOT NULL,
  `level` varchar(2) NOT NULL,
  `age` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Indexes for dumped tables
--

--
-- Indexes for table `class`
--
ALTER TABLE `class`
  ADD PRIMARY KEY (`name`),
  ADD KEY `fid` (`fid`);

--
-- Indexes for table `enrolled`
--
ALTER TABLE `enrolled`
  ADD KEY `snum` (`snum`),
  ADD KEY `cname` (`cname`);

--
-- Indexes for table `faculty`
```

```
--
ALTER TABLE `faculty`
  ADD PRIMARY KEY (`fid`);

--
-- Indexes for table `student`
--
ALTER TABLE `student`
  ADD PRIMARY KEY (`snum`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `faculty`
--
ALTER TABLE `faculty`
  MODIFY `fid` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `student`
--
ALTER TABLE `student`
  MODIFY `snum` int(11) NOT NULL AUTO_INCREMENT;

--
-- Constraints for dumped tables
--
```

```
--
-- Constraints for table `class`
--
ALTER TABLE `class`
  ADD CONSTRAINT `class_ibfk_1` FOREIGN KEY (`fid`) REFERENCES
`faculty` (`fid`);


--
-- Constraints for table `enrolled`
--
ALTER TABLE `enrolled`
  ADD CONSTRAINT `enrolled_ibfk_1` FOREIGN KEY (`snum`) REFERENCES
`student` (`snum`),
  ADD CONSTRAINT `enrolled_ibfk_2` FOREIGN KEY (`cname`) REFERENCES
`class` (`name`);


INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("janish","cs","sr","19");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("kin","cv","so","18");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("vedika","is","jr","19");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("tarun","cs","sr","20");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("ram","me","fr","18");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("john","ec","sr","19");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("ahmed","tc","jr","20");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("raveena","ec","so","18");
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("kripal","is","jr","18");
```

```
INSERT INTO `student`(`sname`, `major`, `level`, `age`) VALUES
("ananya","cs","sr","20");


INSERT INTO `faculty`( `fname`, `deptid`) VALUES ("ravi","1");

INSERT INTO `faculty`( `fname`, `deptid`) VALUES ("priyam","1");

INSERT INTO `faculty`( `fname`, `deptid`) VALUES ("chander","2");

INSERT INTO `faculty`( `fname`, `deptid`) VALUES ("satish","5");

INSERT INTO `faculty`( `fname`, `deptid`) VALUES ("guru","4");

INSERT INTO `faculty`( `fname`, `deptid`) VALUES ("rahul","3");

INSERT INTO `faculty`( `fname`, `deptid`) VALUES ("abdul","6");


INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("ccp","09:00:00","r125","1");

INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("ece","09:15:00","5","3");

INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("som","09:50:00","4","6");

INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("nwt","09:00:00","2","4");

INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("dec","10:00:00","3","5");

INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("dst","11:00:00","r125","2");

INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("eed","10:30:00","2","7");

INSERT INTO `class`(`name`, `meets_at`, `room`, `fid`) VALUES
("coa","09:00:00","3","2");


INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("6","ccp");

INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("13","ccp");

INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("7","dst");

INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("4","coa");

INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("8","eed");
```

```
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("5","som");
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("9","ece");
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("12","ccp");
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("9","nwt");
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("13","dst");
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("11","ece");
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("8","som");
INSERT INTO `enrolled`(`snum1`, `cname1`) VALUES ("4","nwt");
```

TABLES

```
SELECT * FROM `class`
```

Show all | Number of rows: 25 Filter rows: Sear

+ Options

	name	meets_at	room	fid
Edit Copy Delete	ccp	09:00:00	r125	1
Edit Copy Delete	ccph	12:00:00	r127	1
Edit Copy Delete	coa	09:00:00	r127	2
Edit Copy Delete	cp	10:00:00	r126	1
Edit Copy Delete	cpi	04:00:00	r129	1
Edit Copy Delete	cpr	03:00:00	r128	1
Edit Copy Delete	dec	10:00:00	r127	5
Edit Copy Delete	dst	11:00:00	r125	2
Edit Copy Delete	ece	09:15:00	r129	3
Edit Copy Delete	eed	10:30:00	r126	7
Edit Copy Delete	nwt	09:00:00	r126	4
Edit Copy Delete	som	09:50:00	r128	6

Showing rows 0 - 12 (13 to

```
SELECT * FROM `enrolled`
```

Show all | Number of

+ Options

snum1	cname1
6	ccp
13	ccp
7	dst
4	coa
8	eed
5	som
9	ece
12	ccp
9	nwt
13	dst
11	ece
8	som
4	nwt

Showing rows 0 - 6 (7 total, Query took 0.0015 seconds.)

```
SELECT * FROM `faculty`
```

Show all | Number of rows: 25 Filter rov

+ Options

	fid	fname	deptid
Edit Copy Delete	1	ravi	1
Edit Copy Delete	2	priyam	1
Edit Copy Delete	3	chander	2
Edit Copy Delete	4	satish	5
Edit Copy Delete	5	guru	4
Edit Copy Delete	6	rahul	3
Edit Copy Delete	7	abdul	6

```
SELECT * FROM `student`
```

Show all | Number of rows: 25 Filter rows: Search this t

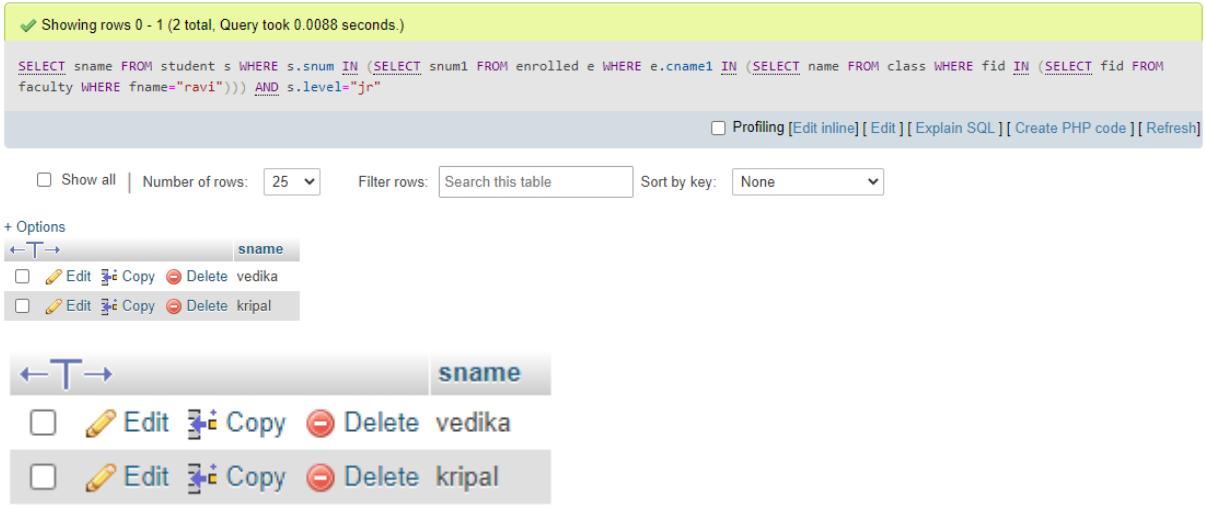
+ Options

	snum	sname	major	level	age
Edit Copy Delete	4	janish	cs	sr	19
Edit Copy Delete	5	kin	cv	so	18
Edit Copy Delete	6	vedika	is	jr	19
Edit Copy Delete	7	tarun	cs	sr	20
Edit Copy Delete	8	ram	me	fr	18
Edit Copy Delete	9	john	ec	sr	19
Edit Copy Delete	10	ahmed	tc	jr	20
Edit Copy Delete	11	raveena	ec	so	18
Edit Copy Delete	12	kripal	is	jr	18
Edit Copy Delete	13	ananya	cs	sr	20

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

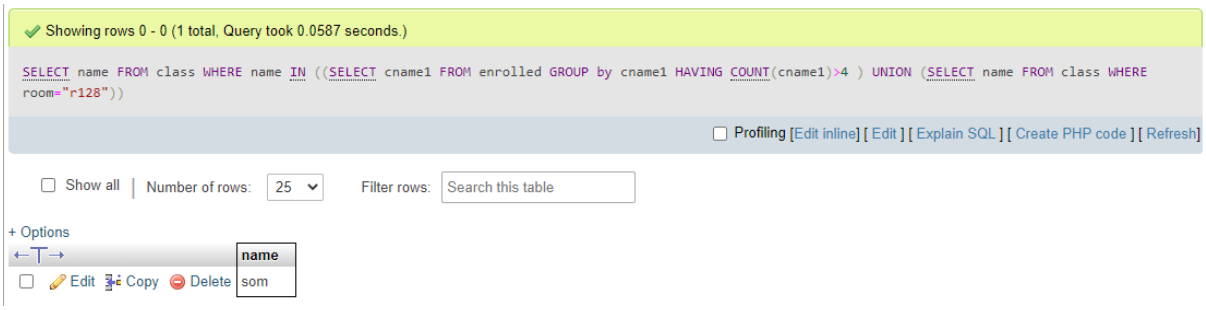
i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by

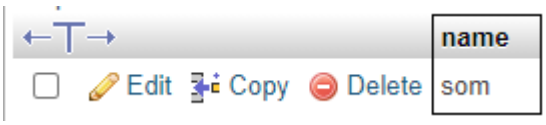
```
SELECT sname

FROM student s

WHERE s.snum IN

(SELECT snum1 FROM enrolled e WHERE e.cname1 IN

 (SELECT name FROM class WHERE fid IN

  (SELECT fid FROM faculty WHERE fname="ravi")))

AND s.level="jr";
```



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

```
SELECT name FROM class WHERE name IN ((SELECT cname1 FROM enrolled GROUP by cname1 HAVING COUNT(cname1)>4 ) UNION (SELECT name FROM class WHERE room="r128"))
```



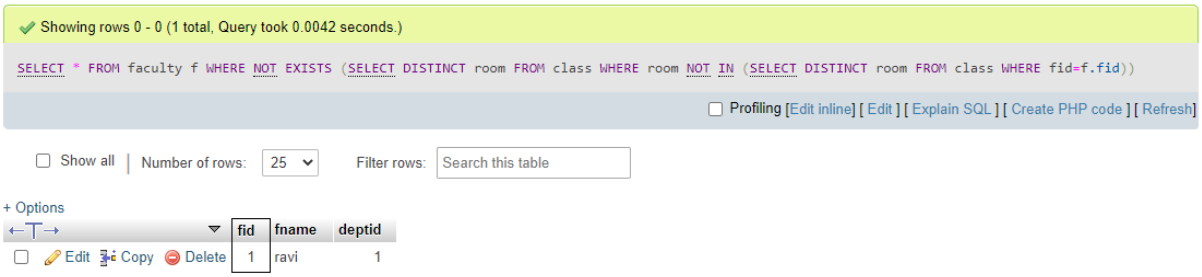


iii. Find the names of all students who are enrolled in two classes that meet at the same time.

```
SELECT sname FROM student WHERE NOT EXISTS(

(SELECT e1.cname1,e2.cname1 FROM enrolled e1, enrolled e2 WHERE
e1.cname1<>e2.cname1 AND e1.snum1=e2.snum1) NOT IN

(SELECT c1.name,c2.name FROM class c1, class c2 WHERE
c1.name!=c2.name AND c1.meets_at=c2.meets_at)

);
```

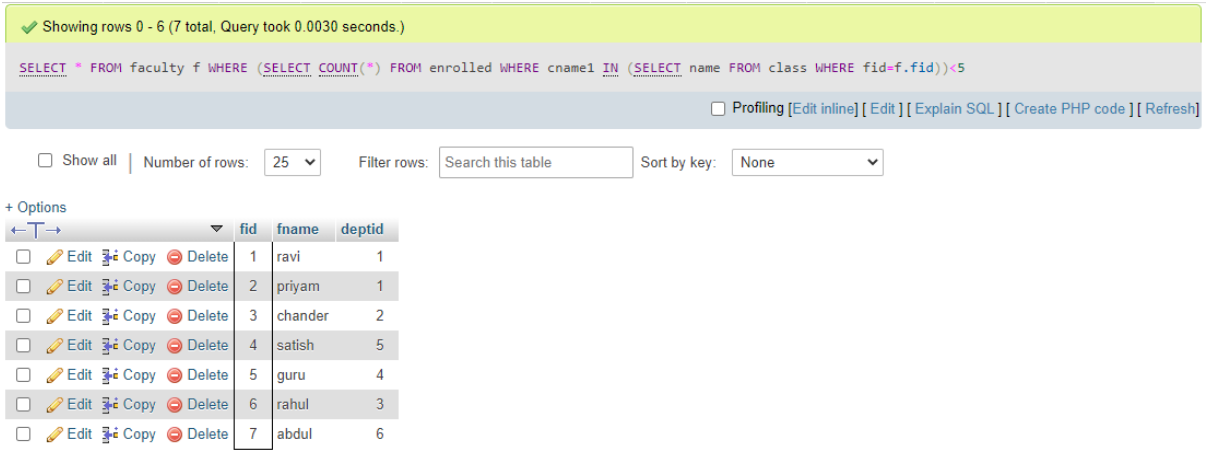
iv. Find the names of faculty members who teach in every room in which some class is taught.

```
SELECT * FROM faculty f WHERE NOT EXISTS (SELECT DISTINCT room FROM
class WHERE room NOT IN (SELECT DISTINCT room FROM class WHERE
fid=f.fid));
```



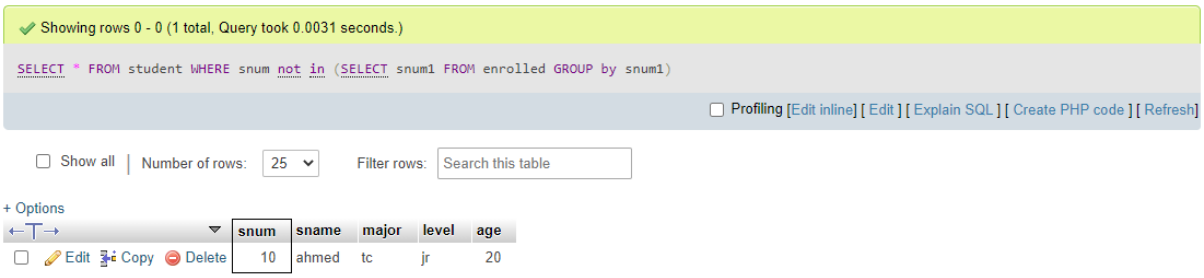
v. Find the names of faculty members for whom the combined enrolment of the courses that they teach is less than five.

```
SELECT * FROM faculty f WHERE (SELECT COUNT(*) FROM enrolled
WHERE cname1 IN (SELECT name FROM class WHERE fid=f.fid))<5;
```



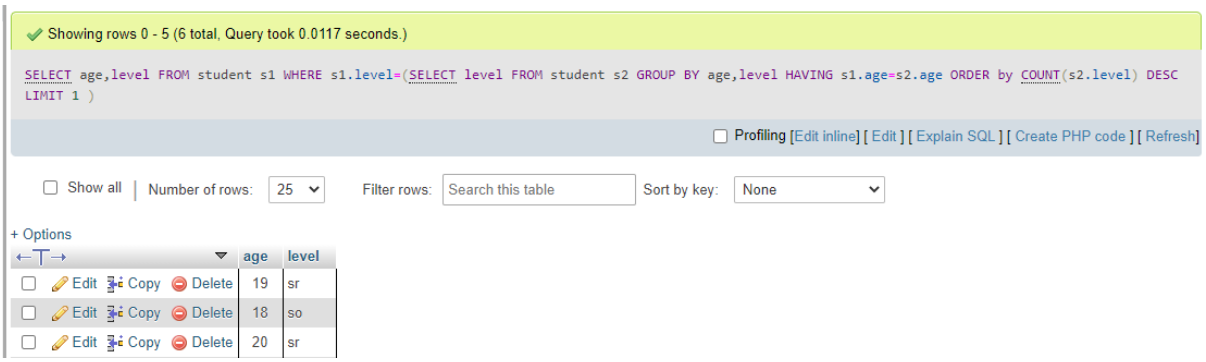
vi. Find the names of students who are not enrolled in any class.

SELECT * FROM student WHERE snum not in (SELECT snum1 FROM enrolled GROUP by snum1);



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

SELECT age,level FROM student s1 WHERE s1.level=(SELECT level FROM student s2 GROUP BY age,level HAVING s1.age=s2.age ORDER by COUNT(s2.level) DESC LIMIT 1)



****