

Chandigarh College of Engineering & Technology (Degree Wing)



Department of Computer Science and Engineering

Database Systems (Practical)

CS 352

Practical - 9

DOP: - 27/09/2024

DOS:- /10/2024

Submitted By:

Abhay Pratap Singh
(Roll No: - CO23306)

Submitted To:

Dr. Dheerendra Singh
CSE Department

Aim : - Implementation of at least five different SQL nested sub queries in from clause on each “Banking, University and database Project assigned to each student”, by running on WAMP/ LAMP /XAMPP /SQL server.

1. Query to get the total loan amounts per customer, joined with customer data

```
SELECT c.Customer_Name, c.Customer_Street, c.Customer_City, t.total_loan
FROM customer c
JOIN (
    SELECT b.customer_name, SUM(l.amount) AS total_loan
    FROM borrower b
    JOIN loan l ON b.loan_number = l.loan_number
    GROUP BY b.customer_name
) t ON c.Customer_Name = t.customer_name;
```

Customer_Name	Customer_Street	Customer_City	total_loan
Curry	North	Rye	500
Hayes	Main	Harrison	1500
Smith	North	Rye	2900
Williams	Nassaw	Princeton	1000
Adams	Spring	Pittsfield	1300

2. Query to find customers who have accounts in branches with assets greater than \$5,000,000

```
SELECT c.Customer_Name, c.Customer_Street, c.Customer_City, a.account_number
FROM customer c
JOIN (
    SELECT d.customer_name, a.account_number
    FROM depositor d
    JOIN account a ON d.account_number = a.account_number
    JOIN branch b ON a.branch_name = b.branch_name
    WHERE b.assets > 5000000
) a ON c.Customer_Name = a.customer_name;
```

Customer_Name	Customer_Street	Customer_City	account_number
Johnson	Alma	Palo Alto	A-201
Johnson	Alma	Palo Alto	A-101
Jones	Main	Harrison	A-217
Turner	Putnam	Stamford	A-305

3. Query to find the number of accounts per branch

```
SELECT b.branch_name, count(*) AS account_count
FROM branch b
JOIN (
    SELECT a.account_number, a.branch_name
    FROM account a
) a ON b.branch_name = a.branch_name
GROUP BY b.branch_name;
```

branch_name	account_count
Brighton	2
Downtown	1
Mianus	1
Perryridge	1
Redwood	1
Round Hill	1

4. Query to find customers who have loans at branches with assets less than \$2,000,000

```
SELECT c.Customer_Name, c.Customer_Street, c.Customer_City, l.loan_number
FROM customer c
JOIN (
    SELECT b.customer_name, l.loan_number
    FROM borrower b
    JOIN loan l ON b.loan_number = l.loan_number
    JOIN branch br ON l.branch_name = br.branch_name
    WHERE br.assets < 2000000
) l ON c.Customer_Name = l.customer_name;
```

Customer_Name	Customer_Street	Customer_City	loan_number
Curry	North	Rye	L-93
Hayes	Main	Harrison	L-15
Adams	Spring	Pittsfield	L-16

5. Query to find customers who have accounts but no loans

```

SELECT c.Customer_Name, c.Customer_Street, c.Customer_City
FROM customer c
JOIN (
    SELECT d.customer_name
    FROM depositor d
    LEFT JOIN borrower b ON d.customer_name = b.customer_name
    WHERE b.loan_number IS NULL
) t ON c.Customer_Name = t.customer_name;

```




















Customer_Name	Customer_Street	Customer_City
Johnson	Alma	Palo Alto
Johnson	Alma	Palo Alto
Jones	Main	Harrison
Lindsay	Park	Pittsfield
Turner	Putnam	Stamford

6. List the Names of Students Who Are Taking Courses in the 'Comp. Sci.' Department

```

SELECT s.name
FROM student s
WHERE s.ID IN (
    SELECT t.ID
    FROM takes t
    JOIN course c ON t.course_id = c.course_id
    WHERE c.dept_name = 'Comp. Sci.'
);

```













	name
<input type="checkbox"/>  Edit  Copy  Delete Zhang	
<input type="checkbox"/>  Edit  Copy  Delete Shankar	
<input type="checkbox"/>  Edit  Copy  Delete Levy	
<input type="checkbox"/>  Edit  Copy  Delete Williams	
<input type="checkbox"/>  Edit  Copy  Delete Brown	
<input type="checkbox"/>  Edit  Copy  Delete Bourikas	

7. Find the Average Salary of Instructors in Each Department

```

SELECT dept_name, AVG(salary) AS avg_salary
FROM instructor
WHERE dept_name IN (
    SELECT dept_name
    FROM department
    WHERE budget > 80000
)
GROUP BY dept_name;

```

				dept_name	avg_salary
<input type="checkbox"/>				Biology	96486.880000
<input type="checkbox"/>				Comp. Sci.	100591.256667
<input type="checkbox"/>				Elec. Eng.	105165.610000
<input type="checkbox"/>				Finance	109506.375000

8. Find the Students Who Have Taken the Most Courses in the Spring 2010 Semester

```

SELECT s.name, course_count
FROM student s
JOIN (
    SELECT t.ID, COUNT(*) AS course_count
    FROM takes t
    WHERE t.semester = 'Spring' AND t.year = 2010
    GROUP BY t.ID
    ORDER BY course_count DESC
    LIMIT 1
) AS max_courses ON s.ID = max_courses.ID;

```





name	course_count
Levy	2

9. List the Courses That Have No Students Enrolled in Any Section

```

SELECT c.title
FROM course c
WHERE NOT EXISTS (
    SELECT 1
    FROM section s
    LEFT JOIN takes t ON s.course_id = t.course_id AND s.sec_id = t.sec_id
    WHERE s.course_id = c.course_id
    AND t.ID IS NOT NULL
);

```

	title
<input type="checkbox"/>  Edit	 Copy
 Delete	Computational Biology

10. Find the Students Who Have Taken the Maximum Number of Credits in a Single Semester

```
SELECT s.name, t.semester, t.year, SUM(c.credits) AS total_credits
FROM student s
JOIN takes t ON s.ID = t.ID
JOIN course c ON t.course_id = c.course_id
GROUP BY s.ID, t.semester, t.year
HAVING SUM(c.credits) = (
    SELECT MAX(credit_sum)
    FROM (
        SELECT SUM(c.credits) AS credit_sum
        FROM takes t
        JOIN course c ON t.course_id = c.course_id
        GROUP BY t.ID, t.semester, t.year
    ) AS credit_totals
);
```

name	semester	year	total_credits
Zhang	Fall	2009	7
Shankar	Fall	2009	7
Levy	Spring	2010	7

11. Find the names of non-teaching staff who have the same department ID as the teaching staff with the highest experience months.

```
SELECT DISTINCT ns.NAME
FROM
    (SELECT DepartmentID
     FROM details_teaching_official
     ORDER BY EXPERIENCE_MONTHS DESC
     LIMIT 1) AS subquery
JOIN
    details_non_teaching_staff ns
ON
    subquery.DepartmentID = ns.DEPARTMENT_ID;
```

NAME

Mr. Gurpreet Singh

Mr. Maninder Singh

12. Find the maximum experience months of teaching staff in departments with more than 5 teaching staff.

```
SELECT MAX(t.EXPERIENCE_MONTHS)
FROM
(SELECT DEPARTMENT, EXPERIENCE_MONTHS
FROM details_teaching_official
WHERE DEPARTMENT IN
(SELECT DEPARTMENT
FROM details_teaching_official
GROUP BY DEPARTMENT
HAVING COUNT(NAME) > 5)) t;
```

MAX(t.EXPERIENCE_MONTHS)

360

13. Find the top 5 departments with the highest average experience months of teaching staff.

```
SELECT DEPARTMENT, AVG(t.EXPERIENCE_MONTHS)
FROM
(SELECT DEPARTMENT, EXPERIENCE_MONTHS
FROM details_teaching_official) t
GROUP BY DEPARTMENT
ORDER BY AVG(t.EXPERIENCE_MONTHS) DESC
LIMIT 5;
```

DEPARTMENT	AVG(t.EXPERIENCE_MONTHS) ▼ 1
Civil Engg.	229.1429
Computer Science Engg.	225.7000
Mechanical Engg.	224.3750
Electronics and Communications Engg.	217.5000
Applied Science	164.5000

14. Find the top 5 teaching staff with the highest experience months in the Mechanical department.

```
SELECT NAME, EXPERIENCE_MONTHS
FROM
(SELECT NAME, EXPERIENCE_MONTHS
 FROM details_teaching_official
 WHERE DEPARTMENT = 'Mechanical Engg.') t
ORDER BY EXPERIENCE_MONTHS DESC
LIMIT 5;
```

NAME	EXPERIENCE_MONTHS ▼ 1
Jatinder Madan	348
Radhey Sham	241
Vettivel S C	228
Ashwani Kumar	216
Nipun Sharma	213

15. Find the maximum experience months of teaching staff in departments with more than 5 teaching staff.

```
SELECT MAX(t.EXPERIENCE_MONTHS)
FROM
(SELECT DEPARTMENT, EXPERIENCE_MONTHS
 FROM details_teaching_official) t
WHERE t.DEPARTMENT IN
(SELECT DEPARTMENT
 FROM details_teaching_official
 GROUP BY DEPARTMENT
 HAVING COUNT(NAME) > 5);
```

MAX(t.EXPERIENCE_MONTHS)

360