

Ramanujan College

University of Delhi

Name: Prashant Mishra

Rollno: 20221481

Course: B.Sc.(h) Computer Science

Subject: Database Management System

```
PS C:\Users\91895> mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.37 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current in
put statement.
```

```
mysql> Create database StudentSociety;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> use StudentSociety;
Database changed
mysql> create table student (
    -> rollno char(6) primary key,
    -> StudentName varchar(20) NOT NULL,
    -> Course Varchar(10) NOT NULL,
    -> DOB date NOT NULL
    -> );
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> show tables;
+-----+
| Tables_in_studentsociety |
+-----+
| enrollment                |
| society                   |
| student                   |
+-----+
3 rows in set (0.00 sec)
```

```
mysql> CREATE TABLE Enrollment (
    -> Rollno CHAR(6),
    -> SID CHAR(6),
    -> DateOfEnrollment DATE,
    -> PRIMARY KEY (Rollno, SID),
    -> FOREIGN KEY (Rollno) REFERENCES Student(rollno),
    -> FOREIGN KEY (SID) REFERENCES Society(SocId)
    -> );
Query OK, 0 rows affected (0.12 sec)
```

```
mysql> CREATE TABLE Society (
->     SocId CHAR(6) PRIMARY KEY,
->     SocName VARCHAR(20),
->     MentorName VARCHAR(15),
->     TotalSeats INT UNSIGNED
-> );
Query OK, 0 rows affected (0.05 sec)
```

```
mysql> DESCRIBE enrollment;
```

Field	Type	Null	Key	Default	Extra
Rollno	char(6)	NO	PRI	NULL	
SID	char(6)	NO	PRI	NULL	
DateOfEnrollment	date	YES		NULL	

3 rows in set (0.01 sec)

```
mysql> DESCRIBE student;
```

Field	Type	Null	Key	Default	Extra
rollno	char(6)	NO	PRI	NULL	
StudentName	varchar(20)	NO		NULL	
Course	varchar(10)	NO		NULL	
DOB	date	NO		NULL	

4 rows in set (0.00 sec)

```
mysql> DESCRIBE society;
```

Field	Type	Null	Key	Default	Extra
SocId	char(6)	NO	PRI	NULL	
SocName	varchar(20)	YES		NULL	
MentorName	varchar(15)	YES		NULL	
TotalSeats	int unsigned	YES		NULL	

4 rows in set (0.00 sec)

```
mysql> select * from Enrollment;
```

Rollno	SocId	DateOfEnrollment
S1001	Soc001	2024-03-06
S1001	Soc004	2022-02-24
S1002	Soc003	2022-02-04
S1002	Soc005	2023-03-05
S1003	Soc004	2023-03-05
S1006	Soc006	2024-03-22
S1007	Soc007	2023-08-05
S1008	Soc008	2022-12-12
S1009	Soc009	2024-01-25
S1010	Soc010	2023-04-08
S1011	Soc011	2022-07-14
S1012	Soc012	2024-02-01
S1013	Soc013	2023-09-10
S1014	Soc014	2022-10-20
S1015	Soc015	2023-11-30
S1016	Soc001	2023-04-01
S1017	Soc002	2023-04-02
S1018	Soc003	2023-04-03
S1019	Soc004	2023-04-04
S1020	Soc005	2023-04-05
S1021	Soc006	2023-04-06
S1022	Soc007	2023-04-07
S1023	Soc008	2023-04-08
S1024	Soc009	2023-04-09
S1025	Soc010	2023-04-10
S1026	Soc011	2023-04-11
S1027	Soc012	2023-04-12
S1028	Soc013	2023-04-13
S1029	Soc014	2023-04-14
S1030	Soc015	2023-04-15

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

rollno	StudentName	COURSE	DOB
S1001	Rahul Kumar	Computer Science	2004-05-10
S1002	Priya Sharma	Commerce	2005-02-15
S1003	Amit Patel	English	2004-09-20
S1004	Neha Gupta	Economics	2005-04-05
S1005	Ravi Verma	Philosophy	2004-11-30
S1006	Ananya Singh	Psychology	2004-07-12
S1007	Vikram Yadav	Statistics	2005-03-25
S1008	Kavya Reddy	Environmental Science (EVS)	2004-08-18
S1009	Aditya Joshi	Mathematics	2005-01-22
S1010	Sneha Das	Computer Science	2004-10-08
S1011	Rohan Banerjee	Commerce	2004-06-14
S1012	Ishita Choudhary	English	2005-04-28
S1013	Arjun Singhania	Economics	2004-09-03
S1014	Meera Iyer	Philosophy	2005-03-17
S1015	Karthik Nair	Psychology	2004-12-01
S1016	Anjali Rao	Statistics	2004-07-15
S1017	Aryan Desai	Environmental Science (EVS)	2005-02-10
S1018	Pooja Mehta	Mathematics	2004-08-23
S1019	Rajesh Khanna	Computer Science	2005-01-07
S1020	Smita Patel	Commerce	2004-10-22
S1021	Vivek Iyer	English	2004-06-28
S1022	Nisha Kapoor	Economics	2005-04-12
S1023	Rahul Mehta	Philosophy	2004-09-27
S1024	Anushka Chatterjee	Psychology	2005-03-03
S1025	Amita Rao	Statistics	2004-11-18
S1026	Vijay Sharma	Environmental Science (EVS)	2004-07-02
S1027	Sakshi Verma	Mathematics	2005-02-17
S1028	Rajeev Kumar	Computer Science	2004-08-12
S1029	Ananya Tiwari	Commerce	2005-01-29
S1030	Prashant mishra	Computer Science	2004-04-08

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

SocId	SocName	MENTORNAME	TotalSeats
Soc001	Debating Club	Dr. Raj Gupta	50
Soc002	Dancing Society	Ms. Ridhima Yadav	40
Soc003	NSS	Prof. Ramesh Kumar	60
Soc004	Music Club	Mrs. Shelly Menon	30
Soc005	Photography Club	Mr. Arav	35
Soc006	Chess Club	Ms. Kalyani	25
Soc007	Coding Society	Mr. Rahul	70
Soc008	Artistic Expression	Prof. Sohil	45
Soc009	Literary Circle	Dr. Kundra	55
Soc010	Environmental Warriors	Ms. Gaurav	50
Soc011	Science Explorers	Dr. Zehnab Khan	40
Soc012	Film Enthusiasts	Mr. Harsh Mehta	30
Soc013	Health and Fitness	Ms. Jagriti	65
Soc014	Adventure Club	Prof. Suman	35
Soc015	Cooking Society	Mr. Anish	20
Soc016	Sashakt	Adarsh Rana	20

```
16 rows in set (0.00 sec)
```

```
mysql> |
```

1. Retrieve names of students enrolled in any society.

```
Windows PowerShell
mysql> SELECT StudentName
      -> FROM Student
      -> WHERE rollno IN (SELECT Rollno FROM Enrollment);
```

StudentName
Rahul Kumar
Priya Sharma
Amit Patel
Ananya Singh
Vikram Yadav
Kavya Reddy
Aditya Joshi
Sneha Das
Rohan Banerjee
Ishita Choudhary
Arjun Singhania
Meera Iyer
Karthik Nair
Anjali Rao
Aryan Desai
Pooja Mehta
Rajesh Khanna
Smita Patel
Vivek Iyer
Nisha Kapoor
Rahul Mehta
Anushka Chatterjee
Amita Rao
Vijay Sharma
Sakshi Verma
Rajeev Kumar
Ananya Tiwari
Prashant mishra

2. Retrieve all society names.

```
mysql> SELECT SocName  
-> FROM Society;
```

SocName
Debating Club
Dancing Society
NSS
Music Club
Photography Club
Chess Club
Coding Society
Artistic Expression
Literary Circle
Environmental Warriors
Science Explorers
Film Enthusiasts
Health and Fitness
Adventure Club
Cooking Society
Sashakt

```
16 rows in set (0.00 sec)
```

```
mysql> |
```


3. Retrieve students' names starting with the letter 'A'.

```
mysql> SELECT StudentName  
-> FROM Student  
-> WHERE StudentName LIKE 'A%';
```

StudentName
Amit Patel
Ananya Singh
Aditya Joshi
Arjun Singhanian
Anjali Rao
Aryan Desai
Anushka Chatterjee
Amita Rao
Ananya Tiwari

9 rows in set (0.00 sec)

```
mysql> |
```

4. Retrieve students' details studying in courses 'computer science' or 'chemistry'.

```
mysql> SELECT *  
      -> FROM Student  
      -> WHERE course IN ('computer science', 'chemistry');  
+-----+-----+-----+-----+  
| rollno | StudentName | COURSE | DOB |  
+-----+-----+-----+-----+  
| S1001 | Rahul Kumar | Computer Science | 2004-05-10 |  
| S1010 | Sneha Das | Computer Science | 2004-10-08 |  
| S1019 | Rajesh Khanna | Computer Science | 2005-01-07 |  
| S1028 | Rajeev Kumar | Computer Science | 2004-08-12 |  
| S1030 | Prashant mishra | Computer Science | 2004-04-08 |  
+-----+-----+-----+-----+  
5 rows in set (0.00 sec)  
  
mysql> |
```

5. Retrieve students' names whose roll no either starts with 'X' or 'Z' and ends with '9'.

```
mysql> SELECT StudentName  
      -> FROM Student  
      -> WHERE rollno LIKE 'X%9' OR rollno LIKE 'Z%9';  
Empty set (0.00 sec)  
  
mysql> |
```

6. Find society details with more than N TotalSeats where N is to be input by the user

```
mysql> SELECT *
      -> from Society
      -> where totalseats > 20;
```

SocId	SocName	MENTORNAME	TotalSeats
Soc001	Debating Club	Dr. Raj Gupta	50
Soc002	Dancing Society	Ms. Ridhima Yadav	40
Soc003	NSS	Prof. Ramesh Kumar	60
Soc004	Music Club	Mrs. Shelly Menon	30
Soc005	Photography Club	Mr. Arav	35
Soc006	Chess Club	Ms. Kalyani	25
Soc007	Coding Society	Mr. Rahul	70
Soc008	Artistic Expression	Prof. Sohil	45
Soc009	Literary Circle	Dr. Kundra	55
Soc010	Environmental Warriors	Ms. Gaurav	50
Soc011	Science Explorers	Dr. Zehnab Khan	40
Soc012	Film Enthusiasts	Mr. Harsh Mehta	30
Soc013	Health and Fitness	Ms. Jagriti	65
Soc014	Adventure Club	Prof. Suman	35

```
14 rows in set (0.00 sec)

mysql> |
```

7. Update society table for mentor name of a specific society.

```
mysql> UPDATE SOCIETY
      -> SET MENTORNAME = 'MR.RAJESH'
      -> WHERE SocId = 'Soc007';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

SocId	SocName	MENTORNAME	TotalSeats
Soc001	Debating Club	Dr. Raj Gupta	50
Soc002	Dancing Society	Ms. Ridhima Yadav	40
Soc003	NSS	Prof. Ramesh Kumar	60
Soc004	Music Club	Mrs. Shelly Menon	30
Soc005	Photography Club	Mr. Arav	35
Soc006	Chess Club	Ms. Kalyani	25
Soc007	Coding Society	MR.RAJESH	70
Soc008	Artistic Expression	Prof. Sohil	45
Soc009	Literary Circle	Dr. Kundra	55
Soc010	Environmental Warriors	Ms. Gaurav	50
Soc011	Science Explorers	Dr. Zehnab Khan	40
Soc012	Film Enthusiasts	Mr. Harsh Mehta	30
Soc013	Health and Fitness	Ms. Jagriti	65
Soc014	Adventure Club	Prof. Suman	35
Soc015	Cooking Society	Mr. Anish	20
Soc016	Sashakt	Adarsh Rana	20

16 rows in set (0.00 sec)

8. Find society names in which more than half students have enrolled.

```
mysql> SELECT SocName
-> FROM Society
-> WHERE SocId IN (SELECT SocId FROM Enrollment GROUP BY SocId HAVING COUNT(*) > 5);
Empty set (0.01 sec)

mysql> |
```

9. Find the name of youngest student enrolled in society 'NSS'.

```
mysql> Select StudentName
-> From Student S
-> join Enrollment E on S.rollno = E.rollno
-> where E.SocId = 'Soc003'
-> order by S.DOB DESC
-> limit 1;
+-----+
| StudentName |
+-----+
| Priya Sharma |
+-----+
1 row in set (0.00 sec)

mysql> |
```

10. Find the name of most popular society(on the basis of enrolled student).

```
mysql> SELECT SocName
-> FROM Society S
-> LEFT JOIN Enrollment E ON E.SocId = S.SocId
-> GROUP BY S.SocName
-> ORDER BY COUNT(E.Rollno) ASC
-> LIMIT 1;
+-----+
| SocName |
+-----+
| Sashakt |
+-----+
1 row in set (0.01 sec)

mysql> |
```

11. Find the name of least popular societies.

```
mysql> SELECT SocName
-> FROM Society S
-> LEFT JOIN Enrollment E ON E.SocId = S.SocId
-> GROUP BY S.SocName
-> ORDER BY COUNT(E.Rollno) ASC
-> LIMIT 1;
+-----+
| SocName |
+-----+
| Sashakt |
+-----+
1 row in set (0.01 sec)

mysql> |
```

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

```
mysql> SELECT StudentName
-> FROM Student S
-> LEFT JOIN Enrollment E ON S.rollno = E.Rollno
-> WHERE E.Rollno IS NULL;
+-----+
| StudentName |
+-----+
| Neha Gupta  |
| Ravi Verma  |
+-----+
2 rows in set (0.00 sec)

mysql> |
```

13. Find the student names enrolled in at least two Societies.

```
mysql> SELECT StudentName
-> FROM Student S
-> JOIN Enrollment E1 ON S.rollno = E1.Rollno
-> JOIN Enrollment E2 ON S.rollno = E2.Rollno
-> WHERE E1.SocId <> E2.SocId
-> GROUP BY S.StudentName
-> HAVING COUNT(DISTINCT E1.SocId) >= 2;
+-----+
| StudentName |
+-----+
| Priya Sharma |
| Rahul Kumar  |
+-----+
2 rows in set (0.01 sec)

mysql> |
```

14. Find society names in which maximum students are enrolled.

```
mysql> SELECT SocName, COUNT(E.Rollno) AS TotalEnrolled  
-> FROM Society S  
-> JOIN Enrollment E ON E.SocId = S.SocId  
-> GROUP BY S.SocName  
-> ORDER BY TotalEnrolled DESC  
-> LIMIT 1;
```

```
+-----+-----+  
| SocName | TotalEnrolled |  
+-----+-----+  
| Music Club |          3 |  
+-----+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> |
```

15. Find names of all students who have enrolled in any society and society named in which at least one student has enrolled.

```
Windows PowerShell
mysql> SELECT StudentName, SocName
      -> FROM Student
      -> JOIN Enrollment ON Student.rollno = Enrollment.rollno
      -> JOIN Society ON Enrollment.SocId = Society.SocId;
```

StudentName	SocName
Rahul Kumar	Debating Club
Anjali Rao	Debating Club
Aryan Desai	Dancing Society
Priya Sharma	NSS
Pooja Mehta	NSS
Rahul Kumar	Music Club
Amit Patel	Music Club
Rajesh Khanna	Music Club
Priya Sharma	Photography Club
Smita Patel	Photography Club
Ananya Singh	Chess Club
Vivek Iyer	Chess Club
Vikram Yadav	Coding Society
Nisha Kapoor	Coding Society
Kavya Reddy	Artistic Expression
Rahul Mehta	Artistic Expression
Aditya Joshi	Literary Circle
Anushka Chatterjee	Literary Circle
Sneha Das	Environmental Warriors
Amita Rao	Environmental Warriors
Rohan Banerjee	Science Explorers
Vijay Sharma	Science Explorers
Ishita Choudhary	Film Enthusiasts
Sakshi Verma	Film Enthusiasts
Arjun Singhania	Health and Fitness
Rajeev Kumar	Health and Fitness
Meera Iyer	Adventure Club
Ananya Tiwari	Adventure Club

16. Find names of students who are enrolled in any of the three society Debating, dancing, and Sashakt.


```
mysql> SELECT StudentName
-> FROM Student
-> WHERE rollno IN (SELECT Rollno FROM Enrollment WHERE SocId IN ('Soc001', 'Soc002', 'Soc016'));
+-----+
| StudentName |
+-----+
| Rahul Kumar |
| Anjali Rao  |
| Aryan Desai |
+-----+
3 rows in set (0.00 sec)

mysql> |
```

17. Find society names such that its mentor has a name with Gupta in it.

```
mysql> SELECT SocName
-> FROM Society
-> WHERE MentorName LIKE '%Gupta%';
+-----+
| SocName |
+-----+
| Debating Club |
+-----+
1 row in set (0.00 sec)

mysql> |
```

18. Find the society names in which the number of enrolled students is only 10% of its capacity.

```
mysql> SELECT SocName
-> FROM Society
-> WHERE (SELECT COUNT(*) FROM Enrollment WHERE SocId = Society.SocId) <= 0.1 * TotalSeats;
```

SocName
Debating Club
Dancing Society
NSS
Music Club
Photography Club
Chess Club
Coding Society
Artistic Expression
Literary Circle
Environmental Warriors
Science Explorers
Film Enthusiasts
Health and Fitness
Adventure Club
Cooking Society
Sashakt

```
16 rows in set (0.01 sec)

mysql> |
```

19. Display the vacant seats for each society.

```
mysql> SELECT SocName, TotalSeats - (SELECT COUNT(*) FROM Enrollment WHERE SocId = Society.SocId) AS VacantSeats
-> from Society;
```

SocName	VacantSeats
Debating Club	48
Dancing Society	39
NSS	58
Music Club	27
Photography Club	33
Chess Club	23
Coding Society	68
Artistic Expression	43
Literary Circle	53
Environmental Warriors	48
Science Explorers	38
Film Enthusiasts	28
Health and Fitness	63
Adventure Club	33
Cooking Society	18
Sashakt	20

```
16 rows in set (0.00 sec)

mysql> |
```

20. Increment totals leads of each society by 10%.

```
mysql> UPDATE Society
      -> SET TotalSeats = TotalSeats * 1.1;
Query OK, 16 rows affected (0.01 sec)
Rows matched: 16  Changed: 16  Warnings: 0

mysql> select * from Society;
```

SocId	SocName	MENTORNAME	TotalSeats
Soc001	Debating Club	Dr. Raj Gupta	55
Soc002	Dancing Society	Ms. Ridhima Yadav	44
Soc003	NSS	Prof. Ramesh Kumar	66
Soc004	Music Club	Mrs. Shelly Menon	33
Soc005	Photography Club	Mr. Arav	39
Soc006	Chess Club	Ms. Kalyani	28
Soc007	Coding Society	MR.RAJESH	77
Soc008	Artistic Expression	Prof. Sohil	50
Soc009	Literary Circle	Dr. Kundra	61
Soc010	Environmental Warriors	Ms. Gaurav	55
Soc011	Science Explorers	Dr. Zehnab Khan	44
Soc012	Film Enthusiasts	Mr. Harsh Mehta	33
Soc013	Health and Fitness	Ms. Jagriti	72
Soc014	Adventure Club	Prof. Suman	39
Soc015	Cooking Society	Mr. Anish	22
Soc016	Sashakt	Adarsh Rana	22

```
16 rows in set (0.00 sec)

mysql> |
```

21. Add the enrolment fee paid('yes', 'no') field in the enrolment table.

```
mysql> desc enrollment;
```

Field	Type	Null	Key	Default	Extra
Rollno	char(6)	NO	PRI	NULL	
SocId	char(6)	NO	PRI	NULL	
DateOfEnrollment	date	YES		NULL	
EnrollmentFEE	enum('yes','no')	NO		no	

```
4 rows in set (0.01 sec)
mysql>
```

```
mysql> ALTER TABLE Enrollment
-> ADD EnrollmentFEE ENUM('yes', 'no') NOT NULL DEFAULT 'no'
-> ;
```

```
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc enrollment;
```

Field	Type	Null	Key	Default	Extra
Rollno	char(6)	NO	PRI	NULL	
SocId	char(6)	NO	PRI	NULL	
DateOfEnrollment	date	YES		NULL	
EnrollmentFEE	enum('yes','no')	NO		no	

```
4 rows in set (0.01 sec)
mysql>
```

22. Show me the mum update date of enrolment of society id 'S1001' to '2018-01-15', 'S1002' to current date and 'S1003' to '2018-01-02'.

```
mysql> UPDATE Enrollment
      -> SET DateOfEnrollment = '2018-01-15'
      -> where SocId = 'Soc001';
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2  Changed: 2  Warnings: 0
```

```
mysql> UPDATE Enrollment
      -> SET DateOfEnrollment = CURDATE()
      -> where SocId = 'Soc002';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> UPDATE Enrollment
      -> SET DateOfEnrollment = '2018-01-02'
      -> where SocId = 'Soc003';
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2  Changed: 2  Warnings: 0
```

```
mysql> Select * from Enrollment;
```

Rollno	SocId	DateOfEnrollment	EnrollmentFEE
S1001	Soc001	2018-01-15	no
S1001	Soc004	2022-02-24	no
S1002	Soc003	2018-01-02	no
S1002	Soc005	2023-03-05	no
S1003	Soc004	2023-03-05	no
S1006	Soc006	2024-03-22	no
S1007	Soc007	2023-08-05	no
S1008	Soc008	2022-12-12	no
S1009	Soc009	2024-01-25	no
S1010	Soc010	2023-04-08	no

```
mysql> Select * from Enrollment;
```

Rollno	SocId	DateOfEnrollment	EnrollmentFEE
S1001	Soc001	2018-01-15	no
S1001	Soc004	2022-02-24	no
S1002	Soc003	2018-01-02	no
S1002	Soc005	2023-03-05	no
S1003	Soc004	2023-03-05	no
S1006	Soc006	2024-03-22	no
S1007	Soc007	2023-08-05	no
S1008	Soc008	2022-12-12	no
S1009	Soc009	2024-01-25	no
S1010	Soc010	2023-04-08	no
S1011	Soc011	2022-07-14	no
S1012	Soc012	2024-02-01	no
S1013	Soc013	2023-09-10	no
S1014	Soc014	2022-10-20	no
S1015	Soc015	2023-11-30	no
S1016	Soc001	2018-01-15	no
S1017	Soc002	2024-05-05	no
S1018	Soc003	2018-01-02	no
S1019	Soc004	2023-04-04	no
S1020	Soc005	2023-04-05	no
S1021	Soc006	2023-04-06	no
S1022	Soc007	2023-04-07	no
S1023	Soc008	2023-04-08	no
S1024	Soc009	2023-04-09	no
S1025	Soc010	2023-04-10	no
S1026	Soc011	2023-04-11	no
S1027	Soc012	2023-04-12	no
S1028	Soc013	2023-04-13	no
S1029	Soc014	2023-04-14	no
S1030	Soc015	2023-04-15	no

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

```
Windows PowerShell

mysql> CREATE VIEW SOCIETYENROLLMENTCOUNT AS
-> SELECT Society.SocName, count(Enrollment.rollno) AS Total
StusentsEnrolled
-> FROM Society
-> LEFT JOIN Enrollment on Enrollment.SocId = Society.SocId
-> GROUP BY Society.SocName;
Query OK, 0 rows affected (0.02 sec)

mysql> desc view
-> ;
ERROR 1146 (42S02): Table 'studentsociety.view' doesn't exist
mysql> DESC SOCIETYENROLLMENTCOUNT;
+-----+-----+-----+-----+-----+-----+
| Field                | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| SocName              | char(30)  | YES  |     | NULL    |       |
| TotalStusentsEnrolled | bigint    | NO   |     | 0       |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> Select * from Societyenrollmentcount;
+-----+-----+
| SocName                | TotalStusentsEnrolled |
+-----+-----+
| Debating Club          | 2                     |
| Dancing Society        | 1                     |
| NSS                    | 2                     |
| Music Club             | 3                     |
| Photography Club       | 2                     |
| Chess Club             | 2                     |
| Coding Society         | 2                     |
| Artistic Expression    | 2                     |
| Literary Circle        | 2                     |
| Environmental Warriors | 2                     |
+-----+-----+
```

24. Find student names enrolled in all the societies.

```
mysql> SELECT StudentName
-> FROM Student S
-> WHERE NOT EXISTS (
->   SELECT 1
->   FROM Society Soc
->   WHERE NOT EXISTS (
->     SELECT 1
->     FROM Enrollment E
->     WHERE E.Rollno = S.rollno AND E.SocId = Soc.SocId)
-> );
Empty set (0.00 sec)

mysql> |
```

25. Count the number of societies with more than 5 students enrolled in it.

```
mysql> SELECT COUNT(*) AS SocietiesWithMoreThan5Students
-> FROM Society S
-> JOIN Enrollment E ON E.SocId = S.SocId
-> GROUP BY S.SocName
-> HAVING COUNT(E.Rollno) > 2;
+-----+
| SocietiesWithMoreThan5Students |
+-----+
|                               3 |
+-----+
1 row in set (0.00 sec)

mysql> |
```

26. Add column Mobile number in student table with default value '9999999999'.

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


```

mysql> ALTER TABLE Student
      -> ADD Mobile_number CHAR(10) NOT NULL DEFAULT '9999999999';
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> SELECT COUNT(*) AS TotalStudentsAbove20
      -> FROM Student
      -> WHERE YEAR(CURDATE()) - YEAR(DOB) > 20;
+-----+
| TotalStudentsAbove20 |
+-----+
|                      0 |
+-----+
1 row in set (0.00 sec)

mysql> |

```

28. Find names of students who are born in 2001 and are enrolled in at least one society.

```
mysql> select StudentName
-> from student
-> join enrollment on student.rollno = enrollment.rollno
-> where Year(student.dob) = 2004
-> Group By student.StudentName
-> Having Count(enrollment.rollno) > 0;
```

StudentName
Rahul Kumar
Anjali Rao
Pooja Mehta
Amit Patel
Smita Patel
Ananya Singh
Vivek Iyer
Kavya Reddy
Rahul Mehta
Sneha Das
Amita Rao
Rohan Banerjee
Vijay Sharma
Arjun Singhanian
Rajeev Kumar
Karthik Nair
Prashant mishra

17 rows in set (0.00 sec)

```
mysql> |
```

29. Count all societies whose name starts with 'S' and ends with 't' and at least 5 students are enrolled in the society.

```
mysql> SELECT COUNT(*) AS MatchingSocieties
-> FROM Society S
-> JOIN Enrollment E ON E.SocId = S.SocId
-> WHERE S.SocName LIKE 'S%' -- Starts with 'S'
-> AND S.SocName LIKE '%t' -- Ends with 't'
-> GROUP BY S.SocName
-> HAVING COUNT(E.Rollno) > 5;
```

Empty set (0.00 sec)

```
mysql> |
```

30. 30. Display the following information: Society
name Mentor name Total Capacity Total Enrolled
Unfilled Seats

II. Do the following database administration
commands: create user, create role, grant privileges to
a role, revoke privileges from a role, create index

III. Execute queries given in part I through a high-level
language using ODBC connection.

IV. Students should implement the COMPANY database
schema from Chapter 3 [1] and execute the solved
queries of Chapter 7 [1].