

Task 6: Procedures, Functions and Loops

Aim: To write a programming PL/SQL procedures, functions and loops on number theory and business scenario like.

1. Write a PL/SQL block that calculates the average age

of players and displays the result.

2. Write a PL/SQL block that inserts a new player record into a Player table.

3. Create a function that returns the total number of even-numbered players registered for any tournament.

Write a PL/SQL block that calculates the average age of players and displays the result:

DECLARE

total-age NUMBER := 0;

num-players NUMBER := 0;

avg-age NUMBER := 0;

BEGIN

-- USING a cursor to loop through all players.

FOR player_rec IN (SELECT Age FROM Player) LOOP

total-age := total-age + player_rec.Age; -- summing up the ages.

num-players := num-players + 1; -- Counting the numbers of players

END LOOP;

-- Calculating the average age

If num-players > 0 THEN

avg-age := total-age / num-players;

END IF;

-- Displaying the result

DBMS-OUTPUT.PUT-LINE('Total Players: ' || num-players);

DBMS-OUTPUT.PUT-LINE('Total Age: ' || total-age);

DBMS_OUTPUT.PUT_LINE('Average Age: '|| avg_age);
END;

Output:

Total Players : 14
Total Age : 342
Average Age : 24.42

Write a PL/SQL block that inserts a new player record into a Player table.

DECLARE

V-PlayerID VARCHAR(6) := '&PlayerID'; -- You can generate
a unique playerID as needed

V-TeamID VARCHAR(6) := '&TeamID'; -- Replace with
with the actual TeamID.

V-FName VARCHAR(30) := '&Fname';

V-LName VARCHAR(30) := '&Lname';

V-Age NUMBER(5,2) := &Age;

V-DateOfBirth DATE := TO_DATE('&DOB', 'YYYY-MM-DD');

-- Replace with the actual Date of Birth.

V-PlayingRole VARCHAR(25) := '&PlayingRole';

V-email VARCHAR(40) := '&email';

V-Contact_no NUMBER := &Phone; -- Replace with the actual

Contact number.

BEGIN

INSERT INTO Player(PlayerID, TeamID, FName, LName,
Age, DateOfBirth, PlayingRole, email, Contact_no);

VALUES (V-PlayerID, V-TeamID, V-FName, V-LName, V-Age,
V-DateOfBirth, V-PlayingRole, V-email, V-Contact-no);

Commit;

DBMS_OUTPUT.PUT_LINE('Player record inserted successfully');

EXCEPTION

WHEN OTHERS THEN

DBMS_OUTPUT.PUT_LINE('Error: '|| SQLERRM);

ROLLBACK;

END;

Enter the PlayerID : 696

Enter the TeamID : CCB01

Enter the Fname : Rahul.

Enter the LName : Sharma

Enter the Age : 23

Enter the player's Role : All Rounder

Enter the email : rahul.sharma@gmail.com

Enter the contact_no : 99997181815

Player record inserted successfully.

To create a function that returns the total number of teams in a particular cricket Board.

CREATE OR REPLACE FUNCTION

GetTotalTeamsInBoard(BoardID VARCHAR2) RETURN NUMBER IS

V_TotalTeams NUMBER := 0;

BEGIN
SELECT COUNT(*) INTO V_TotalTeams FROM Teams
WHERE BoardID = BoardID;

RETURN V_TotalTeams;

EXCEPTION

WHEN OTHERS THEN

-- Handle other exception like other case when the board doesn't exist or has no teams.

RETURN 0;

WHEN OTHERS THEN

-- Handle other exceptions as needed.

Return -1; -- Return a negative value to indicate an error

END Get Team In Board;

/

Function successfully created.

SQL >

Declare

Numbers res;

Begin

res := GetTotalTeamInBoard('B1D01');

DBMS_OUTPUT.PUT-LINE ('No of teams: ' || res);

END;

/
No of teams: 2

To write a non-recursive PL/SQL procedure to retrieve even-numbered player IDs registered for any tournament.

CREATE OR REPLACE PROCEDURE

Get Even Numbered Player IDs IS

BEGIN

FOR player_rec IN (SELECT playerID FROM Player WHERE TO_NUMBER(playerID) MOD 2 = 0)

LOOP

DBMS_OUTPUT.PUT-LINE ('Even-numbered player ID: ' ||

player_rec.playerID);

END LOOP;

END GetEven Numbered player IDs;

/

VEL TECH CSE	
EX NO	30
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	4
RECORD (5)	5
TOTAL (20)	18
SIGN WITH DATE	9/12/2023

Result:

thus the PL/SQL Procedures, Functions and loops on Number theory and business scenarios. experiment was successfully completed and results are verified.