**DBMS Practicals**

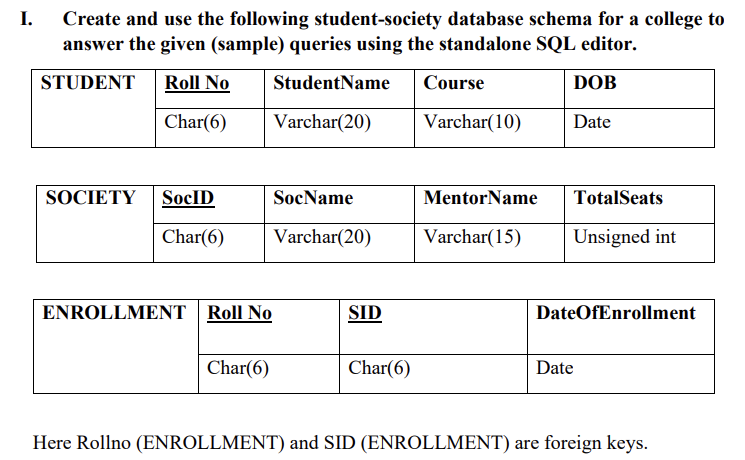
**Name – APEKSHA SHARMA**

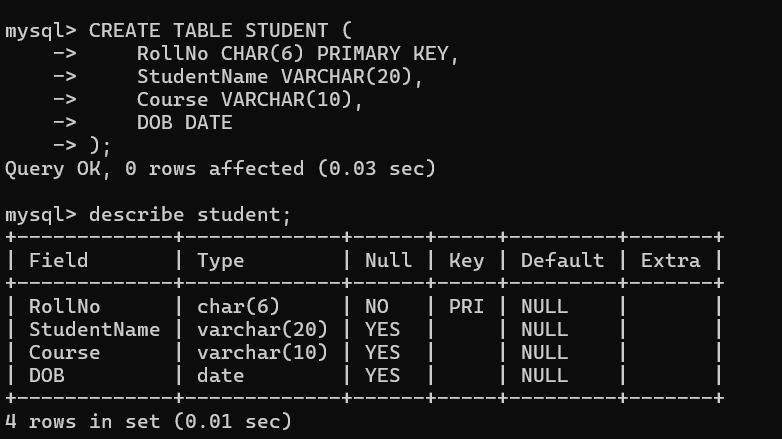
**Roll no – 22HCS4180**

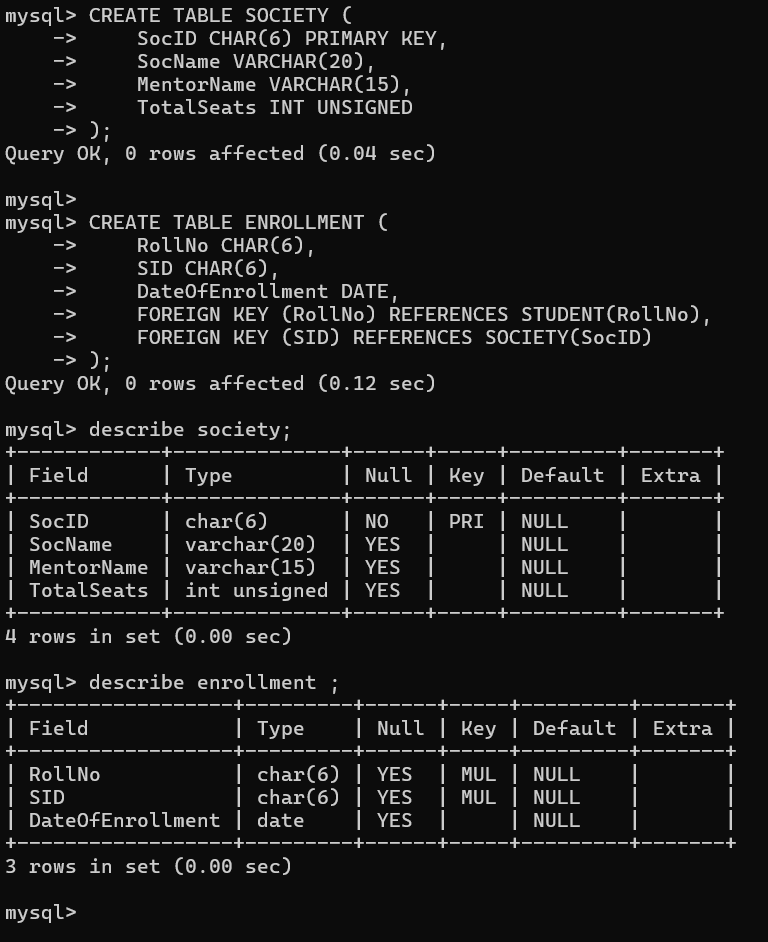
**Course – BSc CS(H)**

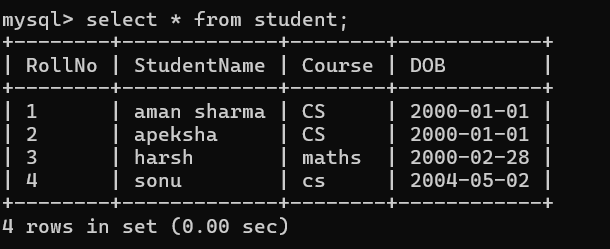
**Subject - DBMS**

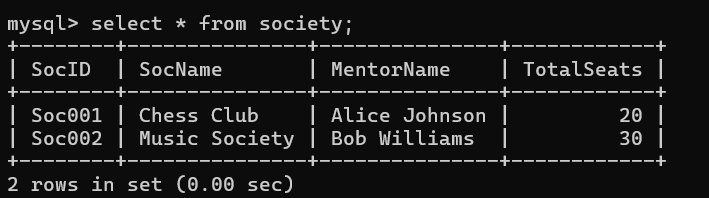
## Tables ->

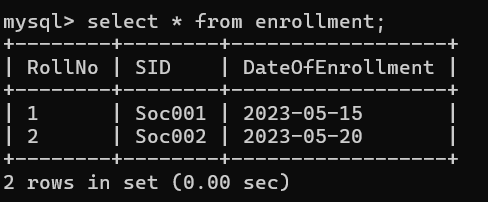




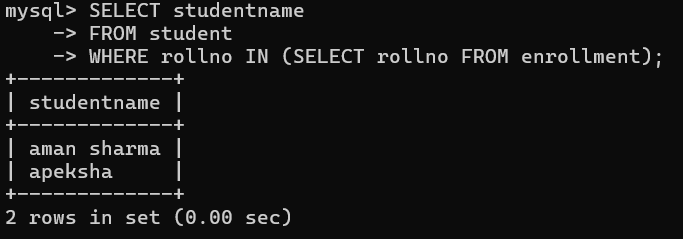




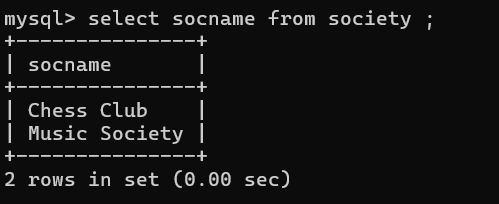




1. Retrieve names of students enrolled in any society

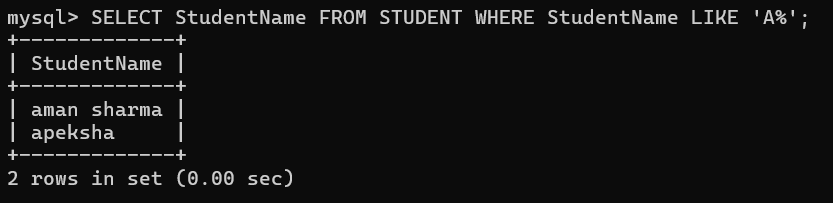


1. Retrieve all society names

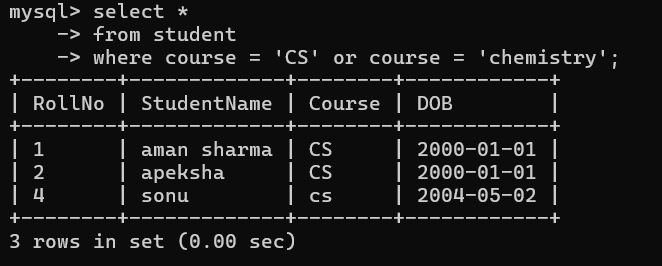


Because society Id is a primary key ,therefore we select direct socname

1. Retrieve students' names starting with letter ‘A’

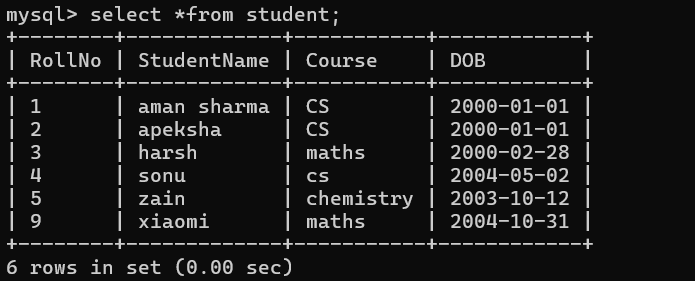


1. Retrieve students' details studying in courses ‘computer science’ or ‘chemistry’.

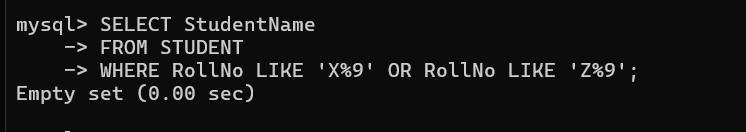


1. Retrieve students’ names whose roll no either starts with ‘X’ or ‘Z’ and ends with ‘9’

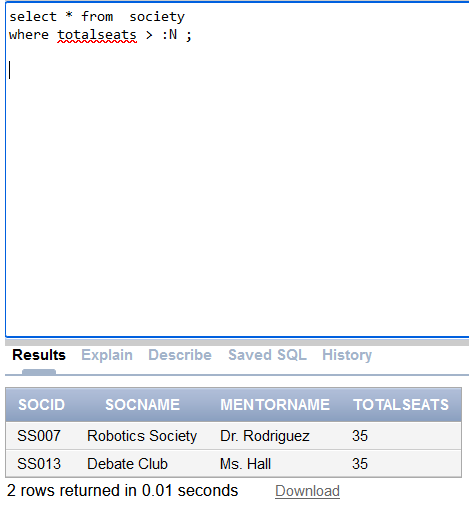
Table before query -----



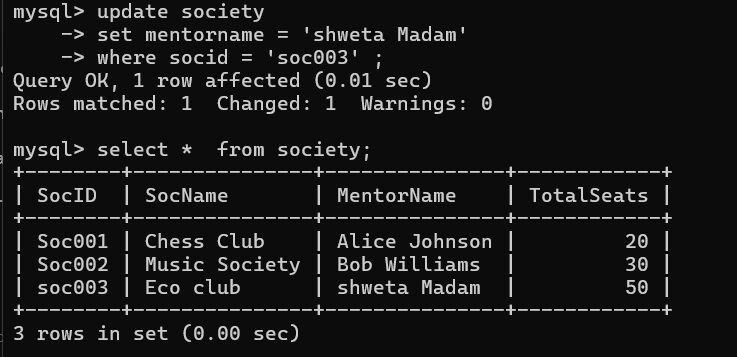
Query –



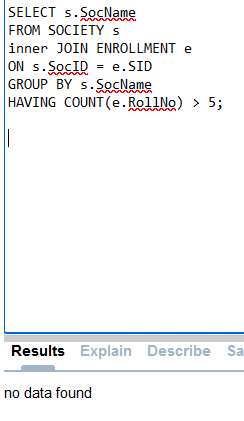
6. Find society details with more than N TotalSeats where N is to be input by the user



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



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



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

Table enrollment -🡪

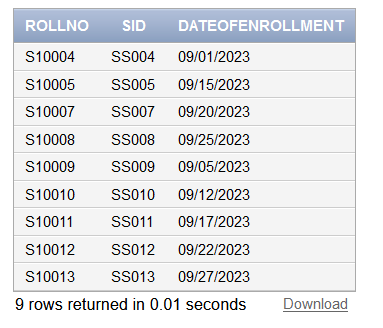


Table society 🡪

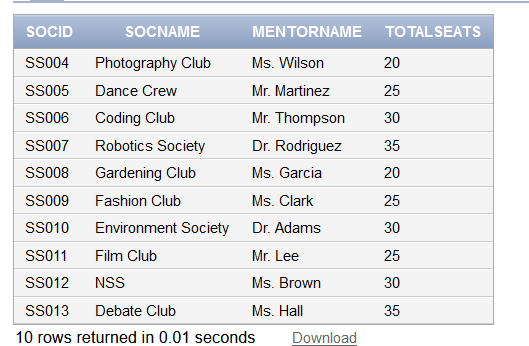
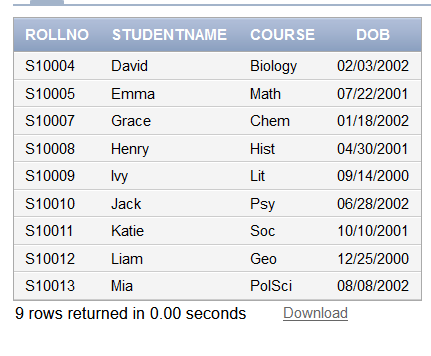
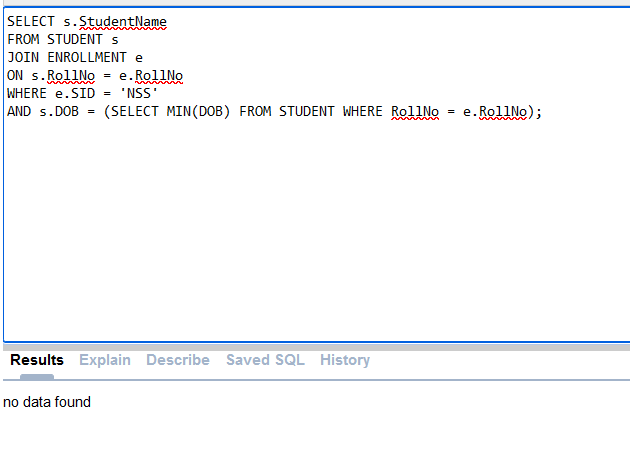


Table student ->



Query->

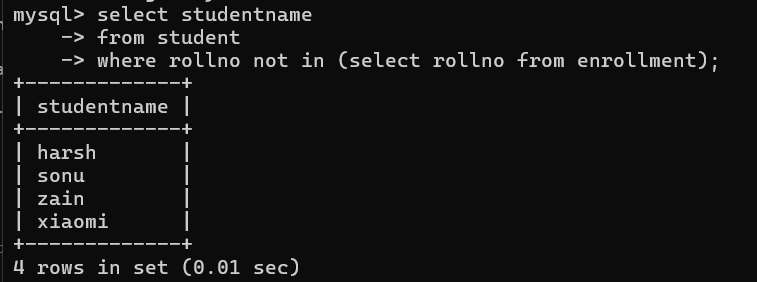


10. Find the name of most popular society (on the basis of enrolled students)

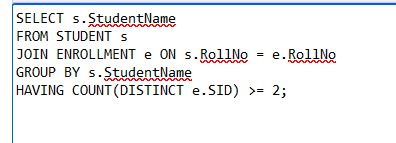
11. Find the name of two least popular societies (on the basis of enrolled students)

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

Query –



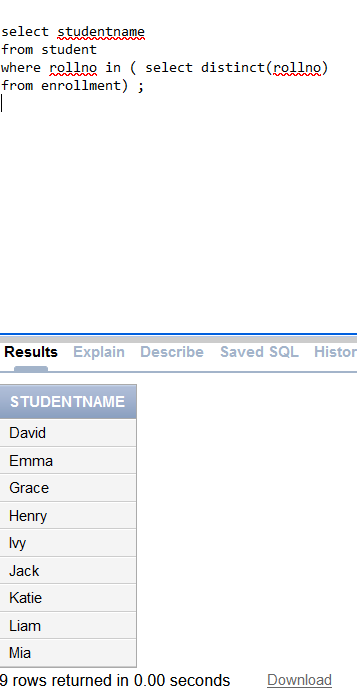
13. Find the student names enrolled in at least two societies



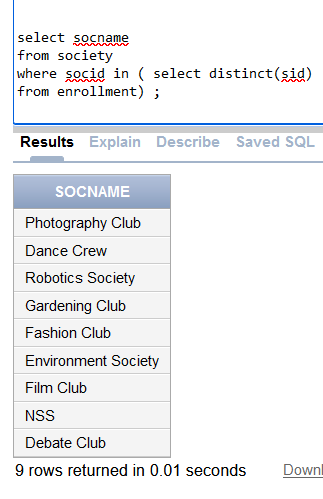
14. Find society names in which maximum students are enrolled

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

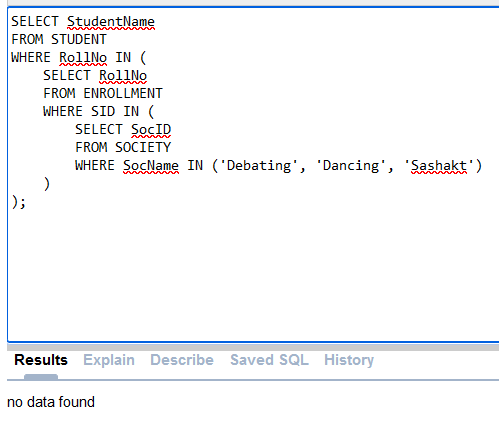
a)



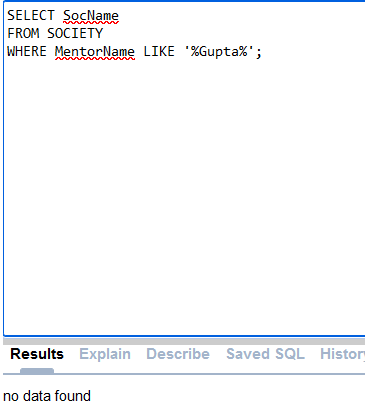
b)



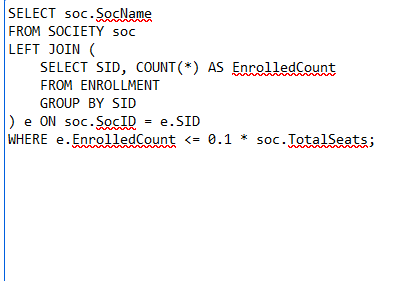
16. Find names of students who are enrolled in any of the three societies ‘Debating’, ‘Dancing’ and ‘Sashakt’.

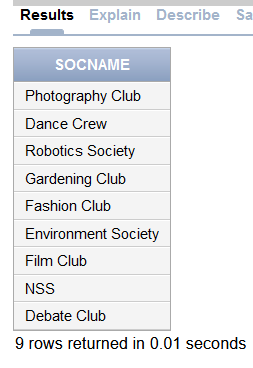


17. Find society names such that its mentor has a name with ‘Gupta’ in it.

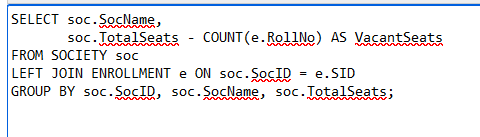


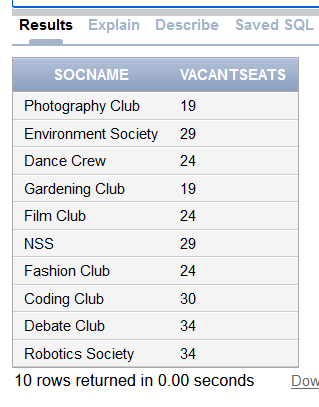
18. Find the society names in which the number of enrolled students is only 10% of its capacity.





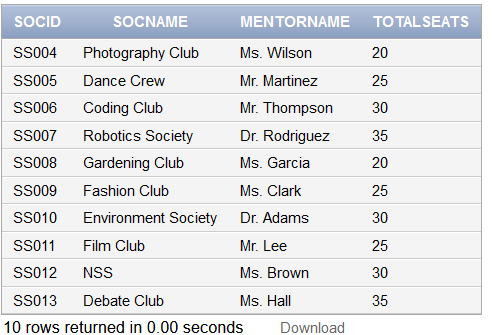
19. Display the vacant seats for each society.



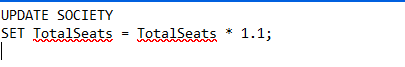


20. Increment Total Seats of each society by 10%

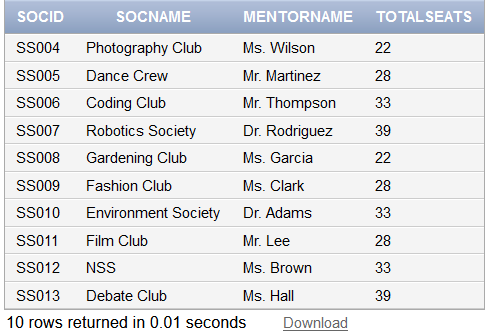
Before –



Query –

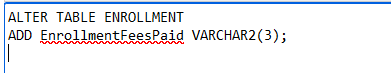


After –

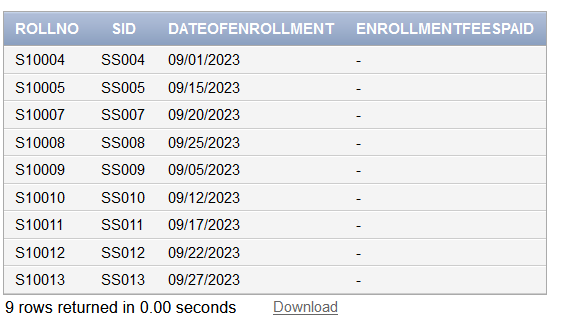


21. Add the enrollment fees paid (‘yes’/’No’) field in the enrollment table.

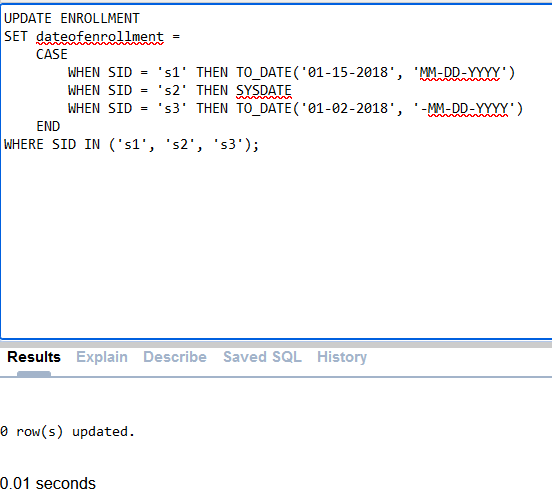
Query –



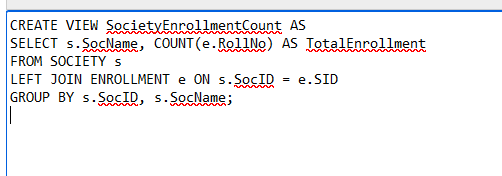
After -



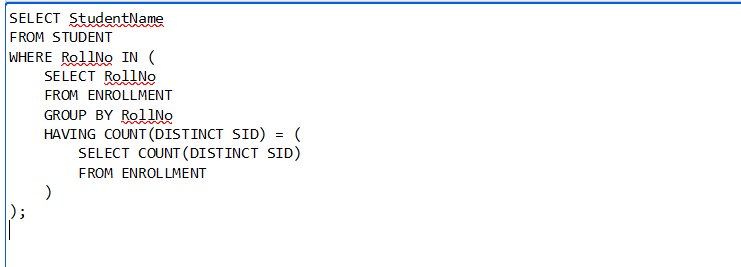
22. Update date of enrollment of society id ‘s1’ to ‘2018-01-15’, ‘s2’ to current date and ‘s3’ to ‘2018-01-02’.



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

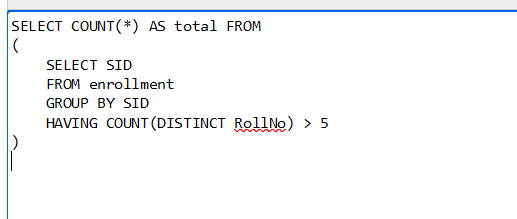


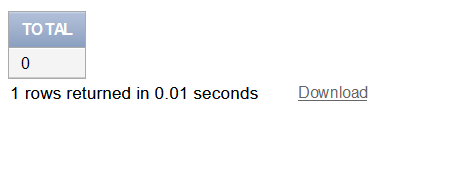
24. Find student names enrolled in all the societies.



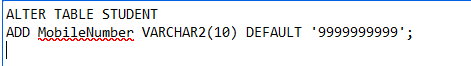


25. Count the number of societies with more than 5 students enrolled in it

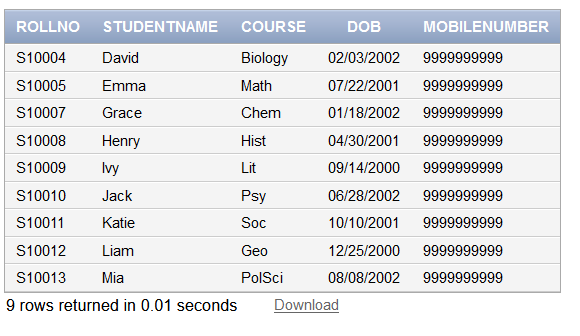




26. Add column Mobile number in student table with default value ‘9999999999’

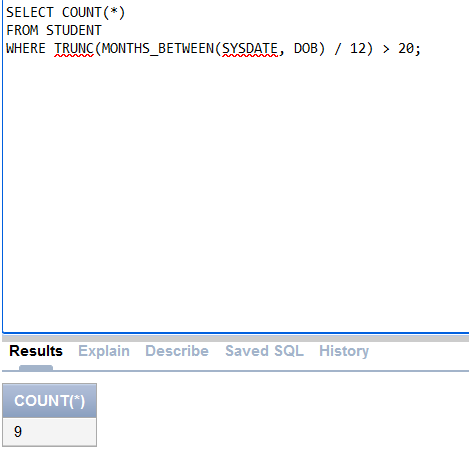


After –



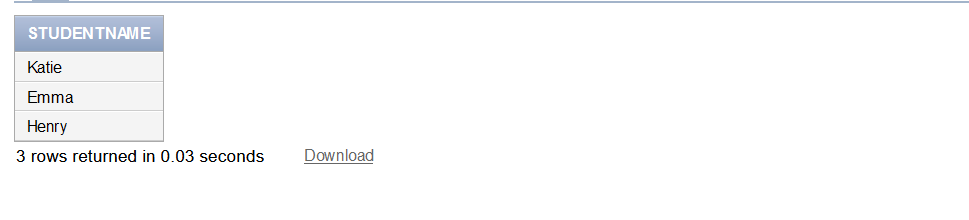
27. Find the total number of students whose age is > 20 years.

* SYSDATE returns the current date.
* MONTHS\_BETWEEN(SYSDATE, DOB) calculates the difference in months between the current date and the date of birth (DOB).
* TRUNC(... / 12) converts the difference in months to years and truncates any decimal places.
* The WHERE clause filters the students whose age (in years) is greater than 20.
* COUNT(\*) counts the number of rows that meet the criteria.

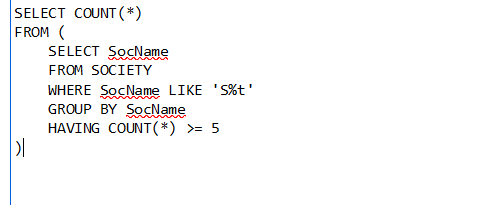


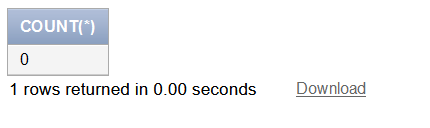
28. Find names of students who are born in 2001 and are enrolled in at least one society.





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





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