Ex.No.: 7		USING SET OPERATORS
Date:	27/8/24	

Objectives

After the completion this exercise, the students should be able to do the following:

- Describe set operators
- Use a set operator to combine multiple queries into a single query
- Control the order of rows returned

The set operators combine the results of two or more component queries into one result.

Queries containing set operators are called *compound queries*.

Operator	Returns
UNION	All distinct rows selected by either query
UNION ALL	All rows selected by either query, including all duplicates
INTERSECT	All distinct rows selected by both queries
MINUS	All distinct rows that are selected by the first SELECT statement and not selected in the second SELECT statement

The tables used in this lesson are:

- EMPLOYEES: Provides details regarding all current employees

- JOB_HISTORY: Records the details of the start date and end date of the former job, and the job identification number and department when an employee switches jobs

UNION Operator

Guidelines

- The number of columns and the data types of the columns being selected must be identical in all the SELECT statements used in the query. The names of the columns need not be identical.

- UNION operates over all of the columns being selected.

- NULL values are not ignored during duplicate checking.

- The IN operator has a higher precedence than the UNION operator.

1) Select department_id
From employees
Minus
Select department_id
From employees
where upper (job_id) = upper ('ST_CLERK')
Order by 1;

2) Select country_id, country_name
From countries
Minus
Select country_id, country_name
From Countries c
join locations l
Using (country_id)
join departments d
Using (location_id)
where department_id is not null;

3) Select distinct job_id, department_id
From employees
where department_id = 10
Union all
Select distinct job_id, department_id
From employees
where department_id = 50
Union all
Select distinct job_id, department_id
From employees
where department_id = 20;

4) Select employee_id, job_id
From employees
Intersect
Select employee_id, job_id
From job_history
Order by 1;

5) Select last_name, department_id, To_char('null')
From employees
Union
select to_char('null'), department_id, department_name
From departments
order by 1;

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

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).

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.

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