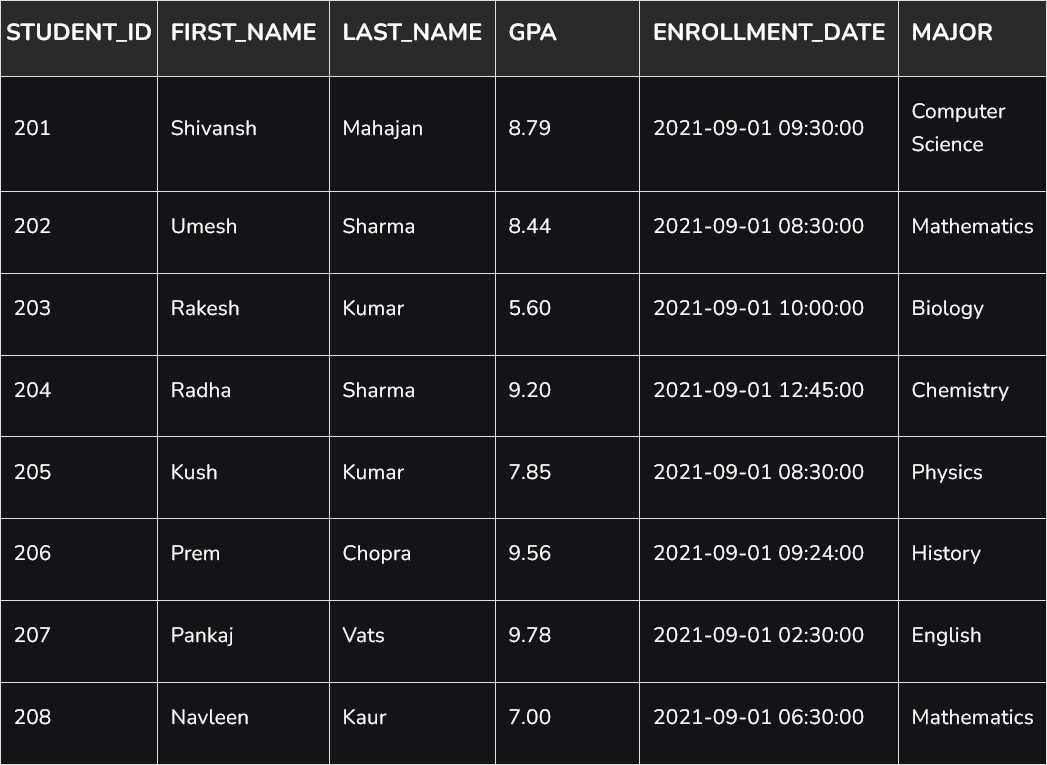
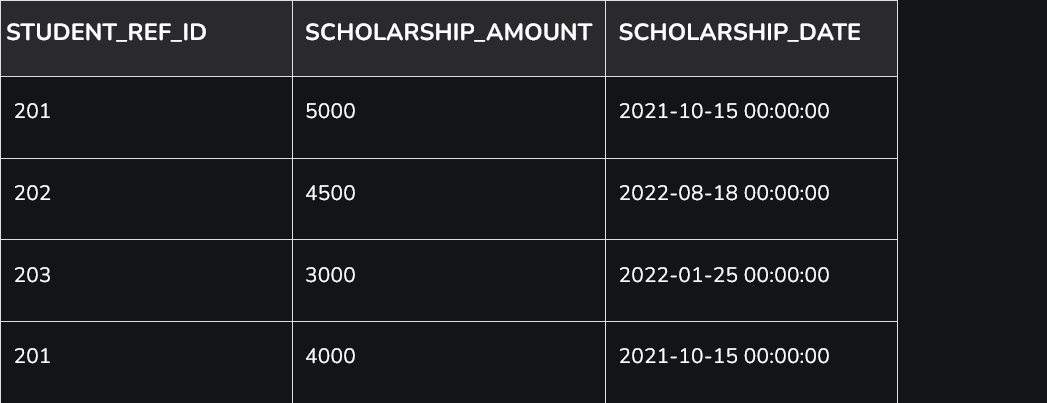
**Student Table**

****

**Scholarship Table**

****

**1. Write a SQL query to fetch “FIRST\_NAME” from the Student table in upper case and use ALIAS name as STUDENT\_NAME.**

* select upper(first\_name) as student\_name from student;

**2. Write a SQL query to fetch unique values of MAJOR Subjects from Student table.**

* select distinct major from student;

**3. Write a SQL query to print the first 3 characters of FIRST\_NAME from Student table.**

* select substring(first\_name, 1, 3) from student;

**4. Write a SQL query to find the position of alphabet (‘a’) int the first name column ‘Shivansh’ from Student table.**

* select instr(first\_name, 'a') from student where first\_name = 'Shivansh';

**5. Write a SQL query that fetches the unique values of MAJOR Subjects from Student table and print its length.**

* select major,length(major) from student group by (major);

**6. Write a SQL query to print FIRST\_NAME from the Student table after replacing ‘a’ with ‘A’.**

* select replace(first\_name, 'a', 'A') from student;

**7. Write a SQL query to print the FIRST\_NAME and LAST\_NAME from Student table into single column COMPLETE\_NAME.**

* select concat(first\_name, '', last\_name) as complete\_name from student;

**8. Write a SQL query to print all Student details from Student table order by FIRST\_NAME Ascending and MAJOR Subject descending .**

* select \* from student order by first\_name, major desc;

**9. Write a SQL query to print details of the Students with the FIRST\_NAME as ‘Prem’ and ‘Shivansh’ from Student table.**

* select \* from student where first\_name = 'Prem' or first\_name = 'Shivansh';

**10. Write a SQL query to print details of the Students excluding FIRST\_NAME as ‘Prem’ and ‘Shivansh’ from Student table.**

* select \* from Student where first\_name not in ('Prem' , 'Shivansh');

**11. Write a SQL query to print details of the Students whose FIRST\_NAME ends with ‘a’.**

* select \* from Student where first\_name like '%a';

**12. Write an SQL query to print details of the Students whose FIRST\_NAME ends with ‘a’ and contains six alphabets.**

* select \* from Student where first\_name like '\_\_\_\_\_a';

**13. Write an SQL query to print details of the Students whose GPA lies between 9.00 and 9.99.**

* select \* from Student where gpa between 9.00 and 9.99;

**14. Write an SQL query to fetch the count of Students having Major Subject ‘Computer Science’.**

* select major, count(\*) as total\_count from student where major = 'Computer Science';

**15. Write an SQL query to fetch Students full names with GPA >= 8.5 and <= 9.5.**

* select concat(first\_name, '', last\_name) as full\_name from student where gpa between 8.5 and 9.5;

**16. Write an SQL query to fetch the no. of Students for each MAJOR subject in the descending order.**

* select major, count(major) from student group by major order by count(major) desc;

**17. Display the details of students who have received scholarships, including their names, scholarship amounts, and scholarship dates.**

* select concat(s.first\_name, '', s.last\_name) as name, sc.scholarship\_amount, sc.scholarship\_date from student s inner join scholarship sc

on s.student\_id = sc.studentref\_id;

**18. Write an SQL query to show only odd rows from Student table.**

* select \* from student where student\_id % 2 != 0;

**19. Write an SQL query to show only even rows from Student table.**

* select \* from student where student\_id % 2 = 0;

**20. List all students and their scholarship amounts if they have received any. If a student has not received a scholarship, display NULL for the scholarship details.**

* select s.first\_name, s.last\_name, sc.scholarship\_amount from student s left join

scholarhip sc on s.student\_id = sc.student\_ref\_id;

**21. Write an SQL query to show the top n (say 5) records of Student table order by descending GPA.**

* select \* from student order by gpa desc limit 5;

**22. Write an SQL query to determine the nth (say n=5) highest GPA from a table.**

* select \* from student order by gpa desc limit 1 offset 4;

**23. Write an SQL query to determine the 5th highest GPA without using LIMIT keyword.**

* select \* from student s1

where 5 = (select count(distinct gpa) from student s2 where s2.gpa >= s1.gpa);

**24. Write an SQL query to fetch the list of Students with the same GPA.**

* select \* from student s1 inner join student s2

on s1.gpa = s2.gpa and s1.student\_id <> s2.student\_id;

**25. Write an SQL query to show the second highest GPA from a Student table using sub-query.**

* select max(gpa) from student where gpa not in (select max(gpa) from student);

**26. Write an SQL query to show one row twice in results from a table.**

* select \* from student union all select \* from student order by student\_id;

**27. Write an SQL query to list STUDENT\_ID who does not get Scholarship.**

* select student\_id from student where not in (select student\_ref\_id from scholarship);

**28. Write an SQL query to fetch the first 50% records from a table.**

* select \* from student limit floor((select count(\*) from student)/2);

**29. Write an SQL query to fetch the MAJOR subject that have less than 4 people in it.**

* select major, count(major) as major\_count from student group by major having count(major) < 4;

**30. Write an SQL query to show all MAJOR subject along with the number of people in there.**

* select major, count(major) as major\_count from student group by major;

**31. Write an SQL query to show the last record from a table.**

* select \* from student where student\_id = (select max(student\_id) from student);

**32. Write an SQL query to fetch the first row of a table.**

* select \* from student where student\_id = (select min(student\_id) from student);

**33. Write an SQL query to fetch the last five records from a table.**

* select \* from (select \* from student order by student\_id desc) as subq order by student\_id;

**34. Write an SQL query to fetch three max GPA from a table using co-related subquery.**

* select distinct gpa from student s1 where 3 >= (select count(distinct gpa) from student s2 where s2.gpa >= s1.gpa) order by s1.gpa desc;

**35. Write an SQL query to fetch three min GPA from a table using co-related subquery.**

* select distinct gpa from student s1 where 3 >= (select count(distinct gpa) from student s2 where s2.gpa =< s1.gpa) order by s1.gpa;

**36. Write an SQL query to fetch nth max GPA from a table.**

* select distinct gpa from student s1 where n >= (select count(distinct gpa) from student s2 where s2.gpa >= s1.gpa) order by s1.gpa desc;

**37. Write an SQL query to fetch MAJOR subjects along with the max GPA in each of these MAJOR subjects.**

* select major, max(gpa) as maxgpa from student group by major;

**38. Write an SQL query to fetch the names of Students who has highest GPA.**

* select first\_name, gpa from student where gpa = (select max(gpa) from student);

**39. Write an SQL query to show the current date and time.**

* select curdate();
* select now();

**40. Write a query to create a new table which consists of data and structure copied from the other table (say Student) or clone the table named Student.**

* create table clonetable as select \* from student;

**41. Write an SQL query to update the GPA of all the students in ‘Computer Science’ MAJOR subject to 7.5.**

* update student set gpa = 7.5 where major = 'Computer Science';

**42. Write an SQL query to find the average GPA for each major.**

* select major, avg(gpa) as avg\_major from student group by major;

**43. Write an SQL query to show the top 3 students with the highest GPA.**

* select \* from student order by gpa desc limit 3;

**44. Write an SQL query to find the number of students in each major who have a GPA greater than 7.5.**

* select major, count(student\_id) as cnt from student group by major having gpa > 7.5;

**45. Write an SQL query to find the students who have the same GPA as ‘Shivansh Mahajan’.**

* select \* from student

where gpa = (select gpa from student where first\_name = 'Shivansh' and last\_name = 'Mahajan');