Islamic University of Technology

Database Management Systems Lab

CSE 4308 / CSE 4174

Lab 8

Author:
Ishmam Tashdeed
Md. Tariquzzaman
Zannatun Naim Sristy
CSE, IUT

Contents

1 PL/SQL

PL/SQL is a block-structured language where we can write code organized into blocks similar to Java/C/C++, although the coding style differs here due to not having any curly braces to define blocks, rather we define blocks using the BEGIN and END keywords. You can define PROCEDUREs (which execute some code without returning anything) and FUNCTIONs (which execute code and return some variable/record). And you can even define unnamed blocks in PL/SQL, otherwise known as anonymous blocks that can be called immediately upon defining using the / at the end.

1.1 Structure of PL/SQL Block

PL/SQL extends SQL by adding constructs found in procedural languages, resulting in a structural language that is more powerful than SQL. The basic unit in PL/SQL is a block. All PL/SQL programs are made up of blocks, which can be nested within each other.

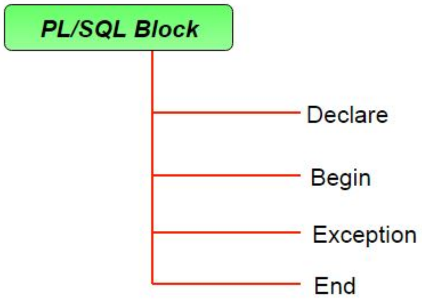


Figure 1: Structure of PL/SQL Block

1.2 Syntax of PL/SQL Block

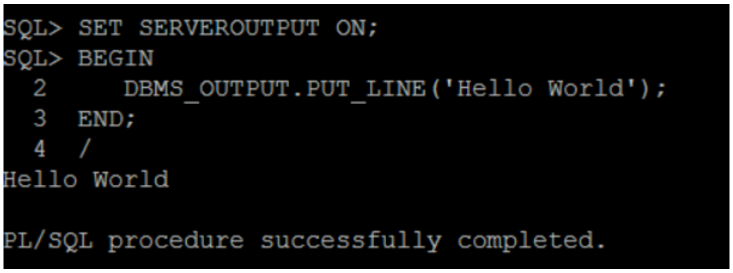
```
DECLARE
    declaration statements;

BEGIN
    executable statements;
```

```
EXCEPTION
    exception handling statements;

END;
/
```

Note: Remember to `SET SERVEROUTPUT ON` to see the results of the blocks.



```
SQL> SET SERVEROUTPUT ON;
SQL> BEGIN
  2      DBMS_OUTPUT.PUT_LINE('Hello World');
  3  END;
  4  /
Hello World

PL/SQL procedure successfully completed.
```

Figure 2: PL/SQL Syntax to show output

1.3 Example of PL/SQL Block

```
DECLARE
   v_salary NUMBER;
   v_bonus  NUMBER := 1000;

BEGIN
   SELECT salary INTO v_salary FROM employees WHERE
   employee_id = 101;

   IF v_salary > 5000 THEN
      v_salary := v_salary + v_bonus;
   END IF;

   DBMS_OUTPUT.PUT_LINE('Updated salary: ' || v_salary);

EXCEPTION
   WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.PUT_LINE('Employee not found.');
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('An error occurred.');

END;
/
```

Note: The list of all exceptions can be found here.

2 Function

A function in PL/SQL is a subprogram designed to perform a specific task and return a single value. Functions are commonly used when a value needs to be computed and returned, such as the result of a calculation, string manipulation, or database query. Unlike procedures, a function must always return a value.

2.1 Example of a Function

Define a function that, given the name of a department, returns the count of the number of instructors in that department.

```
CREATE FUNCTION dept_count (dept_name VARCHAR(20))
RETURNS INTEGER
BEGIN
    DECLARE d_count INTEGER;
    SELECT COUNT(*)
    INTO d_count
    FROM instructor
    WHERE instructor.dept_name = dept_name;
    RETURN d_count;
END;
/
```

Running the function:

```
SELECT dept_name, budget
FROM department
WHERE dept_count(dept_name) > 12;
```

2.2 Table Functions

```
CREATE FUNCTION instructor_of (dept_name CHAR(20))
RETURNS TABLE (
    ID VARCHAR(5),
    name VARCHAR(20),
    dept_name VARCHAR(20),
    salary NUMERIC(8,2))
RETURN TABLE
    (SELECT ID, name, dept_name, salary
     FROM instructor
     WHERE instructor.dept_name = dept_name);
/
```

Running the function:

```
SELECT *
FROM TABLE (instructor_of(’Music’));
```

3 Procedure

A procedure in PL/SQL is also a subprogram. It can accept input and output parameters. It uses parameters to return results, but it does not use the RETURN statement. It is used to perform an action (eg. modifying data in the database).

3.1 Example of a Procedure

```
CREATE PROCEDURE findMin(x IN NUMBER, y IN NUMBER) IS
   z NUMBER; -- Declare the variable z
BEGIN
   IF x < y THEN
      z := x;
   ELSE
      z := y;
   END IF;
   dbms_output.put_line(’ Minimum is : ’ || z);
END;
/
```

Running the procedure:

```
DECLARE
    a NUMBER;
    b NUMBER;
BEGIN
   a := 23;
   b := 45;
   findMin(a, b);
```

```
END;
/
```

4 Language Constructs

PL/SQL syntax for conditionals, loops, functions, and mathematical operations.

4.1 Conditionals

```
DECLARE
    num NUMBER := 7;
BEGIN
    IF MOD(num, 2) = 0 THEN
        DBMS_OUTPUT.PUT_LINE('The number is even.');
    ELSE
        DBMS_OUTPUT.PUT_LINE('The number is odd.');
    END IF;
END;
/
```

4.2 Loops

Print numbers from 1 to 5 using a `WHILE` loop.

```
DECLARE
    counter NUMBER := 1;
BEGIN
    WHILE counter <= 5 LOOP
        DBMS_OUTPUT.PUT_LINE('Number: ' || counter);
        counter := counter + 1;
    END LOOP;
END;
/
```

Print numbers from 1 to 5 using a `FOR` loop

```
BEGIN
    FOR i IN 1..5 LOOP
        DBMS_OUTPUT.PUT_LINE('Number: ' || i);
    END LOOP;
END;
/
```

4.3 Function inside a Procedure

```
CREATE OR REPLACE FUNCTION calculate_square(n NUMBER) RETURNS
    NUMBER IS
BEGIN
    RETURN n * n;
END calculate_square;
/

CREATE OR REPLACE PROCEDURE print_square(num IN NUMBER) IS
    result NUMBER;
BEGIN
    result := calculate_square(num);
    DBMS_OUTPUT.PUT_LINE(’The square of ’ || num || ’ is: ’
   || result);
END print_square;
/

BEGIN
    print_square(5);
END;
/
```

4.4 Mathematical Functions

```
DECLARE
    num NUMBER := 16;
    result NUMBER;
BEGIN
    result := SQRT(num);
    DBMS_OUTPUT.PUT_LINE(’The square root of ’ || num || ’ is
    ’ || result);
END;
/
```

```
DECLARE
    num1 NUMBER := 10;
    num2 NUMBER := 3;
BEGIN
    DBMS_OUTPUT.PUT_LINE(’The remainder when ’ || num1 || ’
   is divided by ’ || num2 || ’ is ’ || MOD(num1, num2));
END;
/
```

5 Lab Tasks

1. Warm-up

   (a) Print your name and student ID.

   (b) Take two numbers as input and print their product.

   (c) Take a number as input and print whether it is a whole number or a fraction.

2. Execute the provided `movie.sql` file and consider the following schema for a movie database:

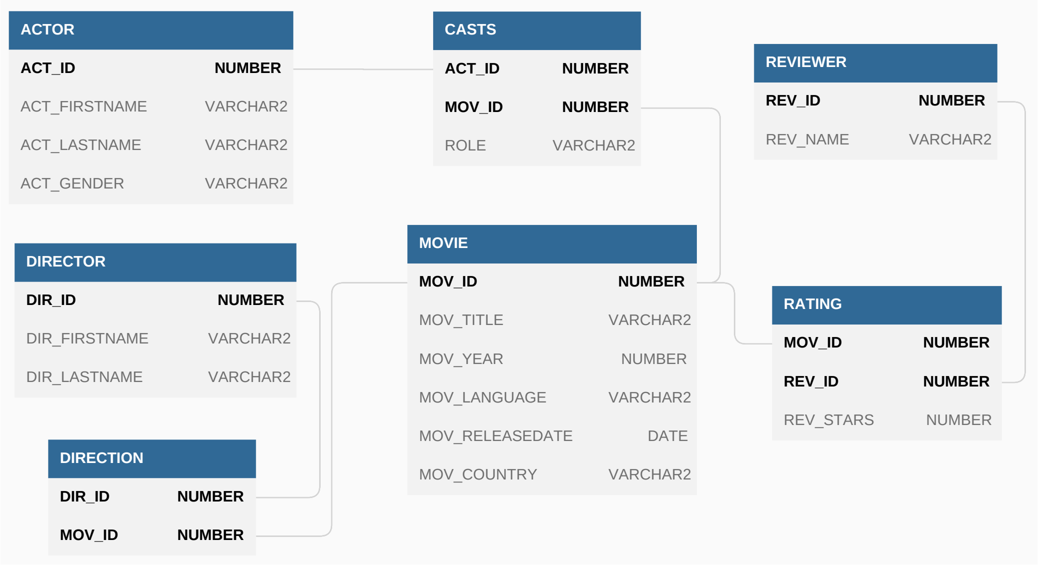


Figure 3: ER Diagram for a movie database

3. Write a procedure to find the N movies with the highest rating and their details (highest rating means the highest average rating). The procedure will take N as input and print the details up to N movies. If N is greater than the number of movies, then it will print an error message.

4. Write a function to find the movie status (“Solo”or “Ensemble”). If the total number of actors/actresses in a movie is 1, then the status should

be “Solo”, otherwise it should be “Ensemble”. The function will take the title of the movie as input and return the status.

5. Write a procedure to find possible nominees for the Oscars. A director is eligible for an Oscar if at least one of their movies has an average rating of at least 7. In addition, the movie should be reviewed by more than 10 reviewers.