**RAMANUJAN COLLEGE**

(University of Delhi)

**DATABASE MANAGEMENT SYSTEM**

**(MySQL PRACTICALS)**

|  |  |  |
| --- | --- | --- |
| **Name :** | Devansh Tyagi |  |
| **Roll No. :** | 20221416 |  |
| **Course :** | B.Sc. (Hons.) Computer Science |  |
| **Year :** | Second |  |

**CREATING THE DATABASE**

CREATE DATABASE College;

USE College;

CREATE TABLE Student (

Roll\_No CHAR(6) PRIMARY KEY,

StudentName VARCHAR(20),

Course VARCHAR(10),

DOB DATE

);

CREATE TABLE Society (

SocID CHAR(6) PRIMARY KEY,

SocName VARCHAR(20),

MentorName VARCHAR(15),

TotalSeats INT UNSIGNED

);

CREATE TABLE Enrollment (

Roll\_No CHAR(6),

SID CHAR(6),

DateOfEnrollment DATE,

FOREIGN KEY(Roll\_No) REFERENCES Student(Roll\_No),

FOREIGN KEY(SID) REFERENCES Society(SocID)

ON UPDATE CASCADE

ON DELETE CASCADE

);

INSERT INTO Student VALUES

("BY2M1G", "Aarav Patel", "Chemistry", "2004-08-15"),

("E41246", "Aisha Shah", "Computer", "2011-02-18"),

("H7I8H9", "Advik Singh", "Physics", "2008-06-21"),

("K1L2H3", "Ananya Mishra", "Chemistry", "2010-10-04"),

("G4O5P6", "Ishaan Kumar", "History", "2009-04-17"),

("Z7R8S9", "Kavya Gupta", "Computer", "2013-01-20"),

("T1F2V3", "Diya Sharma", "Biology", "2007-07-10"),

("W4XFY6", "Vihaan Joshi", "Maths", "2006-11-23"),

("Z7A869", "Riya Patel", "English", "2012-05-06"),

("X1D2S9", "Shaurya Reddy", "Chemistry", "2005-09-28"),

("F4G5F6", "Myra Kumar", "Computer", "2011-03-12"),

("I7J8KB", "Rudra Malhotra", "Physics", "2008-07-25"),

("L1M2NW", "Zoya Khan", "History", "2009-01-18"),

("Z4P5Q9", "Aryan Sharma", "Biology", "2010-09-11"),

("R7SST9", "Anika Singh", "Maths", "2007-03-24"),

("U1VDW3", "Ishan Patel", "English", "2013-08-07"),

("X4YNZ6", "Ananya Gupta", "Chemistry", "2005-12-15"),

("A7B3C9", "Arjun Singh", "Computer", "2011-06-28"),

("D1R2F3", "Anaya Verma", "Physics", "2008-10-31"),

("G4Y5I6", "Saanvi Kapoor", "History", "2009-04-04"),

("Z7I8L9", "Kabir Sharma", "Biology", "2010-11-27"),

("M1H2O3", "Pari Mishra", "Maths", "2002-06-10"),

("P455R6", "Ahaan Patel", "English", "2013-11-23"),

("B224D6", "Aditi Singhania", "Chemistry", "2004-08-25"),

("E345G7", "Aryan Kapoor", "Computer", "2011-02-28"),

("H4IRJ8", "Diya Mehra", "Physics", "2008-06-12"),

("K5LRM9", "Ishaan Gupta", "Chemistry", "2010-10-24"),

("N6O4P1", "Kavya Reddy", "History", "2009-04-07"),

("Q7R9S2", "Rohan Sharma", "Computer", "2013-01-10"),

("T8U1F3", "Ananya Rajput", "Biology", "2007-07-15"),

("W9XJY4", "Arnav Khanna", "Maths", "2006-11-28"),

("Z1AGB9", "Ishita Mehta", "English", "2012-05-11"),

("C6DEE1", "Riya Verma", "Chemistry", "2000-09-30"),

("F7F9H2", "Samarth Jain", "Computer", "2011-03-22"),

("I8F1K3", "Tanvi Agrawal", "Physics", "2001-07-05"),

("L9MFN4", "Vivaan Gupta", "History", "2009-01-28"),

("O1P3S5", "Zara Khan", "Biology", "2010-09-21"),

("R4S6S8", "Aahana Joshi", "Maths", "2002-04-04"),

("U7V9F1", "Aarav Khatri", "English", "2013-08-17"),

("X2Y4Z9", "Isha Sharma", "Chemistry", "2005-12-22"),

("A3E5C7", "Ritvik Malhotra", "Computer", "2011-06-05"),

("D6T8F1", "Shreya Patel", "Physics", "2008-10-18"),

("G7T9I2", "Vihaan Malhotra", "History", "2009-04-01"),

("X8W1L9", "Zoya Gupta", "Biology", "2010-11-14"),

("M9NGO4", "Aaradhya Patel", "Maths", "2007-06-27"),

("P3Q5E7", "Ayaan Khurana", "English", "2013-11-30"),

("B4C7E2", "Ishani Gupta", "Chemistry", "2004-08-18"),

("E8F3S9", "Aarav Singh", "Computer", "2000-02-21"),

("H2I5JH", "Aanya Sharma", "Physics", "2008-06-14"),

("K7L3MD", "Arjun Patel", "Chemistry", "2010-10-07"),

("X1O6P9", "Kiara Malhotra", "History", "2009-04-20"),

("Q5S9S3", "Rohan Kapoor", "Computer", "2013-01-13"),

("T2A7V4", "Aditi Rana", "Biology", "2007-07-20"),

("W5XGY8", "Arnav Singhania", "Maths", "2006-12-03"),

("Z9A4G6", "Ishika Gupta", "English", "2012-05-16"),

("C8D3EG", "Ria Sharma", "Chemistry", "2005-09-25"),

("F2G722", "Samaira Malhotra", "Computer", "2001-03-18"),

("I5J12W", "Tanisha Verma", "Physics", "2008-07-01"),

("Z3M9N9", "Vihaan Mehra", "History", "2002-01-24"),

("O6QFQ7", "Zara Gupta", "Biology", "2010-09-17"),

("R1SAT3", "Aaradhya Kapoor", "Maths", "2007-04-10"),

("U8V4W9", "Aarav Sharma", "English", "2013-08-23"),

("X3YAZ1", "Ishita Singh", "Chemistry", "2005-12-28"),

("A7B2CN", "Ritvik Choudhary", "Computer", "2011-06-11"),

("D3E9FS", "Shreya Verma", "Physics", "2008-10-04"),

("G7H9ID", "Vivaan Verma", "History", "2009-05-01"),

("J4K9LE", "Zara Mehta", "Biology", "2010-11-20"),

("X1N6O9", "Aaradhya Gupta", "Maths", "2007-06-13"),

("P5Q1R7", "Ayaan Malhotra", "English", "2013-12-03"),

("B3Q8D2", "Ishaan Gupta", "Chemistry", "2004-08-20"),

("E9A4G7", "Aadhya Patel", "Computer", "2011-02-25"),

("H2M7J3", "Aaradhya Sharma", "Physics", "2008-06-18"),

("K833M9", "Aryan Kumar", "Chemistry", "2003-10-11"),

("N2H7P4", "Kriti Reddy", "History", "2009-04-24"),

("Q6M1S8", "Rohan Kumar", "Computer", "2013-01-17"),

("T3S8V5", "Advaita Gupta", "Biology", "2007-07-25"),

("W6C1Y9", "Arpit Singh", "Maths", "2006-12-08"),

("Z1B6B2", "Ishika Malhotra", "English", "2012-05-21"),

("C8R3E9", "Ria Verma", "Chemistry", "2005-09-28"),

("F2I7H4", "Samaira Choudhary", "Computer", "2011-03-21"),

("I5J1K8", "Tanisha Singh", "Physics", "2008-07-04"),

("O6Q2Q7", "Zara Gupta", "Biology", "2010-09-20"),

("R1A1T3", "Aaradhya Malhotra", "Maths", "2007-04-13"),

("U8H4W9", "Aarav Verma", "English", "2013-08-26"),

("X3Z8Z1", "Ishita Sharma", "Chemistry", "2005-12-31"),

("A7V2C6", "Ritvik Singh", "Computer", "2011-06-14"),

("D3S9F4", "Shreya Malhotra", "Physics", "2008-10-07"),

("G7H2I6", "Vivaan Verma", "History", "2003-05-01"),

("J4Q9L3", "Zara Gupta", "Biology", "2010-11-24"),

("X136O9", "Aaradhya Malhotra", "Maths", "2007-06-17"),

("P5R1R7", "Ayaan Singh", "English", "2000-12-06");

INSERT INTO Society VALUES

("S1", "Prakritik Sangathan", "Rohan Gupta", 31),

("S2", "Dancing", "Aditya Sharma", 26),

("S3", "Cinema Samaj", "Aaradhya Patel", 35),

("S4", "Coding Samiti", "Gupta Singh", 28),

("S5", "NSS", "Sneha Gupta", 17),

("S6", "Rasoi Mandal", "Manvi Singh", 29),

("S7", "Vigyan Kalp", "Aryan Sharma", 36),

("S8", "Debating", "Ananya Gupta", 28),

("S9", "Sashakt", "Ishaan Joshi", 27),

("S10", "Robotics Parishad", "Aarav Singh", 19);

INSERT INTO Enrollment (Roll\_No, SID, DateOfEnrollment) VALUES

("BY2M1G", "S1", "2024-05-03"),

("BY2M1G", "S2", "2024-05-03"),

("E41246", "S2", "2024-05-03"),

("E41246", "S3", "2024-05-03"),

("H7I8H9", "S4", "2024-05-03"),

("H7I8H9", "S5", "2024-05-03"),

("K1L2H3", "S6", "2024-05-03"),

("G4O5P6", "S8", "2024-05-03"),

("G4O5P6", "S9", "2024-05-03"),

("Z7R8S9", "S10", "2024-05-03"),

("Z7R8S9", "S1", "2024-05-03"),

("T1F2V3", "S2", "2024-05-03"),

("T1F2V3", "S3", "2024-05-03"),

("W4XFY6", "S4", "2024-05-03"),

("W4XFY6", "S5", "2024-05-03"),

("Z7A869", "S6", "2024-05-03"),

("Z7A869", "S7", "2024-05-03"),

("X1D2S9", "S8", "2024-05-03"),

("X1D2S9", "S9", "2024-05-03"),

("F4G5F6", "S10", "2024-05-03"),

("F4G5F6", "S1", "2024-05-03"),

("I7J8KB", "S2", "2024-05-03"),

("I7J8KB", "S3", "2024-05-03"),

("L1M2NW", "S4", "2024-05-03"),

("L1M2NW", "S5", "2024-05-03"),

("U7V9F1", "S7", "2024-05-03"),

("X2Y4Z9", "S8", "2024-05-03"),

("X2Y4Z9", "S9", "2024-05-03"),

("A3E5C7", "S10", "2024-05-03"),

("A3E5C7", "S1", "2024-05-03"),

("D6T8F1", "S2", "2024-05-03"),

("D6T8F1", "S3", "2024-05-03"),

("G7T9I2", "S4", "2024-05-03"),

("G7T9I2", "S5", "2024-05-03"),

("X8W1L9", "S6", "2024-05-03"),

("B4C7E2", "S2", "2024-05-03"),

("U8H4W9", "S3", "2024-05-03"),

("U8H4W9", "S4", "2024-05-03"),

("U8H4W9", "S5", "2024-05-03"),

("U8H4W9", "S7", "2024-05-03"),

("B4C7E2", "S3", "2024-05-03"),

("E8F3S9", "S4", "2024-05-03"),

("E8F3S9", "S5", "2024-05-03"),

("H2I5JH", "S6", "2024-05-03"),

("H2I5JH", "S7", "2024-05-03"),

("K7L3MD", "S8", "2024-05-03"),

("K7L3MD", "S9", "2024-05-03"),

("X1O6P9", "S10", "2024-05-03"),

("X1O6P9", "S1", "2024-05-03"),

("Q5S9S3", "S2", "2024-05-03"),

("Q5S9S3", "S3", "2024-05-03"),

("T2A7V4", "S4", "2024-05-03"),

("T2A7V4", "S5", "2024-05-03"),

("W5XGY8", "S6", "2024-05-03"),

("W5XGY8", "S7", "2024-05-03"),

("Z9A4G6", "S8", "2024-05-03"),

("Z9A4G6", "S9", "2024-05-03"),

("C8D3EG", "S10", "2024-05-03"),

("F2G722", "S3", "2024-05-03"),

("I5J12W", "S4", "2024-05-03"),

("I5J12W", "S5", "2024-05-03"),

("Z3M9N9", "S6", "2024-05-03"),

("Z3M9N9", "S7", "2024-05-03"),

("O6QFQ7", "S8", "2024-05-03"),

("X3YAZ1", "S5", "2024-05-03"),

("U8H4W9", "S1", "2024-05-03"),

("U8H4W9", "S2", "2024-05-03"),

("O6QFQ7", "S9", "2024-05-03"),

("X3YAZ1", "S4", "2024-05-03"),

("U8H4W9", "S8", "2024-05-03"),

("U8H4W9", "S9", "2024-05-03"),

("A7B2CN", "S6", "2024-05-03"),

("A7B2CN", "S7", "2024-05-03"),

("D3E9FS", "S8", "2024-05-03"),

("D3E9FS", "S9", "2024-05-03"),

("G7H9ID", "S10", "2024-05-03"),

("G7H9ID", "S1", "2024-05-03"),

("J4K9LE", "S2", "2024-05-03"),

("J4K9LE", "S3", "2024-05-03"),

("X1N6O9", "S4", "2024-05-03"),

("X1N6O9", "S5", "2024-05-03"),

("B3Q8D2", "S8", "2024-05-03"),

("B3Q8D2", "S9", "2024-05-03"),

("E9A4G7", "S10", "2024-05-03"),

("E9A4G7", "S1", "2024-05-03"),

("H2M7J3", "S2", "2024-05-03"),

("H2M7J3", "S3", "2024-05-03"),

("K833M9", "S4", "2024-05-03"),

("K833M9", "S5", "2024-05-03"),

("N2H7P4", "S6", "2024-05-03"),

("N2H7P4", "S7", "2024-05-03"),

("Q6M1S8", "S8", "2024-05-03"),

("Q6M1S8", "S9", "2024-05-03"),

("T3S8V5", "S10", "2024-05-03"),

("T3S8V5", "S1", "2024-05-03"),

("W6C1Y9", "S2", "2024-05-03"),

("W6C1Y9", "S3", "2024-05-03"),

("C8R3E9", "S6", "2024-05-03"),

("C8R3E9", "S7", "2024-05-03"),

("F2I7H4", "S8", "2024-05-03"),

("F2I7H4", "S9", "2024-05-03"),

("I5J1K8", "S10", "2024-05-03"),

("I5J1K8", "S1", "2024-05-03"),

("O6Q2Q7", "S2", "2024-05-03"),

("O6Q2Q7", "S3", "2024-05-03"),

("R1A1T3", "S4", "2024-05-03"),

("R1A1T3", "S5", "2024-05-03"),

("U8H4W9", "S6", "2024-05-03"),

("U8H4W9", "S7", "2024-05-03"),

("X3Z8Z1", "S8", "2024-05-03"),

("X3Z8Z1", "S9", "2024-05-03"),

("A7V2C6", "S10", "2024-05-03"),

("A7V2C6", "S1", "2024-05-03"),

("D3S9F4", "S2", "2024-05-03"),

("D3S9F4", "S3", "2024-05-03"),

("G7H2I6", "S4", "2024-05-03"),

("G7H2I6", "S5", "2024-05-03"),

("J4Q9L3", "S6", "2024-05-03"),

("J4Q9L3", "S7", "2024-05-03"),

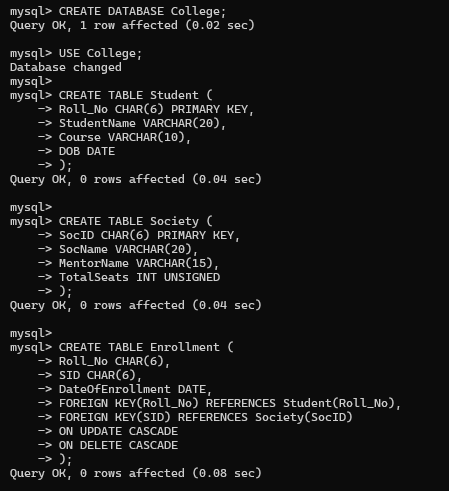
("X136O9", "S8", "2024-05-03"),

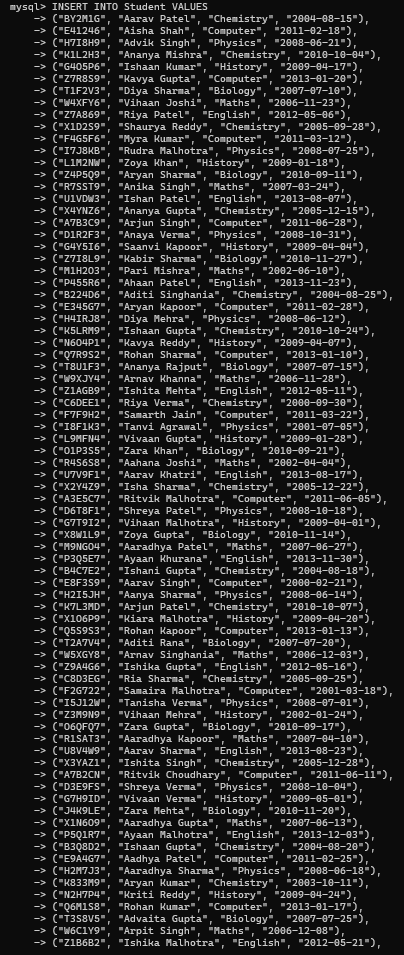
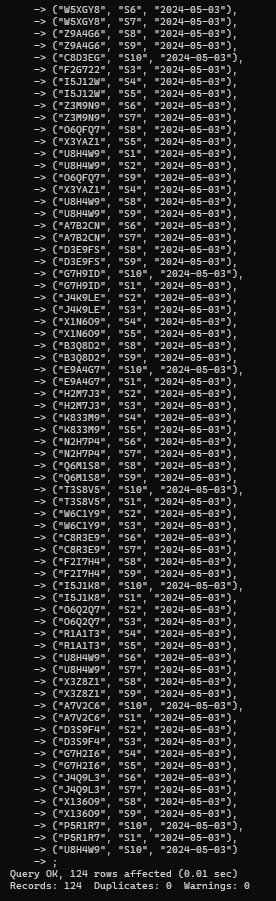
("X136O9", "S9", "2024-05-03"),

("P5R1R7", "S10", "2024-05-03"),

("P5R1R7", "S1", "2024-05-03"),

("U8H4W9", "S10", "2024-05-03");



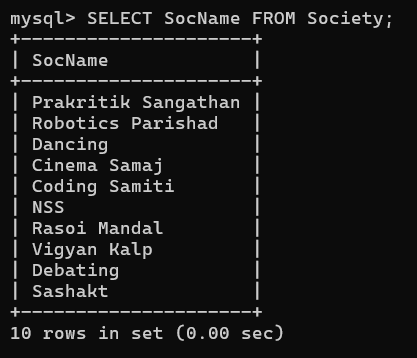
**1. Retrieve names of students enrolled in any society.**

SELECT DISTINCT StudentName FROM Student AS Stu INNER JOIN Enrollment AS Enr ON Stu.Roll\_No = Enr.Roll\_No;



**2. Retrieve all society names.**

SELECT SocName FROM Society;



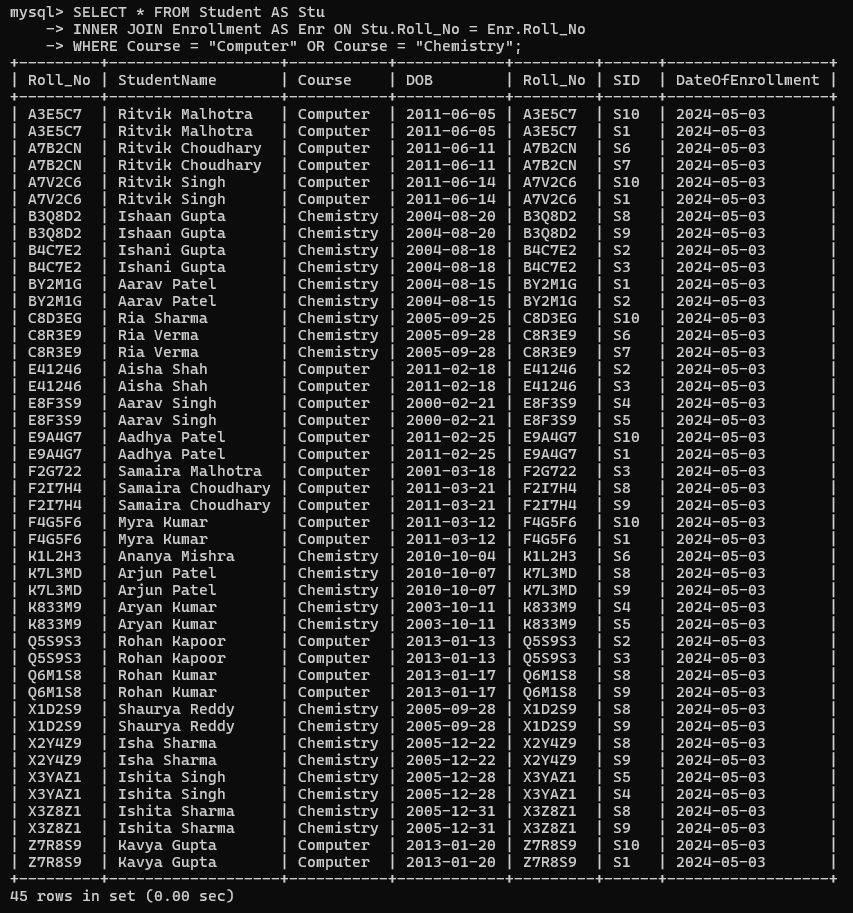
**3. Retrieve students' names starting with letter ‘A’.**

SELECT StudentName FROM Student WHERE StudentName LIKE "A%";



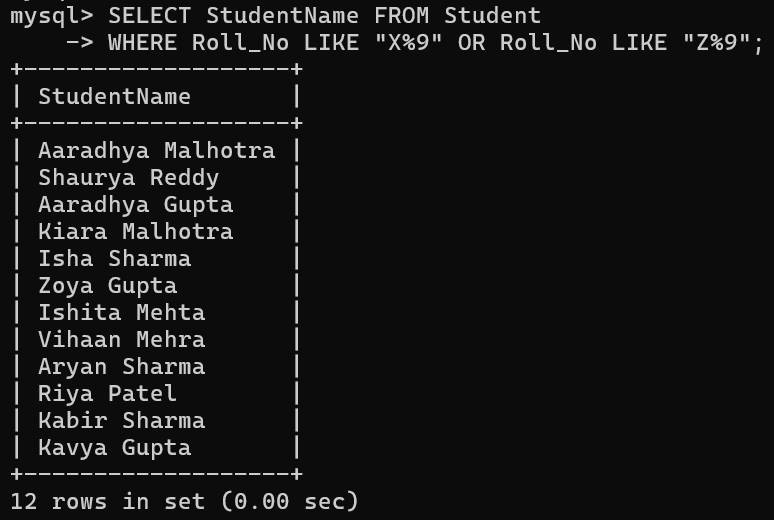
**4. Retrieve students' details studying in courses ‘computer science’ or ‘chemistry’.**

SELECT \* FROM Student AS Stu INNER JOIN Enrollment AS Enr ON Stu.Roll\_No = Enr.Roll\_No WHERE Course = "Computer" OR Course = "Chemistry";



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

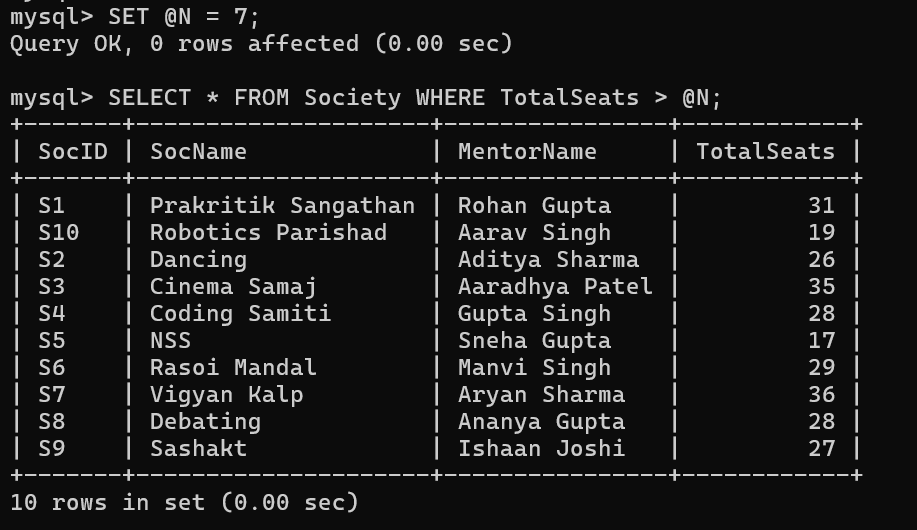
SELECT StudentName FROM Student WHERE Roll\_No LIKE "X%9" OR Roll\_No LIKE "Z%9";



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

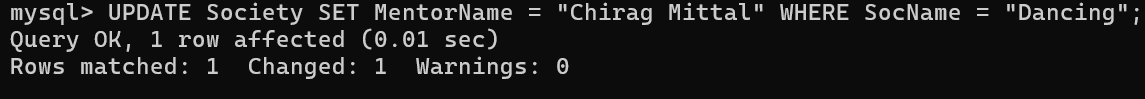
SET @N = 7;

SELECT \* FROM Society WHERE TotalSeats > @N;



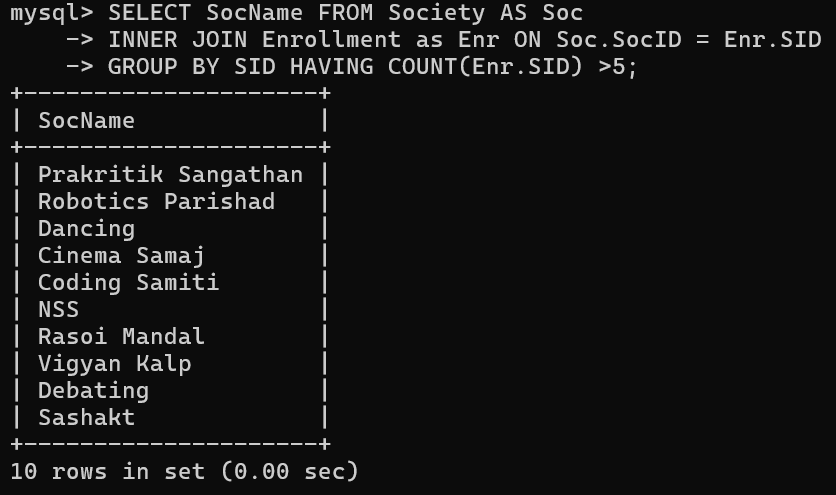
**7. Update society table for mentor name of a specific society**

UPDATE Society SET MentorName = "Chirag Mittal" WHERE SocName = "Dancing";



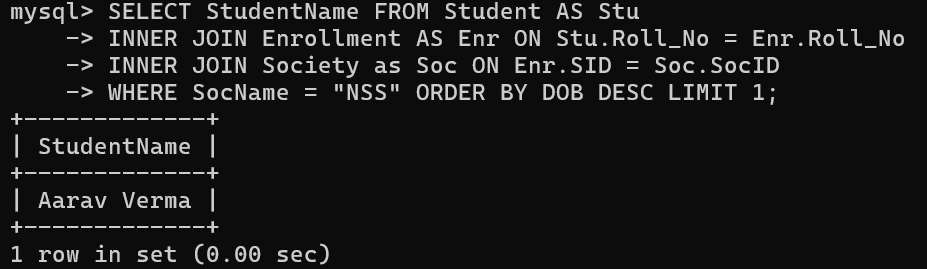
**8. Find society names in which more than five students have enrolled**

SELECT SocName FROM Society AS Soc INNER JOIN Enrollment as Enr ON Soc.SocID = Enr.SID GROUP BY SID HAVING COUNT(Enr.SID) >5;



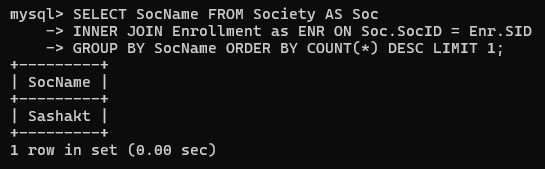
**9. Find the name of youngest student enrolled in society ‘NSS’**

SELECT StudentName FROM Student AS Stu INNER JOIN Enrollment AS Enr ON Stu.Roll\_No = Enr.Roll\_No INNER JOIN Society as Soc ON Enr.SID = Soc.SocID WHERE SocName = "NSS" ORDER BY DOB DESC LIMIT 1;



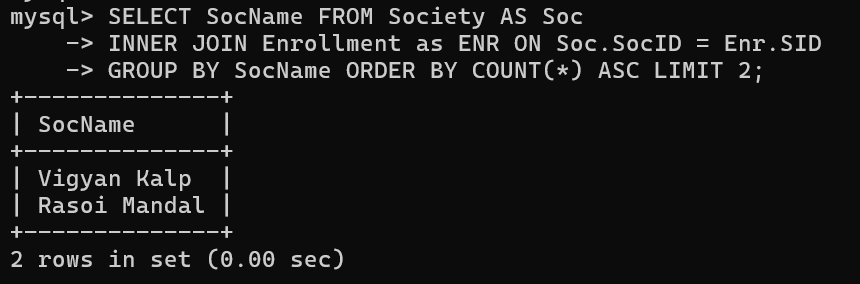
**10. Find the name of most popular society (on the basis of enrolled students)**

SELECT SocName FROM Society AS Soc INNER JOIN Enrollment as ENR ON Soc.SocID = Enr.SID GROUP BY SocName ORDER BY COUNT(\*) DESC LIMIT 1;



**11. Find the name of two least popular societies (on the basis of enrolled students)**

SELECT SocName FROM Society AS Soc INNER JOIN Enrollment as ENR ON Soc.SocID = Enr.SID GROUP BY SocName ORDER BY COUNT(\*) ASC LIMIT 2;



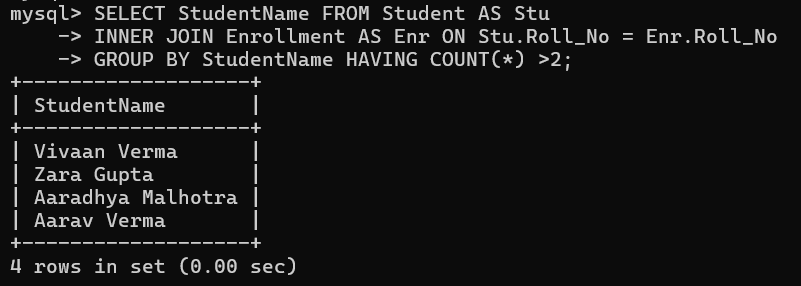
**12. Find the student names who are not enrolled in any society**

SELECT StudentName FROM Student AS Stu LEFT JOIN Enrollment AS Enr ON Stu.Roll\_No = Enr.Roll\_No WHERE SID IS NULL;



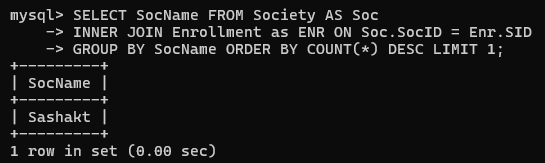
**3. Find the student names enrolled in at least two societies**

SELECT StudentName FROM Student AS Stu INNER JOIN Enrollment AS Enr ON Stu.Roll\_No = Enr.Roll\_No GROUP BY StudentName HAVING COUNT(\*) >2;



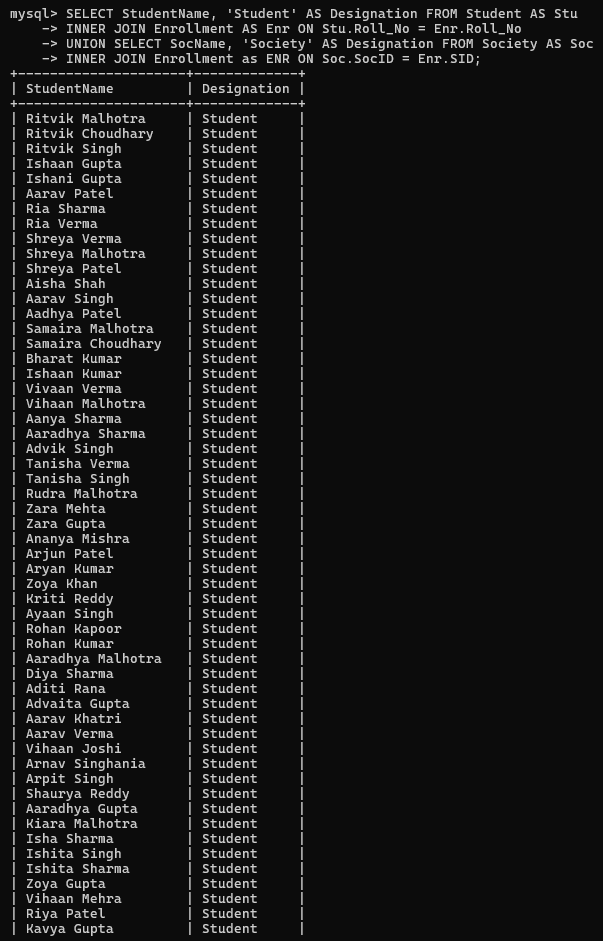
**14. Find society names in which maximum students are enrolled**

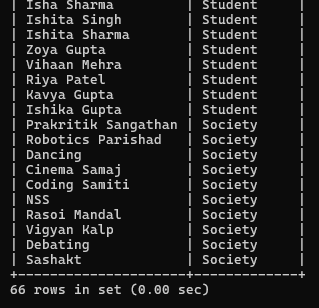
SELECT SocName FROM Society AS Soc INNER JOIN Enrollment AS Enr ON Soc.SocID = Enr.SID GROUP BY SocName ORDER BY COUNT(\*) DESC LIMIT 1;



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

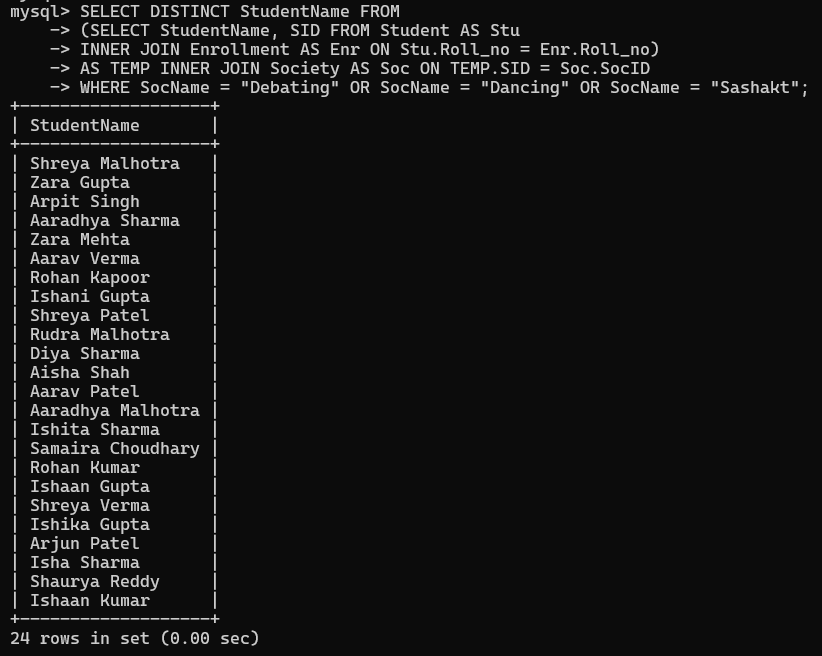
SELECT StudentName, 'Student' AS Designation FROM Student AS Stu INNER JOIN Enrollment AS Enr ON Stu.Roll\_no = Enr.Roll\_no UNION SELECT SocName, 'Society' AS Designation FROM Society AS Soc INNER JOIN Enrollment AS Enr ON Soc.SocID = Enr.SID;





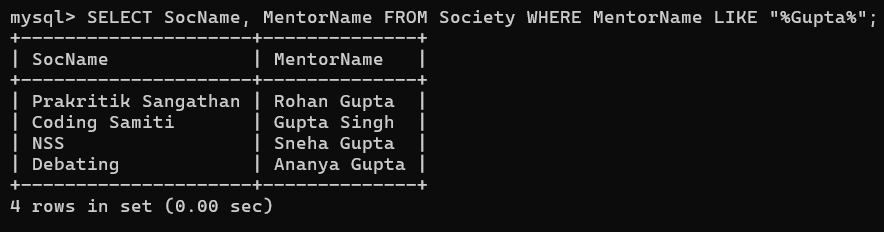
**16. Find names of students who are enrolled in any of the three societies ‘Debating’, ‘Dancing’ and ‘Sashakt’.**

SELECT DISTINCT StudentName FROM (SELECT StudentName, SID FROM Student AS Stu INNER JOIN Enrollment AS Enr ON Stu.Roll\_no = Enr.Roll\_no) AS TEMP INNER JOIN Society AS Soc ON TEMP.SID = Soc.SocID WHERE SocName = "Debating" OR SocName = "Dancing" OR SocName = "Sashakt";



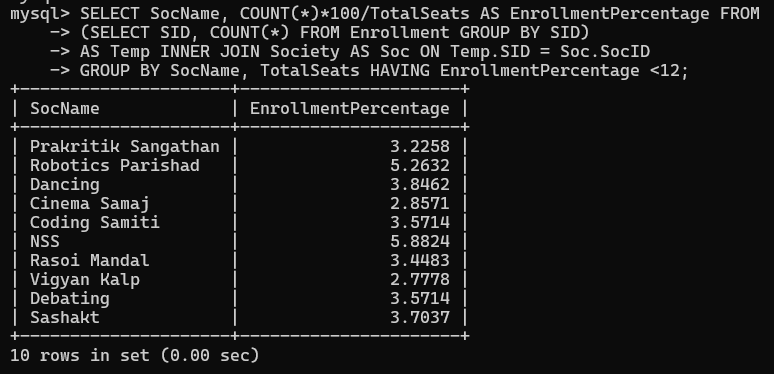
**17. Find society names such that its mentor has a name with ‘Gupta’ in it.**

SELECT SocName, MentorName FROM Society WHERE MentorName LIKE "%Gupta%";



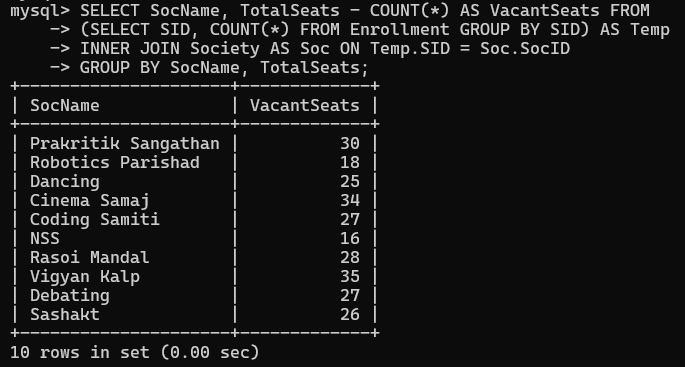
**18. Find the society names in which the number of enrolled students is only 12% of its capacity.**

SELECT SocName, COUNT(\*)\*100/TotalSeats AS EnrollmentPercentage FROM (SELECT SID, COUNT(\*) FROM Enrollment GROUP BY SID) AS Temp INNER JOIN Society AS Soc ON Temp.SID = Soc.SocID GROUP BY SocName, TotalSeats HAVING EnrollmentPercentage <12;



**19. Display the vacant seats for each society.**

SELECT SocName, TotalSeats - COUNT(\*) AS VacantSeats FROM (SELECT SID, COUNT(\*) FROM Enrollment GROUP BY SID) AS Temp INNER JOIN Society AS Soc ON Temp.SID = Soc.SocID GROUP BY SocName, TotalSeats;



**20. Increment Total Seats of each society by 10%**

UPDATE Society SET TotalSeats = 1.1\*TotalSeats;

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

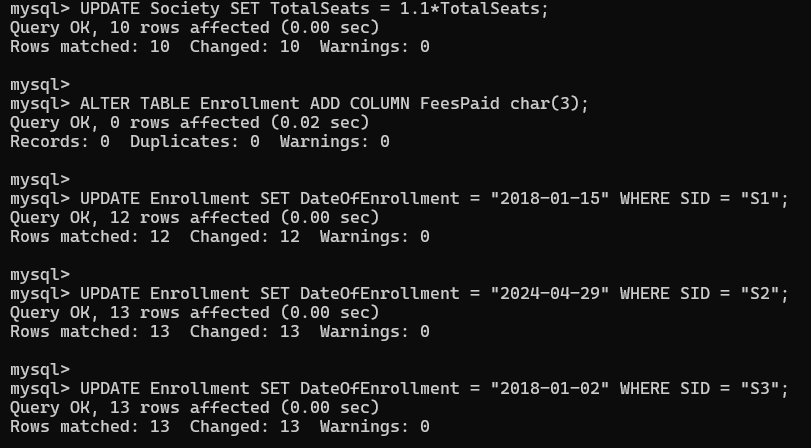
ALTER TABLE Enrollment ADD COLUMN FeesPaid char(3);

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

UPDATE Enrollment SET DateOfEnrollment = "2018-01-15" WHERE SID = "S1";

UPDATE Enrollment SET DateOfEnrollment = "2024-04-29" WHERE SID = "S2";

UPDATE Enrollment SET DateOfEnrollment = "2018-01-02" WHERE SID = "S3";

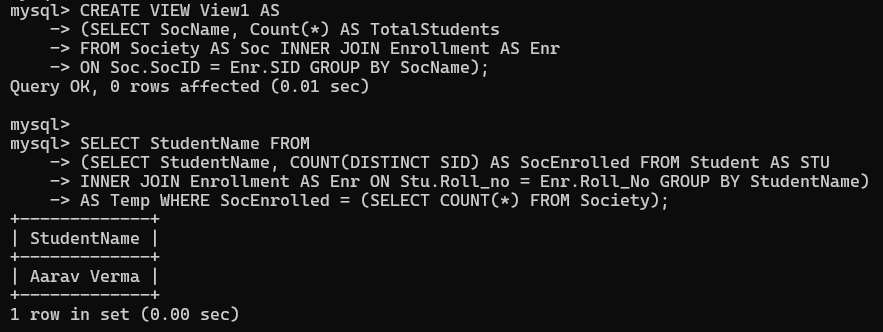


**23. Create a view to keep track of society names with the total number of students enrolled in it.**

CREATE VIEW View1 AS (SELECT SocName, Count(\*) AS TotalStudents FROM Society AS Soc INNER JOIN Enrollment AS Enr ON Soc.SocID = Enr.SID GROUP BY SocName);

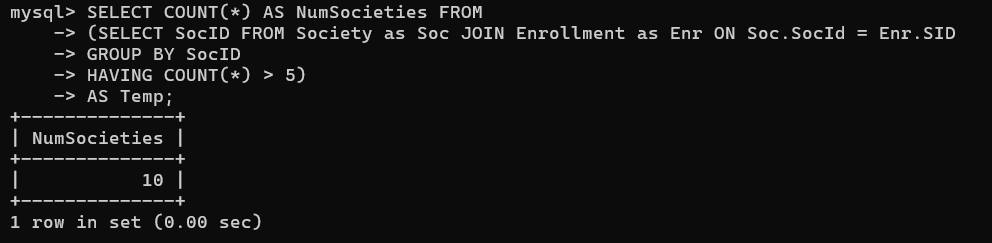
**24. Find student names enrolled in all the societies.**

SELECT StudentName FROM (SELECT StudentName, COUNT(DISTINCT SID) AS SocEnrolled FROM Student AS STU INNER JOIN Enrollment AS Enr ON Stu.Roll\_no = Enr.Roll\_No GROUP BY StudentName) AS Temp WHERE SocEnrolled = (SELECT COUNT(\*) FROM Society);



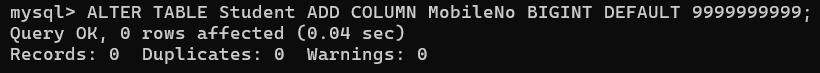
**25. Count the number of societies with more than 5 students enrolled in it**

SELECT COUNT(\*) AS NumSocieties FROM (SELECT SocID FROM Society as Soc JOIN Enrollment as Enr ON Soc.SocId = Enr.SIDGROUP BY SocIDHAVING COUNT(\*) > 5) AS Temp;



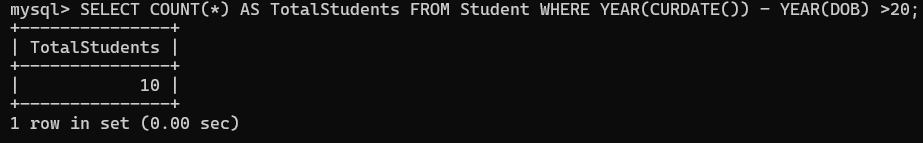
**26. Add column Mobile number in student table with default value ‘9999999999’**

ALTER TABLE Student ADD COLUMN MobileNo BIGINT DEFAULT 9999999999;

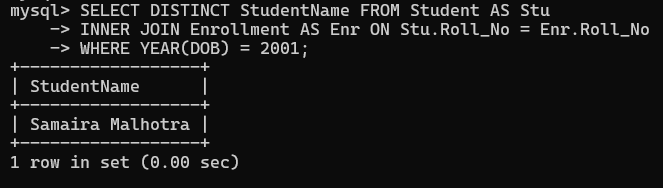
****

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

SELECT COUNT(\*) AS TotalStudents FROM Student WHERE YEAR(CURDATE()) - YEAR(DOB) >20;

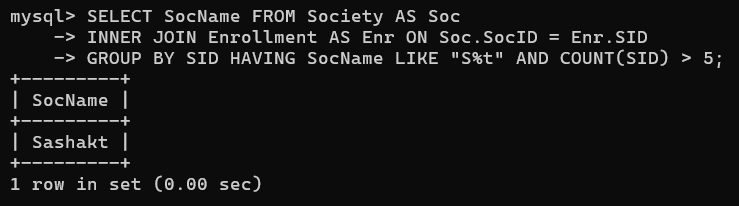
****

**28. Find names of students who are born in 2001 and are enrolled in at least one society.**SELECT DISTINCT StudentName FROM Student AS Stu INNER JOIN Enrollment AS Enr ON Stu.Roll\_No = Enr.Roll\_No WHERE YEAR(DOB) = 2001;

****

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

SELECT SocName FROM Society AS Soc INNER JOIN Enrollment AS Enr ON Soc.SocID = Enr.SID GROUP BY SID HAVING SocName LIKE "S%t" AND COUNT(SID) > 5;

****

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

SELECT SocName AS SocietyName, MentorName, TotalSeats AS TotalCapacity,TotalSeats - COUNT(\*) AS TotalEnrolledUnfilledSeat FROM Society AS Soc INNER JOIN Enrollment

AS Enr ON Soc.SocID = Enr.SID GROUP BY SocName, MentorName, TotalSeats;

