John Thomas A. Paringit 12 – Computer Programming A

Quiz (Maximum Score: 70)

Instruction: Kindly put your answers in a Word Document and pass it as a PDF format. **File name: SURNAME_FinalQuiz.pdf (**Minus 5pts if file name is not followed**)**

1. Make a Venn diagram about Procedures and Functions.
2. In your own words, describe parameters and the purpose they serve in PL/SQL subprograms.
3. What is the difference between the following two pieces of code?

CODE SAMPLE A

```
DECLARE
        v_empid employees.employee_id%TYPE := 100;
v_percent_increase NUMBER(2,2) := .05;
BEGIN
        UPDATE employees
        SET salary = (salary * v_percent_increase) + salary
        WHERE employee_id = v_empid;
END;
```

CODE SAMPLE B

```
CREATE PROCEDURE pay_raise (p_empid employees employee_id%TYPE,
                                        p_percent_increase NUMBER)
IS
BEGIN
        UPDATE employees
        SET salary = (salary * p_percent_increase) + salary
        WHERE employee_id = p_empid;
END pay_raise;
```

4. Using the COUNTRIES table:
   a) Create a procedure that accepts a country_id as a parameter and displays the name of the country and its region id. Name your procedure get_country_info. Save your procedure definition for later use.
   b) Execute your procedure from an anonymous block, using country_id CA.
   c) Re-execute the procedure from the anonymous block, this time using country_id NE. What happens? (screenshot the result)

d) Retrieve your procedure code from Saved SQL and modify it to trap the NO_DATA_FOUND exception in an exception handler. Execute the modified procedure using country_id NE again. Now what happens? (Screenshot the result)

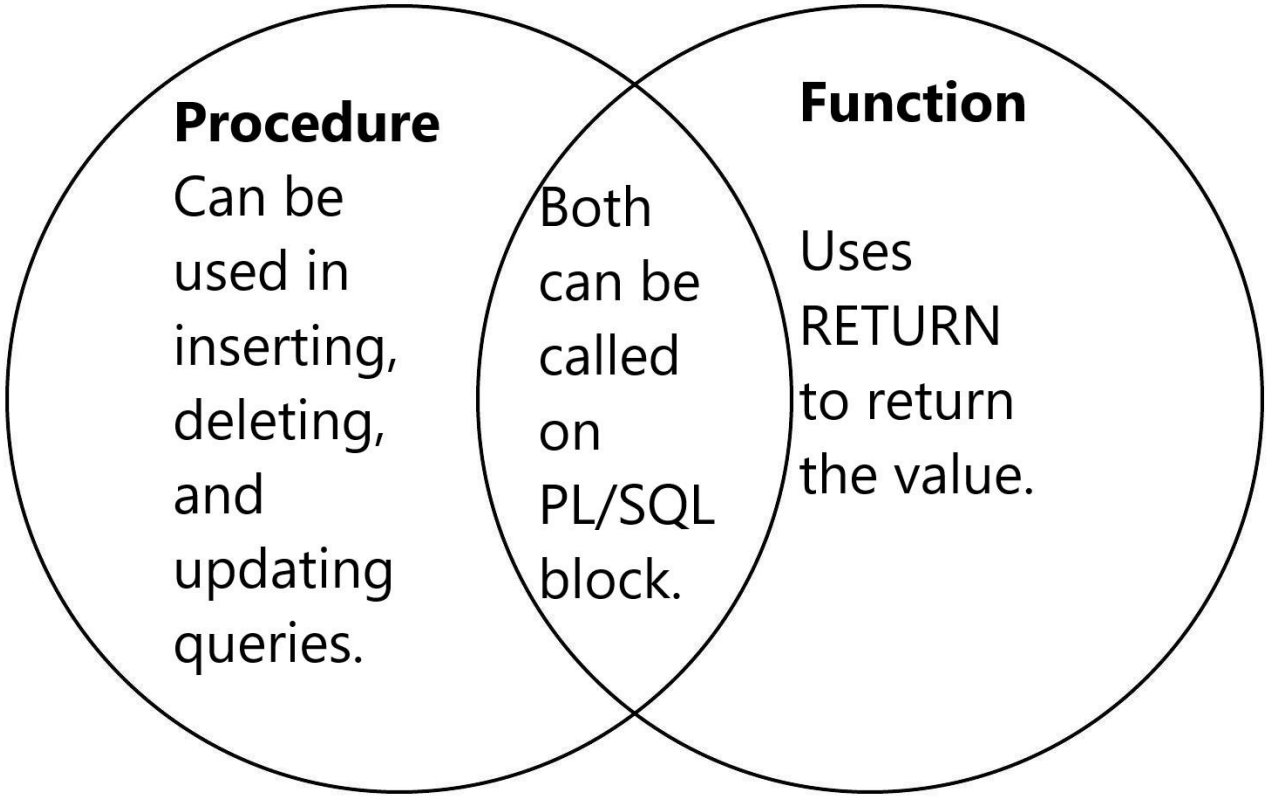
5. Create a procedure that receives a country_id as an IN parameter and returns the name and population of that country as OUT parameters. Include an exception handler to trap the NO_DATA_FOUND exception if the country does not exist. The procedure should not display the returned values; this will be done in the next step. Name your procedure find_area_pop. Save your code.
    a) Test your procedure by creating and executing an anonymous block which invokes the procedure and displays the returned OUT values. Save your code. Run the block twice, with country_ids CA (Canada) and NE (does not exist). (screenshot the result)

6. Create a procedure which accepts an integer as an IN OUT parameter and returns the square of that integer, for example the square of 4 is 16. Save your code.
    a) Test your procedure from an anonymous block three times, using integer values 4, 7, and –20 (negative 20). (Screenshot the result)

7. Write a procedure that displays the number of countries in a given region whose highest elevations exceed a given value. The procedure should accept two formal parameters, one for a region_id and the other for an elevation value for comparison. Use DBMS_OUTPUT.PUT_LINE to display the results in a message. Save your procedure code.
    a) Execute your procedure using the value 5 for the region_id and 2000 for the highest elevation. (screenshot the result)

Note: You will still pass the sql code for the execution of the procedure.

1.)



2.) Parameters holds any datatype in PL/SQL. This helps give input in the PL/SQL subprograms.

3.) The Sample A is an anonymous block, a block that has no name. The Sample B is procedure.

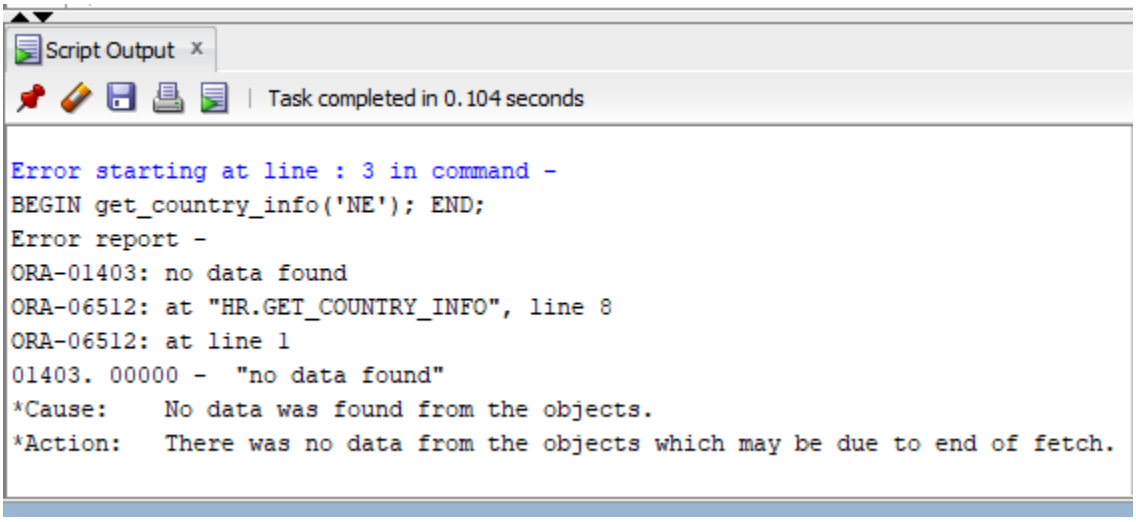
4.a)

```
CREATE OR REPLACE PROCEDURE get_country_info(
        prm_country_id IN countries.country_id%TYPE)
IS
        v_country_name countries.country_name%TYPE;
        v_region_id countries.region_id%TYPE;
BEGIN
        SELECT country_name, region_id
                INTO v_country_name, v_region_id
                FROM countries
                WHERE country_id = prm_country_id;
DBMS_OUTPUT.PUT_LINE('Country Name: ' || v_country_name || ', Region ID: ' || v_region_id);

END get_country_info;
```

4.b)

```
SET SERVEROUTPUT ON

EXEC get_country_info('CA');
```

4.c)



4.d)

```
CREATE OR REPLACE PROCEDURE get_country_info(
        prm_country_id IN countries.country_id%TYPE
)
IS
        v_country_name countries.country_name%TYPE;
        v_region_id countries.region_id%TYPE;
BEGIN
        SELECT country_name, region_id
                INTO v_country_name, v_region_id
                FROM countries
                WHERE country_id = prm_country_id;
        DBMS_OUTPUT.PUT_LINE('Country Name: ' || v_country_name || ', Region ID: ' || v_region_id);
EXCEPTION
        WHEN NO_DATA_FOUND THEN
                DBMS_OUTPUT.PUT_LINE('No data found for this country_id.');
        WHEN OTHERS THEN
                DBMS_OUTPUT.PUT_LINE('Error');
END
get_country_info;
```

```
Procedure GET_COUNTRY_INFO compiled

No data found for this country_id.


PL/SQL procedure successfully completed.
```