Name: Nishant Kumar Giri
Roll No: AC-1254
Subject: DBMS
Course: BSc (Hons) Computer Science
Semester: IV

Problem Statement

College is maintaining data about students registered in different societies with the objective of availability of information as per requirement. **E.g.**
Total students registered in all societies. Total students in each society? Popular society? Least popular society? Popularity each year?? list of students registered in a society?? which course students are opting 'society A' in majority??

Tables to be Created

STUDENT

`RollNo char(7), SName varchar(20), Course varchar(10), DOB date` **Primary Key:** RollNo

SOCIETY

`SID char(6), SocietyName varchar(20), Mentor varchar(10), TotalSeats int unsigned`

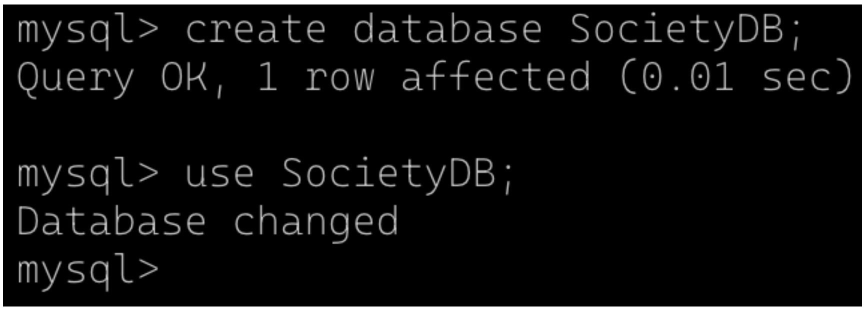
Primary Key: SID

ENROLL

`RollNo, SID, DOE`
Primary Key: RollNo & SID, Foreign Key: RollNo & SID

Other constraints: Name of society and student cannot be NULL. By default total seats in each society is 10

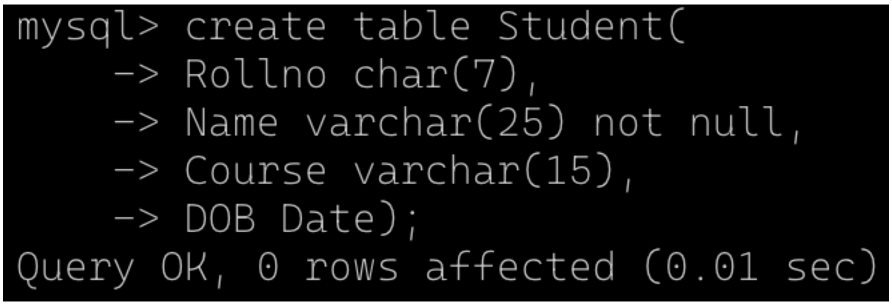
Create Database



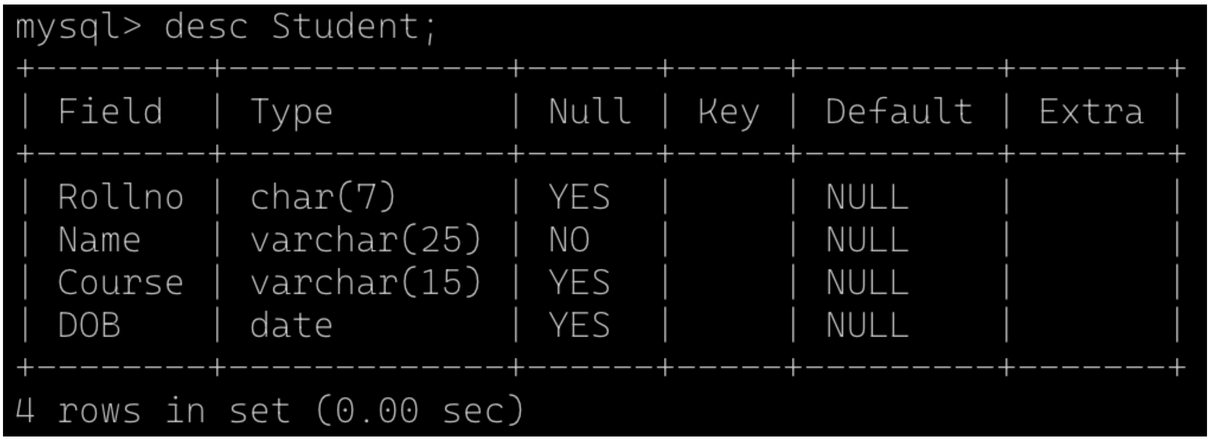
```
mysql> create database SocietyDB;
Query OK, 1 row affected (0.01 sec)

mysql> use SocietyDB;
Database changed
mysql>
```

Create Tables

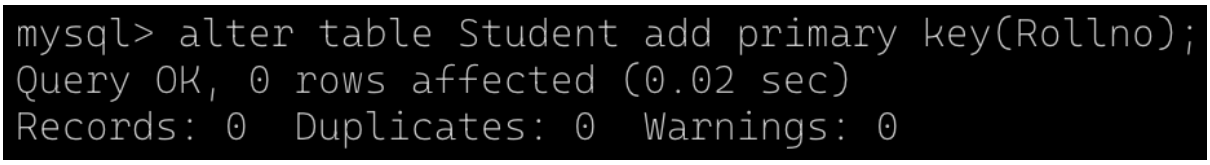
Student Table



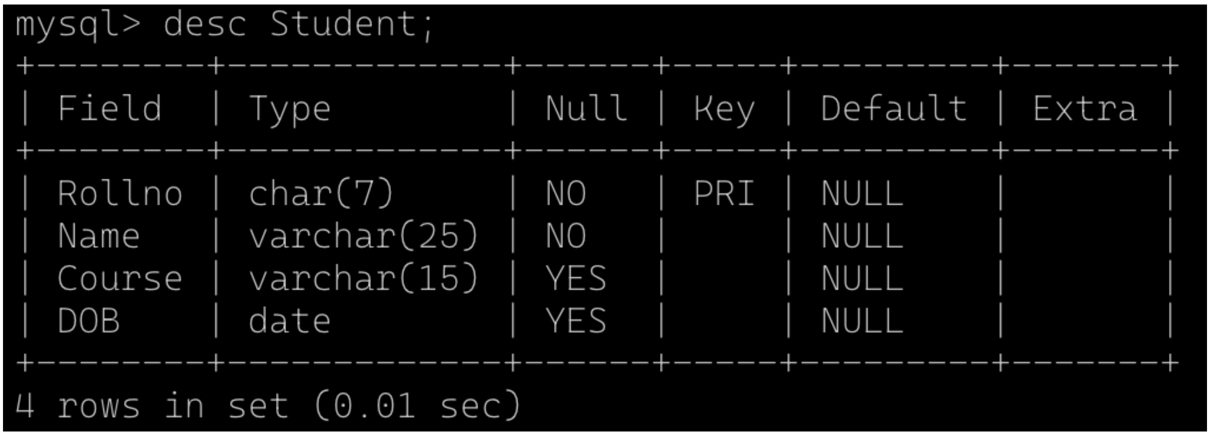
```
mysql> create table Student(
    -> Rollno char(7),
    -> Name varchar(25) not null,
    -> Course varchar(15),
    -> DOB Date);
Query OK, 0 rows affected (0.01 sec)
```



```
mysql> desc Student;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| Rollno | char(7)     | YES  |     | NULL    |       |
| Name   | varchar(25) | NO   |     | NULL    |       |
| Course | varchar(15) | YES  |     | NULL    |       |
| DOB    | date        | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

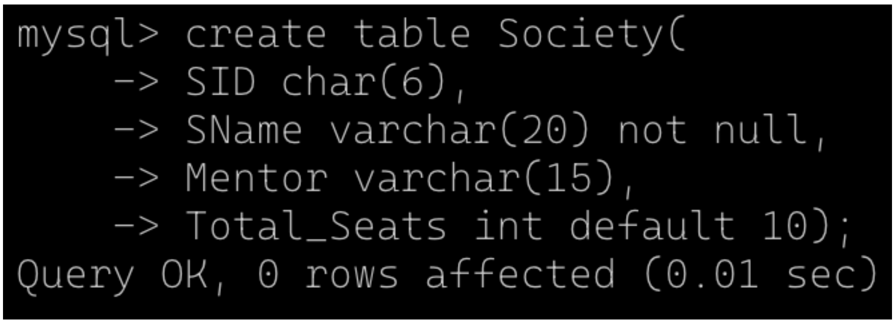


```
mysql> alter table Student add primary key(Rollno);
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

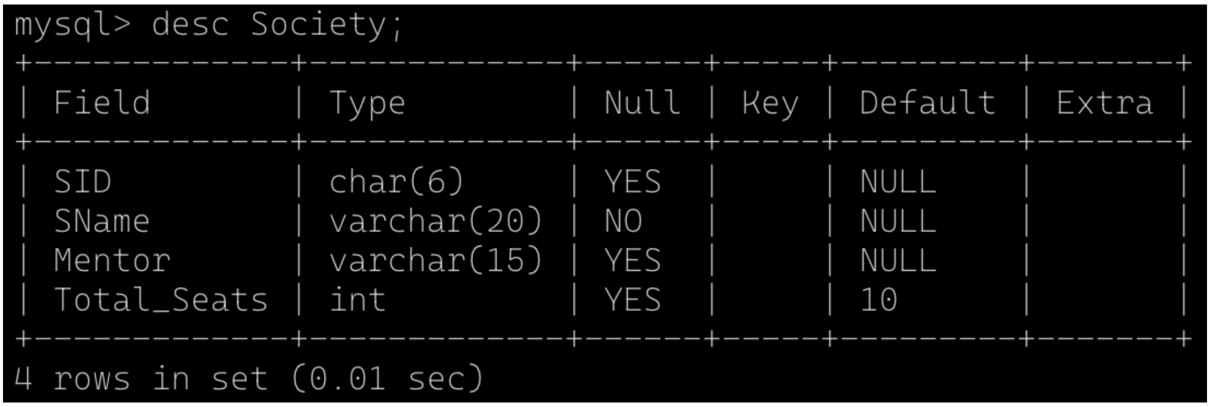


```
mysql> desc Student;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| Rollno | char(7)     | NO   | PRI | NULL    |       |
| Name   | varchar(25) | NO   |     | NULL    |       |
| Course | varchar(15) | YES  |     | NULL    |       |
| DOB    | date        | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```

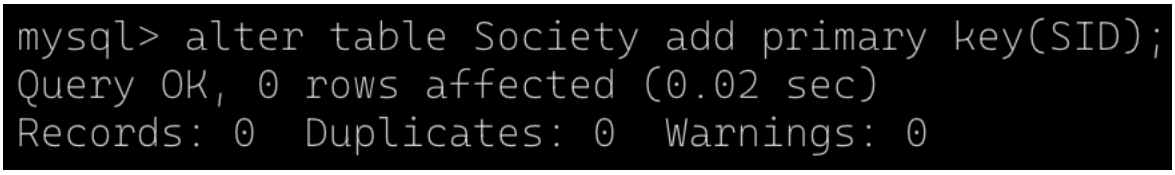
Society Table



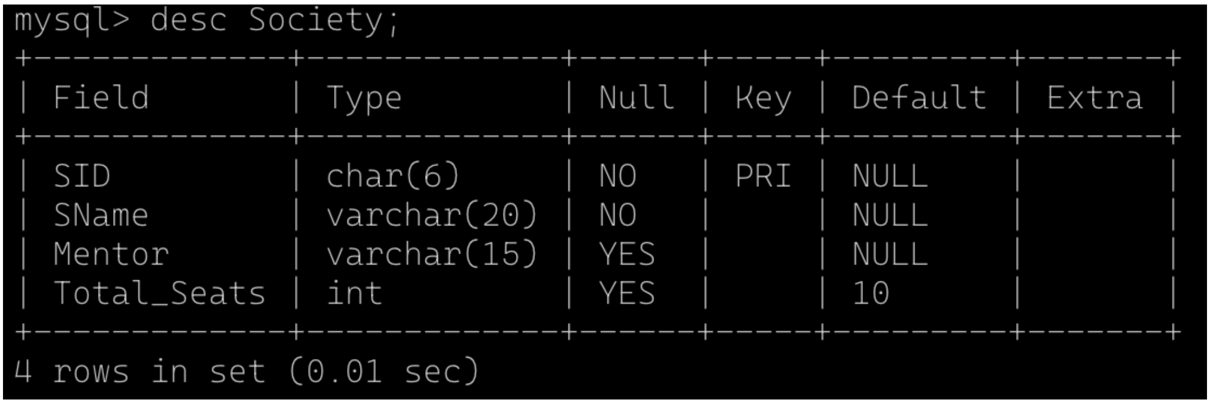
```
mysql> create table Society(
    -> SID char(6),
    -> SName varchar(20) not null,
    -> Mentor varchar(15),
    -> Total_Seats int default 10);
Query OK, 0 rows affected (0.01 sec)
```



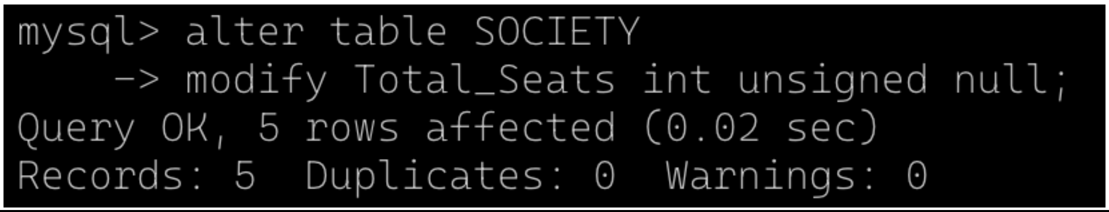
```
mysql> desc Society;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| SID         | char(6)     | YES  |     | NULL    |       |
| SName       | varchar(20) | NO   |     | NULL    |       |
| Mentor      | varchar(15) | YES  |     | NULL    |       |
| Total_Seats | int         | YES  |     | 10      |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```



```
mysql> alter table Society add primary key(SID);
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

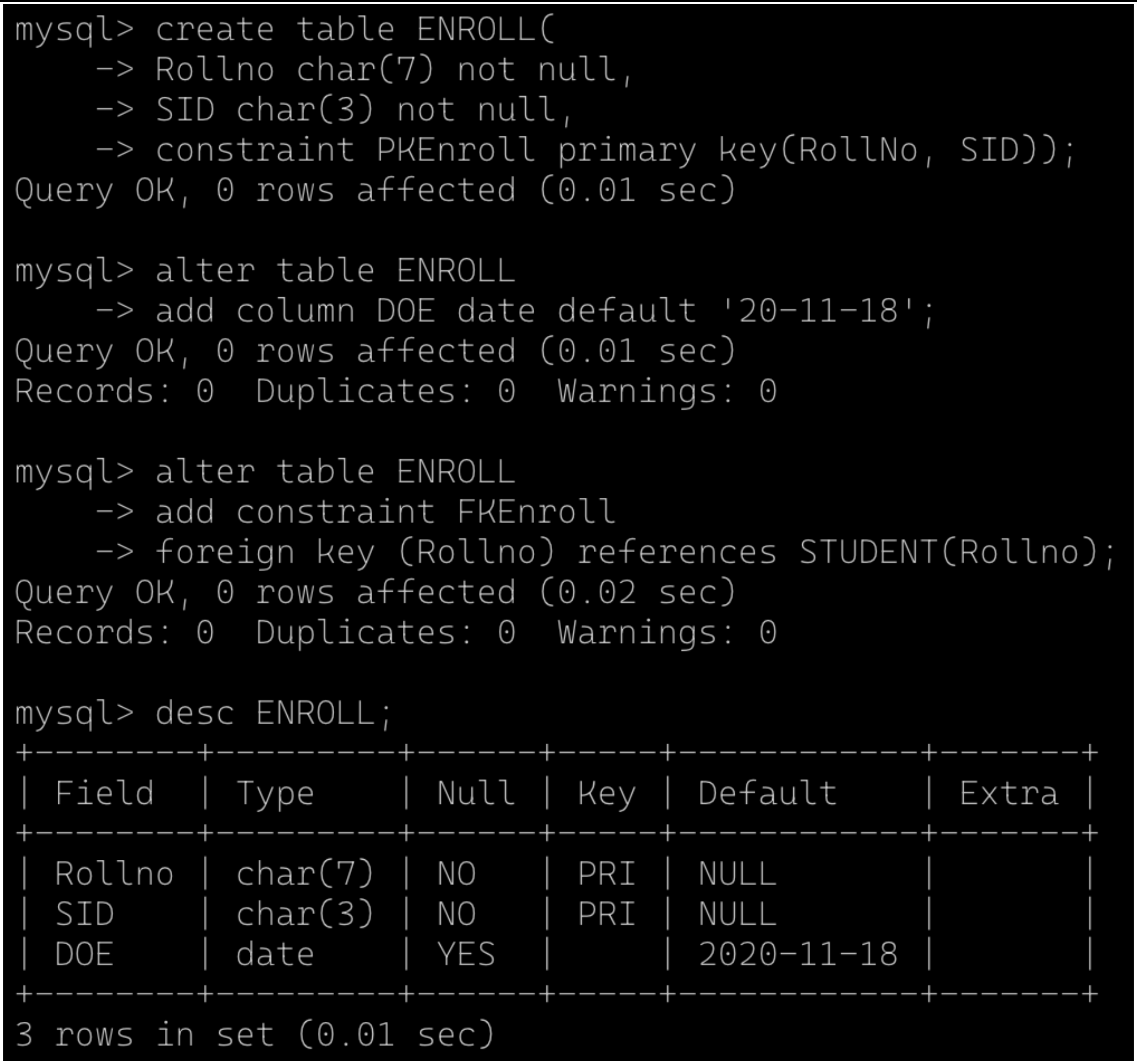


```
mysql> desc Society;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| SID         | char(6)     | NO   | PRI | NULL    |       |
| SName       | varchar(20) | NO   |     | NULL    |       |
| Mentor      | varchar(15) | YES  |     | NULL    |       |
| Total_Seats | int         | YES  |     | 10      |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```



```
mysql> alter table SOCIETY
    -> modify Total_Seats int unsigned null;
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

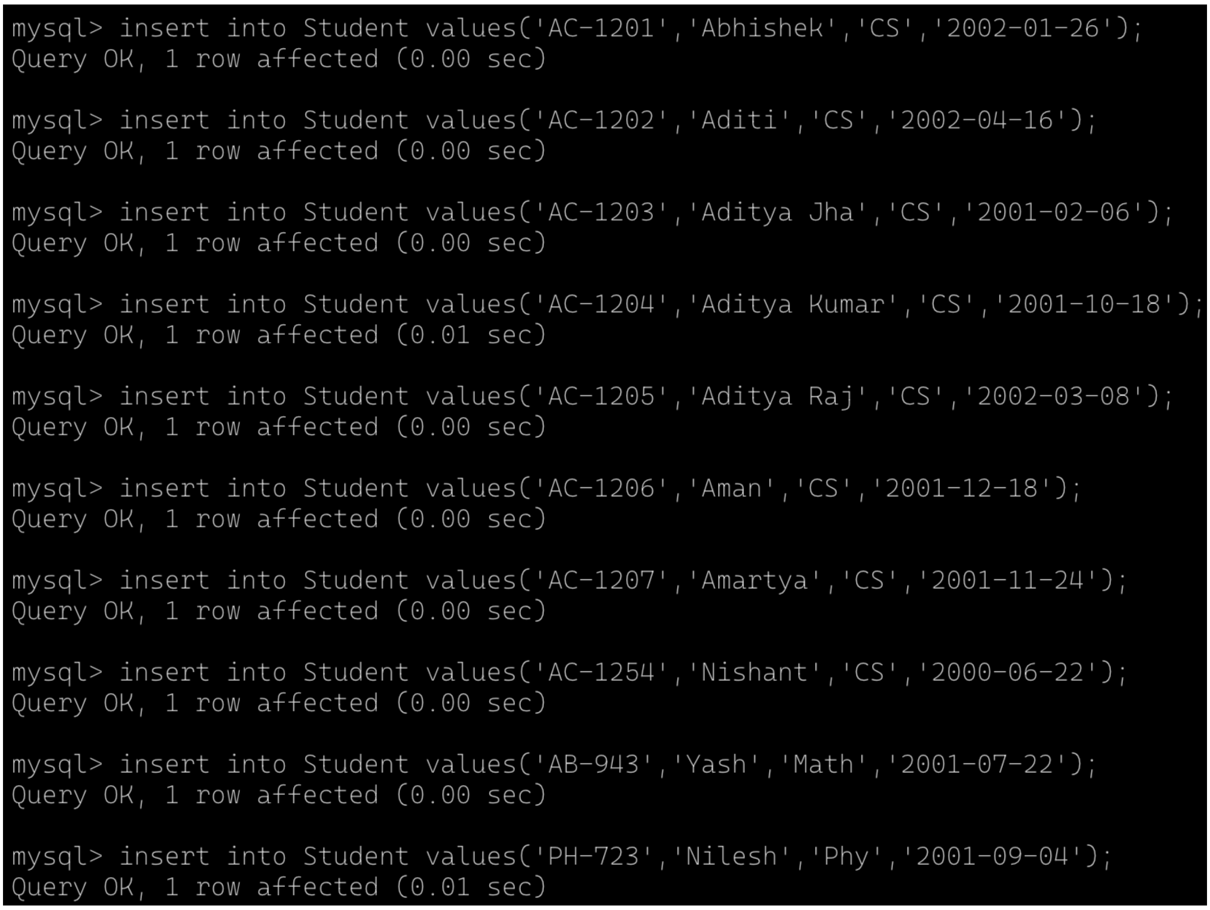
Enrollment Table



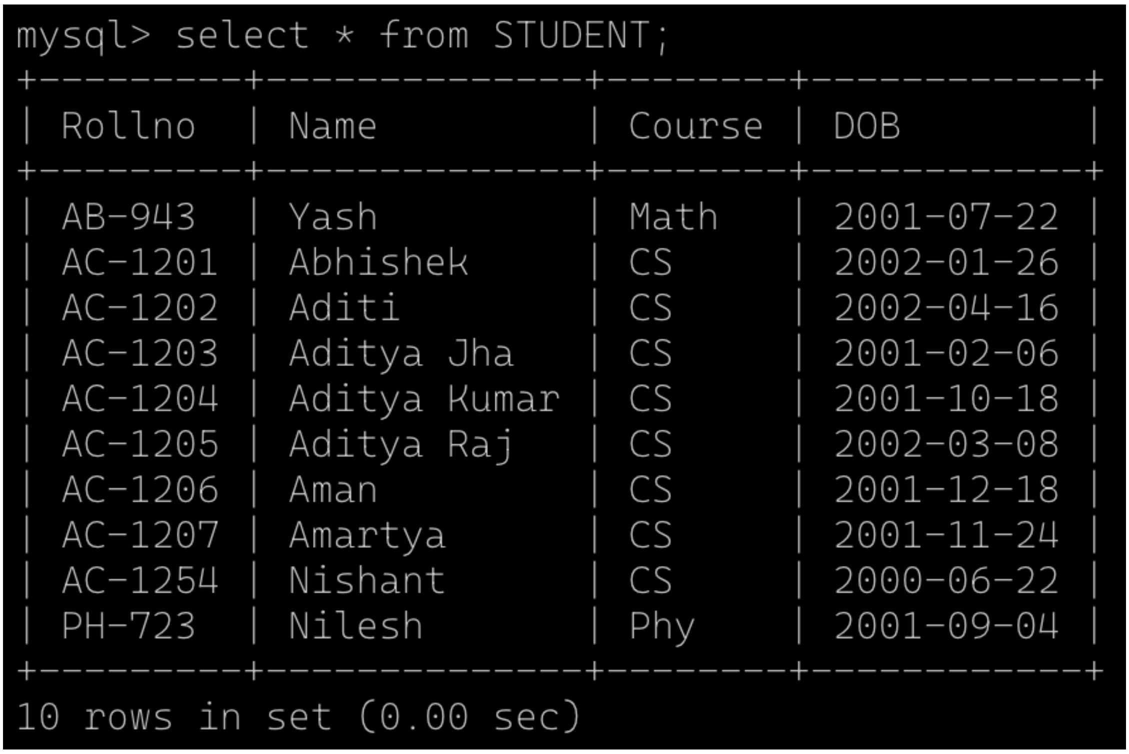
```
mysql> create table ENROLL(
    -> Rollno char(7) not null,
    -> SID char(3) not null,
    -> constraint PKEnroll primary key(RollNo, SID));
Query OK, 0 rows affected (0.01 sec)

mysql> alter table ENROLL
    -> add column DOE date default '20-11-18';
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> alter table ENROLL
    -> add constraint FKEnroll
    -> foreign key (Rollno) references STUDENT(Rollno);
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc ENROLL;
+--------+---------+------+-----+------------+-------+
| Field  | Type    | Null | Key | Default    | Extra |
+--------+---------+------+-----+------------+-------+
| Rollno | char(7) | NO   | PRI | NULL       |       |
| SID    | char(3) | NO   | PRI | NULL       |       |
| DOE    | date    | YES  |     | 2020-11-18 |       |
+--------+---------+------+-----+------------+-------+
3 rows in set (0.01 sec)
```

Populate Data

Inserting values in STUDENT table:

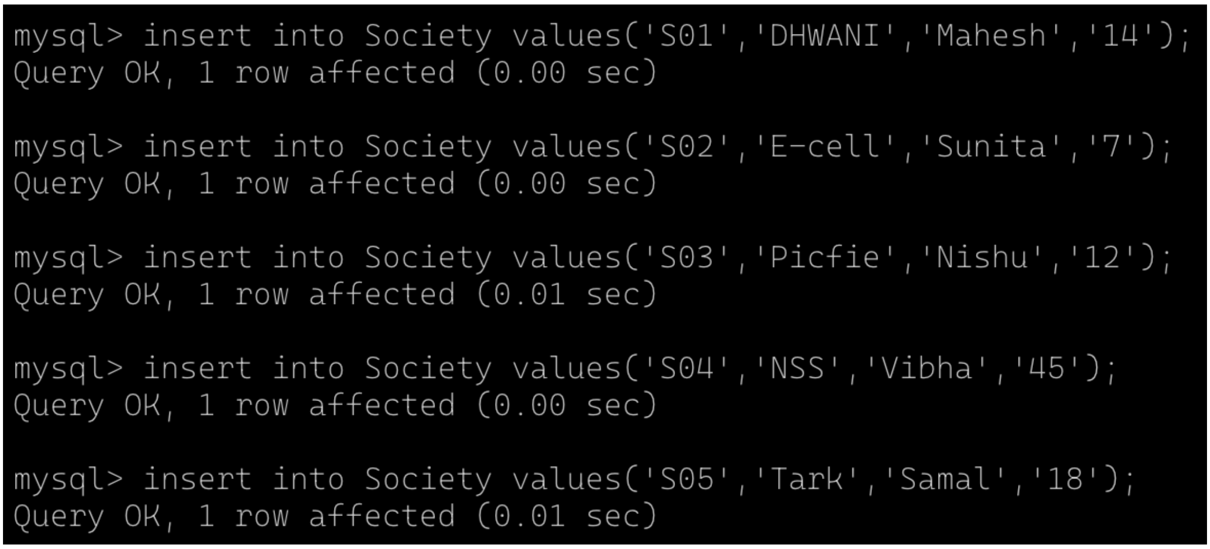


```
mysql> insert into Student values('AC-1201','Abhishek','CS','2002-01-26');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('AC-1202','Aditi','CS','2002-04-16');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('AC-1203','Aditya Jha','CS','2001-02-06');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('AC-1204','Aditya Kumar','CS','2001-10-18');
Query OK, 1 row affected (0.01 sec)

mysql> insert into Student values('AC-1205','Aditya Raj','CS','2002-03-08');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('AC-1206','Aman','CS','2001-12-18');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('AC-1207','Amartya','CS','2001-11-24');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('AC-1254','Nishant','CS','2000-06-22');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('AB-943','Yash','Math','2001-07-22');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Student values('PH-723','Nilesh','Phy','2001-09-04');
Query OK, 1 row affected (0.01 sec)
```

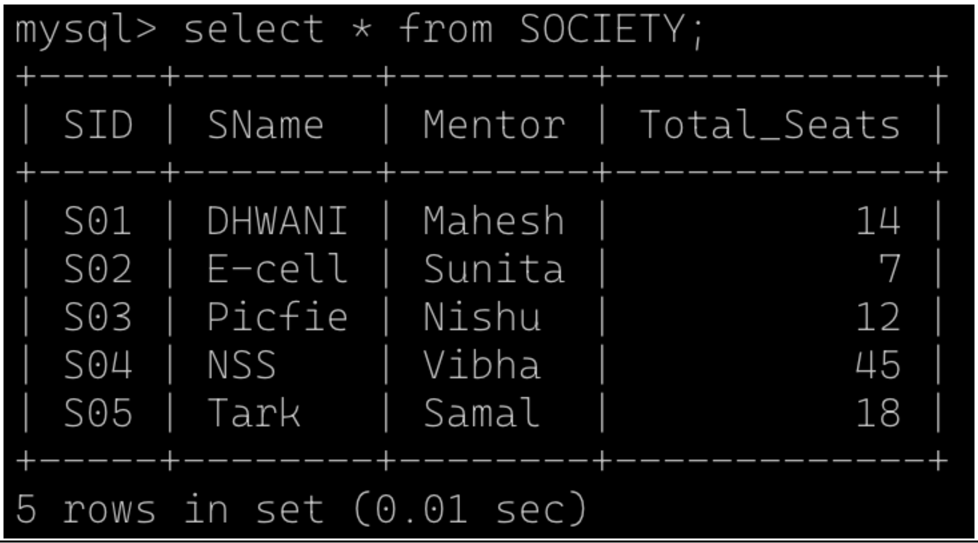


```
mysql> select * from STUDENT;
+---------+--------------+--------+------------+
| Rollno  | Name         | Course | DOB        |
+---------+--------------+--------+------------+
| AB-943  | Yash         | Math   | 2001-07-22 |
| AC-1201 | Abhishek     | CS     | 2002-01-26 |
| AC-1202 | Aditi        | CS     | 2002-04-16 |
| AC-1203 | Aditya Jha   | CS     | 2001-02-06 |
| AC-1204 | Aditya Kumar | CS     | 2001-10-18 |
| AC-1205 | Aditya Raj   | CS     | 2002-03-08 |
| AC-1206 | Aman         | CS     | 2001-12-18 |
| AC-1207 | Amartya      | CS     | 2001-11-24 |
| AC-1254 | Nishant      | CS     | 2000-06-22 |
| PH-723  | Nilesh       | Phy    | 2001-09-04 |
+---------+--------------+--------+------------+
10 rows in set (0.00 sec)
```

Inserting values in SOCIETY table:

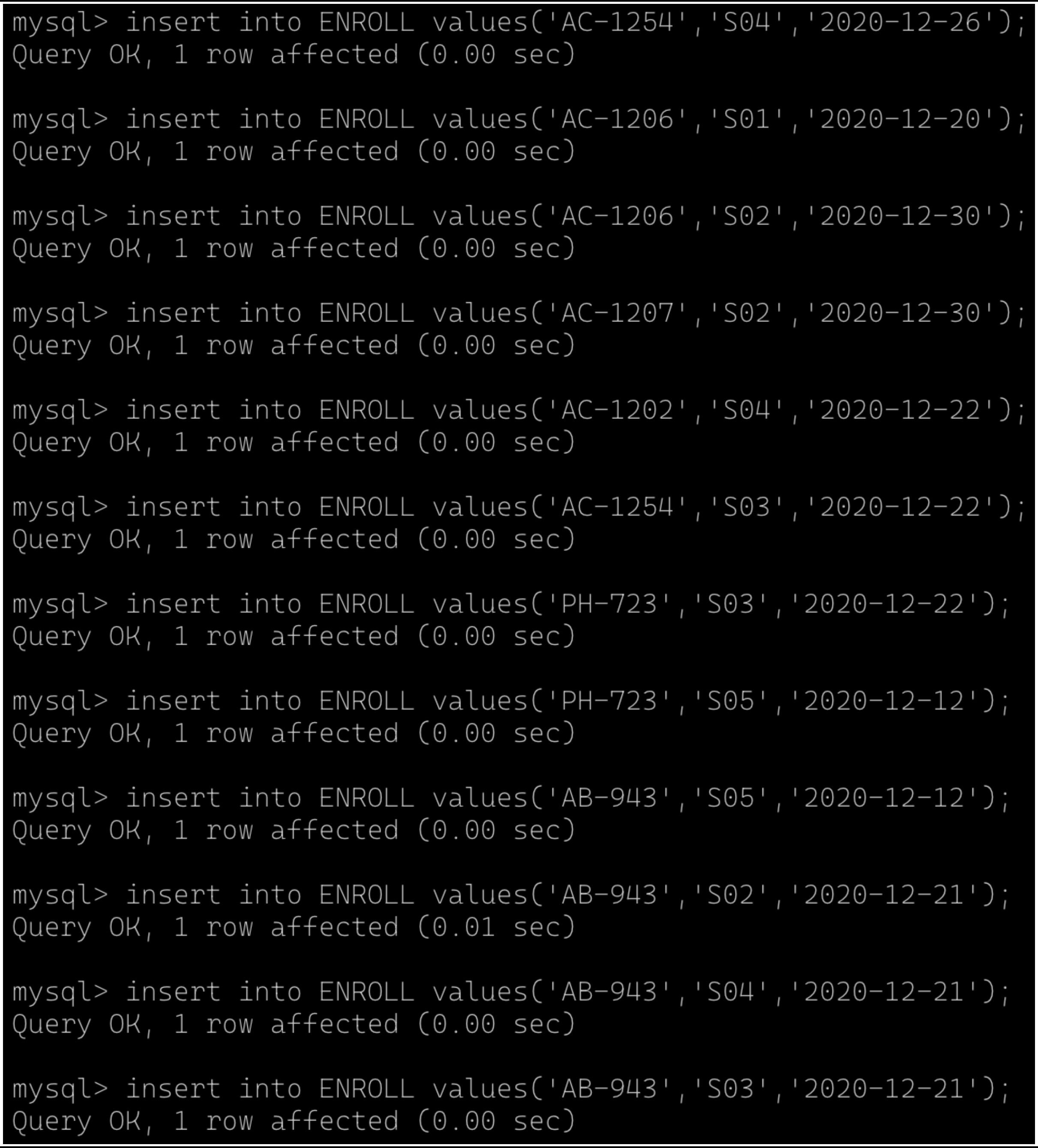


```
mysql> insert into Society values('S01','DHWANI','Mahesh','14');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Society values('S02','E-cell','Sunita','7');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Society values('S03','Picfie','Nishu','12');
Query OK, 1 row affected (0.01 sec)

mysql> insert into Society values('S04','NSS','Vibha','45');
Query OK, 1 row affected (0.00 sec)

mysql> insert into Society values('S05','Tark','Samal','18');
Query OK, 1 row affected (0.01 sec)
```



```
mysql> select * from SOCIETY;
+-----+--------+--------+-------------+
| SID | SName  | Mentor | Total_Seats |
+-----+--------+--------+-------------+
| S01 | DHWANI | Mahesh |          14 |
| S02 | E-cell | Sunita |           7 |
| S03 | Picfie | Nishu  |          12 |
| S04 | NSS    | Vibha  |          45 |
| S05 | Tark   | Samal  |          18 |
+-----+--------+--------+-------------+
5 rows in set (0.01 sec)
```

Inserting values in ENROLL table:

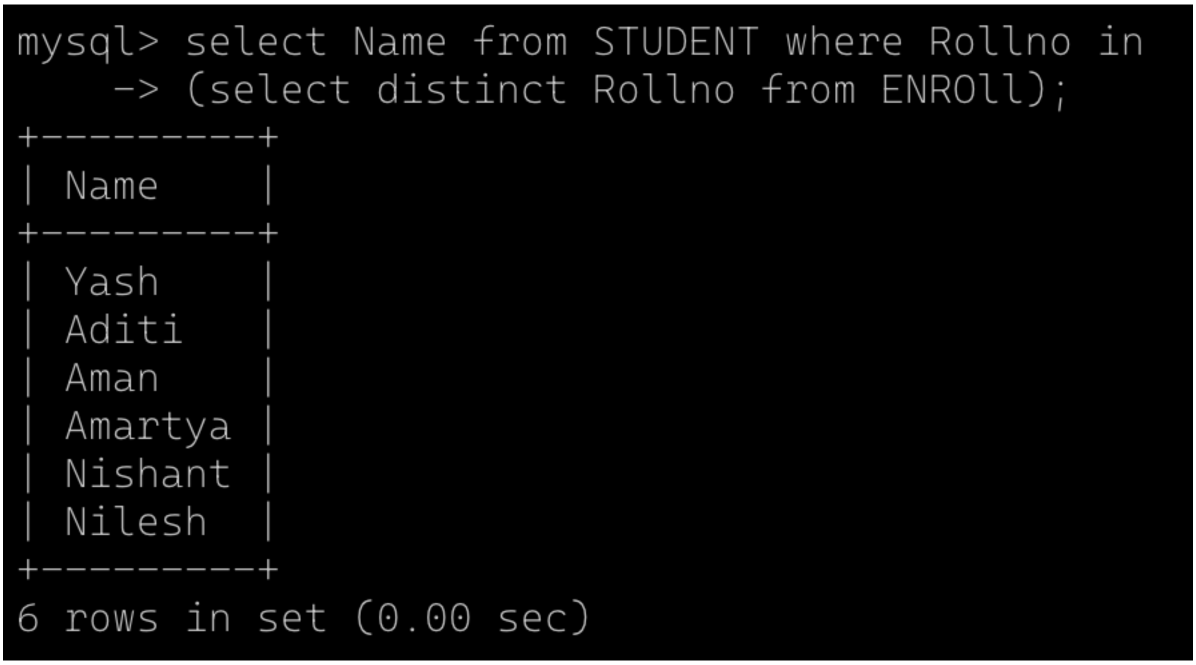


```
mysql> insert into ENROLL values('AC-1254','S04','2020-12-26');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AC-1206','S01','2020-12-20');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AC-1206','S02','2020-12-30');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AC-1207','S02','2020-12-30');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AC-1202','S04','2020-12-22');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AC-1254','S03','2020-12-22');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('PH-723','S03','2020-12-22');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('PH-723','S05','2020-12-12');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AB-943','S05','2020-12-12');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AB-943','S02','2020-12-21');
Query OK, 1 row affected (0.01 sec)

mysql> insert into ENROLL values('AB-943','S04','2020-12-21');
Query OK, 1 row affected (0.00 sec)

mysql> insert into ENROLL values('AB-943','S03','2020-12-21');
Query OK, 1 row affected (0.00 sec)
```

Queries

Q1. Retrieve names of student enrolled in any society.

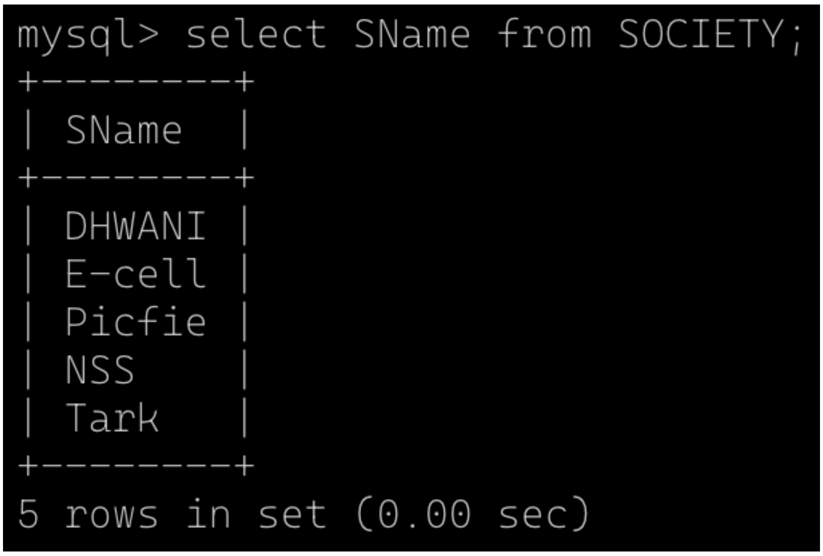
Sol:



```
mysql> select Name from STUDENT where Rollno in
    -> (select distinct Rollno from ENROll);
+---------+
| Name    |
+---------+
| Yash    |
| Aditi   |
| Aman    |
| Amartya |
| Nishant |
| Nilesh  |
+---------+
6 rows in set (0.00 sec)
```

Q2. Retrieve all society names.

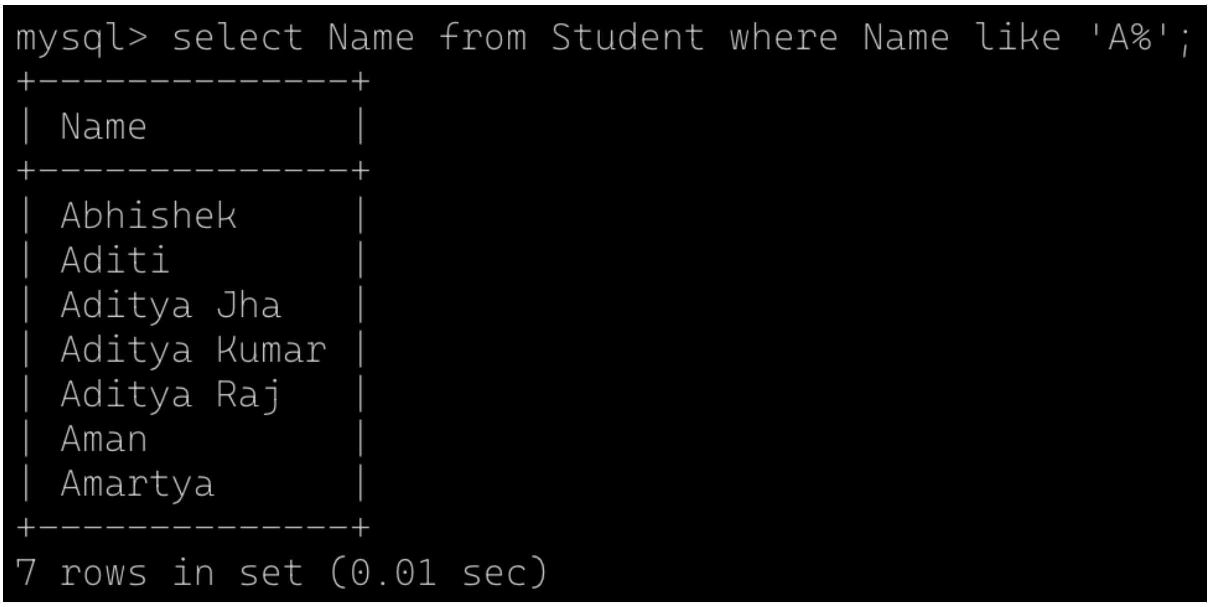
Sol:



```
mysql> select SName from SOCIETY;
+--------+
| SName  |
+--------+
| DHWANI |
| E-cell |
| Picfie |
| NSS    |
| Tark   |
+--------+
5 rows in set (0.00 sec)
```

Q3. Retrieve students names starting with letter ‘A’.

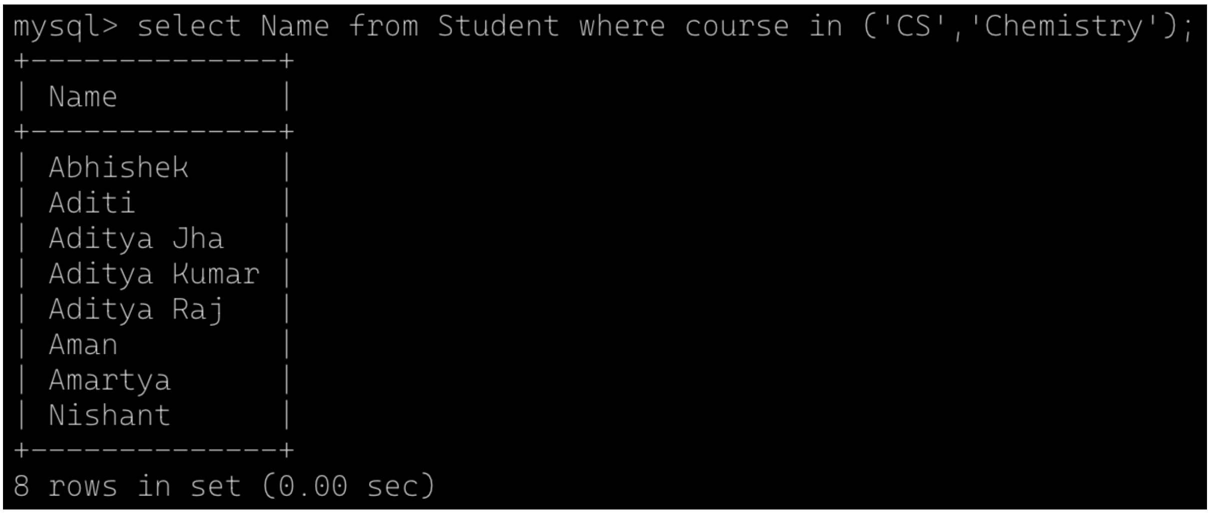
Sol:



```
mysql> select Name from Student where Name like 'A%';
+--------------+
| Name         |
+--------------+
| Abhishek     |
| Aditi        |
| Aditya Jha   |
| Aditya Kumar |
| Aditya Raj   |
| Aman         |
| Amartya      |
+--------------+
7 rows in set (0.01 sec)
```

Q4. Retrieve students studying in course ‘computer sc’ or ‘chemistry’.

Sol:

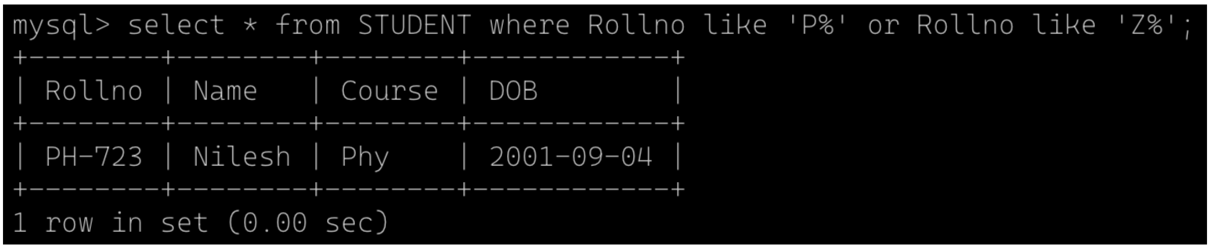


```
mysql> select Name from Student where course in ('CS','Chemistry');
+--------------+
| Name         |
+--------------+
| Abhishek     |
| Aditi        |
| Aditya Jha   |
| Aditya Kumar |
| Aditya Raj   |
| Aman         |
| Amartya      |
| Nishant      |
+--------------+
8 rows in set (0.00 sec)
```

Q5. Retrieve students whose roll no either starts with ‘X’ or ‘Z’.

Sol:

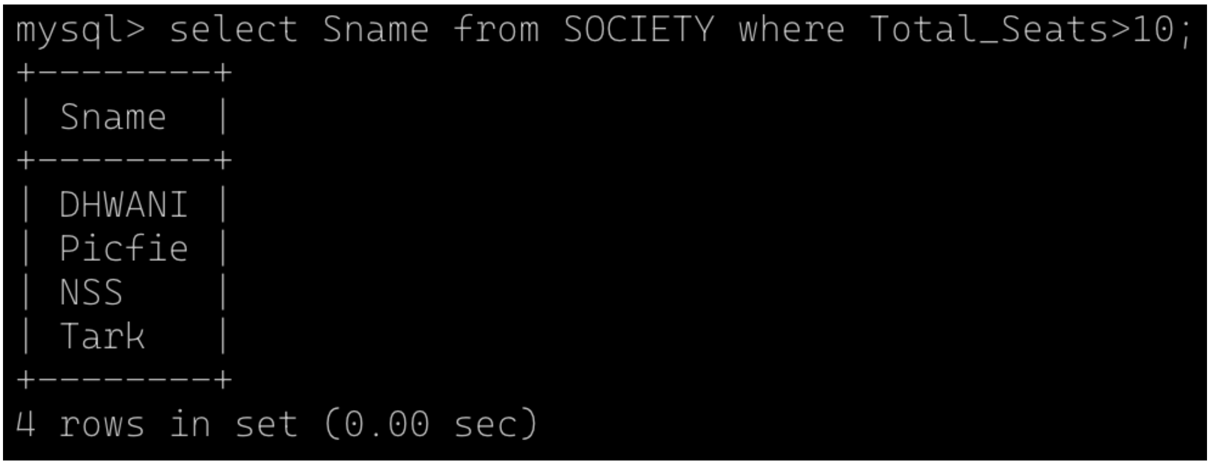
For this command we will use ‘P’ instead of ‘X’



```
mysql> select * from STUDENT where Rollno like 'P%' or Rollno like 'Z%';
+--------+--------+--------+------------+
| Rollno | Name   | Course | DOB        |
+--------+--------+--------+------------+
| PH-723 | Nilesh | Phy    | 2001-09-04 |
+--------+--------+--------+------------+
1 row in set (0.00 sec)
```

Q6. Find society whose capacity is more than 10.

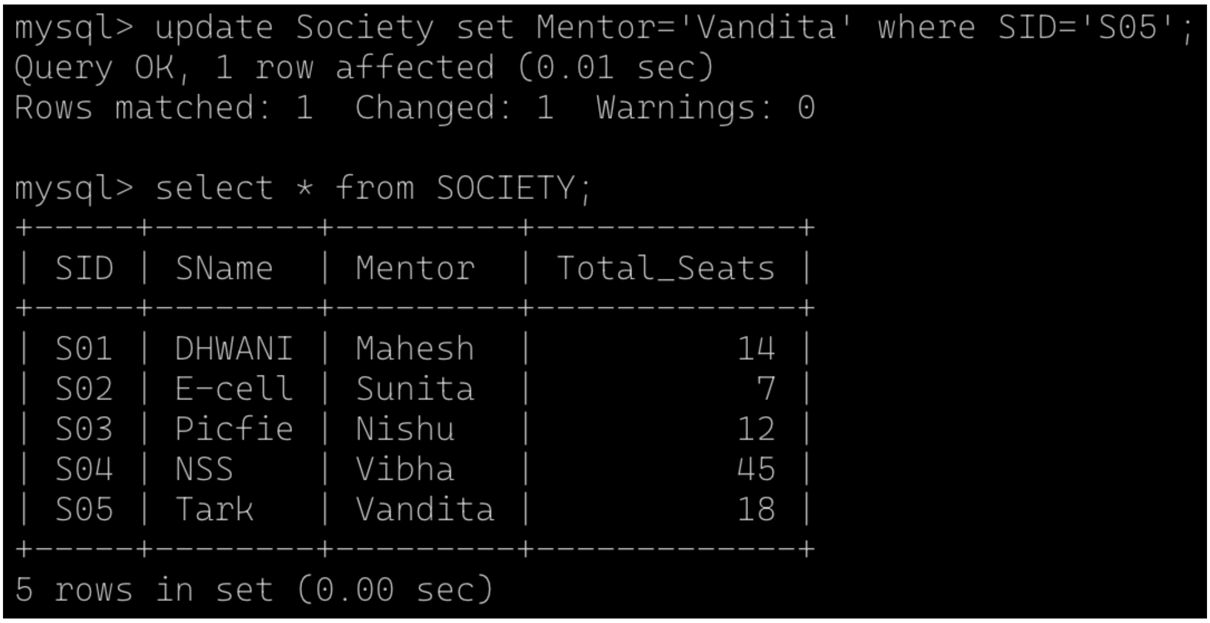
Sol



```
mysql> select Sname from SOCIETY where Total_Seats>10;
+--------+
| Sname  |
+--------+
| DHWANI |
| Picfie |
| NSS    |
| Tark   |
+--------+
4 rows in set (0.00 sec)
```

Q7. Update society table for mentor name for a specific society.

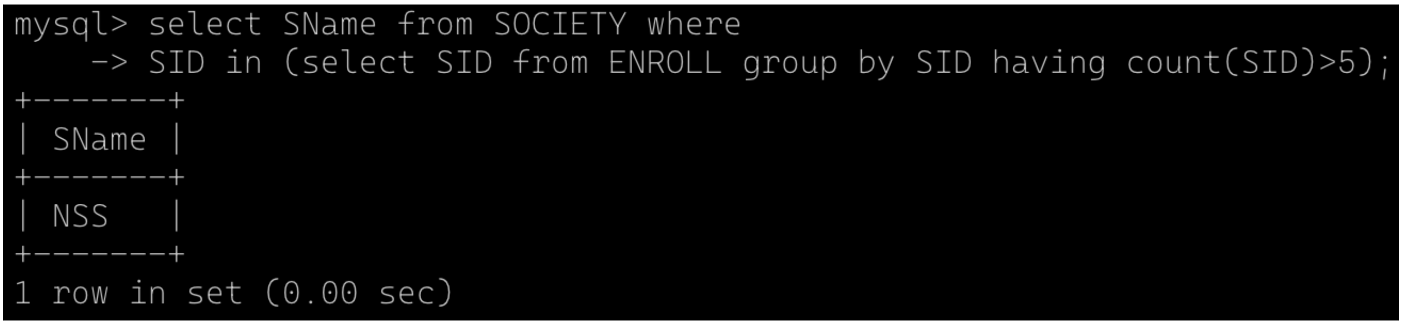
Sol:



```
mysql> update Society set Mentor='Vandita' where SID='S05';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from SOCIETY;
+-----+--------+---------+-------------+
| SID | SName  | Mentor  | Total_Seats |
+-----+--------+---------+-------------+
| S01 | DHWANI | Mahesh  |          14 |
| S02 | E-cell | Sunita  |           7 |
| S03 | Picfie | Nishu   |          12 |
| S04 | NSS    | Vibha   |          45 |
| S05 | Tark   | Vandita |          18 |
+-----+--------+---------+-------------+
5 rows in set (0.00 sec)
```

Q8. Find names of societies with student enrolled > 05.

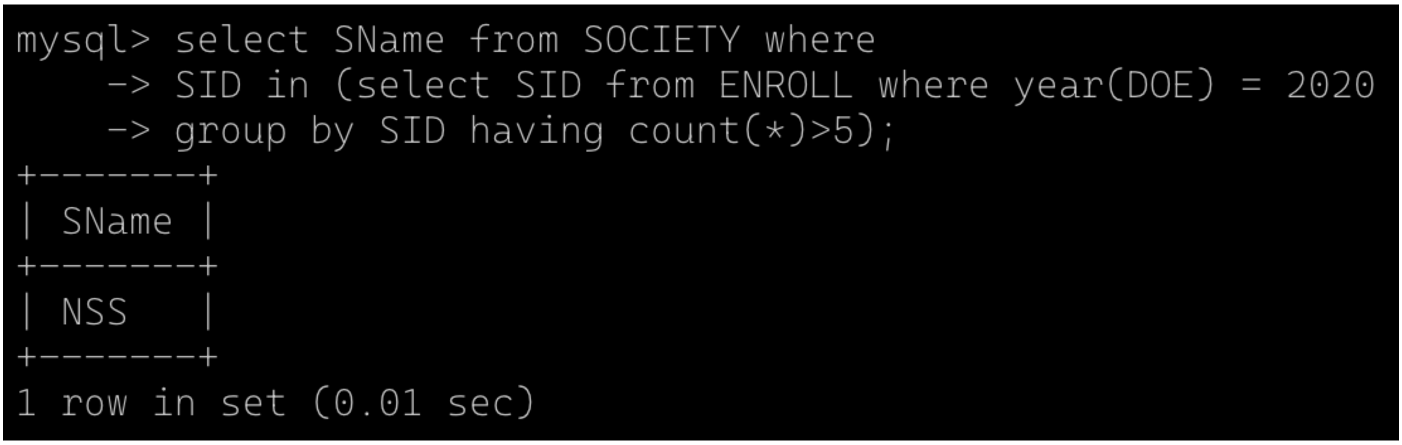
Sol:



```
mysql> select SName from SOCIETY where
    -> SID in (select SID from ENROLL group by SID having count(SID)>5);
+-------+
| SName |
+-------+
| NSS   |
+-------+
1 row in set (0.00 sec)
```

Q9. Find society names in which more than five students have enrolled in a given year.

Sol:

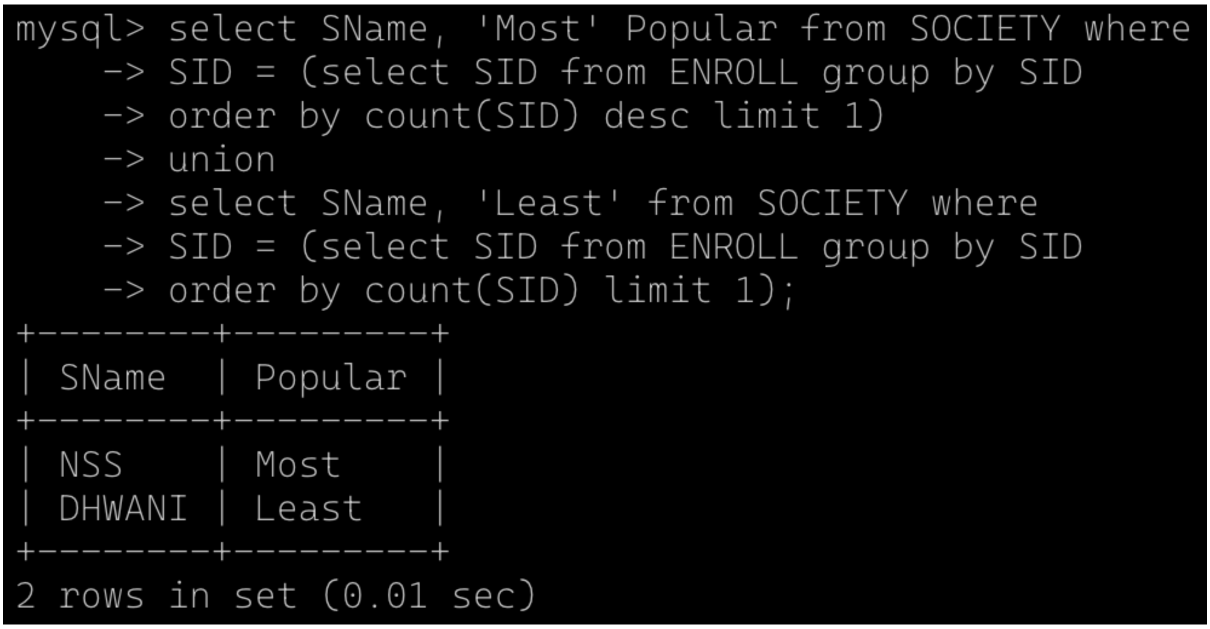


```
mysql> select SName from SOCIETY where
    -> SID in (select SID from ENROLL where year(DOE) = 2020
    -> group by SID having count(*)>5);
+--------+
| SName  |
+--------+
| NSS    |
+--------+
1 row in set (0.01 sec)
```

Q10. Find the most popular and least popular society name (on the basis

of enrolled students).

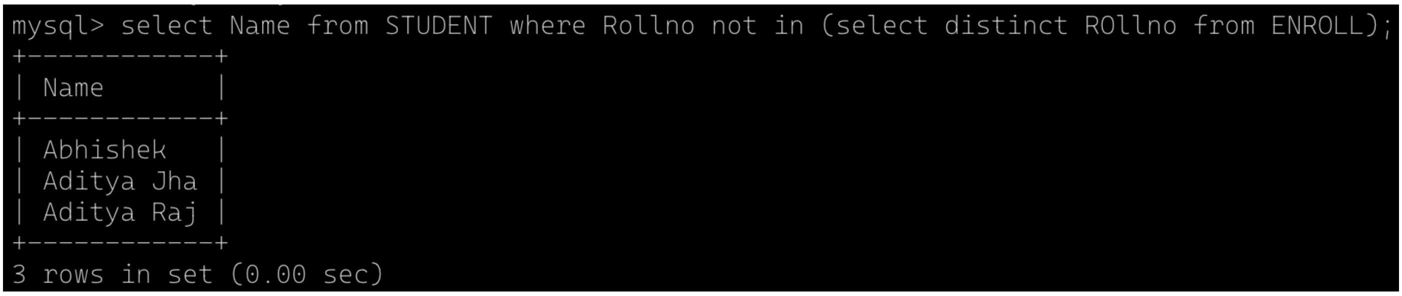
Sol:



```
mysql> select SName, 'Most' Popular from SOCIETY where
    -> SID = (select SID from ENROLL group by SID
    -> order by count(SID) desc limit 1)
    -> union
    -> select SName, 'Least' from SOCIETY where
    -> SID = (select SID from ENROLL group by SID
    -> order by count(SID) limit 1);
+--------+---------+
| SName  | Popular |
+--------+---------+
| NSS    | Most    |
| DHWANI | Least   |
+--------+---------+
2 rows in set (0.01 sec)
```

Q11. Find the student names who are not enrolled in any society.

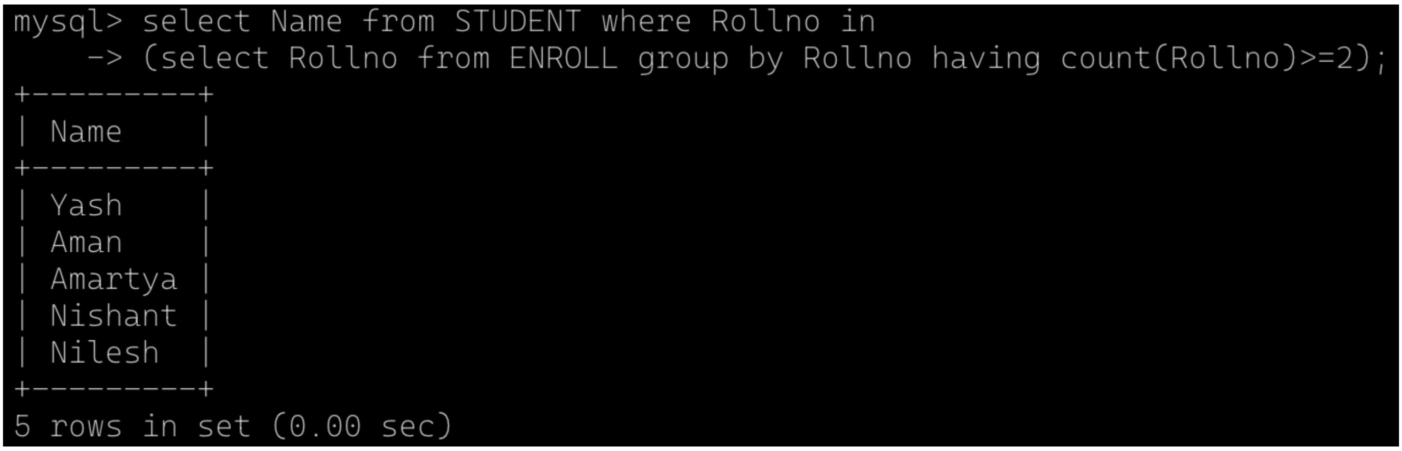
Sol:



```
mysql> select Name from STUDENT where Rollno not in (select distinct ROllno from ENROLL);
+------------+
| Name       |
+------------+
| Abhishek   |
| Aditya Jha |
| Aditya Raj |
+------------+
3 rows in set (0.00 sec)
```

Q12. Find the student names enrolled in at least two societies.

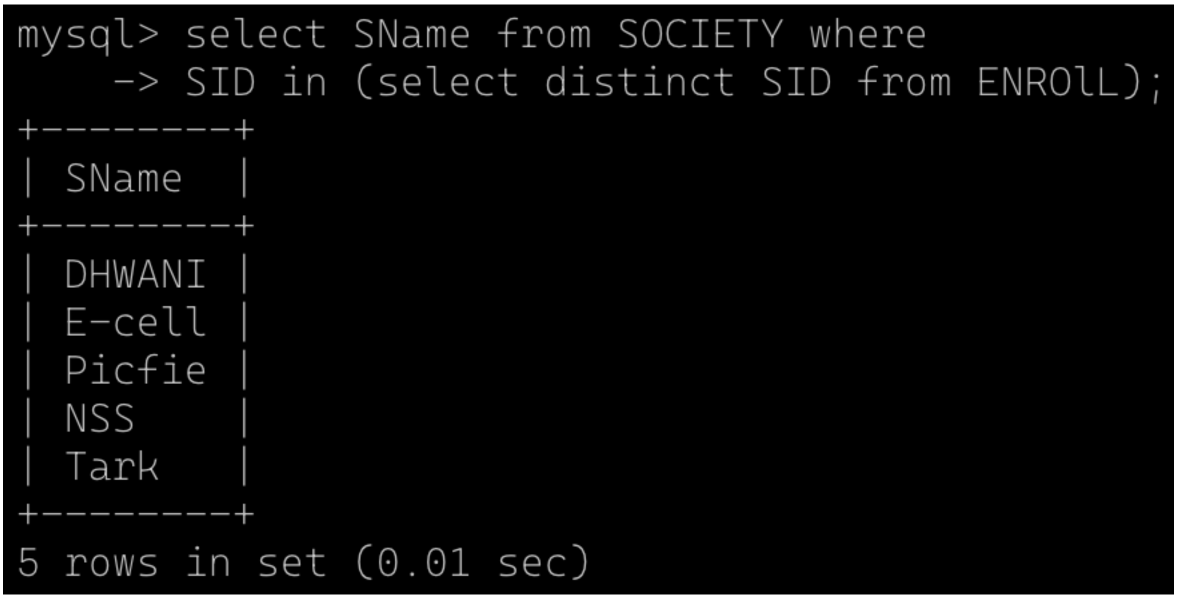
Sol:



```
mysql> select Name from STUDENT where Rollno in
    -> (select Rollno from ENROLL group by Rollno having count(Rollno)>=2);
+---------+
| Name    |
+---------+
| Yash    |
| Aman    |
| Amartya |
| Nishant |
| Nilesh  |
+---------+
5 rows in set (0.00 sec)
```

Q13. Find society names in which any student is enrolled.

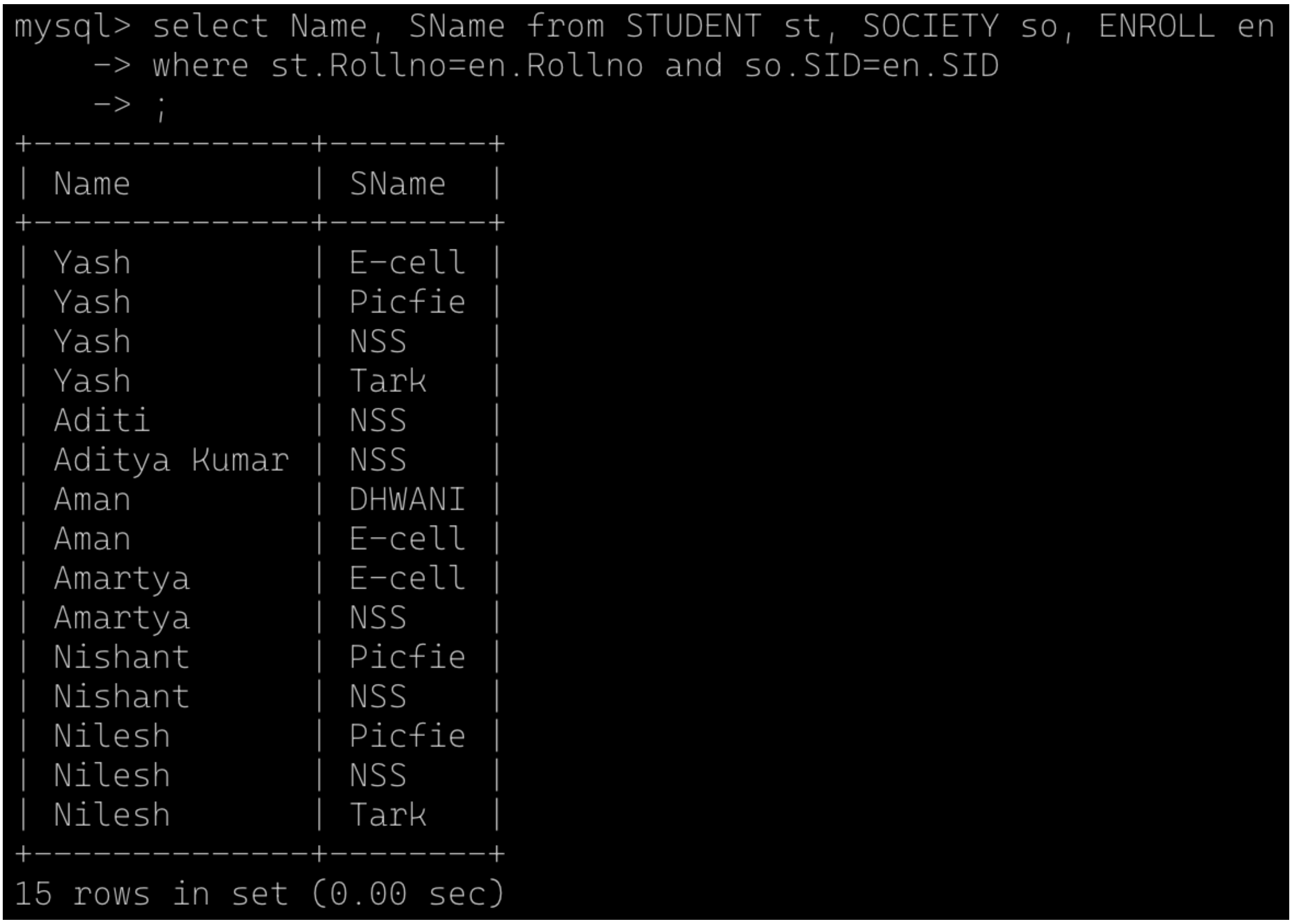
Sol:



```
mysql> select SName from SOCIETY where
    -> SID in (select distinct SID from ENROlL);
+--------+
| SName  |
+--------+
| DHWANI |
| E-cell |
| Picfie |
| NSS    |
| Tark   |
+--------+
5 rows in set (0.01 sec)
```

Q14. Find names of all students enrolled in any society and society names in which any student is enrolled.

Sol:



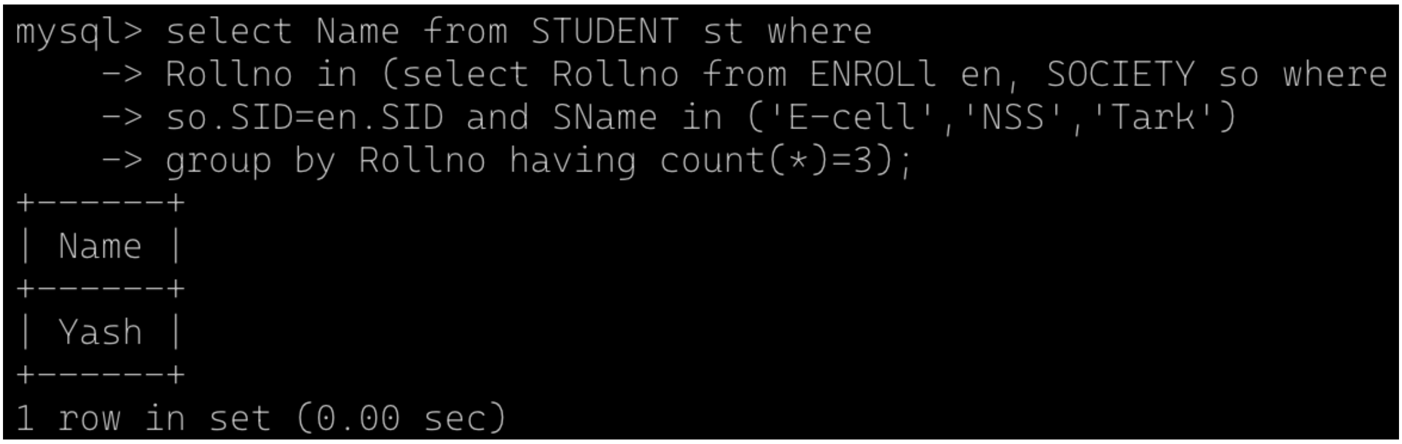
```
mysql> select Name, SName from STUDENT st, SOCIETY so, ENROLL en
    -> where st.Rollno=en.Rollno and so.SID=en.SID
    -> ;
+--------------+--------+
| Name         | SName  |
+--------------+--------+
| Yash         | E-cell |
| Yash         | Picfie |
| Yash         | NSS    |
| Yash         | Tark   |
| Aditi        | NSS    |
| Aditya Kumar | NSS    |
| Aman         | DHWANI |
| Aman         | E-cell |
| Amartya      | E-cell |
| Amartya      | NSS    |
| Nishant      | Picfie |
| Nishant      | NSS    |
| Nilesh       | Picfie |
| Nilesh       | NSS    |
| Nilesh       | Tark   |
+--------------+--------+
15 rows in set (0.00 sec)
```

Q15. Find names of students who are enrolled in all three societies

‘debating’, ‘dancing’ and ‘sashakt’.

Sol:

We will use E-cell, NSS and Tark for this case.



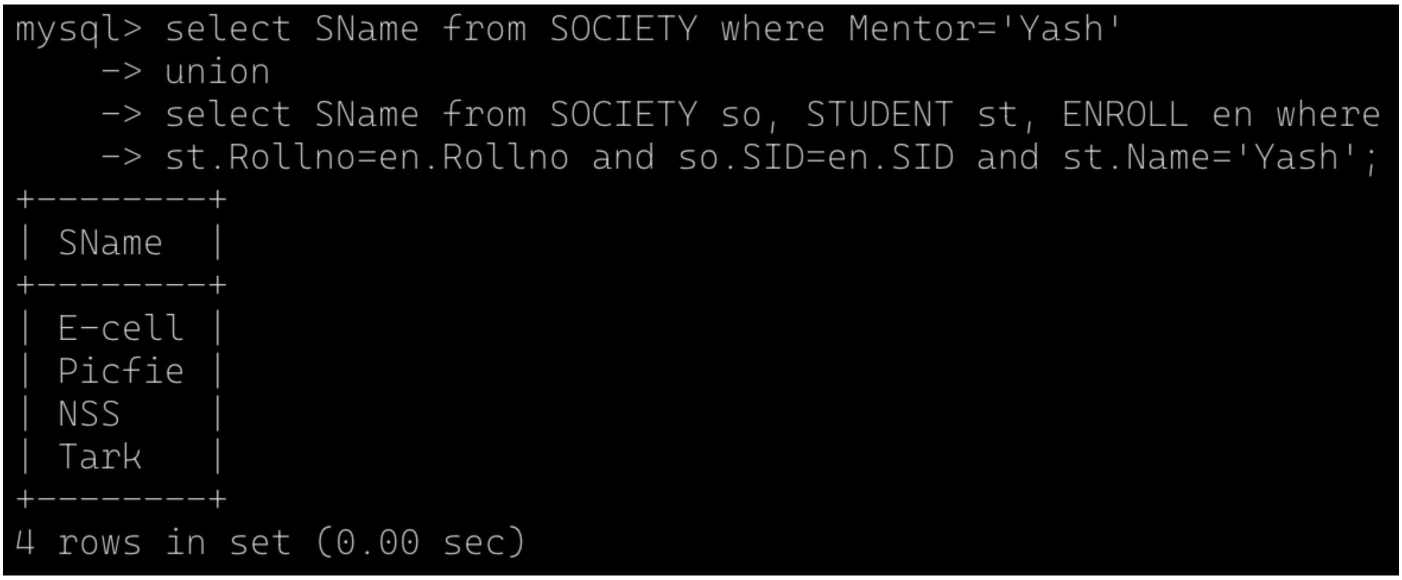
```
mysql> select Name from STUDENT st where
    -> Rollno in (select Rollno from ENROLl en, SOCIETY so where
    -> so.SID=en.SID and SName in ('E-cell','NSS','Tark')
    -> group by Rollno having count(*)=3);
+------+
| Name |
+------+
| Yash |
+------+
1 row in set (0.00 sec)
```

Q16. Find society names that has ‘abc’ as mentor or ‘abc’ as the name

of enrolled student.

Sol:

Here we’ll be using ‘Yash’ as ‘abc’.



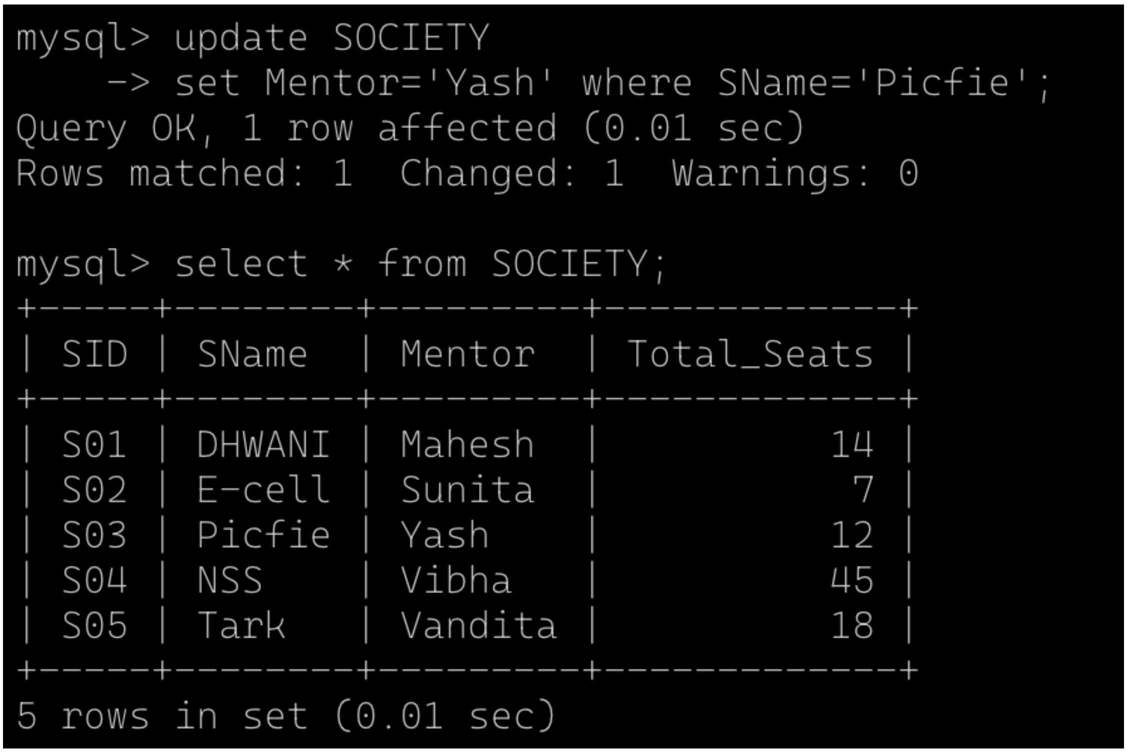
```
mysql> select SName from SOCIETY where Mentor='Yash'
    -> union
    -> select SName from SOCIETY so, STUDENT st, ENROLL en where
    -> st.Rollno=en.Rollno and so.SID=en.SID and st.Name='Yash';
+--------+
| SName  |
+--------+
| E-cell |
| Picfie |
| NSS    |
| Tark   |
+--------+
4 rows in set (0.00 sec)
```

Q17. Find society names whose mentor name is same as that of any

enrolled student in it.

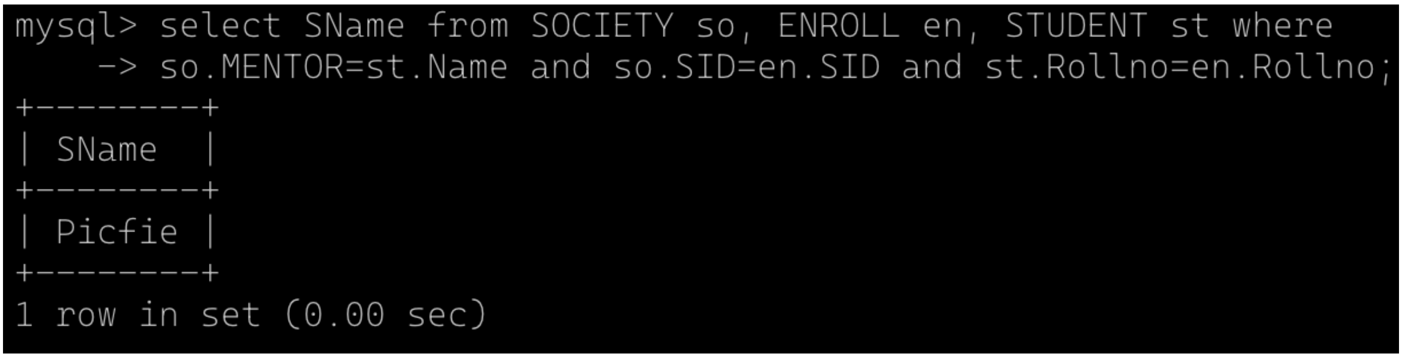
Sol:

Since we don’t have such data, we will update the Mentor of Picfie to Yash. After that, we will run this query.



```
mysql> update SOCIETY
    -> set Mentor='Yash' where SName='Picfie';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from SOCIETY;
+-----+--------+---------+-------------+
| SID | SName  | Mentor  | Total_Seats |
+-----+--------+---------+-------------+
| S01 | DHWANI | Mahesh  |          14 |
| S02 | E-cell | Sunita  |           7 |
| S03 | Picfie | Yash    |          12 |
| S04 | NSS    | Vibha   |          45 |
| S05 | Tark   | Vandita |          18 |
+-----+--------+---------+-------------+
5 rows in set (0.01 sec)
```

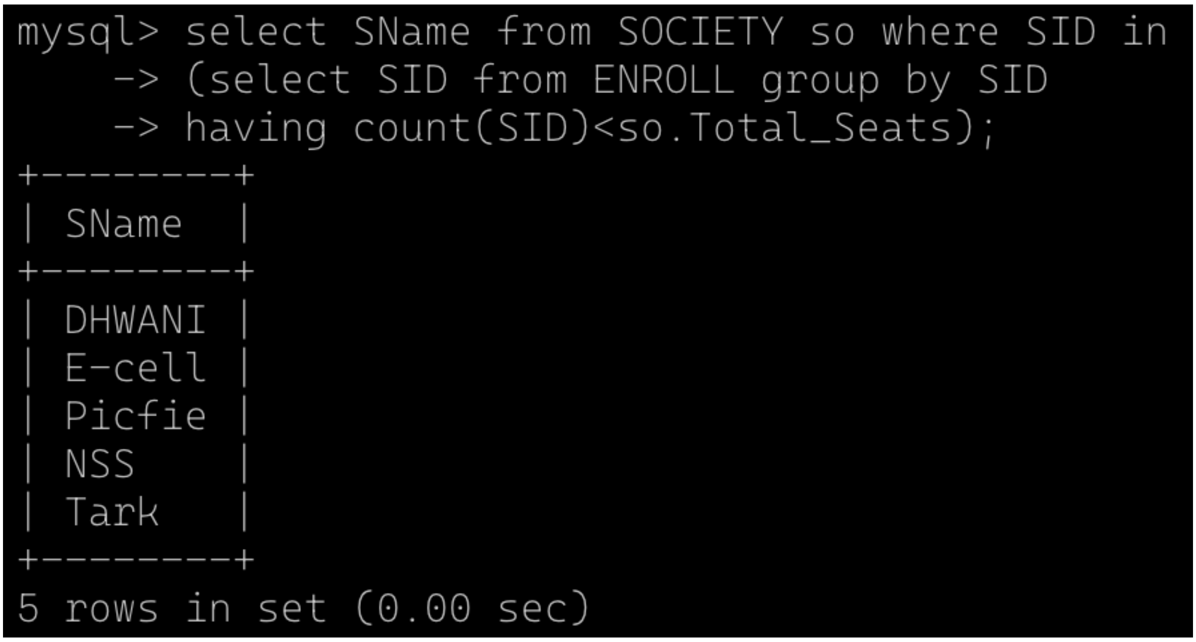
Now, we will perform the required query.



```
mysql> select SName from SOCIETY so, ENROLL en, STUDENT st where
    -> so.MENTOR=st.Name and so.SID=en.SID and st.Rollno=en.Rollno;
+--------+
| SName  |
+--------+
| Picfie |
+--------+
1 row in set (0.00 sec)
```

Q18. Find the society names in which number of enrolled students are less than its capacity.

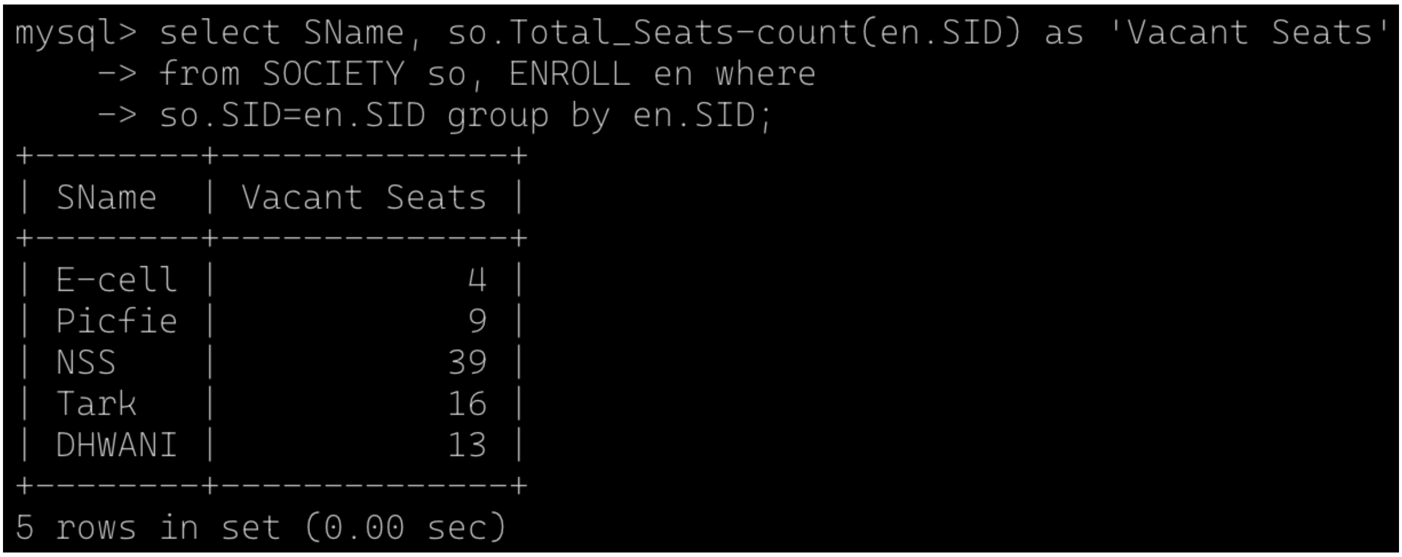
Sol:



```
mysql> select SName from SOCIETY so where SID in
    -> (select SID from ENROLL group by SID
    -> having count(SID)<so.Total_Seats);
+--------+
| SName  |
+--------+
| DHWANI |
| E-cell |
| Picfie |
| NSS    |
| Tark   |
+--------+
5 rows in set (0.00 sec)
```

Q19. Display the vacant seats for each society.

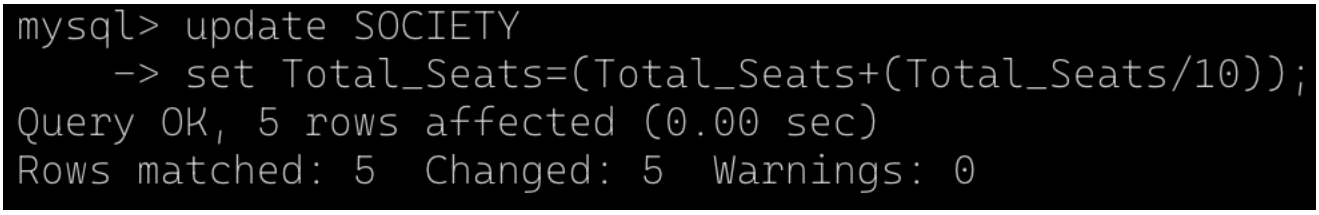
Sol:



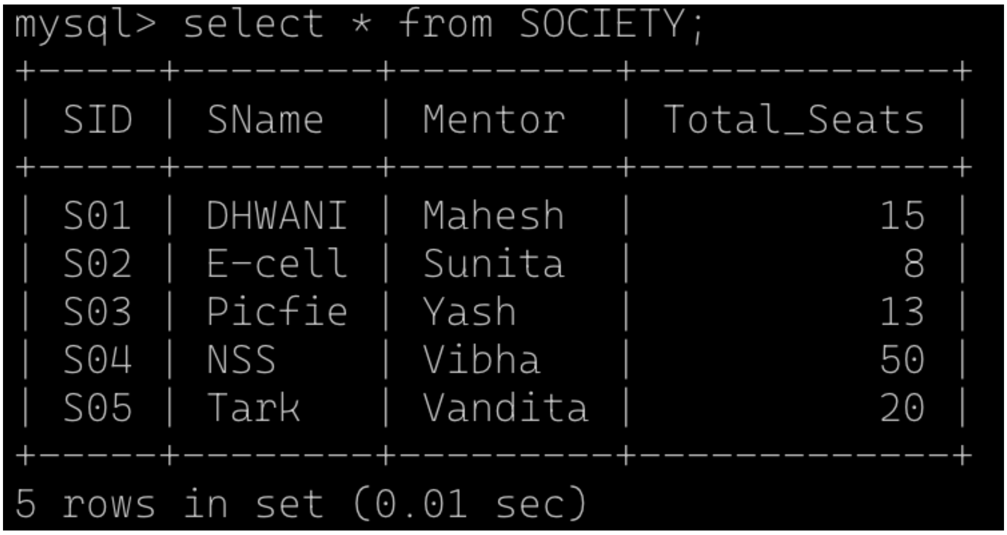
```
mysql> select SName, so.Total_Seats-count(en.SID) as 'Vacant Seats'
    -> from SOCIETY so, ENROLL en where
    -> so.SID=en.SID group by en.SID;
+--------+--------------+
| SName  | Vacant Seats |
+--------+--------------+
| E-cell |            4 |
| Picfie |            9 |
| NSS    |           39 |
| Tark   |           16 |
| DHWANI |           13 |
+--------+--------------+
5 rows in set (0.00 sec)
```

Q20. Increment capacity of each society by 10%.

Sol:



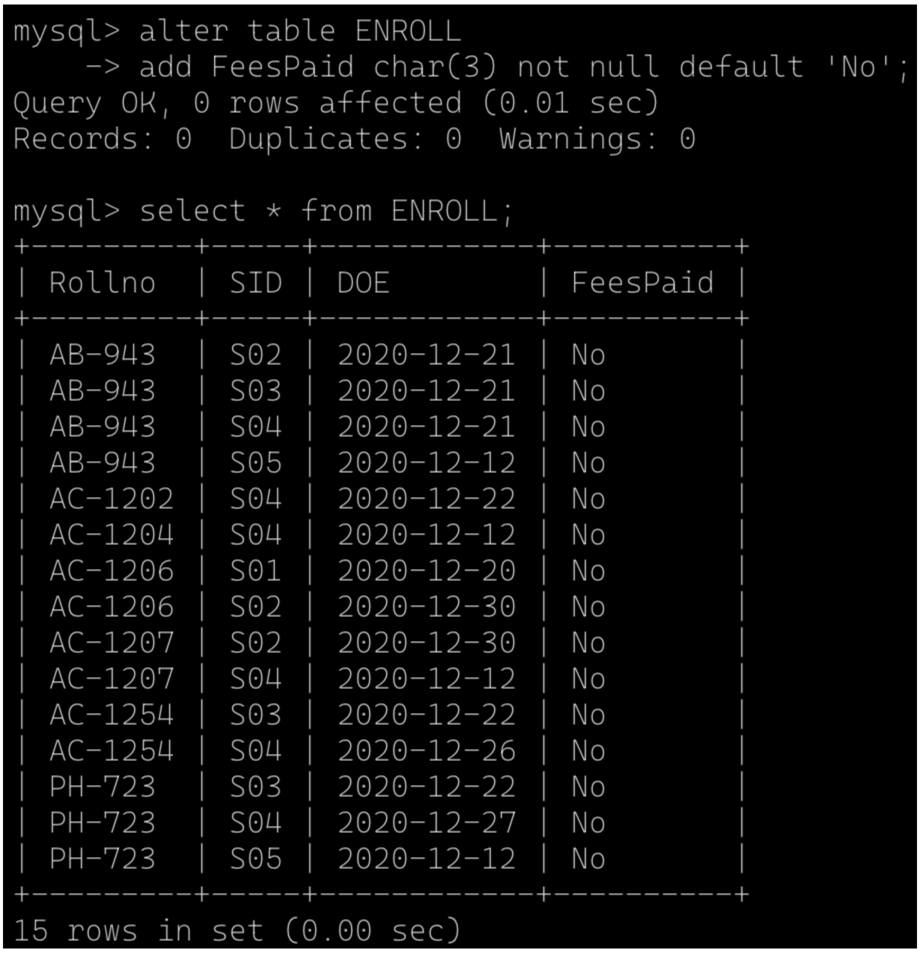
```
mysql> update SOCIETY
    -> set Total_Seats=(Total_Seats+(Total_Seats/10));
Query OK, 5 rows affected (0.00 sec)
Rows matched: 5  Changed: 5  Warnings: 0
```



```
mysql> select * from SOCIETY;
+-----+--------+---------+-------------+
| SID | SName  | Mentor  | Total_Seats |
+-----+--------+---------+-------------+
| S01 | DHWANI | Mahesh  |          15 |
| S02 | E-cell | Sunita  |           8 |
| S03 | Picfie | Yash    |          13 |
| S04 | NSS    | Vibha   |          50 |
| S05 | Tark   | Vandita |          20 |
+-----+--------+---------+-------------+
5 rows in set (0.01 sec)
```

Q21. Add enrollment fees paid (‘yes’/’No’) field in the enrollment table.

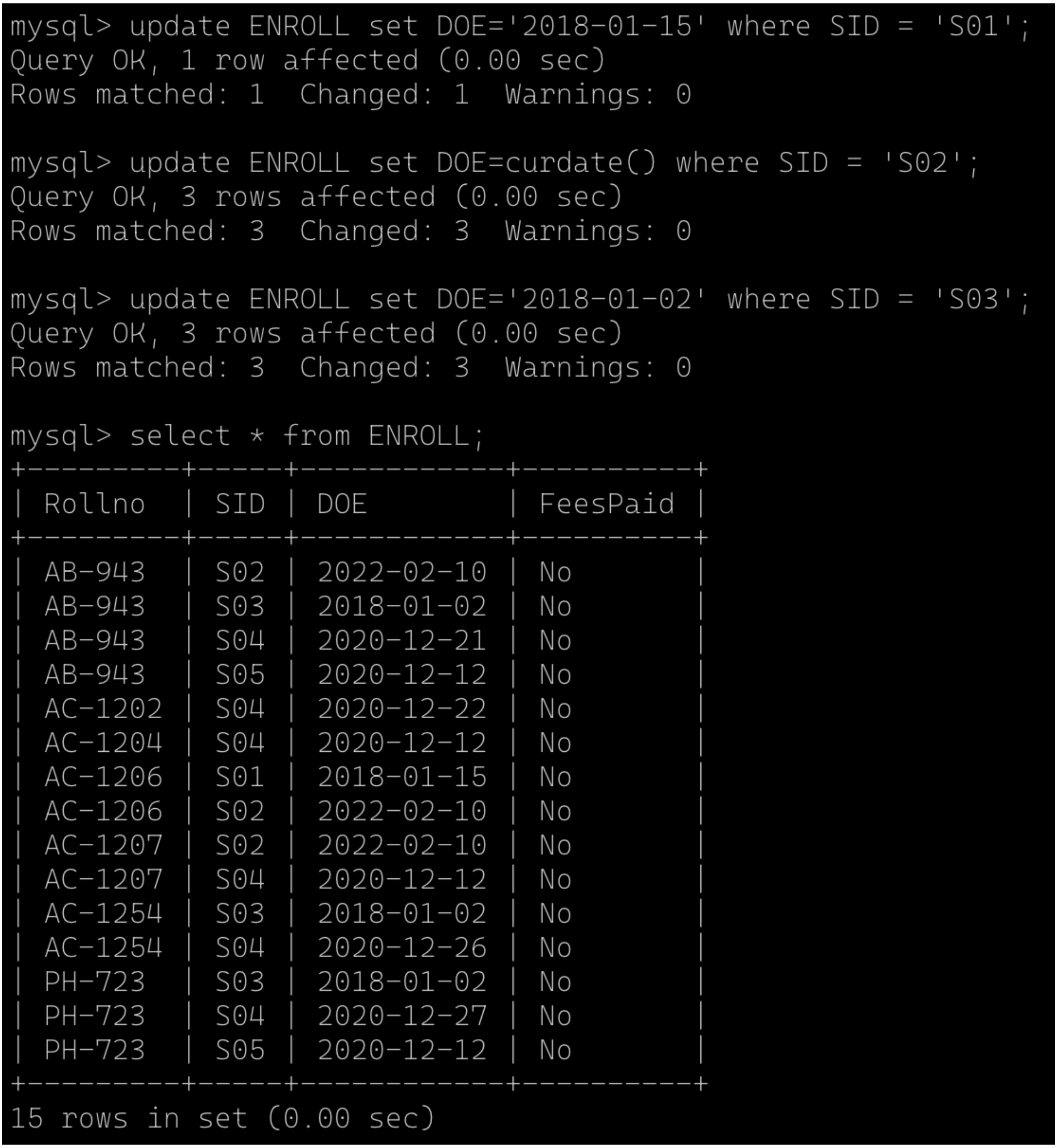
Sol:



```
mysql> alter table ENROLL
    -> add FeesPaid char(3) not null default 'No';
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from ENROLL;
+---------+-----+------------+----------+
| Rollno  | SID | DOE        | FeesPaid |
+---------+-----+------------+----------+
| AB-943  | S02 | 2020-12-21 | No       |
| AB-943  | S03 | 2020-12-21 | No       |
| AB-943  | S04 | 2020-12-21 | No       |
| AB-943  | S05 | 2020-12-12 | No       |
| AC-1202 | S04 | 2020-12-22 | No       |
| AC-1204 | S04 | 2020-12-12 | No       |
| AC-1206 | S01 | 2020-12-20 | No       |
| AC-1206 | S02 | 2020-12-30 | No       |
| AC-1207 | S02 | 2020-12-30 | No       |
| AC-1207 | S04 | 2020-12-12 | No       |
| AC-1254 | S03 | 2020-12-22 | No       |
| AC-1254 | S04 | 2020-12-26 | No       |
| PH-723  | S03 | 2020-12-22 | No       |
| PH-723  | S04 | 2020-12-27 | No       |
| PH-723  | S05 | 2020-12-12 | No       |
+---------+-----+------------+----------+
15 rows in set (0.00 sec)
```

Q22. Update date of enrollment of society s1 to ‘2018-01-15’, s2 to current date and s3 to ‘2018-01-02’.

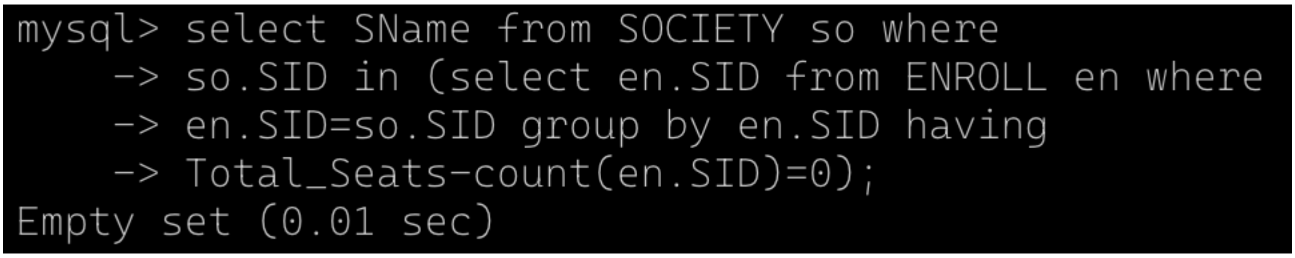
Sol:



```
mysql> update ENROLL set DOE='2018-01-15' where SID = 'S01';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> update ENROLL set DOE=curdate() where SID = 'S02';
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> update ENROLL set DOE='2018-01-02' where SID = 'S03';
Query OK, 3 rows affected (0.00 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> select * from ENROLL;
+---------+-----+------------+----------+
| Rollno  | SID | DOE        | FeesPaid |
+---------+-----+------------+----------+
| AB-943  | S02 | 2022-02-10 | No       |
| AB-943  | S03 | 2018-01-02 | No       |
| AB-943  | S04 | 2020-12-21 | No       |
| AB-943  | S05 | 2020-12-12 | No       |
| AC-1202 | S04 | 2020-12-22 | No       |
| AC-1204 | S04 | 2020-12-12 | No       |
| AC-1206 | S01 | 2018-01-15 | No       |
| AC-1206 | S02 | 2022-02-10 | No       |
| AC-1207 | S02 | 2022-02-10 | No       |
| AC-1207 | S04 | 2020-12-12 | No       |
| AC-1254 | S03 | 2018-01-02 | No       |
| AC-1254 | S04 | 2020-12-26 | No       |
| PH-723  | S03 | 2018-01-02 | No       |
| PH-723  | S04 | 2020-12-27 | No       |
| PH-723  | S05 | 2020-12-12 | No       |
+---------+-----+------------+----------+
15 rows in set (0.00 sec)
```

Q23. Find society names whose enrollment is over.

Sol:



```
mysql> select SName from SOCIETY so where
    -> so.SID in (select en.SID from ENROLL en where
    -> en.SID=so.SID group by en.SID having
    -> Total_Seats-count(en.SID)=0);
Empty set (0.01 sec)
```