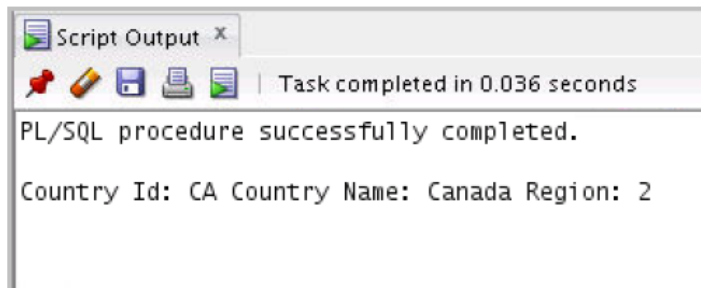


1.

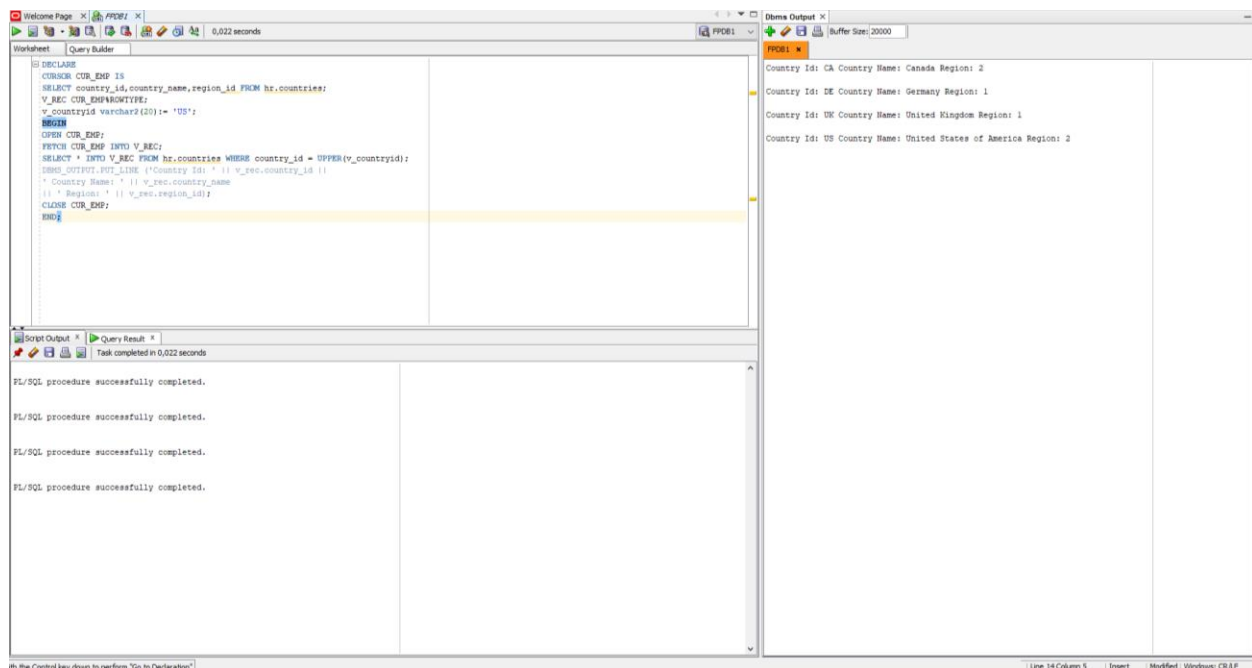
Write a PL/SQL block to print information about a given country.

- Declare a PL/SQL record based on the structure of the `COUNTRIES` table.
- Declare a variable `v_countryid`. Assign CA to `v_countryid`.
- In the declarative section, use the `%ROWTYPE` attribute and declare the `v_country_record` variable of type `countries`.
- In the executable section, get all the information from the `COUNTRIES` table by using `v_countryid`. Display selected information about the country. The sample output is as follows:



- You may want to execute and test the PL/SQL block for countries with the IDs DE, UK, and US.

1-



2.

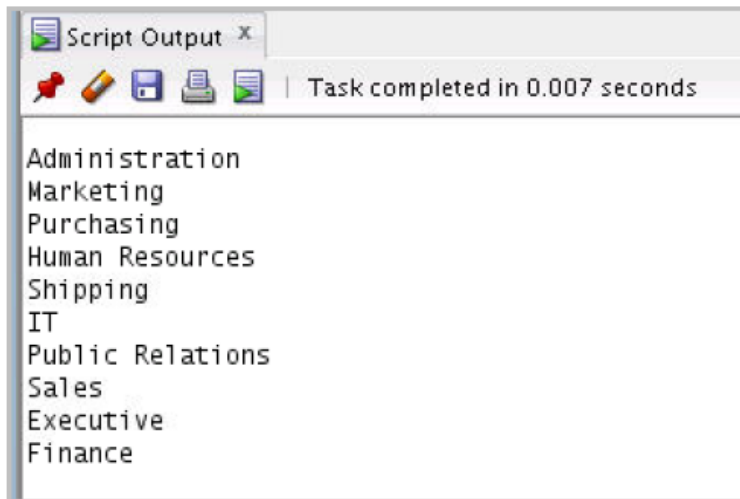
Create a PL/SQL block to retrieve the names of some departments from the `DEPARTMENTS` table and print each department name on the screen, incorporating an associative array. Save the script as `lab_07_02_soln.sql`.

- a. Declare an `INDEX BY table` `dept_table_type` of type `departments.department_name`. Declare a variable `my_dept_table` of type `dept_table_type` to temporarily store the names of the departments.
- b. Declare two variables: `f_loop_count` and `v_deptno` of type `NUMBER`. Assign 10 to `f_loop_count` and 0 to `v_deptno`.
- c. Using a loop, retrieve the names of 10 departments and store the names in the associative array. Start with `department_id` 10. Increase `v_deptno` by 10 for every loop iteration. The following table shows the `department_id` for which you should retrieve the `department_name`.

DEPARTMENT_ID	DEPARTMENT_NAME
10	Administration
20	Marketing
30	Purchasing
40	Human Resources

50	Shipping
60	IT
70	Public Relations
80	Sales
90	Executive
100	Finance

- d. Using another loop, retrieve the department names from the associative array and display them.
- e. Execute and save your script as `lab_07_02_soln.sql`. The output is as follows:



```

Administration
Marketing
Purchasing
Human Resources
Shipping
IT
Public Relations
Sales
Executive
Finance

```

2-

The screenshot displays the Oracle SQL Developer environment. The top toolbar includes icons for Welcome Page, File, Edit, Run, and other standard functions. The main workspace is divided into three panes:

- Worksheet:** Contains a PL/SQL procedure named `FOR_V_REC`. The code is as follows:

```
BEGIN
FOR V_REC IN (SELECT department_name, department_id FROM HR.departments) LOOP
DBMS_OUTPUT.PUT_LINE(V_REC.department_name);
EXIT WHEN V_REC.department_id=100;
END LOOP;
END;
```
- Script Output:** Displays the message "PL/SQL procedure successfully completed." and "Task completed in 0.026 seconds".
- Schema Output:** Lists the departments from the `HR.departments` table:
  - Administration
  - Marketing
  - Purchasing
  - Human Resources
  - Shipping
  - IT
  - Public Relations
  - Sales
  - Executive
  - Finance

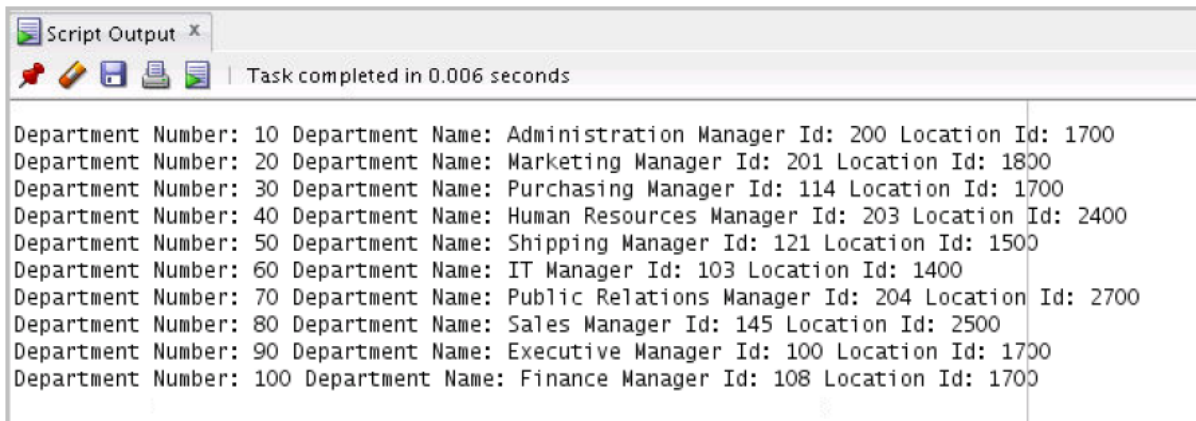
The status bar at the bottom indicates "Line 6 Column 5 | Insert | Modified | Windows: CRLF".

3.

Modify the block that you created in Task 2 to retrieve all information about each department from the `DEPARTMENTS` table and display the information. Use an associative array with the `INDEX BY` table of records method.

- Load the `lab_07_02_soln.sql` script.
- You have declared the associative array to be of type `departments.department_name`. Modify the declaration of the associative array to temporarily store the number, name, and location of all the departments. Use the `%ROWTYPE` attribute.
- Modify the `SELECT` statement to retrieve all department information currently in the `DEPARTMENTS` table and store it in the associative array.
- Using another loop, retrieve the department information from the associative array and display the information.

The sample output is as follows:



```
Script Output x
Task completed in 0.006 seconds

Department Number: 10 Department Name: Administration Manager Id: 200 Location Id: 1700
Department Number: 20 Department Name: Marketing Manager Id: 201 Location Id: 1800
Department Number: 30 Department Name: Purchasing Manager Id: 114 Location Id: 1700
Department Number: 40 Department Name: Human Resources Manager Id: 203 Location Id: 2400
Department Number: 50 Department Name: Shipping Manager Id: 121 Location Id: 1500
Department Number: 60 Department Name: IT Manager Id: 103 Location Id: 1400
Department Number: 70 Department Name: Public Relations Manager Id: 204 Location Id: 2700
Department Number: 80 Department Name: Sales Manager Id: 145 Location Id: 2500
Department Number: 90 Department Name: Executive Manager Id: 100 Location Id: 1700
Department Number: 100 Department Name: Finance Manager Id: 108 Location Id: 1700
```

3-

The screenshot shows the Oracle SQL Developer interface. The 'Script Output' pane on the left contains the following PL/SQL code:

```
DECLARE
CURSOR CUR_EMP IS
SELECT department_id, department_name, manager_id, location_id FROM hr.departments;
V_REC CUR_EMP%ROWTYPE;
F_loop_count NUMBER(2) := 10;
V_deptno NUMBER(4) := 0;
BEGIN
OPEN CUR_EMP;
FETCH CUR_EMP INTO V_REC;
FOR i IN 1..F_loop_count
LOOP
V_deptno := V_deptno + 10;
SELECT * INTO V_REC FROM hr.departments
WHERE department_id = V_deptno;
DBMS_OUTPUT.PUT_LINE ('Department Number: ' ||
V_rec.department_id
|| ' Department Name: ' || V_rec.department_name
|| ' Manager Id: ' || V_rec.manager_id
|| ' Location Id: ' || V_rec.location_id);
END LOOP;
CLOSE CUR_EMP;
END;
```

The 'Items Output' pane on the right displays the results of the procedure, which are 100 rows of department and manager data:

Department Number	Department Name	Manager Id	Location Id
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500
60	IT	103	1400
70	Public Relations	204	2700
80	Sales	145	2500
90	Executive	100	1700
100	Finance	108	1700

The 'Script Output' pane also shows the message: 'PL/SQL procedure successfully completed.'

4.

the salary of the employee is less than 5,000 and if the manager ID is either 101 or 124, display the message “<<last\_name>> Due for a raise.” Otherwise, display the message “<<last\_name>> Not Due for a raise.”

Department ID	Message
10	Whalen Due for a raise
20	Hartstein Not Due for a raise Fay Not Due for a raise
50	Weiss Not Due for a raise Fripp Not Due for a raise Kaufling Not Due for a raise Vollman Not Due for a raise. . . . . . OConnell Due for a raise Grant Due for a raise
80	Russell Not Due for a raise Partners Not Due for a raise Errazuriz Not Due for a raise Cambrault Not Due for a raise . . . Livingston Not Due for a raise Johnson Not Due for a raise

4-

Welcome Page
PDF
Worksheet
Query Builder
Script Output
Query Result
0,037 seconds

```

BEGIN
FOR v_rec IN (SELECT department_id,salary,manager_id,last_name FROM hr.employees) LOOP
IF v_rec.salary>5000 and v_rec.manager_id IN (101,124) THEN
DBMS_OUTPUT.PUT_LINE('Department Id : ' || v_rec.department_id || ' Last Name : ' || UPPER(v_rec.last_name) || ' Due for a raise.' );
ELSE
DBMS_OUTPUT.PUT_LINE('Department Id : ' || v_rec.department_id || ' Last Name : ' || UPPER(v_rec.last_name) || ' Not Due for a raise.' );
END IF;
END LOOP;
END;
```

Script Output
Query Result
0,037 seconds
Task completed in 0,037 seconds

PL/SQL procedure successfully completed.

DMS Output
Buffer Size: 2000

```

Department Id : 90 Last Name : KING Not Due for a raise.
Department Id : 90 Last Name : DECKMAN Not Due for a raise.
Department Id : 90 Last Name : DE SAAN Not Due for a raise.
Department Id : 60 Last Name : HENRIK Not Due for a raise.
Department Id : 40 Last Name : ENIST Not Due for a raise.
Department Id : 60 Last Name : JUSTIN Not Due for a raise.
Department Id : 60 Last Name : FATABALLA Not Due for a raise.
Department Id : 60 Last Name : LORETTI Not Due for a raise.
Department Id : 100 Last Name : GREENBERG Not Due for a raise.
Department Id : 100 Last Name : FAYET Not Due for a raise.
Department Id : 100 Last Name : CHEN Not Due for a raise.
Department Id : 100 Last Name : PCLARRA Not Due for a raise.
Department Id : 100 Last Name : TURNER Not Due for a raise.
Department Id : 100 Last Name : POPP Not Due for a raise.
Department Id : 30 Last Name : RAFAELI Not Due for a raise.
Department Id : 30 Last Name : BROOK Not Due for a raise.
Department Id : 30 Last Name : BALIN Not Due for a raise.
Department Id : 30 Last Name : TOSTIAS Not Due for a raise.
Department Id : 30 Last Name : KIMMO Not Due for a raise.
Department Id : 30 Last Name : COLANDRES Not Due for a raise.
Department Id : 80 Last Name : WEISS Not Due for a raise.
Department Id : 80 Last Name : FRIPP Not Due for a raise.
Department Id : 80 Last Name : BOUTLING Not Due for a raise.
Department Id : 50 Last Name : VULLARD Not Due for a raise.
Department Id : 50 Last Name : MCORRUG Not Due for a raise.
Department Id : 50 Last Name : HAYER Not Due for a raise.
Department Id : 50 Last Name : HIRSHLIMONT Not Due for a raise.
Department Id : 50 Last Name : LANDRY Not Due for a raise.
Department Id : 50 Last Name : MARBLE Not Due for a raise.
Department Id : 50 Last Name : BISSOT Not Due for a raise.
Department Id : 50 Last Name : RIVKIN Not Due for a raise.
Department Id : 50 Last Name : MARLOW Not Due for a raise.
Department Id : 50 Last Name : OLSON Not Due for a raise.
Department Id : 50 Last Name : HALLIN Not Due for a raise.
Department Id : 50 Last Name : ROYERS Not Due for a raise.
Department Id : 50 Last Name : GEE Not Due for a raise.
Department Id : 50 Last Name : PHILLIPPER Not Due for a raise.
Department Id : 50 Last Name : LAMNIS Not Due for a raise.
Department Id : 50 Last Name : STILES Not Due for a raise.
Department Id : 50 Last Name : SEO Not Due for a raise.
Department Id : 50 Last Name : FATEL Not Due for a raise.
Department Id : 50 Last Name : RAJN Not Due for a raise.
Department Id : 50 Last Name : DAVIES Not Due for a raise.
Department Id : 50 Last Name : MATOS Not Due for a raise.
Department Id : 50 Last Name : VARGAS Not Due for a raise.
Department Id : 50 Last Name : RUSSELL Not Due for a raise.
Department Id : 50 Last Name : FARTNESS Not Due for a raise.
Department Id : 50 Last Name : EKKAUTRI Not Due for a raise.
Department Id : 80 Last Name : CAMERALLI Not Due for a raise.
Department Id : 80 Last Name : LOTTEY Not Due for a raise.
Department Id : 30 Last Name : STEVEN Not Due for a raise.
```