**GANESHAN.M**

**230701514**

| Ex.No.: 8 | | WORKING WITH MULTIPLE TABLES |
|---|---|---|
| Date: | 23.08.2024 | |

1. Write a query to display the last name, department number, and department name for all employees.

SELECT e.last_name, e.department_id, d.department_name FROM employees e JOIN departments d ON e.department_id = d.department_id;

| LAST_NAME | DEPARTMENT_ID | DEPARTMENT_NAME |
|---|---|---|
| Miller | 10 | Admin |
| Andrea | 10 | Admin |
| Davis | 20 | ST_CLERK |
| Taylor | 20 | ST_CLERK |
| Matos | 50 | IT |
| Johnson | 50 | IT |
| Austin | 50 | IT |
| Thomas | 60 | ST_CLERK |
| Smith | 70 | Customer Service |
| Wilson | 80 | ST_CLERK |

2. Create a unique listing of all jobs that are in department 80. Include the location of the department in the output.

SELECT DISTINCT e.job_id, d.location_id FROM employees e JOIN departments d ON e.department_id = d.department_id WHERE e.department_id = 80;

| JOB_ID | LOCATION_ID |
|---|---|
| SA_REP | 1007 |

**GANESHAN.M**

**230701514**

3. Write a query to display the employee last name, department name, location ID, and city of all employees who earn a commission

```
SELECT e.last_name, d.department_name, d.location_id, l.city FROM employees e JOIN
departments d ON e.department_id = d.department_id JOIN locations l ON d.location_id =
l.location_id WHERE e.commission_pct IS NOT NULL;
```

| LAST_NAME | DEPARTMENT_NAME | LOCATION_ID | CITY |
|---|---|---|---|
| Johnson | IT | 1004 | London |
| Thomas | ST_CLERK | 1005 | Sydney |
| Wilson | ST_CLERK | 1007 | Dubai |

4. Display the employee last name and department name for all employees who have an a(lowercase) in their last names. P

```
SELECT e.last_name, d.department_name FROM employees e JOIN departments d ON
e.department_id = d.department_id WHERE e.last_name LIKE '%a%';
```

| LAST_NAME | DEPARTMENT_NAME |
|---|---|
| Matos | IT |
| Davis | ST_CLERK |
| Andrea | Admin |
| Taylor | ST_CLERK |
| Thomas | ST_CLERK |

**GANESHAN.M**

**230701514**

5. Write a query to display the last name, job, department number, and department name for all employees who work in Toronto.

SELECT e.last_name, e.job_id, e.department_id, d.department_name FROM employees e JOIN departments d ON e.department_id = d.department_id JOIN locations l ON d.location_id = l.location_id WHERE l.city = 'Toronto';

| LAST_NAME | JOB_ID | DEPARTMENT_ID | DEPARTMENT_NAME |
|---|---|---|---|
| Andrea | IT_PROG | 10 | Admin |
| Miller | ST_CLERK | 10 | Admin |

6. Display the employee last name and employee number along with their manager‘s last name and manager number. Label the columns Employee, Emp#, Manager, and Mgr#, Respectively

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;

| EMPLOYEE | EMP# | MANAGER | MGR# |
|---|---|---|---|
| Andrea | 107 | Matos | 101 |
| Davis | 104 | Matos | 101 |
| Smith | 176 | Matos | 101 |
| Wilson | 106 | Johnson | 103 |
| Thomas | 110 | Miller | 105 |
| Silva | 210 | - | - |
| Wei | 209 | - | - |
| Tanaka | 208 | - | - |
| Wilson | 207 | - | - |
| Miller | 206 | - | - |

**GANESHAN.M**

**230701514**

7. Modify lab4_6.sql to display all employees including King, who has no manager. Order the results by the employee number.

SELECT e.last_name, e.employee_id, m.last_name AS Manager FROM employees e LEFT JOIN employees m ON e.manager_id = m.employee_id ORDER BY e.employee_id;

| LAST_NAME | EMPLOYEE_ID | MANAGER |
|---|---|---|
| Matos | 101 | - |
| Johnson | 103 | - |
| Davis | 104 | Matos |
| Miller | 105 | - |
| Wilson | 106 | Johnson |
| Andrea | 107 | Matos |
| Taylor | 108 | - |
| Austin | 109 | - |
| Thomas | 110 | Miller |
| Smith | 176 | Matos |

8. Create a query that displays employee last names, department numbers, and all the employees who work in the same department as a given employee. Give each column an appropriate label

SELECT e1.last_name AS Employee, e2.last_name AS Colleague FROM employees e1 JOIN employees e2 ON e1.department_id = e2.department_id WHERE e1.employee_id = :employee_id;

| EMPLOYEE | COLLEAGUE |
|---|---|
| Matos | Matos |
| Matos | Johnson |
| Matos | Austin |

**GANESHAN.M**

**230701514**

9. Show the structure of the JOB_GRADES table. Create a query that displays the name, job, department name, salary, and grade for all employees

DESC job_grades;

| Table | Column | Data Type | Length | Precision | Scale | Primary Key | Nullable | Default | Comment |
|---|---|---|---|---|---|---|---|---|---|
| JOB_GRADES | GRADE_LEVEL | VARCHAR2 | 2 | - | - | - | ✓ | - | - |
| | LOWEST_SAL | NUMBER | 22 | - | - | - | ✓ | - | - |
| | HIGHEST_SAL | NUMBER | 22 | - | - | - | ✓ | - | - |
| | DEPTNO | NUMBER | 22 | - | - | - | ✓ | - | - |

SELECT e.last_name, e.job_id, d.department_name, e.salary, j.grade_level FROM employees e JOIN departments d ON e.department_id = d.department_id JOIN job_grades j ON e.salary BETWEEN j.lowest_sal AND j.highest_sal;

| LAST_NAME | JOB_ID | DEPARTMENT_NAME | SALARY | GRADE_LEVEL |
|---|---|---|---|---|
| Davis | AC_ACCOUNT | ST_CLERK | 15000 | G2 |
| Wilson | SA_REP | ST_CLERK | 13500 | G1 |
| Smith | HR_REP | Customer Service | 12500 | F2 |
| Johnson | SA_MAN | IT | 7200 | D1 |
| Austin | AC_MGR | IT | 7100 | D1 |
| Miller | ST_CLERK | Admin | 6200 | C2 |
| Matos | IT_PROG | IT | 6000 | C1 |
| Thomas | ST_CLERK | ST_CLERK | 5300 | C1 |
| Taylor | HR_REP | ST_CLERK | 4600 | B2 |

**GANESHAN.M**

**230701514**

10. Create a query to display the name and hire date of any employee hired after employee Davies.

SELECT last_name, hire_date FROM employees WHERE hire_date > (SELECT hire_date FROM employees WHERE last_name = 'Davies');

| LAST_NAME | HIRE_DATE |
|---|---|
| Smith | 02/20/2019 |
| Johnson | 03/01/1998 |
| Davis | 01/01/1998 |
| Miller | 07/25/2018 |
| Wilson | 03/12/2022 |
| Andrea | 11/05/2017 |
| Taylor | 12/15/2019 |
| Austin | 08/22/2021 |
| Thomas | 04/01/2020 |
| Doe | 10/10/2015 |

11. Display the names and hire dates for all employees who were hired before their managers, along with their manager‘s names and hire dates. Label the columns Employee, Emp Hired, Manager, and Mgr Hired, respectively.

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;

| EMPLOYEE | EMP_HIRED | MANAGER | MGR_HIRED |
|---|---|---|---|
| Smith | 02/20/2019 | Matos | 01/01/1994 |
| Davis | 01/01/1998 | Matos | 01/01/1994 |
| Andrea | 11/05/2017 | Matos | 01/01/1994 |
| Wilson | 03/12/2022 | Johnson | 03/01/1998 |
| Thomas | 04/01/2020 | Miller | 07/25/2018 |