(85)

5. The HR department needs a report with the following specifications:

- Last name and department ID of all the employees from the EMPLOYEES table, regardless of whether or not they belong to a department.

- Department ID and department name of all the departments from the DEPARTMENTS table, regardless of whether or not they have employees working in them Write a compound query to accomplish this.

```
SELECT last_name AS name, department_id
FROM employees
UNION
SELECT department.name AS name, department_id
FROM departments;
```

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

• The number of columns and the data types of the columns being selected by the SELECT statements in the queries must be identical in all the SELECT statements used in the query. The names of the columns need not be identical.
• All of the columns in the WHERE clause must be in the SELECT clause for the MINUS operator to work.

**Example:**

Display the employee IDs of those employees who have not changed their jobs even once.

```
SELECT employee_id,job_id
FROM employees
MINUS
SELECT employee_id,job_id
FROM job_history;
```

**Find the Solution for the following:**

1. The HR department needs a list of department IDs for departments that do not contain the job ID ST_CLERK. Use set operators to create this report.

```
SELECT DISTINCT department_id
FROM employees
WHERE department_id NOT IN ( SELECT DISTINCT department_id
FROM employees
WHERE job_id = 'ST_CLERK' );
```

2. The HR department needs a list of countries that have no departments located in them. Display the country ID and the name of the countries. Use set operators to create this report.

```
Select country_id , country_name
Fwom Countries
where country_id NOT IN ( Select distinct l.country_id
FROM locations l
JOIN departments d ON l.location_id = d.location_id
```

3. Produce a list of jobs for departments 10, 50, and 20, in that order. Display job ID and department ID using set operators.

```
select job_id, department_id
FROM employees
where department_id =10
UNION where department_id =50
UNION where department_id = 20 ;
```

4. Create a report that lists the employee IDs and job IDs of those employees who currently have a job title that is the same as their job title when they were initially hired by the company (that is, they changed jobs but have now gone back to doing their original job).

```
SELECT e.employee_id , e.job_id
FROM employee e
WHERE (e.employee_id , e.job_id) IN (
    SELECT employee_id, job_id
    FROM job_history );
```