

Ramakrishna Mission Vidyamandira

(An Autonomous College Under University of Calcutta)

Computer Science (Honors) Semester IV 2025

Paper: 4CMSMJC3 Practical

|  |
| --- |
| **Submitted by** |
| Class Roll Number: 340  B.Sc.  4th Semester  Batch: 2023-27 |

**INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SI NO.** | **Page Start** | **Page End** | **D-O-S** | **SIGNATURE** |
| **Question no 1** | 3 | 8 |  |  |
| **Question no 2** | 9 | 17 |  |  |
| **Question no 3** | 18 | 21 |  |  |
| **Question no 4** | 22 | 25 |  |  |
| **Question no 5** | 26 | 30 |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Question 1:**

STUDENTS (NAME, ROLL, ADDRESS, MAIN)

ADMISSION (ROLL, COURSE, SEMESTER)

FACULTY (COURSE, FACULTY, SEMESTER)

OFFERING (BRANCH, COURSE)

Write down the following SQL queries:

1. All courses taken by a given student.

2. Names of the students admitted in a particular course in a given semester.

3. Were two students (x and y) ever admitted in the same course in the same semester?

4. Students who have taken all courses offered by a given faculty.

5. Find the name of the faculty who taught maximum courses.

Tables:

Table STUDENTS (NAME, ROLL, ADDRESS, MAIN)

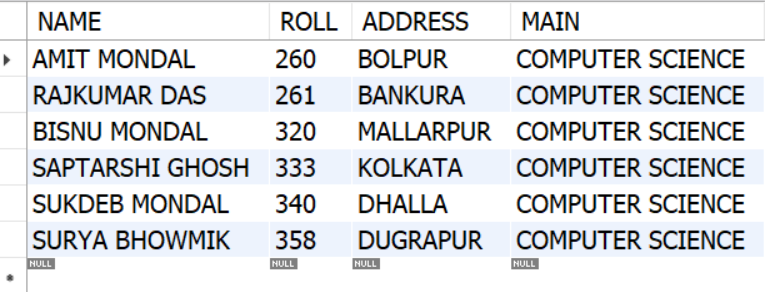


Table ADMISSION (ROLL, COURSE, SEMESTER)

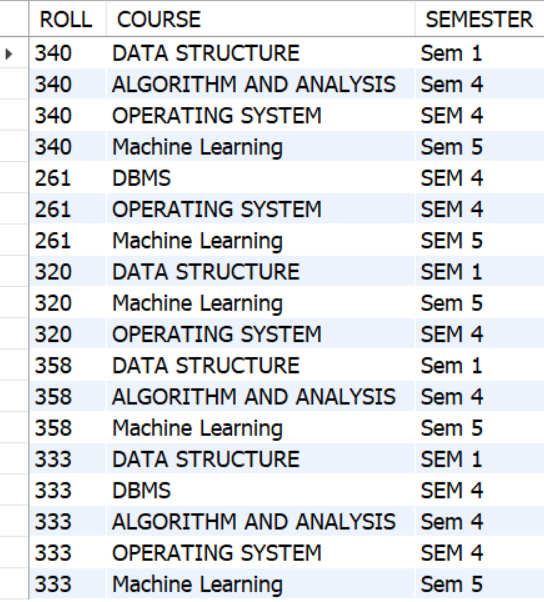


Table FACULTY (COURSE, FACULTY, SEMESTER)

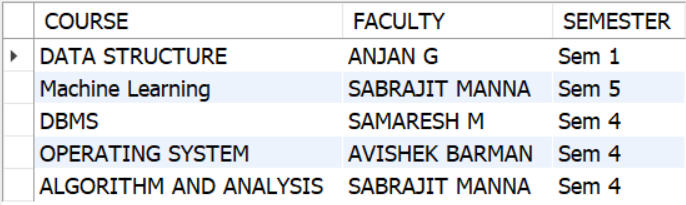
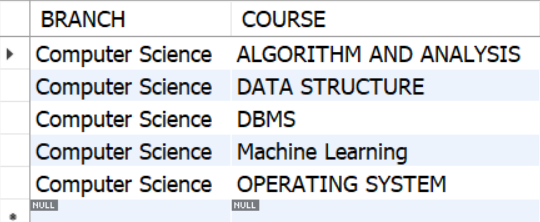


Table OFFERING (BRANCH, COURSE)



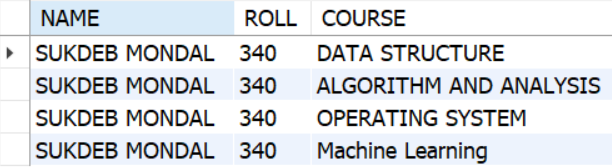
1. All courses taken by a given student.

SELECT S.NAME,A.ROLL,A.COURSE FROM STUDENTS AS S

JOIN ADMISSION AS A

ON S.ROLL = A.ROLL

WHERE S.NAME="sukdeb mondal";



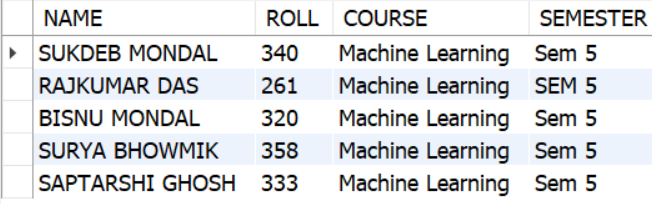
1. Names of the students admitted in a particular course in a given semester.

SELECT S.NAME , A.ROLL,A.COURSE,A.SEMESTER FROM STUDENTS AS S

JOIN ADMISSION AS A

ON S.ROLL = A.ROLL

WHERE A.SEMESTER="SEM 5" AND A.COURSE="Machine Learning";



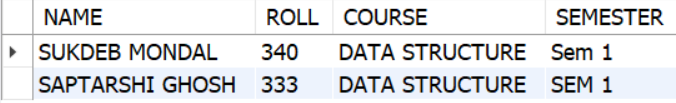
1. Were two students (x and y) ever admitted in the same course in the same semester?

SELECT S.NAME ,A.ROLL,A.COURSE,A.SEMESTER FROM STUDENTS AS S

JOIN ADMISSION AS A

ON S.ROLL = A.ROLL

WHERE A.ROLL IN (333,340) AND A.COURSE="DATA STRUCTURE";



1. Students who have taken all courses offered by a given faculty.

SELECT S.NAME, A.COURSE FROM STUDENTS AS S

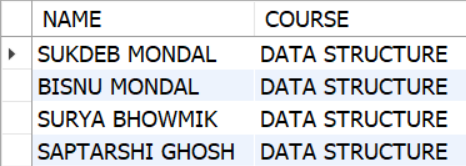
JOIN ADMISSION AS A

ON S.ROLL = A.ROLL

WHERE A.COURSE IN(

SELECT COURSE FROM FACULTY WHERE FACULTY="ANJAN G"

);

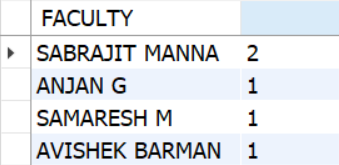


5. Find the name of the faculty who taught maximum courses.

SELECT FACULTY,COUNT(\*)AS COUNT FROM FACULTY

GROUP BY FACULTY

ORDER BY COUNT(COURSE) DESC;



**Question 2:**

STUDENT (SID, SNAME, SEX, AGE, YEAR, GPA)

DEPT (DNAME, NUMPHDS)

PROF (PNAME, DNAME)

COURSE (CNO, CNAME, DNAME)

MAJOR (DNAME, SID)

SECTION (DNAME, CNO, SECTNO, PNAME)

ENROLL (SID, GRADE, DNAME, CNO, SECTNO)

Write down the following SQL queries:

1. List the names of professors who work in department that have fewer than 20 PhD.

2. List the names of students with the lowest GPA.

3. List the name and sid of student enrolled in maximum classes.

4. List the name of student and department name whose sid=’Bsc 2017’

5. List the names of students who are taking both a Computer Science and Mathematics

course.

Table STUDENT (SID, SNAME, SEX, AGE, YEAR, GPA)

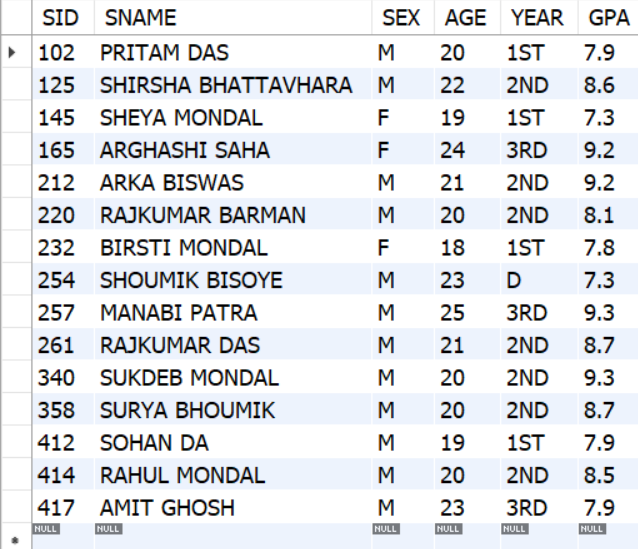


Table DEPT (DNAME, NUMPHDS)

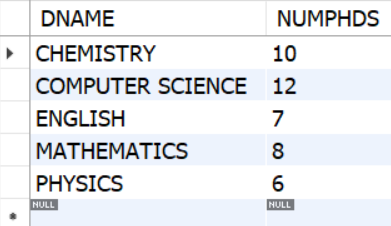


Table PROF (PNAME, DNAME)

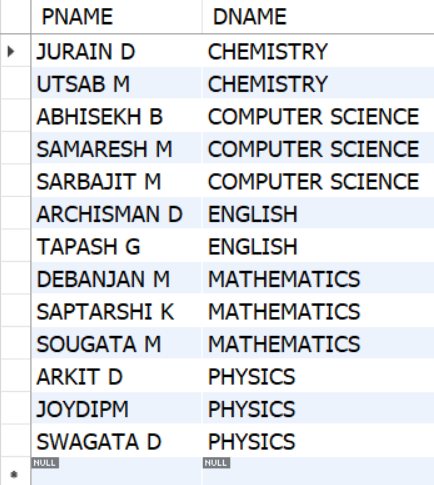


Table COURSE (CNO, CNAME, DNAME)

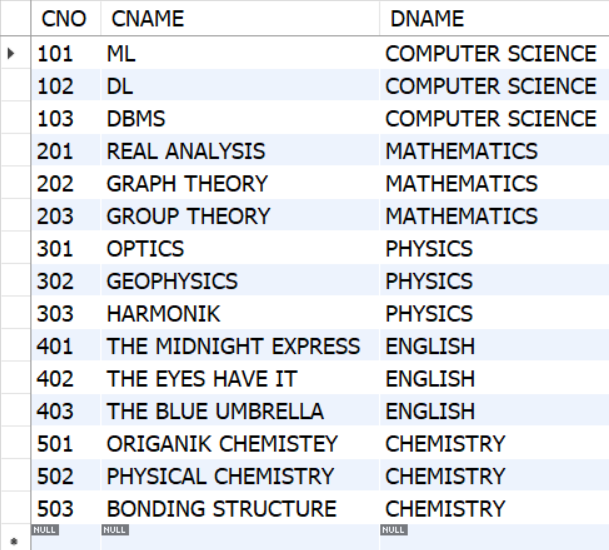


Table MAJOR (DNAME, SID)

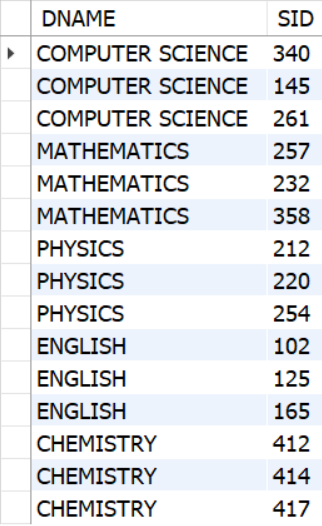


Table SECTION (DNAME, CNO, SECTNO, PNAME)

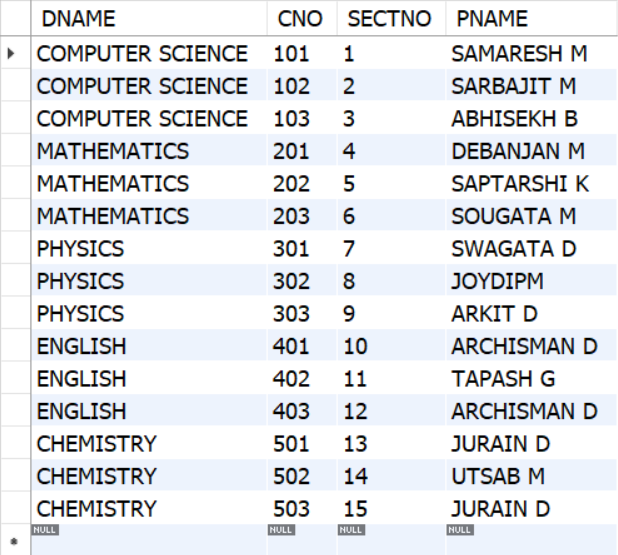
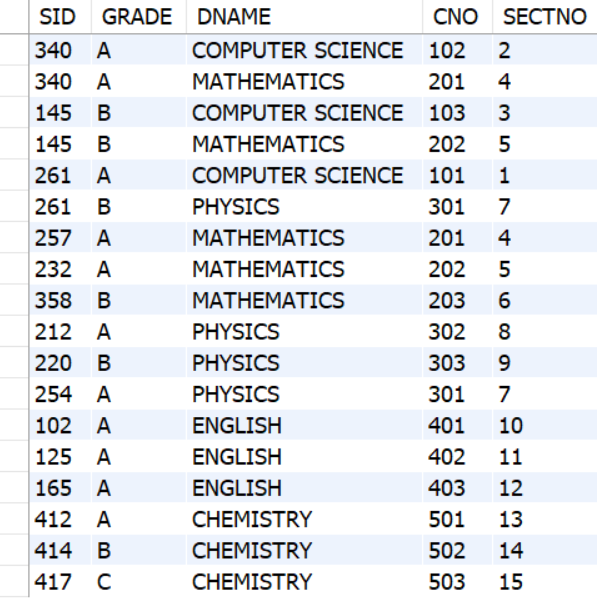


Table ENROLL (SID, GRADE, DNAME, CNO, SECTNO)



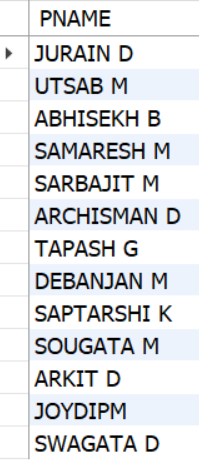
1. List the names of professors who work in department that have fewer than 20 PhD.

SELECT P.PNAME

FROM PROF P

JOIN DEPT D ON P.DNAME = D.DNAME

WHERE D.NUMPHDS < 20;

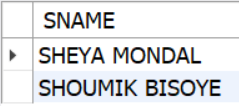


2. List the names of students with the lowest GPA.

SELECT SNAME

FROM STUDENT

WHERE GPA = (SELECT MIN(GPA) FROM STUDENT);



3. List the name and sid of student enrolled in maximum classes.

SELECT S.SID, S.SNAME

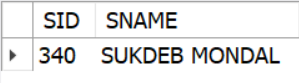
FROM STUDENT S

JOIN ENROLL E ON S.SID = E.SID

GROUP BY S.SID, S.SNAME

ORDER BY COUNT(\*) DESC

LIMIT 1;



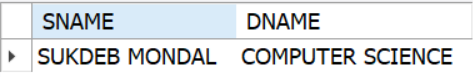
1. List the name of student and department name whose sid=’Bsc 2017’

SELECT S.SNAME, M.DNAME

FROM STUDENT S

JOIN MAJOR M ON S.SID = M.SID

WHERE S.SID = 340;



5. List the names of students who are taking both a Computer Science and Mathematics

course.

SELECT S.SNAME

FROM STUDENT S

WHERE S.SID IN (

SELECT E1.SID

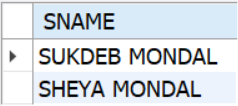
FROM ENROLL E1

JOIN ENROLL E2 ON E1.SID = E2.SID

WHERE E1.DNAME = 'COMPUTER SCIENCE'

AND E2.DNAME = 'MATHEMATICS'

);



**Question 3:**

EMPLOYEE (EMP\_ID, EMP\_NAME, SALARY, DEPT\_ID)

DEPARTMENT (DEPT\_ID, DEPT\_NAME, MGR\_ID, FLOOR\_NO)

Write down the following SQL queries:

1. Display the names of all employees who work on 10th floor and earn less than Rs. 10,000.

2.List names of all managers who manage two or more departments on the same floor.

3. Give 10% hike to the salary to every employee who works in sales department and display the appropriate record.

4.Display the names of employee who earn more than every employee of IT department.

5.Print the names of employees who earn more than Rs. 50,000 and work in either sales or marketing departments.

Question 4:

DEPOSIT (BR\_NAME, ACC\_NO, CUST\_NAME, BALANCE)

CUSTOMER (CUST\_NAME, STREET, CITY)

BRANCH (BR\_NAME, ASSETS, CITY)

BORROW (BR\_NAME, LOAN\_NO, CUST\_NAME, AMOUNT)

Table EMPLOYEE (EMP\_ID, EMP\_NAME, SALARY)

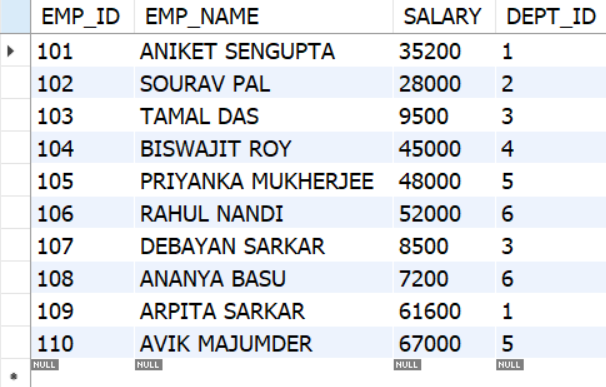
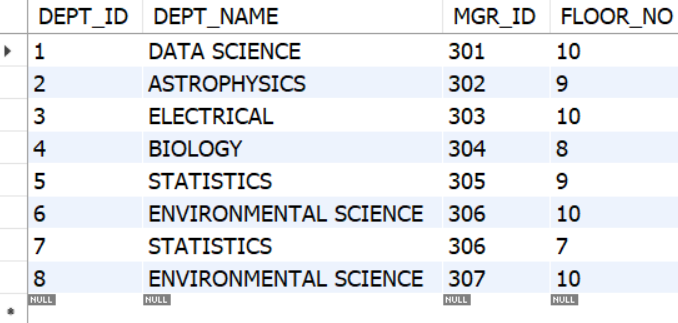


Table DEPARTMENT (DEPT\_ID, DEPT\_NAME, MGR\_ID, FLOOR\_NO)



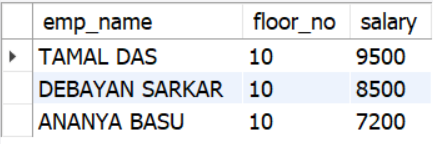
1. Display the names of all employees who work on 10th floor and earn less than Rs. 10,000.

SELECT e.emp\_name, d.floor\_no, e.salary

FROM employee AS e

JOIN department AS d ON e.dept\_id = d.dept\_id

WHERE d.floor\_no = 10 AND e.salary < 10000;



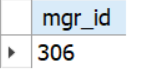
2.List names of all managers who manage two or more departments on the same floor.

SELECT mgr\_id

FROM department

GROUP BY mgr\_id

HAVING COUNT(\*) >= 2;



1. Give 10% hike to the salary to every employee who works in sales department and display the appropriate record.

SELECT e.emp\_name, e.salary, e.salary \* 1.10 AS hiked\_salary

FROM employee e

JOIN department d ON e.dept\_id = d.dept\_id

WHERE d.dept\_name = 'DATA SCIENCE';

UPDATE employee

SET salary = salary \* 1.10

WHERE dept\_id IN (

SELECT dept\_id FROM department WHERE dept\_name = 'DATA SCIENCE'

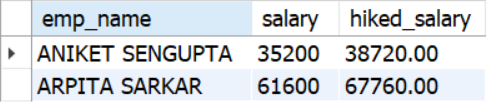
);

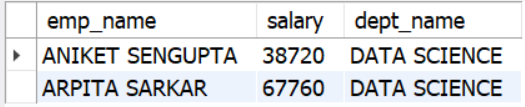
SELECT e.emp\_name, e.salary, d.dept\_name

FROM employee e

JOIN department d ON e.dept\_id = d.dept\_id

WHERE d.dept\_name = 'DATA SCIENCE';



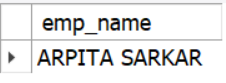


4.Display the names of employee who earn more than every employee of IT department.

SELECT emp\_name

FROM employee

WHERE salary = (SELECT MAX(salary) FROM employee);



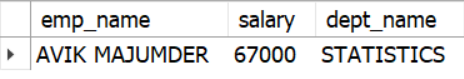
5.Print the names of employees who earn more than Rs. 50,000 and work in either sales or marketing departments.

SELECT e.emp\_name, e.salary, d.dept\_name

FROM employee e

JOIN department d ON e.dept\_id = d.dept\_id

WHERE e.salary > 50000 AND d.dept\_name IN ('BIOLOGY', 'STATISTICS');



**Question 4:**

DEPOSIT (BR\_NAME, ACC\_NO, CUST\_NAME, BALANCE)

CUSTOMER (CUST\_NAME, STREET, CITY)

BRANCH (BR\_NAME, ASSETS, CITY)

BORROW (BR\_NAME, LOAN\_NO, CUST\_NAME, AMOUNT)

Write down the following SQL queries:

1. Find all customers who have either account or loan or both in SBI, Belur Math Branch

2. Find customers with names and cities in which they live who have taken loan from

SBI, Kolkata Branch.

3. Find customers name and amount including name of the branch for loan over

Rs.50,000.

4. Find customer having highest balance.

5. Find branch having highest loan.

Table DEPOSIT (BR\_NAME, ACC\_NO, CUST\_NAME, BALANCE)

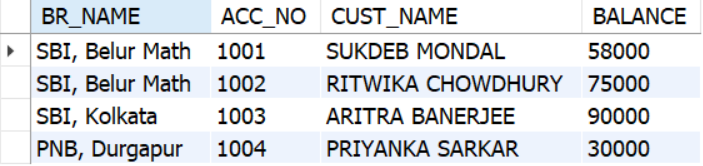


Table CUSTOMER (CUST\_NAME, STREET, CITY)

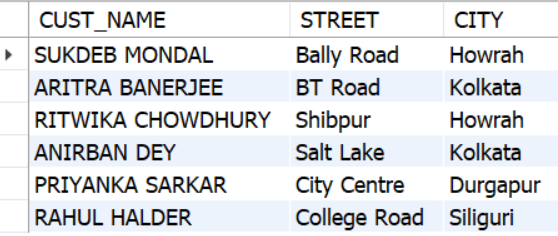


Table BRANCH (BR\_NAME, ASSETS, CITY)

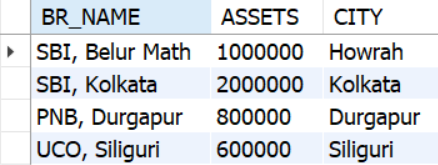
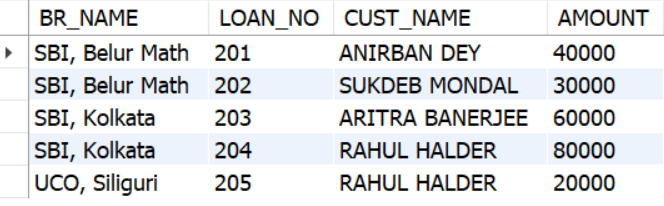


Table BORROW (BR\_NAME, LOAN\_NO, CUST\_NAME, AMOUNT)



1. Find all customers who have either account or loan or both in SBI, Belur Math Branch

SELECT DISTINCT CUST\_NAME

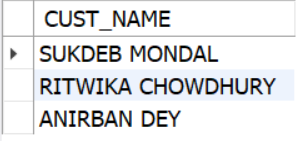
FROM (

SELECT CUST\_NAME FROM DEPOSIT WHERE BR\_NAME = 'SBI, Belur Math'

UNION

SELECT CUST\_NAME FROM BORROW WHERE BR\_NAME = 'SBI, Belur Math'

) AS RESULT;



2. Find customers with names and cities in which they live who have taken loan from

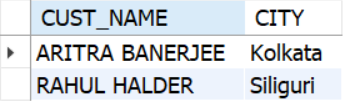
SBI, Kolkata Branch.

SELECT DISTINCT C.CUST\_NAME, C.CITY

FROM CUSTOMER C

JOIN BORROW B ON C.CUST\_NAME = B.CUST\_NAME

WHERE B.BR\_NAME = 'SBI, Kolkata';



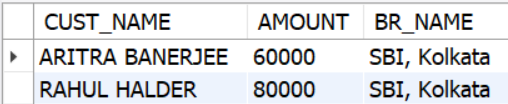
3. Find customers name and amount including name of the branch for loan over

Rs.50,000.

SELECT CUST\_NAME, AMOUNT, BR\_NAME

FROM BORROW

WHERE AMOUNT > 50000;



1. Find customer having highest balance.

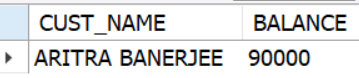
SELECT CUST\_NAME, BALANCE

FROM DEPOSIT

WHERE BALANCE = (

SELECT MAX(BALANCE) FROM DEPOSIT

);



1. Find branch having highest loan.

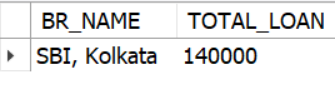
SELECT BR\_NAME, SUM(AMOUNT) AS TOTAL\_LOAN

FROM BORROW

GROUP BY BR\_NAME

ORDER BY TOTAL\_LOAN DESC

LIMIT 1;



**Question 5:**

STUDENT(SNUM: INTEGER, SNAME: STRING, MAJOR: STRING,LEVEL: STRING, AGE:

INTEGER)

CLASS(NAME: STRING, MEETS AT: STRING, ROOM: STRING,FID: INTEGER)

ENROLLED(SNUM: INTEGER, CNAME: STRING)

FACULTY(FID: INTEGER, FNAME: STRING, DEPTID: INTEGER)

The meaning of these relations is straightforward; for example, Enrolled has one record per

student-class pair such that the student is enrolled in the class.

Write the following queries in SQL. No duplicates should be printed in any of the answers.

1. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

2. For each level, print the level and the average age of students for that level.

3. For all levels except JR, print the level and the average age of students for that level.

4. For each faculty member that has taught classes only in room R128, print the faculty member's name and the total number of classes she or he has taught.

5. Find the names of students enrolled in the maximum number of classes.

Table STUDENT(SNUM: INTEGER, SNAME: STRING, MAJOR: STRING,LEVEL: STRING, AGE:INTEGER)

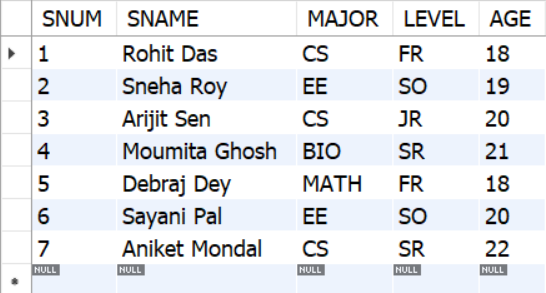


Table CLASS(NAME: STRING, MEETS AT: STRING, ROOM: STRING,FID: INTEGER)

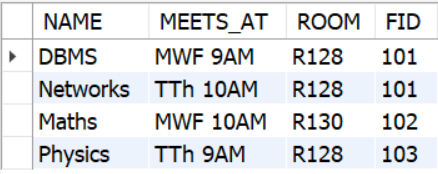


Table ENROLLED(SNUM: INTEGER, CNAME: STRING)

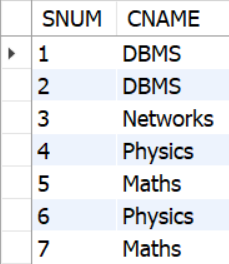
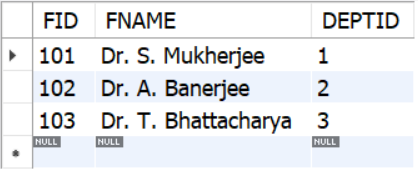


Table FACULTY(FID: INTEGER, FNAME: STRING, DEPTID: INTEGER)



1. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

SELECT DISTINCT F.FNAME

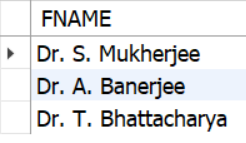
FROM FACULTY F

JOIN CLASS C ON F.FID = C.FID

LEFT JOIN ENROLLED E ON C.NAME = E.CNAME

GROUP BY F.FNAME

HAVING COUNT(E.SNUM) < 5;

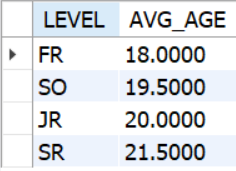


1. For each level, print the level and the average age of students for that level.

SELECT LEVEL, AVG(AGE) AS AVG\_AGE

FROM STUDENT

GROUP BY LEVEL;



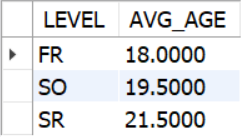
1. For all levels except JR, print the level and the average age of students for that level.

SELECT LEVEL, AVG(AGE) AS AVG\_AGE

FROM STUDENT

WHERE LEVEL <> 'JR'

GROUP BY LEVEL;



1. For each faculty member that has taught classes only in room R128, print the faculty member's name and the total number of classes she or he has taught.

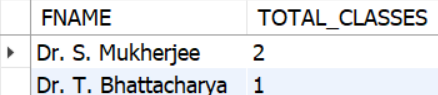
SELECT F.FNAME, COUNT(\*) AS TOTAL\_CLASSES

FROM FACULTY F

JOIN CLASS C ON F.FID = C.FID

GROUP BY F.FNAME

HAVING SUM(CASE WHEN C.ROOM = 'R128' THEN 0 ELSE 1 END) = 0;



5. Find the names of students enrolled in the maximum number of classes.

SELECT S.SNAME

FROM STUDENT S

JOIN ENROLLED E ON S.SNUM = E.SNUM

GROUP BY S.SNUM, S.SNAME

HAVING COUNT(\*) = (

SELECT MAX(CLASS\_COUNT)

FROM (

SELECT COUNT(\*) AS CLASS\_COUNT

FROM ENROLLED

GROUP BY SNUM

) AS CLASS\_COUNTS

);

