

**Name - Priyanshu**

**Roll Number - AD-1274**

**Course - BSc(Hons) Computer Science**

**Semester - III**

**Date of Submission - 24<sup>th</sup> January, 2023**

Q1) Retrieve names of student enrolled in any society.

Ans:-

select name from student where rollno in (select rollno from enrollment);

```
mysql> select name from student where rollno in (select rollno from enrollment);
+-----+
| name  |
+-----+
| Priyam|
| Priyanshu|
+-----+
2 rows in set (0.00 sec)
```

Q2) Retrieve all society names.

Ans:-

select distinct sname from society;

```
mysql> select distinct sname from society;
+-----+
| sname  |
+-----+
| dhawani|
| picfie |
| saksham|
| eco    |
+-----+
4 rows in set (0.00 sec)
```

Q3) Retrieve student names starting with letter 'A'.

Ans:-

select name from student where name like 'a%';

```
mysql> select name from student where name like 'a%';
+-----+
| name  |
+-----+
| Ayush |
| Anshika |
+-----+
2 rows in set (0.00 sec)
```

Q4) Retrieve students studying in course 'computer sc' or 'chemistry'.

Ans:-

select name, cname from student where cname = 'cs' or cname = 'chemistry';

```
mysql> select name, cname from student where cname = 'cs' or cname = 'chemistry';
+-----+-----+
| name  | cname |
+-----+-----+
| Ayush | cs     |
| Priyanshu | cs     |
| Nikhil | cs     |
| Vipin | cs     |
| Ritesh | chemistry |
| Riyansh | chemistry |
+-----+-----+
6 rows in set (0.00 sec)
```

Q5) Retrieve students whose rollno either starts with 'X' or 'Z'.

Ans:-

select name, rollno from student where rollno like 'X%' or rollno like 'Z%';

```
mysql> select name, rollno from student where rollno like 'X%' or rollno like 'Z%';
+-----+-----+
| name  | rollno |
+-----+-----+
| Priyam | X-208  |
| Tanu   | Z-204  |
+-----+-----+
2 rows in set (0.00 sec)
```

Q6) Find society whose capacity is more than 10.

Ans:-

select \* from society where capacity >= 10;

```
mysql> select * from society where capacity > 10;
+-----+-----+-----+-----+
| sid  | sname  | mentorname | capacity |
+-----+-----+-----+-----+
| 01   | dhawani | Sonu Kumar | 15       |
| 08   | dhawani | Sonu Kumar | 15       |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

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

Ans:-

update society set mentorname = 'Monika Sharma' where sname = 'saksham';

```
mysql> update society set mentorname = 'Monika Sharma' where sname = 'saksham';
Query OK, 3 rows affected (0.01 sec)
Rows matched: 3  Changed: 3  Warnings: 0

mysql> select * from society;
+-----+-----+-----+-----+
| sid  | sname  | mentorname | capacity |
+-----+-----+-----+-----+
| 01   | dhawani | Sonu Kumar | 15       |
| 02   | picfie  | Neha Kumari | 10       |
| 03   | picfie  | Neha Kumari | 10       |
| 04   | saksham | Monika Sharma | 8        |
| 05   | picfie  | Neha Kumari | 10       |
| 06   | eco     | Pankaj Thakur | 10       |
| 07   | saksham | Monika Sharma | 8        |
| 08   | dhawani | Sonu Kumar | 15       |
| 09   | picfie  | Neha Kumari | 10       |
| 10   | saksham | Monika Sharma | 8        |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

Q8) Remove details for a student who is not enrolled in any society.

Ans:-

delete from student where rollno not in (select rollno from enrollment);

```
mysql> delete from student where rollno not in (select rollno from enrollment);
Query OK, 8 rows affected (0.01 sec)

mysql> select * from student;
+-----+-----+-----+-----+
| rollno | name      | cname    | dob      |
+-----+-----+-----+-----+
| X-208   | Priyam    | commerce | 2003-05-09 |
| AD-127  | Priyanshu | cs       | 2002-08-03 |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Q9) Increment capacity of each society by 10%.

Ans:-

update society set capacity = capacity + (capacity \* 10) / 100;

```
mysql> select * from society;
+-----+-----+-----+-----+
| sid  | sname  | mentorname | capacity |
+-----+-----+-----+-----+
| 01   | dhawani | Sonu Kumar | 15       |
| 02   | picfie  | Neha Kumari | 10       |
| 03   | picfie  | Neha Kumari | 10       |
| 04   | saksham | Monika Sharma | 8        |
| 05   | picfie  | Neha Kumari | 10       |
| 06   | eco     | Pankaj Thakur | 10       |
| 07   | saksham | Monika Sharma | 8        |
| 08   | dhawani | Sonu Kumar | 15       |
| 09   | picfie  | Neha Kumari | 10       |
| 10   | saksham | Monika Sharma | 8        |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql> update society set capacity = capacity + (capacity * 10) / 100;
Query OK, 10 rows affected (0.01 sec)
Rows matched: 10  Changed: 10  Warnings: 0

mysql> select * from society;
+-----+-----+-----+-----+
| sid  | sname  | mentorname | capacity |
+-----+-----+-----+-----+
| 01   | dhawani | Sonu Kumar | 17       |
| 02   | picfie  | Neha Kumari | 11       |
| 03   | picfie  | Neha Kumari | 11       |
| 04   | saksham | Monika Sharma | 9        |
| 05   | picfie  | Neha Kumari | 11       |
| 06   | eco     | Pankaj Thakur | 11       |
| 07   | saksham | Monika Sharma | 9        |
| 08   | dhawani | Sonu Kumar | 17       |
| 09   | picfie  | Neha Kumari | 11       |
| 10   | saksham | Monika Sharma | 9        |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

Q10) Find the student names who are not enrolled in any society.

Ans:-

select name from student where rollno not in (select rollno from enrollment);

```
mysql> select name from student where rollno not in (select rollno from enrollment);
+-----+
| name |
+-----+
| Ayush |
| Anshika |
| Rishav |
| Nikhil |
| Vipin |
| Tanu |
| Ritesh |
| Riyansh |
+-----+
8 rows in set (0.00 sec)
```

After all the queries the table is as below :-

```
mysql> select * from student;
```

rollno	name	cname	dob
X-208	Priyam	commerce	2003-05-09
AD-127	Priyanshu	cs	2002-08-03

2 rows in set (0.00 sec)

```
mysql> select * from society;
```

sid	sname	mentorname	capacity
01	dhawani	Sonu Kumar	17
02	picfie	Neha Kumari	11
03	picfie	Neha Kumari	11
04	saksham	Monika Sharma	9
05	picfie	Neha Kumari	11
06	eco	Pankaj Thakur	11
07	saksham	Monika Sharma	9
08	dhawani	Sonu Kumar	17
09	picfie	Neha Kumari	11
10	saksham	Monika Sharma	9

10 rows in set (0.00 sec)

```
mysql> select * from enrollment;
```

rollno	sid	doe
101	01	2003-04-05
102	02	2003-05-05
103	03	2001-05-05
X-208	04	2003-05-09
AD-127	05	2002-08-03
126	06	2002-08-08
129	07	2003-06-10
487	08	2005-06-10
105	09	2005-06-04
100	10	2001-06-10

Q11. Find total number of students whose age is > 20 years.

Sol :-

Query :-

```
select
    name,
    rollno,
    cname,
    dob,
    curdate(),
    timestampdiff(year, dob, curdate()) as age
from
    student;
```

Output :-

name	rollno	cname	dob	curdate()	age
Nikhil	X-1263	pmcs	2001-04-02	2023-02-20	21
Vipin	Z-1601	pmcs	2001-04-03	2023-02-20	21
Priyanshu	AD-1274	cs	2003-04-08	2023-02-20	19
Satendar	A-1284	cs	2002-04-07	2023-02-20	20
Deepak	B-1230	zoology	2002-04-03	2023-02-20	20
Bhoomika	P-1276	pscs	2002-11-27	2023-02-20	20
Aman	B-1290	pscs	2002-04-10	2023-02-20	20
Meenakshi	M-1274	physics	2004-01-09	2023-02-20	19

Query :-

```
select
    *
from
    student
where
    timestampdiff(year, dob, curdate()) > 20;
```

Output :-

```
-- OUTPUT
-- +-----+-----+-----+-----+
-- | rollno | name  | cname | dob      |
-- +-----+-----+-----+-----+
-- | X-1263 | Nikhil | pmcs  | 2001-04-02 |
-- | Z-1601 | Vipin  | pmcs  | 2001-04-03 |
-- +-----+-----+-----+-----+
```

Q12. Find the student names enrolled in atleast two societies.

Sol :-

Query :-

```
select
    rollno,
```

```

count(*) as count
from
  enrollment
group by
  rollno
having
  count > 1;

```

**Output :-**

```

+-----+-----+
| rollno | count |
+-----+-----+
| P-1276 |     2 |
| X-1263 |     2 |
+-----+-----+

```

**Query :-**

```

select
  name
from
  student
where
  rollno = 'p-1276'
or rollno = 'X-1263';

```

**Output :-**

```

+-----+
| name   |
+-----+
| Nikhil |
| Bhoomika |
+-----+

```

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

Sol :-

**Query :-**

```

select
  sname
from
  society
where
  sid in (
    select
      sid
    from
      enrollment
  );

```



**Output :-**

```
+-----+
| sname |
+-----+
| dancing |
| shasakt |
| eco |
| tech |
| music |
| hacking |
| coding |
| cricket |
+-----+
```

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

Sol :-

**Query :-**

```
select
    name
from
    student
where
    rollno in (
        select
            distinct(rollno)
        from
            enrollment
    );
```

**Output :-**

```
+-----+
| name |
+-----+
| Nikhil |
| Vipin |
| Priyanshu |
| Satendar |
| Bhoomika |
| Aman |
| Meenakshi |
+-----+
```

**Query**

```
select
    sname
from
```

```

society
where
  sid in (
    select
      sid
    from
      enrollment
  );

```

### Output

```

+-----+
| sname |
+-----+
| dancing |
| shasakt |
| eco |
| tech |
| music |
| hacking |
| coding |
| cricket |
+-----+

```

Q15. Find names of students who are enrolled in all three societies 'debating', 'dancing' and 'sashakt'.

Sol :-

### Query

```

select
  s.name
from
  student s
  inner join enrollment e on s.rollno = e.rollno
  inner join society so on e.sid = so.sid
where
  so.sid in ('debating', 'dancing', 'sashakt')
group by
  s.name
having
  count(distinct so.sid) = 3;

```

### Output

Empty set

Q16. Find society names that has 'abc' as mentor or 'abc' as the name of enrolled student.

Sol :-

### Query

```

select

```

```

distinct s.sid,
s.sname
from
  society s
left join enrollment e on s.sid = e.sid
left join student st on e.rollno = st.rollno
where
  s.mentorname like '%abc%'
or st.name = '%abc%';

```

### Output

```

+-----+-----+
| sid | sname |
+-----+-----+
| 03  | shasakt |
+-----+-----+

```

Q17. Find society names whose mentor name is same as that of any enrolled student in it.

Sol :-

### Query

```

select
  distinct s.sname
from
  society s
inner join enrollment e on s.sid = e.sid
where
  s.mentorname in (
    select
      name
    from
      student
    where
      rollno = e.rollno
  );

```

### Output

```

+-----+
| sname |
+-----+
| debating |
+-----+

```

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

Sol :-

### Query

```

select

```

```

s.sname
from
  society s
  inner join enrollment e on s.sid = e.sid
where
  (
    select
      count(*)
    from
      enrollment e2
    where
      e2.sid = s.sid
  ) < s / capacity;

```

## Output

```

+-----+
| sname |
+-----+
| tech  |
| music |
| dancing |
| dancing |
| tech  |
| shasakt |
| coding |
| eco   |
| hacking |
| cricket |
| debating |
| dancing |
| shasakt |
| debating |
+-----+

```

Q19. Display the vacant seats for each society.

Sol :-

## Query

```

select
  s.sname,
  (
    s.capacity - (
      select
        count(*)
      from
        enrollment e
      where
        e.sid = s.sid
    )
  ) as vacant_seats
from
  society s;

```

## Output

```
+-----+-----+
| sname | vacant_seats |
+-----+-----+
| debating | 15 |
| dancing | 19 |
| shasakt | 9 |
| eco | 27 |
| picfie | 33 |
| tech | 20 |
| music | 5 |
| hacking | 8 |
| coding | 43 |
| cricket | 87 |
+-----+-----+
```

Q20. Find society names in which more than five students have enrolled in the given year.

Sol :-

## Query

```
select
    s.sname
from
    society s
    inner join enrollment e on s.sid = e.sid
where
    year(e.dateOfEnrollment) = 2023
group by
    s.sname
having
    count(*) > 5;
```

## Output

Empty set

Q21. Add enrolment fees paid ('yes'/'No') field in the enrollment table.

Sol :-

## Query

```
alter table
    enrollment
add
    fees_paid enum('yes', 'no') default 'no';

select
    *
from
    enrollment;
```

## Output

rollno	sid	dateOfEnrollment	fees_paid
AD-1274	06	2023-01-15	no
P-1276	07	2023-01-16	no
AD-1294	02	2023-02-15	no
X-1263	02	2023-01-17	no
Z-1601	06	2023-01-17	no
M-1274	03	2023-01-15	no
A-1284	09	2023-01-15	no
P-1276	04	2022-05-19	no
B-1290	08	2023-04-15	no
X-1263	10	2023-08-16	no
B-1290	01	2023-01-17	no
B-1290	02	2023-01-20	no
B-1290	03	2023-01-24	no
M-1274	01	2023-04-06	no

Q22. Update date of enrolment of society s1 to '2018-01-15', s2 to current date and s3 to '2018-01-02'.

Sol :-

## Query

```
update
  enrollment
set
  dateOfEnrollment = '2018-01-15'
where
  sid = '06';

update
  enrollment
set
  dateOfEnrollment = current_date()
where
  sid = '07';
```

```
update
  enrollment
set
  dateOfEnrollment = '2018-01-02'
where
  sid = '08';
```

```
select
  *
from
  enrollment;
```

## Output

```
+-----+-----+-----+-----+
| rollno | sid | dateOfEnrollment | fees_paid |
+-----+-----+-----+-----+
| AD-1274 | 06 | 2018-01-15 | no |
| P-1276 | 07 | 2023-02-20 | no |
| AD-1294 | 02 | 2023-02-15 | no |
| X-1263 | 02 | 2023-01-17 | no |
| Z-1601 | 06 | 2018-01-15 | no |
| M-1274 | 03 | 2023-01-15 | no |
| A-1284 | 09 | 2023-01-15 | no |
| P-1276 | 04 | 2022-05-19 | no |
| B-1290 | 08 | 2018-01-02 | no |
| X-1263 | 10 | 2023-08-16 | no |
| B-1290 | 01 | 2023-01-17 | no |
| B-1290 | 02 | 2023-01-20 | no |
| B-1290 | 03 | 2023-01-24 | no |
| M-1274 | 01 | 2023-04-06 | no |
+-----+-----+-----+-----+
```

Q23. Find society names whose enrolment is over.

Sol :-

## Query

```
select
  sname
from
  society
where
  capacity <= (
    select
      count(*)
    from
      enrollment
    where
      enrollment.sid = society.sid
  );
```

## Output

Empty set

Q24. Find common societies of students of courses 'cs(hons)' and 'pscs'.

Sol :-

## Query

```
select
  s.sname
from
```

```

enrollment e
  inner join society s on e.sid = s.sid
  inner join student st on e.rollno = st.rollno
where
  st.cname in ('cs', 'pssc')
group by
  s.sname
having
  count(distinct st.rollno) = 2;

```

## Output

```

+-----+
| sname |
+-----+
| coding |
+-----+

```

Q25. Create a view to keep track of society names with total number of students enrolled in it.

Sol :-

## Query

```

create view society_enrollment_count as
select
  s.sid,
  s.sname,
  count(distinct e.rollno) as total_enrollment
from
  society s
  inner join enrollment e on s.sid = e.sid
group by
  s.sid,
  s.sname;

```

```

select
  *
from
  society_enrollment_count;

```

## Output

```

+-----+-----+-----+
| sid | sname | total_enrollment |
+-----+-----+-----+
| 01 | debating | 2 |
| 02 | dancing | 3 |
| 03 | shasakt | 2 |
| 04 | eco | 1 |
| 06 | tech | 2 |
| 07 | music | 1 |
| 08 | hacking | 1 |
| 09 | coding | 2 |
| 10 | cricket | 1 |
+-----+-----+-----+

```



Q26. Find student names enrolled in all societies.

Sol :-

#### Query

```
select
  st.name
from
  student st
where
  st.rollno in (
    select
      e.rollno
    from
      enrollment e
    group by
      e.rollno
    having
      count(distinct e.sid) = (
        select
          count(*)
        from
          society
      )
  );
```

#### Output

Empty set

Q27. Count societies with student enrolled > 3.

Sol :-

#### Query

```
select
  count(distinct e.sid),
  e.sid
from
  enrollment e
group by
  e.sid
having
  count(distinct e.rollno) > 3;
```

#### Output

```
+-----+-----+
| count(distinct e.sid) | sid |
+-----+-----+
```

```

/          1 | 02      /
+-----+-----+

```

Q28. add column contact in student with default value.

Sol :-

Query

```

alter table
  student
add
  contact varchar(20) default 'unknown';

```

```

select
  *
from
  student;

```

Output

```

+-----+-----+-----+-----+-----+
| rollno | name      | cname | dob       | contact |
+-----+-----+-----+-----+-----+
| X-1263 | Nikhil    | pmcs  | 2001-04-02 | unknown |
| Z-1601 | Vipin     | pmcs  | 2001-04-03 | unknown |
| AD-1274 | Priyanshu | cs    | 2003-04-08 | unknown |
| A-1284 | Satendar  | cs    | 2002-04-07 | unknown |
| B-1230 | Deepak    | zoology | 2002-04-03 | unknown |
| P-1276 | Bhoomika  | pscs  | 2002-11-27 | unknown |
| B-1290 | Aman      | pscs  | 2002-04-10 | unknown |
| M-1274 | Meenakshi | physics | 2004-01-09 | unknown |
+-----+-----+-----+-----+-----+

```

Q29. Find the name of oldest and youngest student in class along with their age and DOB.

Sol :-

Query

```

select
  name,
  curdate(),
  timestampdiff(year, dob, curdate()) as age,
  dob
from
  student
where
  dob in (
    select
      max(dob)
    from
      student
    union
    select

```

```

    min(dob)
from
    student
);

```

### Output

```

+-----+-----+-----+-----+
| name      | curdate() | age | dob      |
+-----+-----+-----+-----+
| Nikhil    | 2023-02-21 | 21  | 2001-04-02 |
| Meenakshi | 2023-02-21 | 19  | 2004-01-09 |
+-----+-----+-----+-----+

```

Q30. Find the most popular and least popular society name (on the basis of enrolled students).

Sol :-

### Query

```

select
    s.sname,
    count(*) as number_of_students
from
    enrollment e
    join society s on e.sid = s.sid
group by
    sname
order by
    number_of_students desc
limit
    1;

```

### Output

```

+-----+-----+
| sname      | number_of_students |
+-----+-----+
| dancing    | 4                  |
+-----+-----+

```

### Query

```

select
    s.sname,
    count(*) as number_of_students
from
    enrollment e
    join society s on e.sid = s.sid
group by
    sname
order by
    number_of_students asc
limit
    1;

```

### Output

```
+-----+-----+
| sname | number_of_students |
+-----+-----+
| music |           1 |
+-----+-----+
```

Q31. Find names of students born in year 2001 and enrolled in atleast one society.

Sol :-

### Query

```
select
  name
from
  student
where
  year(dob) = 2001
  and rollno in (
    select
      rollno
    from
      enrollment
  );
```

### Output

```
+-----+
| name |
+-----+
| Nikhil |
| Vipin |
+-----+
```

Q32. Remove default value of any field.

Sol :-

### Query

```
alter table
  student
alter column
  contact drop default;
```

Q33. Find society names where students have enrolled in month Jan/Feb.

Sol :-

### Query

```
select
```

```

distinct s.sname,
month(e.dateOfEnrollment)
from
  enrollment e
  inner join society s
where
  month(e.dateOfEnrollment) = '1'
  or month(e.dateOfEnrollment) = '2';

```

## Output

```

+-----+-----+
| sname | month(e.dateOfEnrollment) |
+-----+-----+
| cricket | 1 |
| coding | 1 |
| hacking | 1 |
| music | 1 |
| tech | 1 |
| picfie | 1 |
| eco | 1 |
| shasakt | 1 |
| dancing | 1 |
| debating | 1 |
| cricket | 2 |
| coding | 2 |
| hacking | 2 |
| music | 2 |
| tech | 2 |
| picfie | 2 |
| eco | 2 |
| shasakt | 2 |
| dancing | 2 |
| debating | 2 |
+-----+-----+

```

Q34. Display all students details alongwith society name if they are enrolled in any society.

Sol :-

## Query

```

select
  st.*,
  e.sid,
  s.sname
from
  enrollment e
  join society s on e.sid = s.sid
  inner join student st on e.rollno = st.rollno
order by
  st.name;

```

## Output

```

+-----+-----+-----+-----+-----+-----+-----+-----+

```

rollno	name	cname	dob	contact	sid	sname
B-1290	Aman	pscs	2002-04-10	unknown	01	debating
B-1290	Aman	pscs	2002-04-10	unknown	02	dancing
B-1290	Aman	pscs	2002-04-10	unknown	03	shasakt
B-1290	Aman	pscs	2002-04-10	unknown	08	hacking
B-1290	Aman	pscs	2002-04-10	unknown	09	coding
P-1276	Bhoomika	pscs	2002-11-27	unknown	04	eco
P-1276	Bhoomika	pscs	2002-11-27	unknown	07	music
M-1274	Meenakshi	physics	2004-01-09	unknown	01	debating
M-1274	Meenakshi	physics	2004-01-09	unknown	02	dancing
M-1274	Meenakshi	physics	2004-01-09	unknown	03	shasakt
X-1263	Nikhil	pmcs	2001-04-02	unknown	02	dancing
X-1263	Nikhil	pmcs	2001-04-02	unknown	10	cricket
AD-1274	Priyanshu	cs	2003-04-08	unknown	06	tech
A-1284	Satendar	cs	2002-04-07	unknown	09	coding
Z-1601	Vipin	pmcs	2001-04-03	unknown	06	tech

Q35. Display society names in uppercase and padded with character \* to get a length of 10 characters which are mentored by mentors whose names start with 'M' and ends with 'r' and capacity is between 10 to 20.

Sol :-

Query

```
select
  rpad(upper(sname), 10, '*') 'Right Padded'
from
  society
where
  mentorname like 's%i'
  and capacity > 20;
```

Output

Empty set