

## EXERCISE-10

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 department_id FROM departments MINUS
```

```
SELECT 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 FROM countries MINUS SELECT c.country_id, c.country_name
```

```
FROM countries c JOIN locations l ON c.country_id = l.country_id 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
```

```
(SELECT job_id, department_id FROM employees WHERE department_id = 50) UNION
```

```
(SELECT job_id, department_id FROM employees 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 employee_id, job_id FROM job_history WHERE job_id = (SELECT job_id FROM
```

```
employees WHERE job_history.employee_id = employees.employee_id) AND start_date = (SELECT
```

```
MIN(start_date) FROM job_history WHERE job_history.employee_id = employees.employee_id);
```

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 e.last_name, e.department_id FROM employees e UNION SELECT d.department_id,
```

```
d.department_name FROM departments d;
```

## Practice Exercise

1. Create a report that shows the Global Fast Foods promotional name, start date, and end date from the f\_promotional\_menus table. If there is an end date, temporarily replace it with “end in two weeks”. If there is no end date, replace it with today’s date.

```
SELECT promotional_name, start_date, NVL(TO_CHAR(end_date, 'end in  
two weeks'), TO_CHAR(SYSDATE, 'DD-Mon-YYYY')) AS end_date FROM f_promotional_menus;
```

2. Not all Global Fast Foods staff members receive overtime pay. Instead of displaying a null value for these employees, replace null with zero. Include the employee’s last name and overtime rate in the output. Label the overtime rate as “Overtime Status”.

```
SELECT last_name, NVL(overtime_rate, 0) as "Overtime Status"  
FROM Global_Fast_Foods_Staff;
```

3. The manager of Global Fast Foods has decided to give all staff who currently do not earn overtime an overtime rate of \$5.00. Construct a query that displays the last names and the overtime rate for each staff member, substituting \$5.00 for each null overtime value.

```
SELECT last_name, NVL(overtime_rate, 5.00) as "Overtime Rate"  
FROM Global_Fast_Foods_Staff;
```

4. Not all Global Fast Foods staff members have a manager. Create a query that displays the employee last name and 9999 in the manager ID column for these employees.

```
SELECT last_name, NVL(manager_id, 9999) as "Manager ID"  
FROM Global_Fast_Foods_Staff;
```

5. Which statement(s) below will return null if the value of v\_sal is 50?

- a. SELECT nvl(v\_sal, 50) FROM emp;
- b. SELECT nvl2(v\_sal, 50) FROM emp;
- c. SELECT nullif(v\_sal, 50) FROM emp;
- d. SELECT coalesce(v\_sal, Null, 50) FROM emp;

Statement c will return null if v\_sal is 50. It uses the NULLIF function.

6. What does this query on the Global Fast Foods table return?

```
SELECT COALESCE(last_name, to_char(manager_id)) as NAME  
FROM f_staffs;
```

7a. Create a report listing the first and last names and month of hire for all employees in the EMPLOYEES table (use TO\_CHAR to convert hire\_date to display the month).

```
SELECT first_name, last_name, TO_CHAR(hire_date, 'Month') as month_of_hire
FROM employees;
```

b. Modify the report to display null if the month of hire is September. Use the NULLIF function.

```
SELECT
first_name, last_name, NULLIF(TO_CHAR(hire_date, 'Month'), 'September') as month_of_hire
FROM employees;
```

8. For all null values in the specialty column in the DJs on Demand d\_partners table, substitute “No Specialty.” Show the first name and specialty columns only.

### Conditional Expressions

1. From the DJs on Demand d\_songs table, create a query that replaces the 2-minute songs with “shortest” and the 10-minute songs with “longest”. Label the output column “Play Times”.

```
SELECT
CASE
  WHEN duration = 2 THEN 'Shortest'
  WHEN duration = 10 THEN 'Longest'
  ELSE duration
END as "Play Times"
FROM d_songs;
```

2. Use the Oracle database employees table and CASE expression to decode the department id. Display the department id, last name, salary and a column called “New Salary” whose value is based on the following conditions:

If the department id is 10 then 1.25 \* salary  
If the department id is 90 then 1.5 \* salary  
If the department id is 130 then 1.75 \* salary  
Otherwise, display the old salary.

```
SELECT department_id, last_name, salary,
CASE
  WHEN department_id = 10 THEN 1.25 * salary
  WHEN department_id = 90 THEN 1.5 * salary
  WHEN department_id = 130 THEN 1.75 * salary
  ELSE salary
END as "New Salary"
FROM employees;
```

3. Display the first name, last name, manager ID, and commission percentage of all employees in departments 80 and 90. In a 5th column called “Review”, again display the manager ID. If they don’t have a manager, display the commission percentage. If they don’t have a commission, display 99999.

```
SELECT first_name, last_name, manager_id, commission_pct,
CASE
```

```
WHEN manager_id IS NULL THEN commission_pct  
WHEN commission_pct IS NULL THEN 99999  
ELSE manager_id  
END as "Review"  
FROM employees  
WHERE department_id IN (80, 90);
```

### **Cross Joins and Natural Joins**

Use the Oracle database for problems 1-4.

1. Create a cross-join that displays the last name and department name from the employees and departments tables.

```
SELECT e.last_name, d.department_name  
FROM employees e  
CROSS JOIN departments d;
```

2. Create a query that uses a natural join to join the departments table and the locations table. Display the department id, department name, location id, and city.

```
SELECT d.department_id, d.department_name, l.location_id, l.city  
FROM departments d  
NATURAL JOIN locations l;
```

3. Create a query that uses a natural join to join the departments table and the locations table. Restrict the output to only department IDs of 20 and 50. Display the department id, department name, location id, and city.

```
SELECT d.department_id, d.department_name, l.location_id, l.city FROM departments d  
NATURAL JOIN locations l  
WHERE d.department_id IN (20, 50);
```