EXERCISE-7

Displaying data from multiple tables

Find the Solution for the following:

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;
```

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

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

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;
```

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 LOWER(e.last_name) LIKE '%a%';
```

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, j.job_title, e.department_id, d.department_name FROM employees e
JOIN jobs j ON e.job_id = j.job_id JOIN departments d ON e.department_id = d.department_id
JOIN locations l ON d.location_id = l.location_id WHERE l.city = 'Toronto';
```

6. Display the employee last name and employee number along with their manager‘s last name and 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;
```

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 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;
```

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, e1.department_id AS Dept#, e2.last_name AS Colleague
FROM employees e1
JOIN employees e2 ON e1.department_id = e2.department_id
ORDER BY e1.department_id, e1.last_name, e2.last_name;
```

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;
SELECT e.last_name AS Name,
    e.job_id AS Job,
    d.department_name AS Department,
    e.salary AS Salary,
    j.grade_level AS Grade
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;
```

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

```
SELECT e.last_name AS Name, e.hire_date AS Hire_Date FROM employees e
WHERE e.hire_date > (SELECT hire_date FROM employees WHERE last_name = 'Davies');
```

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;
```