 



Database Programming with PL/SQL 4-3: Iterative Control: Basic Loops Practice Activities

# Vocabulary

Identify the vocabulary word for each definition below:

|  |  |
| --- | --- |
| BASIC LOOPS | Encloses a sequence of statements between the keywords LOOP and END LOOP and must execute at least once. |
| EXIT STATMENT | Statement to terminate a loop. |

# Try It / Solve It

1. What purpose does a loop serve in PL/SQL?

Scopul unui loop este de a executa o secventa pana cand o conditie este indeplinita.

1. List the types of loops in PL/SQL.

BASIC loops

FOR loops

WHILE loops

1. What statement is used to explicitly end a loop?

EXIT

1. Write a PL/SQL block to display the country\_id and country\_name values from the COUNTRIES table for country\_id whose values range from 1 through 3. Use a basic loop. Increment a variable from 1 through 3. Use an IF statement to test your variable and EXIT the loop after you have displayed the first 3 countries.

DECLARE

v\_country\_id countries.country\_id%TYPE :=1;

v\_country\_name countries.country\_name%TYPE;

BEGIN

LOOP

SELECT country\_name INTO v\_country\_name

FROM countries

WHERE country\_id = v\_country\_id;

DBMS\_OUTPUT.PUT\_LINE('Tara cu id-ul: ' || v\_country\_id ||' are numele: '||v\_country\_name);

v\_country\_id := v\_country\_id + 1;

if v\_country\_id > 3 THEN EXIT;

END IF;

END LOOP;

END;

1. Modify your solution to question 4 above, replacing the IF statement with an EXIT WHEN

statement.

DECLARE

v\_country\_id countries.country\_id%TYPE :=1;

v\_country\_name countries.country\_name%TYPE;

BEGIN

LOOP

SELECT country\_name INTO v\_country\_name

FROM countries

WHERE country\_id = v\_country\_id;

DBMS\_OUTPUT.PUT\_LINE('Tara cu id-ul: ' || v\_country\_id ||' are numele: '||v\_country\_name);

v\_country\_id := v\_country\_id + 1;

EXIT WHEN v\_country\_id > 3;

END LOOP;

END;

1. Create a MESSAGES table and insert several rows into it.
   1. To create the messages table.

DROP TABLE messages;

CREATE TABLE messages (results NUMBER(2));

* 1. Write a PL/SQL block to insert numbers into the MESSAGES table. Insert the numbers 1 through 10, excluding 6 and 8.

DECLARE

v\_counter NUMBER(2) := 1;

BEGIN

FOR v\_counter IN 1.. 10 LOOP

IF v\_counter != 6 AND v\_counter != 8 THEN

INSERT INTO messages(results)

VALUES(v\_counter);

END IF;

END LOOP;

END;

* 1. Execute a SELECT statement to verify that your PL/SQL block worked.

SELECT\*

FROM messages;