This query was completed in earlier releases as follows:

```
SELECT e.last_name, e.department_id, d.department_name
FROM   employees e, departments d
WHERE  d.department_id = e.department_id  (+);
```

FULL OUTER JOIN
Example:

```
SELECT e.last_name, e.department_id, d.department_name
FROM   employees e
FULL OUTER JOIN departments d
ON   (e.department_id = d.department_id) ;
```

This query retrieves all rows in the EMPLOYEES table, even if there is no match in the DEPARTMENTS table. It alslso retrieves all rows in the DEPARTMENTS table, even if there is no match in the EMPLOYEES table.

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 e.job_id , d.location_id from employees e 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 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. 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%';

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

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.departmentid 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 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 employeesm 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_id, e2.last_name as colleague from employees e1 Join employees e2 on e.department_id = e2.department_id where e1.employee_id != e2.employee_id order by e1.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, j.job_title, d.department_name, e.salary, g.grade_level from employees e JOIN jobs j on e.job_id = j.job_id JOIN departments d on e.department_id = d.department_id JOIN job_grades g on e.salary BETWEEN g.lowest_sal AND g.highest_sal;

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

```
Select e.last_name, e.hire_date from employees e where e.hire_date > ( select
hiredate 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 EmpHired, m.last_name as
manager and m.hire_date as ManagerHired From Employees e JOIN employees m
On e.manager_id = m.employee_id where e.hire_date < m.hire_date;
```

Evaluation Procedure	Marks awarded
Query(5)	5
Execution (5)	5
Viva(5)	5
Total (15)	15
Faculty Signature	