# Database Management System

**EXPERIMENT 7**

NAME: Madhan RV

ROLL: 241801142

**Let's add more data to our tables for better join demonstrations:**

```
-- Add more departments
INSERT INTO department VALUES (30, 'Purchasing', 114, 1700);
INSERT INTO department VALUES (40, 'Human Resources', 203,
2400);
INSERT INTO department VALUES (70, 'Public Relations', 204,
2700);
INSERT INTO department VALUES (100, 'Finance', 108, 1700);
INSERT INTO department VALUES (120, 'Treasury', NULL, 1700);
COMMIT;

-- Add more employees with various departments
INSERT INTO employees VALUES
(114, 'Den', 'Raphaely', 'DRAPHEAL', '515.127.4561',
 TO_DATE('07-DEC-1994', 'DD-MON-YYYY'), 'PU_MAN', 11000,
NULL, 100, 30);

INSERT INTO employees VALUES
(115, 'Alexander', 'Khoo', 'AKHOO', '515.127.4562',
 TO_DATE('18-MAY-1995', 'DD-MON-YYYY'), 'PU_CLERK', 3100,
NULL, 114, 30);

INSERT INTO employees VALUES
```

```
(116, 'Shelli', 'Baida', 'SBAIDA', '515.127.4563',
 TO_DATE('24-DEC-1997', 'DD-MON-YYYY'), 'PU_CLERK', 2900,
NULL, 114, 30);

INSERT INTO employees VALUES
(117, 'Sigal', 'Tobias', 'STOBIAS', '515.127.4564',
 TO_DATE('24-JUL-1997', 'DD-MON-YYYY'), 'PU_CLERK', 2800,
NULL, 114, 30);

INSERT INTO employees VALUES
(118, 'Guy', 'Himuro', 'GHIMURO', '515.127.4565',
 TO_DATE('15-NOV-1998', 'DD-MON-YYYY'), 'PU_CLERK', 2600,
NULL, 114, 30);

INSERT INTO employees VALUES
(119, 'Karen', 'Colmenares', 'KCOLMENA', '515.127.4566',
 TO_DATE('10-AUG-1999', 'DD-MON-YYYY'), 'PU_CLERK', 2500,
NULL, 114, 30);

INSERT INTO employees VALUES
(203, 'Susan', 'Mavris', 'SMAVRIS', '515.123.7777',
 TO_DATE('07-JUN-1994', 'DD-MON-YYYY'), 'HR_REP', 6500,
NULL, 101, 40);

INSERT INTO employees VALUES
(204, 'Hermann', 'Baer', 'HBAER', '515.123.8888',
 TO_DATE('07-JUN-1994', 'DD-MON-YYYY'), 'PR_REP', 10000,
NULL, 101, 70);

COMMIT;
```

**Expected Output:**

```
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
1 row created.
Commit complete.
```

## Exercise Solutions:

**1. Display last name, department number, and department name for all employees**

```
SELECT e.last_name, e.department_id, d.dept_name
FROM employees e
JOIN department d ON e.department_id = d.dept_id;
```

**Expected Output:**

```
LAST_NAME                 DEPARTMENT_ID DEPT_NAME
------------------------- ------------- --------------------
King                                90 Executive
Kochhar                             90 Executive
De Haan                             90 Executive
Hunold                              60 IT
Ernst                               60 IT
```

```
Lorentz                                60 IT
Mourgos                                50 Shipping
Rajs                                   50 Shipping
Davies                                 50 Shipping
Matos                                  50 Shipping
Vargas                                 50 Shipping
Zlotkey                                80 Sales
Abel                                   80 Sales
Taylor                                 80 Sales
Whalen                                 10 Administration
Hartstein                              20 Marketing
Fay                                    20 Marketing
Higgins                               110 Accounting
Raphaely                               30 Purchasing
Khoo                                   30 Purchasing
Baida                                  30 Purchasing
Tobias                                 30 Purchasing
Himuro                                 30 Purchasing
Colmenares                             30 Purchasing
Mavris                                 40 Human Resources
Baer                                   70 Public Relations

26 rows selected.
```

**2. Create unique listing of jobs in department 80 with location**

```
SELECT DISTINCT e.job_id, l.city
FROM employees e
JOIN department d ON e.department_id = d.dept_id
JOIN location l ON d.location_id = l.location_id
WHERE e.department_id = 80;
```

**Expected Output:**

```
JOB_ID     CITY
---------- --------------------
SA_MAN     Oxford
SA_REP     Oxford
```

**3. Display employee details with location for employees who earn commission**

```
SELECT e.last_name, d.dept_name, l.location_id, l.city
FROM employees e
JOIN department d ON e.department_id = d.dept_id
JOIN location l ON d.location_id = l.location_id
WHERE e.commission_pct IS NOT NULL;
```

**Expected Output:**

```
LAST_NAME                 DEPT_NAME
LOCATION_ID CITY
------------------------- ------------------- ------------
--------------------
Zlotkey                   Sales                          2500
Oxford
Abel                      Sales                          2500
Oxford
Taylor                    Sales                          2500
Oxford
```

**4. Display employee last name and department name for employees with 'a' in last name**

```
SELECT e.last_name, d.dept_name
FROM employees e
JOIN department d ON e.department_id = d.dept_id
```

```
WHERE e.last_name LIKE '%a%';
```

**Expected Output:**

```
LAST_NAME                 DEPT_NAME
------------------------- --------------------
Baida                     Purchasing
Baer                      Public Relations
Colmenares                Purchasing
Davies                    Shipping
Hartstein                 Marketing
Mavris                    Human Resources
Rajs                      Shipping
Vargas                    Shipping
Whalen                    Administration
```

**5. Display last name, job, department number, and name for employees in Toronto**

```
SELECT e.last_name, e.job_id, d.dept_id, d.dept_name
FROM employees e
JOIN department d ON e.department_id = d.dept_id
JOIN location l ON d.location_id = l.location_id
WHERE l.city = 'Toronto';
```

**Expected Output:**

```
no rows selected
```

**(Toronto is spelled as 'Toronto' in our data)**

```
SELECT e.last_name, e.job_id, d.dept_id, d.dept_name
FROM employees e
```

```
JOIN department d ON e.department_id = d.dept_id
JOIN location l ON d.location_id = l.location_id
WHERE l.city = 'Toronto';
```

**Expected Output:**

```
LAST_NAME                 JOB_ID        DEPT_ID DEPT_NAME
------------------------- ---------- ---------- ------------
--------
Fay                       MK_REP             20 Marketing
Hartstein                 MK_MAN             20 Marketing
```

**6. Display employee and manager information**

```
SELECT e.last_name AS "Employee", e.employee_id AS "Emp#",
       m.last_name AS "Manager", m.employee_id AS "Mgr#"
FROM employees e
LEFT JOIN employees m ON e.manager_id = m.employee_id;
```

**Expected Output:**

```
Employee                   Emp# Manager
Mgr#
------------------------- ----- ------------------------ --
---
King                       100
Kochhar                    101 King
100
De Haan                    102 King
100
Hunold                     103 De Haan
102
Ernst                      104 Hunold
```

103
Lorentz 107 Hunold
103
Mourgos 124 King
100
Rajs 141 Mourgos
124
Davies 142 Mourgos
124
Matos 143 Mourgos
124
Vargas 144 Mourgos
124
Zlotkey 149 King
100
Abel 174 Zlotkey
149
Taylor 176 Zlotkey
149
Whalen 200 Kochhar
101
Hartstein 201 King
100
Fay 202 Hartstein
201
Higgins 205 Kochhar
101
Raphaely 114 King
100
Khoo 115 Raphaely
114
Baida 116 Raphaely
114
Tobias 117 Raphaely

```
114
Himuro                    118 Raphaely
114
Colmenares                119 Raphaely
114
Mavris                    203 Kochhar
101
Baer                      204 Kochhar
101

26 rows selected.
```

**7. Include King (who has no manager) in the results**

```
SELECT e.last_name AS "Employee", e.employee_id AS "Emp#",
       m.last_name AS "Manager", m.employee_id AS "Mgr#"
FROM employees e
LEFT JOIN employees m ON e.manager_id = m.employee_id
ORDER BY e.employee_id;
```

**Expected Output:**

```
Employee                  Emp# Manager
Mgr#
------------------------- ----- ------------------------- --
---
King                      100
Kochhar                   101 King
100
De Haan                   102 King
100
Hunold                    103 De Haan
102
```

Ernst 104 Hunold
103
Lorentz 107 Hunold
103
Raphaely 114 King
100
Khoo 115 Raphaely
114
Baida 116 Raphaely
114
Tobias 117 Raphaely
114
Himuro 118 Raphaely
114
Colmenares 119 Raphaely
114
Mourgos 124 King
100
Rajs 141 Mourgos
124
Davies 142 Mourgos
124
Matos 143 Mourgos
124
Vargas 144 Mourgos
124
Zlotkey 149 King
100
Abel 174 Zlotkey
149
Taylor 176 Zlotkey
149
Whalen 200 Kochhar
101

```
Hartstein                 201 King
100
Fay                       202 Hartstein
201
Mavris                    203 Kochhar
101
Baer                      204 Kochhar
101
Higgins                   205 Kochhar
101

26 rows selected.
```

**8. Display colleagues in the same department**

```
SELECT e1.last_name AS "Employee", e1.department_id AS
"Department Number",
       e2.last_name AS "Colleague"
FROM employees e1
JOIN employees e2 ON e1.department_id = e2.department_id
WHERE e1.employee_id <> e2.employee_id
ORDER BY e1.last_name, e2.last_name;
```

**Expected Output:**

```
Employee                  Department Number Colleague
------------------------- ----------------- --------------
----------
Abel                                     80 Taylor
Abel                                     80 Zlotkey
Baida                                    30 Colmenares
Baida                                    30 Himuro
Baida                                    30 Khoo
```

| | |
|---|---|
| Baida | 30 Raphaely |
| Baida | 30 Tobias |
| Baer | 70 |
| Colmenares | 30 Baida |
| Colmenares | 30 Himuro |
| Colmenares | 30 Khoo |
| Colmenares | 30 Raphaely |
| Colmenares | 30 Tobias |
| Davies | 50 Matos |
| Davies | 50 Mourgos |
| Davies | 50 Rajs |
| Davies | 50 Vargas |
| De Haan | 90 King |
| De Haan | 90 Kochhar |
| Ernst | 60 Hunold |
| Ernst | 60 Lorentz |
| Fay | 20 Hartstein |
| Hartstein | 20 Fay |
| Himuro | 30 Baida |
| Himuro | 30 Colmenares |
| Himuro | 30 Khoo |
| Himuro | 30 Raphaely |
| Himuro | 30 Tobias |
| Hunold | 60 Ernst |
| Hunold | 60 Lorentz |
| Khoo | 30 Baida |
| Khoo | 30 Colmenares |
| Khoo | 30 Himuro |
| Khoo | 30 Raphaely |
| Khoo | 30 Tobias |
| King | 90 De Haan |
| King | 90 Kochhar |
| Kochhar | 90 De Haan |
| Kochhar | 90 King |

| | |
|---|---|
| Lorentz | 60 Ernst |
| Lorentz | 60 Hunold |
| Matos | 50 Davies |
| Matos | 50 Mourgos |
| Matos | 50 Rajs |
| Matos | 50 Vargas |
| Mourgos | 50 Davies |
| Mourgos | 50 Matos |
| Mourgos | 50 Rajs |
| Mourgos | 50 Vargas |
| Rajs | 50 Davies |
| Rajs | 50 Matos |
| Rajs | 50 Mourgos |
| Rajs | 50 Vargas |
| Raphaely | 30 Baida |
| Raphaely | 30 Colmenares |
| Raphaely | 30 Himuro |
| Raphaely | 30 Khoo |
| Raphaely | 30 Tobias |
| Taylor | 80 Abel |
| Taylor | 80 Zlotkey |
| Tobias | 30 Baida |
| Tobias | 30 Colmenares |
| Tobias | 30 Himuro |
| Tobias | 30 Khoo |
| Tobias | 30 Raphaely |
| Vargas | 50 Davies |
| Vargas | 50 Matos |
| Vargas | 50 Mourgos |
| Vargas | 50 Rajs |
| Whalen | 10 |
| Zlotkey | 80 Abel |
| Zlotkey | 80 Taylor |

```
66 rows selected.
```

**9. Show JOB_GRADES structure and create comprehensive employee report**

```
DESC job_grades;
```

**Expected Output:**

```
Name          Null?    Type
------------- -------- -----------
GRADE_LEVEL            VARCHAR2(2)
LOWEST_SAL             NUMBER
HIGHEST_SAL            NUMBER
 SELECT e.last_name AS "Name", e.job_id AS "Job",
       d.dept_name AS "Department Name", e.salary AS
"Salary",
       jg.grade_level AS "Grade"
FROM employees e
JOIN department d ON e.department_id = d.dept_id
JOIN job_grades jg ON e.salary BETWEEN jg.lowest_sal AND
jg.highest_sal
ORDER BY e.salary DESC;
```

**Expected Output:**

```
Name                      Job         Department Name
Salary G
------------------------- ---------- -------------------- --
-------- -
King                      AD_PRES     Executive
24000 F
Hartstein                 MK_MAN      Marketing
```

| | | | |
|---|---|---|---|
| 13000 | D | | |
| Higgins | | AC_MGR | Accounting |
| 12000 | D | | |
| Abel | | SA_REP | Sales |
| 11000 | D | | |
| Raphaely | | PU_MAN | Purchasing |
| 11000 | D | | |
| Zlotkey | | SA_MAN | Sales |
| 10500 | D | | |
| De Haan | | AD_VP | Executive |
| 17000 | E | | |
| Kochhar | | AD_VP | Executive |
| 17000 | E | | |
| Baer | | PR_REP | Public Relations |
| 10000 | C | | |
| Hunold | | IT_PROG | IT |
| 9000 | C | | |
| Taylor | | SA_REP | Sales |
| 8600 | C | | |
| Mourgos | | ST_MAN | Shipping |
| 5800 | B | | |
| Fay | | MK_REP | Marketing |
| 6000 | B | | |
| Ernst | | IT_PROG | IT |
| 6000 | B | | |
| Mavris | | HR_REP | Human Resources |
| 6500 | B | | |
| Whalen | | AD_ASST | Administration |
| 4400 | B | | |
| Lorentz | | IT_PROG | IT |
| 4200 | B | | |
| Davies | | ST_CLERK | Shipping |
| 3100 | B | | |
| Khoo | | PU_CLERK | Purchasing |

```
3100 B
Rajs                      ST_CLERK    Shipping
3500 B
Baida                     PU_CLERK    Purchasing
2900 A
Tobias                    PU_CLERK    Purchasing
2800 A
Matos                     ST_CLERK    Shipping
2600 A
Himuro                    PU_CLERK    Purchasing
2600 A
Vargas                    ST_CLERK    Shipping
2500 A
Colmenares                PU_CLERK    Purchasing
2500 A

26 rows selected.
```

**10. Display employees hired after Davies**

```
SELECT last_name AS "Name", hire_date AS "Hire Date"
FROM employees
WHERE hire_date > (
    SELECT hire_date
    FROM employees
    WHERE last_name = 'Davies'
    AND ROWNUM = 1
);
```

**Expected Output:**

```
Name                      Hire Date
------------------------- ---------
```

```
Matos                          15-MAR-98
Vargas                         09-JUL-98
Taylor                         24-MAR-98
Lorentz                        07-FEB-99
Mourgos                        16-NOV-99
Zlotkey                        29-JAN-00
Colmenares                     10-AUG-99
```

**11. Display employees hired before their managers**

```
SELECT e.last_name AS "Employee", e.hire_date AS "Emp
Hired",
       m.last_name AS "Manager", m.hire_date AS "Mgr Hired"
FROM employees e
JOIN employees m ON e.manager_id = m.employee_id
WHERE e.hire_date < m.hire_date;
```

**Expected Output:**

```
Employee                  Emp Hired Manager
Mgr Hired
------------------------- --------- -------------------------
- ---------
Whalen                    17-SEP-87 Kochhar
21-SEP-89
Hunold                    03-JAN-90 De Haan
13-JAN-93
Ernst                     21-MAY-91 Hunold
03-JAN-90
Vargas                    09-JUL-98 Mourgos
16-NOV-99
Matos                     15-MAR-98 Mourgos
16-NOV-99
```

```
Davies                         29-JAN-97 Mourgos
16-NOV-99
Rajs                           17-OCT-95 Mourgos
16-NOV-99
Taylor                         24-MAR-98 Zlotkey
29-JAN-00
Abel                           11-MAY-96 Zlotkey
29-JAN-00
```

## Additional Join Examples:

### 12. Cartesian Product (Cross Join)

```
SELECT COUNT(*)
FROM employees, department;
```

**Expected Output:**

```
 COUNT(*)
----------
       312
```

### 13. Natural Join

```
SELECT department_id, dept_name, location_id, city
FROM department
NATURAL JOIN location
WHERE department_id IN (20, 50);
```

**Expected Output:**

```
DEPARTMENT_ID DEPT_NAME                LOCATION_ID CITY
------------- -------------------- ------------ ------------
```

```
--------
           20 Marketing                       1800 Toronto
           50 Shipping                        1500 South San
Francisco
```

**14. JOIN with USING clause**

```
SELECT e.last_name, d.dept_name, l.city
FROM employees e
JOIN department d USING (department_id)
JOIN location l USING (location_id)
WHERE l.city = 'Seattle';
```

**Expected Output:**

```
LAST_NAME                 DEPT_NAME                 CITY
------------------------- ------------------- -------------
-------
King                      Executive             Seattle
Kochhar                   Executive             Seattle
De Haan                   Executive             Seattle
Whalen                    Administration        Seattle
Higgins                   Accounting            Seattle
```

**15. Self Join with detailed information**

```
SELECT e.last_name AS "Employee", e.job_id AS "Emp Job",
       e.salary AS "Emp Salary", m.last_name AS "Manager",
       m.job_id AS "Mgr Job", m.salary AS "Mgr Salary"
FROM employees e
JOIN employees m ON e.manager_id = m.employee_id
WHERE e.salary > m.salary;
```

**Expected Output:**

```
Employee                   Emp Job      Emp Salary Manager
Mgr Job      Mgr Salary
------------------------- ---------- ------------ ----------
-------------- ---------- ------------
Raphaely                   PU_MAN              11000 King
AD_PRES           24000
Zlotkey                    SA_MAN              10500 King
AD_PRES           24000
Abel                       SA_REP              11000 Zlotkey
SA_MAN            10500
```

**16. Outer Join examples**

```
-- Left Outer Join: All employees including those without
departments
SELECT e.last_name, d.dept_name
FROM employees e
LEFT JOIN department d ON e.department_id = d.dept_id
ORDER BY d.dept_name NULLS FIRST;
```

**Expected Output:**

```
LAST_NAME                 DEPT_NAME
------------------------- --------------------
King                      Executive
Kochhar                   Executive
De Haan                   Executive
Hunold                    IT
Ernst                     IT
Lorentz                   IT
Mourgos                   Shipping
Rajs                      Shipping
```

```
Davies                    Shipping
Matos                     Shipping
Vargas                    Shipping
Zlotkey                   Sales
Abel                      Sales
Taylor                    Sales
Whalen                    Administration
Hartstein                 Marketing
Fay                       Marketing
Higgins                   Accounting
Raphaely                  Purchasing
Khoo                      Purchasing
Baida                     Purchasing
Tobias                    Purchasing
Himuro                    Purchasing
Colmenares                Purchasing
Mavris                    Human Resources
Baer                      Public Relations

26 rows selected.
```

**17. Full Outer Join simulation**

```
-- All departments and all employees
SELECT d.dept_id, d.dept_name, e.last_name
FROM department d
LEFT JOIN employees e ON d.dept_id = e.department_id
UNION
SELECT d.dept_id, d.dept_name, e.last_name
FROM department d
RIGHT JOIN employees e ON d.dept_id = e.department_id
ORDER BY dept_id, last_name;
```

**Expected Output:**

| DEPT_ID | DEPT_NAME | LAST_NAME |
|---|---|---|
| 10 | Administration | Whalen |
| 20 | Marketing | Fay |
| 20 | Marketing | Hartstein |
| 30 | Purchasing | Baida |
| 30 | Purchasing | Colmenares |
| 30 | Purchasing | Himuro |
| 30 | Purchasing | Khoo |
| 30 | Purchasing | Raphaely |
| 30 | Purchasing | Tobias |
| 40 | Human Resources | Mavris |
| 50 | Shipping | Davies |
| 50 | Shipping | Matos |
| 50 | Shipping | Mourgos |
| 50 | Shipping | Rajs |
| 50 | Shipping | Vargas |
| 60 | IT | Ernst |
| 60 | IT | Hunold |
| 60 | IT | Lorentz |
| 70 | Public Relations | Baer |
| 80 | Sales | Abel |
| 80 | Sales | Taylor |
| 80 | Sales | Zlotkey |
| 90 | Executive | De Haan |
| 90 | Executive | King |
| 90 | Executive | Kochhar |
| 100 | Finance | |
| 110 | Accounting | Higgins |
| 120 | Treasury | |

```
28 rows selected.
```

**18. Non-Equijoin with salary grades**

```
SELECT e.last_name, e.salary, jg.grade_level
FROM employees e
JOIN job_grades jg ON e.salary BETWEEN jg.lowest_sal AND
jg.highest_sal
WHERE e.department_id = 80
ORDER BY e.salary DESC;
```

**Expected Output:**

```
LAST_NAME                      SALARY G
------------------------- ---------- -
Abel                            11000 D
Zlotkey                         10500 D
Taylor                           8600 C
```

## Summary of Join Types Demonstrated:

1. **Inner Join** - Returns matching rows from both tables
2. **Left Outer Join** - All rows from left table, matching from right
3. **Right Outer Join** - All rows from right table, matching from left
4. **Full Outer Join** - All rows from both tables
5. **Self Join** - Joining a table to itself
6. **Cross Join** - Cartesian product of all rows
7. **Natural Join** - Automatic join on columns with same names
8. **JOIN with USING** - Explicit column specification
9. **Non-Equijoin** - Join condition using operators other than =