

Phakeovilay Andrew

PL/SQL Compte rendu

TP#06 Création d'un paquetage PL/SQL

Table des matières

| | |
|--|---|
| 1 Introduction..... | 3 |
| 2 Mise en place d'un paquetage PL/SQL..... | 3 |
| 2.1 Créez un paquetage PL/SQL Gestion_Rugby dans laquelle vous placerez les fonctions PL/SQL répondant aux spécifications ci-dessous..... | 3 |
| 2.2 Testez les deux fonctions PL/SQL en vérifiant le cas nominal et les cas d'erreur | 4 |
| 2.3 Modifiez la paquetage PL/SQL en ajoutant une procédure classement qui permet d'afficher le classement des équipes en utilisant les fonctions précédentes..... | 6 |
| 3 Conclusion..... | 7 |

PL/SQL Compte rendu

TP#06 Création d'un paquetage PL/SQL

1 Introduction

Ce TP traite du développement d'un paquetage PL/SQL. Nous allons regrouper plusieurs procédures et fonctions PL/SQL au sein d'un paquetage.

2 Mise en place d'un paquetage PL/SQL

2.1 Créez un paquetage PL/SQL Gestion_Rugby dans laquelle vous placerez les fonctions PL/SQL répondant aux spécifications ci-dessous.

```
create or replace PACKAGE Gestion_Rugby IS

    FUNCTION retournePointsMarques(p_nomE EQUIPE.nomE%TYPE) RETURN NUMBER;

    FUNCTION retournePointsAttribues(p_nomE EQUIPE.nomE%TYPE) RETURN NUMBER;

END Gestion_Rugby;
```

```
create or replace PACKAGE BODY Gestion_Rugby IS

    FUNCTION retournePointsMarques(p_nomE EQUIPE.nomE%TYPE) RETURN NUMBER IS
        v_totalPoints NUMBER := 0;
    BEGIN

        SELECT SUM(point) INTO v_totalPoints
        FROM MARQUER MA, EQUIPE E, JOUEUR J
        WHERE MA.nj = J.nj AND J.ne = E.ne AND E.nomE = p_nomE GROUP BY E.ne;

        RETURN(v_totalPoints);

    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            DBMS_OUTPUT.PUT_LINE('Erreur : Equipe inconnu');
            RETURN NULL;
    END retournePointsMarques;
```

```

FUNCTION retournePointsAttribues(p_nomE EQUIPE.nomE%TYPE) RETURN NUMBER IS
    v_totalPointsAttribues NUMBER := 0;
    nb_victoires NUMBER := 0;
    nb_essais NUMBER := 0;
BEGIN
    FOR tuple IN (SELECT nm, ne1, scr1, scr2 FROM MATCH, EQUIPE WHERE ne1 = ne AND nomE = p_nomE
                  UNION
                  SELECT nm, ne2, scr2, scr1 FROM MATCH, EQUIPE WHERE ne2 = ne AND nomE = p_nomE) LOOP
        IF(tuple.scr1 > tuple.scr2) THEN
            v_totalPointsAttribues := v_totalPointsAttribues + 4;
            nb_victoires := nb_victoires + 1;
        ELSIF(tuple.scr1 = tuple.scr2) THEN
            v_totalPointsAttribues := v_totalPointsAttribues + 2;
        ELSIF(tuple.scr1 < tuple.scr2) AND ((tuple.scr1 - tuple.scr2) < 0) THEN
            v_totalPointsAttribues := v_totalPointsAttribues + 1;
        END IF;
        SELECT COUNT(*) INTO nb_essais
        FROM Marquer, Joueur
        WHERE Marquer.nj = Joueur.nj
            AND nm = tuple.nm
            AND ne = tuple.ne1
            AND action = 'Essais';
        IF(nb_essais >= 4) THEN
            v_totalPointsAttribues := v_totalPointsAttribues + 1;
        END IF;
    END LOOP;

    IF(nb_victoires = 5) THEN
        v_totalPointsAttribues := v_totalPointsAttribues + 3;
    END IF;

    RETURN(v_totalPointsAttribues);

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('Erreur : Equipe inconnu');
        RETURN NULL;
END retournePointsAttribues;

```

2.2 Testez les deux fonctions PL/SQL en vérifiant le cas nominal et les cas d'erreur

| Feuille de calcul | | Query Builder |
|-------------------|--|---------------|
| 1 | ACCEPT vNomEquipe PROMPT 'Donnez nom equipe :' | |
| 2 | SELECT Gestion_Rugby.retournePointsMarques('&vNomEquipe') as total_points_marques FROM dual; | |

Sortie de script x

Tâche terminée en 2,131 secondes

```

ancien :SELECT Gestion_Rugby.retournePointsMarques('&vNomEquipe') as total_points_marques FROM dual
nouveau :SELECT Gestion_Rugby.retournePointsMarques('France') as total_points_marques FROM dual

TOTAL_POINTS_MARQUES
-----
                        141
    
```

| Feuille de calcul | | Query Builder |
|-------------------|--|---------------|
| 1 | SET SERVEROUTPUT ON; | |
| 2 | ACCEPT vNomEquipe PROMPT 'Donnez nom equipe :' | |
| 3 | SELECT Gestion_Rugby.retournePointsMarques('&vNomEquipe') as total_points_marques FROM dual; | |
| 4 | | |

Sortie de script x

Tâche terminée en 5,489 secondes

```

ancien :SELECT Gestion_Rugby.retournePointsMarques('&vNomEquipe') as total_points_marques FROM dual
nouveau :SELECT Gestion_Rugby.retournePointsMarques('FRANCE') as total_points_marques FROM dual

TOTAL_POINTS_MARQUES
-----

Erreur : Equipe inconnu
    
```

| Feuille de calcul | | Query Builder |
|-------------------|--|---------------|
| 1 | SET SERVEROUTPUT ON; | |
| 2 | ACCEPT vNomEquipe PROMPT 'Donnez nom equipe :' | |
| 3 | SELECT Gestion_Rugby.retournePointsAttribues('&vNomEquipe') as total_points_marques FROM dual; | |
| 4 | | |

Sortie de script x

Tâche terminée en 2,322 secondes

```

ancien :SELECT Gestion_Rugby.retournePointsAttribues('&vNomEquipe') as total_points_marques FROM dual
nouveau :SELECT Gestion_Rugby.retournePointsAttribues('France') as total_points_marques FROM dual

TOTAL_POINTS_MARQUES
-----
                        25
    
```

```

Feuille de calcul  Query Builder
1 SET SERVEROUTPUT ON;
2 ACCEPT vNomEquipe PROMPT 'Donnez nom equipe : '
3 SELECT Gestion_Rugby.retournePointsAttribues('&vNomEquipe') as total_points_marques FROM dual;
4

Sortie de script x
Tâche terminée en 2,396 secondes

-----
25

ancien :SELECT Gestion_Rugby.retournePointsAttribues('&vNomEquipe') as total_points_marques FROM dual
nouveau :SELECT Gestion_Rugby.retournePointsAttribues('FRANCE') as total_points_marques FROM dual

TOTAL_POINTS_MARQUES
-----

Erreur : Equipe inconnu

```

2.3 Modifiez la paquetage PL/SQL en ajoutant une procédure classement qui permet d'afficher le classement des équipes en utilisant les fonctions précédentes.

```

create or replace PACKAGE Gestion_Rugby IS

    FUNCTION retournePointsMarques(p_nomE EQUIPE.nomE%TYPE) RETURN NUMBER;

    FUNCTION retournePointsAttribues(p_nomE EQUIPE.nomE%TYPE) RETURN NUMBER;

    PROCEDURE classementEquipe;

END Gestion_Rugby;

PROCEDURE classementEquipe IS
BEGIN
    DBMS_OUTPUT.PUT_LINE('Nation | Points Attribués | Points Marqués');
    FOR ligne IN (SELECT nome, retournePointsMarques(nome) as pm, retournePointsAttribues(nome) as pa
                  FROM Equipe ORDER BY pa DESC) LOOP
        DBMS_OUTPUT.PUT_LINE(ligne.nome || ' : ' || ligne.pa || ' | ' || ligne.pm);
    END LOOP;
END classementEquipe;

```

```

Nation | Points Attribués | Points Marqués
France : 25 | 141
Irlande : 21 | 168
Ecosse : 12 | 85
Angleterre : 12 | 101
Italie : 8 | 60
Galles : 8 | 70

```

3 Conclusion

Nous avons appris comment créer des paquetages en 2 parties, la spécification et le body, nous avons également écrit des fonctions.