**Name:** Nishant Kumar Giri

**Roll No:** AC-1254

**Subject:** DBMS

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

**Semester:** IV

**Problem Statement**

College is maintaining data about students registered in different societies with the objective of availability of information as per requirement. **E.g.**Total students registered in all societies. Total students in each society? Popular society? Least popular society? Popularity each year?? list of students registered in a society?? which course students are opting ‘society A’ in majority?? \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tables to be Created

**STUDENT**

**RollNo char**(7), **SName varchar**(20), **Course varchar**(10), **DOB date Primary Key:** RollNo

**SOCIETY**

SID **char**(6), SocietyName varchar(20), Mentor varchar(10), TotalSeats **int unsigned**

**Primary Key:** SID

**ENROLL**

RollNo, SID, DOE  
**Primary Key:** RollNo & SID, Foreign Key: RollNo & SID

**Other constraints:** Name of society and student cannot be NULL. By default total seats in each society is 10 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Create Database

Graphical user interface, text

Description automatically generated

Create Tables

Student Table

Text

Description automatically generated

Text

Description automatically generated with medium confidence

Graphical user interface, text

Description automatically generated

Text

Description automatically generated with low confidence

Society Table

Text

Description automatically generated

Text

Description automatically generated

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated with medium confidence

Enrollment Table

Text

Description automatically generated

Populate Data

Inserting values in STUDENT table:

Text

Description automatically generated

Text

Description automatically generated with low confidence

Inserting values in SOCIETY table:

Text

Description automatically generated

Text

Description automatically generated with low confidence

Inserting values in ENROLL table:

Text

Description automatically generated

Queries

Q1. Retrieve names of student enrolled in any society.

**Sol:**

**Graphical user interface, text

Description automatically generated**

Q2. Retrieve all society names.

**Sol:**

**Text

Description automatically generated**

Q3. Retrieve students names starting with letter ‘A’.

**Sol:**

Text

Description automatically generated with low confidence

Q4. Retrieve students studying in course ‘computer sc’ or ‘chemistry’.

**Sol:**

Text

Description automatically generated

Q5. Retrieve students whose roll no either starts with ‘X’ or ‘Z’.

**Sol:**

For this command we will use ‘P’ instead of ‘X’

Text

Description automatically generated

Q6. Find society whose capacity is more than 10.

**Sol**

Graphical user interface, text

Description automatically generated

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

**Sol: Text

Description automatically generated**

Q8. Find names of societies with student enrolled > 05.

**Sol:**

Text

Description automatically generated

Q9. Find society names in which more than five students have enrolled in a given year.

**Sol:**

**Text

Description automatically generated**

Q10. Find the most popular and least popular society name (on the basis

of enrolled students).

**Sol:**

**Text

Description automatically generated**

Q11. Find the student names who are not enrolled in any society.

**Sol:**

Text

Description automatically generated

Q12. Find the student names enrolled in at least two societies.

**Sol:**

Text

Description automatically generated

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

**Sol:**

Graphical user interface, text

Description automatically generated

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

**Sol:**

A picture containing table

Description automatically generated

Q15. Find names of students who are enrolled in all three societies

‘debating’, ‘dancing’ and ‘sashakt’.

**Sol:**

**We will use E-cell, NSS and Tark for this case.**

**Text

Description automatically generated**

Q16. Find society names that has ‘abc’ as mentor or ‘abc’ as the name

of enrolled student.

**Sol:**

**Here we’ll be using ‘Yash’ as ‘abc’.**

Text

Description automatically generated

Q17. Find society names whose mentor name is same as that of any

enrolled student in it.

**Sol:**

**Since we don’t have such data, we will update the Mentor of Picfie to Yash. After that, we will run this query.**

Text

Description automatically generated

Now, we will perform the required query.

Text

Description automatically generated

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

**Sol:**

Text

Description automatically generated

Q19. Display the vacant seats for each society.

**Sol:**

Text

Description automatically generated

Q20. Increment capacity of each society by 10%.

**Sol:**

Graphical user interface, text

Description automatically generated

Text

Description automatically generated with low confidence

Q21. Add enrollment fees paid (‘yes’/’No’) field in the enrollment table.

**Sol:**

A picture containing table

Description automatically generated

Q22. Update date of enrollment of society s1 to ‘2018-01-15’, s2 to current date and s3 to ‘2018-01-02’.

**Sol:**

Text

Description automatically generated with medium confidence

Q23. Find society names whose enrollment is over.

**Sol:**

Text

Description automatically generated

Q24. Display all Students details alongwith Society name if they are enrolled in any society.  
**Sol:**

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

Q26. Find student names enrolled in all socities.

**Sol:**

Q27. Count societies with student enrolled > 3.

**Sol:**

Q28. add column contact in student with default value.

**Sol:**

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

Q30. Find total number of students whose age is > 20 years.

**Sol:**

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

Q32. Remove default value of any field.

**Sol:**

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

Q34. Find common societies of students of courses ‘BSc CS’ and ‘BSc Chem’  
**Sol:**

Q35. Display society names in uppercase and padded with character \* to get a length of 15 characters which are mentored by mentors whose names start with ‘S’ and ends with ‘y’ and capacity is between 20 to 40.

**Sol:**