

TD 2

Exercice 1

Écrire un bloc PL/SQL contenant un curseur implicite qui affiche le nom de chercheurs du labo numéro 10. Le Bloc affiche le message « Plusieurs chercheurs Trouvés » si le curseur retourne plusieurs lignes.

DECLARE

V_Cnom Chercheur.Cnom%TYPE;

BEGIN

SELECT Cnom INTO V_Cnom FROM Chercheur WHERE Labno=10;

DBMS_OUTPUT.PUT_LINE('Nom Chercheur' || V_Cnom);

Exception

WHEN TOO_MANY_ROWS THEN

DBMS_OUTPUT.PUT_LINE('Plusieurs chercheurs Trouvés.');

END;

Exercice 2

Écrire un bloc PL/SQL contenant un curseur implicite qui retourne toutes les informations relatives au cours d'identifiant 777 de la table Cours. Le bloc affiche le message « Aucune donnée retournée » si la table ne contient pas ce cours.

DECLARE

Course_Rec Cours%ROWTYPE;

BEGIN

SELECT * INTO Course_Rec FROM Course WHERE Course_ID = '777';

EXCEPTION

WHEN No_Data_Found THEN

dbms_output.put_line('Aucune donnée retournée');

WHEN others THEN

null;

END;

/

Aucune donnée retournée

Procédure PL/SQL terminé avec succès.

Exercice 3

Écrire un bloc PL/SQL qui utilise un curseur explicite pour parcourir la table "departments". Pour chaque département, essayez d'insérer une nouvelle ligne dans la table "employees" avec un salaire nul. Gérez l'exception lorsque le salaire est nul et affichez un message approprié.

Exercice 4

Écrire un script qui permet à un employé saisi par l'utilisateur de modifier le salaire. Contrôler que le salaire ne soit pas inférieur au salaire actuel; si c'est le cas, lancer une exception.

Exercice 5

Créer un script PL/SQL qui permet de saisir les informations d'un employé et de les insérer dans la table EMPLOYES. Si l'âge de l'employé est inférieur à 18 ans, une exception doit être lancée.

Exercice 6

1. Ecrire un bloc PL/SQL qui parcourt les produits d'une table Produit et s'arrête au premier produit dont le prix dépasse 10000, affiche le produit et son prix, puis affiche le message « prix élevé » et supprime la ligne de la table.

DECLARE

```
CURSOR c IS SELECT * FROM produits;
```

```
depassement EXCEPTION;
```

```
articleId NUMBER(4);
```

BEGIN

```
FOR e IN c LOOP
```

```
  IF e.prix > 10000 THEN
```

```
    dbms_output.put_line('Article: ' || e.article || ' Prix: ' || e.prix);
```

```
    articleId := e.id;
```

```
    RAISE depassement;
```

```
  END IF;
```

```
END LOOP;
```

EXCEPTION

```
  WHEN depassement THEN
```

```
    DBMS_OUTPUT.PUT_LINE ('prix élevé');
```

```
    DELETE FROM produit WHERE id=articleId;
```

```
END;  
/
```

2. Modifie le code pour afficher un message d'erreur « prix élevé »

```
DECLARE
```

```
CURSOR c IS SELECT * FROM produits;
```

```
depassement EXCEPTION;
```

```
articleId NUMBER(4);
```

```
BEGIN
```

```
FOR e IN c LOOP
```

```
IF e.prix > 10000 THEN
```

```
dbms_output.put_line('Article: ' || e.article || ' Prix: ' || e.prix);
```

```
articleId := e.id;
```

```
RAISE depassement;
```

```
END IF;
```

```
END LOOP;
```

```
EXCEPTION
```

```
WHEN depassement THEN
```

```
DELETE FROM produit WHERE id=articleId;
```

```
RAISE_APPLICATION_ERROR(-20100, 'prix élevé');
```

```
END;  
/
```