

DATABASE MANAGEMENT SYSTEMS

Course Code: CSE2007

Slot: L13+L14

Class Number: AP2023246000686

Venue: CB-102

Assignment No.: 7

Date: 14-03-2024

Reg. No: 22BCE8609

Name: Gourab Choudhury

School & Programme: B.Tech CSE core

Instructions

1. Submit Your Assignment at the following link. [Click Here](https://forms.gle/YH1bBwpKnM8LMSVM8) or <https://forms.gle/YH1bBwpKnM8LMSVM8>
2. Submit the Assignment in PDF format.
3. Submit the assignment on or before 04/March/2024.
4. Copy the code, take a picture of the output, and paste it into a Word document. Convert the word doc into PDF and submit.
5. The file name will be your registration number and name.
6. **Assignments Programs:**

Solve the following using machine learning techniques.

- | | |
|----|---|
| A. | Create Faculty and student Tables like following:
<pre>CREATE TABLE Student (
 StudentID INT PRIMARY KEY,
 Name VARCHAR(100),
 Major VARCHAR(100),
 FacultyAdvisorID INT,
 FOREIGN KEY (FacultyAdvisorID) REFERENCES Faculty(FacultyID)
);

CREATE TABLE Faculty (
 FacultyID INT PRIMARY KEY,
 Name VARCHAR(100),
 Department VARCHAR(100)
);</pre> |
| B. | INNER JOIN: Write a SQL query to list all students and their respective faculty advisors. Use the INNER JOIN to combine data from the <code>Students</code> and <code>Faculty</code> tables where the <code>FacultyAdvisorID</code> matches the <code>FacultyID</code> . |
| C. | LEFT JOIN: Write a SQL query to list all faculty members and any students they advise. Include faculty members who do not advise any students. |

D.	RIGHT JOIN: Construct a SQL query to show all students and their faculty advisors. If a student does not have a faculty advisor, their name should still appear. Use the RIGHT JOIN for this task.
E.	<ul style="list-style-type: none"> FULL OUTER JOIN - (Note: MySQL does not support FULL OUTER JOIN directly, but you can achieve the same result using a combination of LEFT JOIN and UNION.) <p>Write a SQL query to list all <code>Student</code> and <code>Faculty</code> members, showing the advisory relationship where applicable. Include students without advisors and faculty members who do not advise any students.</p>
F.	<p>NATURAL JOIN</p> <p>Write a SQL query that performs a NATURAL JOIN on the <code>Students</code> and <code>Faculty</code> tables. Assume there are columns with the exact same name that should be matched.</p>
G.	EQUI JOIN Write a SQL query to join the <code>Student</code> and <code>Faculty</code> tables on the faculty advisor ID and return the student's name along with the advisor's name and department.

Program No:

Title:

```
CREATE TABLE Student (
    StudentID INT PRIMARY KEY,
    Name VARCHAR(100),
    Major VARCHAR(100),
    FacultyAdvisorID INT,
    FOREIGN KEY (FacultyAdvisorID) REFERENCES Faculty(FacultyID)
);
```

```
CREATE TABLE Faculty (
    FacultyID INT PRIMARY KEY,
    Name VARCHAR(100),
    Department VARCHAR(100)
);
```

ENTRIES

-- Insert 4 entries into the Faculty table

```
INSERT INTO Faculty (FacultyID, Name, Department) VALUES
(101, 'Dr. Gupta', 'Physics'),
(102, 'Prof. Sharma', 'Chemistry'),
```

```
(103, 'Dr. Verma', 'Biology'),  
(104, 'Prof. Singh', 'Geology');
```

-- Insert 12 entries into the Student table

INSERT INTO Student (StudentID, Name, Major, FacultyAdvisorID) VALUES

```
(101, 'Amit Kumar', 'Physics', 101),  
(102, 'Deepika Singh', 'Chemistry', 102),  
(103, 'Rajesh Verma', 'Biology', 103),  
(104, 'Anjali Sharma', 'Geology', 104),  
(105, 'Suresh Gupta', 'Physics', 101),  
(106, 'Kritika Singh', 'Chemistry', 102),  
(107, 'Neeraj Verma', 'Biology', 103),  
(108, 'Riya Sharma', 'Geology', 104),  
(109, 'Rahul Kumar', 'Physics', 101),  
(110, 'Priya Singh', 'Chemistry', 102),  
(111, 'Manoj Verma', 'Biology', 103),  
(112, 'Anu Sharma', 'Geology', 104);
```

Output:

Show query box

✓ 12 rows inserted. (Query took 0.0005 seconds.)

```
INSERT INTO Student (StudentID, Name, Major, FacultyAdvisorID) VALUES (101, 'Amit Kumar', 'Physics', 101), (102, 'Deepika Singh', 'Chemistry', 102), (103, 'Rajesh  
Verma', 'Biology', 103), (104, 'Anjali Sharma', 'Geology', 104), (105, 'Suresh Gupta', 'Physics', 101), (106, 'Kritika Singh', 'Chemistry', 102), (107, 'Neeraj Verma',  
'Biology', 103), (108, 'Riya Sharma', 'Geology', 104), (109, 'Rahul Kumar', 'Physics', 101), (110, 'Priya Singh', 'Chemistry', 102), (111, 'Manoj Verma', 'Biology',  
103), (112, 'Anu Sharma', 'Geology', 104);
```

[Edit inline] [Edit] [Create PHP code]

Program No:B

Title:INNER JOIN

```
SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName
```

```
FROM Student s
```

```
INNER JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID;
```

Output:

Showing rows 0 - 11 (12 total, Query took 0.0004 seconds.)

```
SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName FROM Student s INNER JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID;
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

StudentName	FacultyAdvisorName
Amit Kumar	Dr. Gupta
Suresh Gupta	Dr. Gupta
Rahul Kumar	Dr. Gupta
Deepika Singh	Prof. Sharma
Kritika Singh	Prof. Sharma
Priya Singh	Prof. Sharma
Rajesh Verma	Dr. Verma
Neeraj Verma	Dr. Verma
Manoj Verma	Dr. Verma
Anjali Sharma	Prof. Singh
Riya Sharma	Prof. Singh
Anu Sharma	Prof. Singh

Program No:C

Title:LEFT JOIN

SELECT f.Name AS FacultyName, s.Name AS StudentName

FROM Faculty f

LEFT JOIN Student s ON f.FacultyID = s.FacultyAdvisorID;

Output:

Showing rows 0 - 11 (12 total, Query took 0.0003 seconds.)

```
SELECT f.Name AS FacultyName, s.Name AS StudentName FROM Faculty f LEFT JOIN Student s ON f.FacultyID = s.FacultyAdvisorID;
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

FacultyName	StudentName
Dr. Gupta	Amit Kumar
Dr. Gupta	Suresh Gupta
Dr. Gupta	Rahul Kumar
Prof. Sharma	Deepika Singh
Prof. Sharma	Kritika Singh
Prof. Sharma	Priya Singh
Dr. Verma	Rajesh Verma
Dr. Verma	Neeraj Verma
Dr. Verma	Manoj Verma
Prof. Singh	Anjali Sharma
Prof. Singh	Riya Sharma
Prof. Singh	Anu Sharma

Program No:D

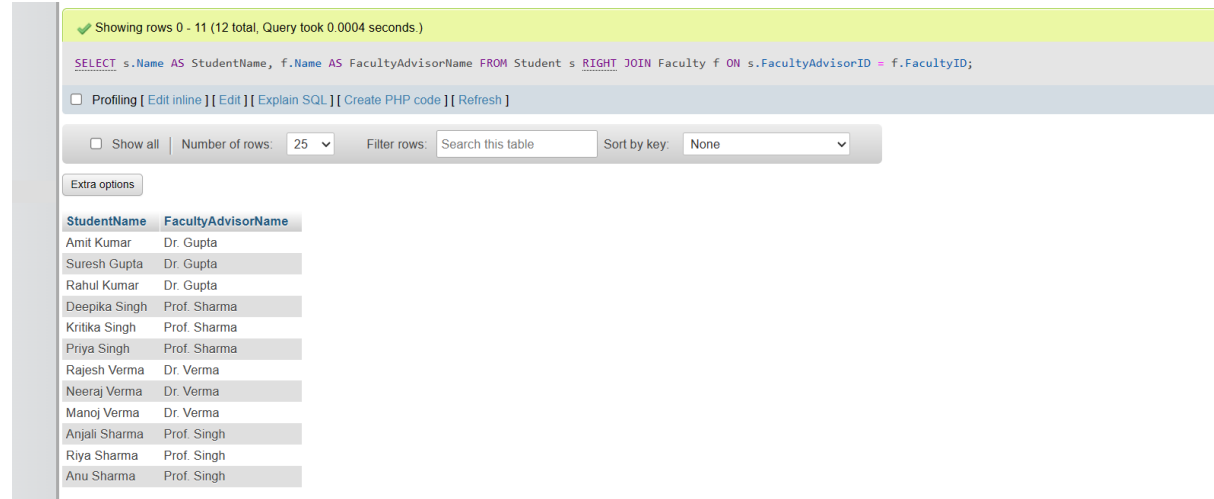
Title:Right JOIN

```
SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName
```

```
FROM Student s
```

```
RIGHT JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID;
```

Output:



The screenshot displays a database query interface. At the top, a green status bar indicates 'Showing rows 0 - 11 (12 total, Query took 0.0004 seconds.)'. Below this, the SQL query is shown: `SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName FROM Student s RIGHT JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID;`. A toolbar contains links for 'Profiling', 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'. Below the toolbar, there are controls for 'Show all', 'Number of rows' (set to 25), 'Filter rows' (a search box), and 'Sort by key' (set to None). An 'Extra options' button is also present. The main area shows a table with two columns: 'StudentName' and 'FacultyAdvisorName'. The table contains 12 rows of data, listing students and their corresponding faculty advisors.

StudentName	FacultyAdvisorName
Amit Kumar	Dr. Gupta
Suresh Gupta	Dr. Gupta
Rahul Kumar	Dr. Gupta
Deepika Singh	Prof. Sharma
Kritika Singh	Prof. Sharma
Priya Singh	Prof. Sharma
Rajesh Verma	Dr. Verma
Neeraj Verma	Dr. Verma
Manoj Verma	Dr. Verma
Anjali Sharma	Prof. Singh
Riya Sharma	Prof. Singh
Anu Sharma	Prof. Singh

Program No:E

Title:Full

```
SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName
```

```
FROM Student s
```

```
LEFT JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID
```

```
UNION
```

```
SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName
```

```
FROM Faculty f
```

```
LEFT JOIN Student s ON f.FacultyID = s.FacultyAdvisorID
```

```
WHERE s.Name IS NULL;
```

Output:

Showing rows 0 - 11 (12 total, Query took 0.0008 seconds.)

```
SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName FROM Student s LEFT JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID UNION SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName FROM Faculty f LEFT JOIN Student s ON f.FacultyID = s.FacultyAdvisorID WHERE s.Name IS NULL;
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

StudentName	FacultyAdvisorName
Amit Kumar	Dr. Gupta
Deepika Singh	Prof. Sharma
Rajesh Verma	Dr. Verma
Anjali Sharma	Prof. Singh
Suresh Gupta	Dr. Gupta
Kritika Singh	Prof. Sharma
Neeraj Verma	Dr. Verma
Riya Sharma	Prof. Singh
Rahul Kumar	Dr. Gupta
Priya Singh	Prof. Sharma
Manoj Verma	Dr. Verma
Anu Sharma	Prof. Singh

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Results operations

Console

Program No:F

Title:Nutral

SELECT *

FROM Student

NATURAL JOIN Faculty;

Output:

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0003 seconds.)

```
SELECT * FROM Student NATURAL JOIN Faculty;
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

Name	StudentID	Major	FacultyAdvisorID	FacultyID	Department
------	-----------	-------	------------------	-----------	------------

Query results operations

[Create view](#)

[Bookmark this SQL query](#)

Label: ☐ Let every user access this bookmark

[Bookmark this SQL query](#)

Program No:G

Title:Equi join

SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName, f.Department

FROM Student s

JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID;

Output:

✓ Showing rows 0 - 11 (12 total, Query took 0.0004 seconds.)

```
SELECT s.Name AS StudentName, f.Name AS FacultyAdvisorName, f.Department FROM Student s JOIN Faculty f ON s.FacultyAdvisorID = f.FacultyID;
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Sort by key: None

Extra options

StudentName	FacultyAdvisorName	Department
Amit Kumar	Dr. Gupta	Physics
Suresh Gupta	Dr. Gupta	Physics
Rahul Kumar	Dr. Gupta	Physics
Deepika Singh	Prof. Sharma	Chemistry
Kritika Singh	Prof. Sharma	Chemistry
Priya Singh	Prof. Sharma	Chemistry
Rajesh Verma	Dr. Verma	Biology
Neeraj Verma	Dr. Verma	Biology
Manoj Verma	Dr. Verma	Biology
Anjali Sharma	Prof. Singh	Geology
Riya Sharma	Prof. Singh	Geology
Anu Sharma	Prof. Singh	Geology

☐ Show all | Number of rows: 25 | Filter rows: Sort by key: None