

EXERCISE-10

USING THE SET OPERATORS

Name: Vedhasree S
Register Number: 240701580
Department: CSE

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.

The screenshot shows the APEX SQL Workshop interface. The SQL Commands tab is active, displaying the following query:

```
1 SELECT department_id
2 FROM departments
3 MINUS
4 SELECT department_id
5 FROM employees
6 WHERE job_id = 'ST_CLERK';
```

The Results tab is selected, showing a table with the column **DEPARTMENT_ID**. The table contains 8 rows with values: 10, 20, 30, 40, 60, 70, 80, and 90. At the bottom, it states "8 rows returned in 0.01 seconds" and provides a "Download" link.

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.

The screenshot shows the APEX SQL Workshop interface. The SQL Commands tab is active, displaying the following query:

```
1 SELECT country_id, country_name
2 FROM countries
3 MINUS
4 SELECT c.country_id, c.country_name
5 FROM countries c
6 JOIN locations l ON c.country_id = l.country_id
7 JOIN departments d ON l.location_id = d.location_id;
```

The Results tab is selected, showing the text "no data found".

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

The screenshot shows the APEX SQL Workshop interface. The SQL Commands tab is active, displaying a query that uses UNION ALL to combine results from the employees table for departments 10, 50, and 20. The query is as follows:

```
1 SELECT job_id, department_id
2 FROM employees
3 WHERE department_id = 10
4 UNION ALL
5 SELECT job_id, department_id
6 FROM employees
7 WHERE department_id = 50
8 UNION ALL
9 SELECT job_id, department_id
10 FROM employees
11 WHERE department_id = 20
12 ORDER BY department_id;
```

The Results tab shows the output of the query, which is a table with two columns: JOB_ID and DEPARTMENT_ID. The results are as follows:

JOB_ID	DEPARTMENT_ID
AD_PRE	10
AD_VP	10
AD_VP	10
ST_CLERK	50

4 rows returned in 0.01 seconds

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).

The screenshot shows the APEX SQL Workshop interface. The SQL Commands tab is active, displaying a query that uses INTERSECT to find employees who have returned to their original job. The query is as follows:

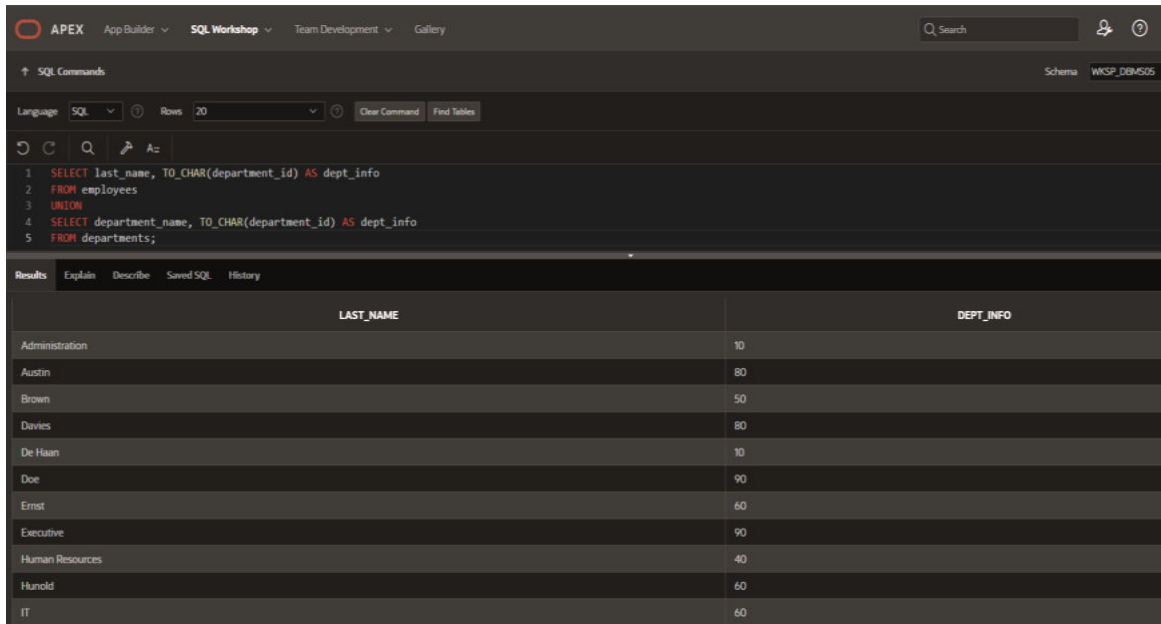
```
1 SELECT employee_id, job_id
2 FROM employees
3 INTERSECT
4 SELECT employee_id, job_id
5 FROM job_history;
```

The Results tab shows the output of the query, which is a table with two columns: EMPLOYEE_ID and JOB_ID. The results are as follows:

EMPLOYEE_ID	JOB_ID
104	IT_PROG
105	SA_REP

2 rows returned in 0.01 seconds

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.



The screenshot shows the APEX SQL Workshop interface. The SQL command window contains the following query:

```
1 SELECT last_name, TO_CHAR(department_id) AS dept_info
2 FROM employees
3 UNION
4 SELECT department_name, TO_CHAR(department_id) AS dept_info
5 FROM departments;
```

The Results tab is active, displaying a table with two columns: LAST_NAME and DEPT_INFO. The table contains 14 rows of data, combining employee information and department information.

LAST_NAME	DEPT_INFO
Administration	10
Austin	80
Brown	50
Davies	80
De Haan	10
Doe	90
Ernst	60
Executive	90
Human Resources	40
Hunold	60
IT	60