

11/9/25

## Task 6: Procedures, Functions and Loops

Aim: To write a programming PL/SQL procedures, functions and loops on number theory and business scenarios 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 to retire 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 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', 'YYMM-DD');  
-- Replace with the actual Date of Birth.

V-PlayingRole VARCHAR(15) := '&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 playing Role : All rounder

Enter the email : rahul.sharma@gmail.com

Enter the contact\_no : 9997181815

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 exceptions the 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 EN

END GetTeamInBoard;

/

Function successfully created.

SQL >

Declare

numbers res;

Begin

res := GetTotalTeamInBoard('BID 01');

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 PlayerIDs registered for any tournament.

CREATE OR REPLACE PROCEDURE

GetEvenNumberedPlayerIDs 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 GetEvenNumberedPlayerIDs;

/

VEL TECH CSE	
EX NO	30
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	4
RECORD (5)	5
TOTAL (20)	19
SIGN WITH DATE	22

Result:

Thus the PL, SQL Procedures, Functions and loops non number theory and business scenarios. experiments was successfully completed and results are verified.