Database Programming with PL/SQL

4-1: Conditional Control: IF Statements

Practice Activities

Vocabulary

Identify the vocabulary word for each definition below:

|  |  |
| --- | --- |
| IF | Statement that enables PL/SQL to perform actions selectively  based on conditions. |
| LOOP | Control structures – Repetition statements that enable you to  execute statements in a PL/SQL block repeatedly. |
| Condition | An expression with a TRUE or FALSE value that is used to  make a decision. |
| CASE | An expression that determines a course of action based on  conditions and can be used outside a PL/SQL block in a SQL  statement. |

Try It / Solve It

1. What is the purpose of a conditional control structure in PL/SQL?

Sa analizeze niste date in functie de conditii si sa trimita programul pe o ramura sau alta

2. List the three categories of control structures in PL/SQL.

IF

CASE

LOOP

3. List the keywords that can be part of an IF statement.

IF, THEN, ELSIF, ELSE, END IF

4. List the keywords that are a required part of an IF statement.

IF

THEN

END IF

5. Write a PL/SQL block to find the population of a given country in the countries table. Display a

message indicating whether the population is greater than or less than 1 billion (1,000,000,000).

Test your block twice using India (country\_id = 91) and United Kingdom (country\_id = 44). India’s

population should be greater than 1 billion, while United Kingdom’s should be less than 1 billion.

DECLARE

v\_populatie

wf\_countries.population%TYPE

BEGIN

SELECT population INTO v\_populatie

FROM wf\_countries

WHERE country\_id = 91;--apoi ii dai 44 in loc de 91

DBMS\_OUTPUT.PUT\_LINE(v\_populatie);

IF(v\_populatie > 1 000 000 000) THEN

DBMS\_OUTPUT.PUT\_LINE(‘more than 1mld’);

ELSE

DBMS\_OUTPUT.PUT\_LINE(‘less than 1mld’);

END;

6. Modify the code from the previous exercise so that it handles all the following cases:

A. Population is greater than 1 billion.

B. Population is greater than 0.

C. Population is 0.

D. Population is null. (Display: No data for this country.)

Run your code using the following country ids. Confirm the indicated results.

• China (country\_id = 86): Population is greater than 1 billion.

• United Kingdom (country\_id = 44): Population is greater than 0.

• Antarctica (country\_id = 672): Population is 0.

• Europa Island (country\_id = 15): No data for this country.

Asta o fac pe loc

7. Examine the following code:

DECLARE

v\_country\_id countries.country\_name%TYPE := <a value>;

v\_ind\_date

countries.date\_of\_independence%TYPE;

v\_natl\_holiday countries.national\_holiday\_date%TYPE;

BEGIN

SELECT date\_of\_independence, national\_holiday\_date

INTO v\_ind\_date, v\_natl\_holiday

FROM countries

WHERE country\_id = v\_country\_id;

IF v\_ind\_date IS NOT NULL THEN

DBMS\_OUTPUT.PUT\_LINE('A');

ELSIF v\_natl\_holiday IS NOT NULL THEN

DBMS\_OUTPUT.PUT\_LINE('B');

ELSIF v\_natl\_holiday IS NULL AND v\_ind\_date IS NULL THEN

DBMS\_OUTPUT.PUT\_LINE('C');

END IF;

END;

A. What would print if the country has an independence date equaling NULL and a national

holiday date equaling NULL?

Print C

B. What would print if the country has an independence date equaling NULL and a national

holiday date containing a value?

Print B

C. What would print if the country has an independence date equaling a value and a national

holiday date equaling NULL?

Print A

D. Run a SELECT statement against the COUNTRIES table to determine whether the following

countries have independence dates or national holiday dates, or both. Predict the output of

running the anonymous block found at the beginning of this question.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Country | Country\_ID | Independence Date | National Holiday Date | Output should be |
| Antarctica | 672 | no | no | C |
| Iraq | 964 | yes | no | A |
| Spain | 34 | no | yes | B |
| United States | 1 | yes | no | A |

E. Finally, run the anonymous block found at the beginning of this question using each of the

above country ids as input. Check whether your output answers are correct.

8. Examine the following code. What output do you think it will produce?

DECLARE

v\_num1 NUMBER(3) := 123;

v\_num2 NUMBER;

BEGIN

IF v\_num1 <> v\_num2 THEN

DBMS\_OUTPUT.PUT\_LINE('The two numbers are not equal');

ELSE

DBMS\_OUTPUT.PUT\_LINE('The two numbers are equal');

END IF;

END;

Run the code to check if your prediction was correct. What was the result and why? Modify the

code to use various comparison operators to see different results.

Cele 2 numere nu sunt egale, dar al doilea numar e NULL. Orice test care implica NULL va merge pe False. Deci printezaza ca numerele sunt egale

9. Write a PL/SQL block to accept a year and check whether it is a leap year. For example, if the

year entered is 1990, the output should be “1990 is not a leap year.”

Hint: A leap year should be exactly divisible by 4, but not exactly divisible by 100. However, any

year exactly divisible by 400 is a leap year.

Test your solution with the following years:

Declare

v\_an number(20):=1990;

BEGIN

if mod(v\_an, 400)=0 then

DBMS\_OUTPUT.PUT\_LINE(‘An bisect’);

elsif mod(v\_an, 100) =0 then

DBMS\_OUTPUT.PUT\_LINE(‘An ne-bisect’);

elsif mod(v\_an, 4)=0 then

DBMS\_OUTPUT.PUT\_LINE(‘An bisect’);

else DBMS\_OUTPUT.PUT\_LINE(‘An ne-bisect’);

end if;

end;

