

TD Bloc PL/SQL

Partie I.

1. Créer un bloc anonyme qui affiche à l'écran la phrase "Mon bloc PL/SQL marche".

```
G_MESSAGE
```

```
-----
```

```
Mon bloc PL/SQL marche
```

2. Créer un bloc qui déclare deux variables. Affectez la valeur de ces variables à des variables hôtes SQL*PLUS et affichez le résultat des variables PL/SQL à l'écran. Exécutez votre bloc et enregistrez le sous p16q4.sql.

```
V_CHAR Character (variable length)
```

```
V_NUM Number
```

Affectez des valeurs aux variables suivantes :

```
Variable Value
```

```
-----
```

```
V_CHAR la constante '42 is the answer'
```

```
V_NUM Les deux premières caractères de V_CHAR
```

```
G_CHAR
```

```
-----
```

```
42 is the answer
```

```
G_NUM
```

```
-----
```

```
42
```

Partie II.

1. Créez et exécutez un bloc PL/SQL qui prenne en compte deux nombres grâce à des variables de substitution SQL*Plus. Il faut diviser le premier par le second et ajouter le résultat au second nombre. Le résultat doit être stocké dans une variable PL/SQL et affiché à l'écran.

- a. Quand une variable PL/SQL est utilisée :

```
Please enter the first number: 2
```

```
Please enter the second number: 4
```

```
4.5
```

```
PL/SQL procedure successfully completed.
```

- b. Quand une variable SQL*Plus SQL est utilisée :

```
Please enter the first number: 2
```

```
Please enter the second number: 4
```

```
PL/SQL procedure successfully completed.
```

```
G_RESULT
```

```
-----
```

```
4.5
```

Partie III :

1. Créez un bloc PL/SQL qui sélectionne le plus grand numéro de département (DEPTNO) de la table DEPT et le stocke dans une variable SQL*Plus. Affichez le résultat à l'écran. enregistrez votre bloc PL/SQL dans le fichier p18q1.sql.

```
G_MAX_DEPTNO
```

```
-----
```

```
40
```

2. Modifier le bloc PL/SQL que vous venez de créer afin d'insérer à nouveau département dans la table DEPT. Sauvegarder sous p18q2.sql.

- a. Plutôt que d'ajouter le numéro de département extrait dans la question précédente, ajoutez 10 à ce numéro et utilisez ce nouveau numéro pour créer le nouveau département.

- b. Utilisez une variable de substitution SQL*Plus pour le nom de département.

- c. Laissez la localité (LOC) à la valeur NULL pour l'instant.

- d. Exécutez le bloc PL/SQL.

```
Please enter the department name: EDUCATION
```

```
PL/SQL procedure successfully completed.
```

- e. Affichez le nouveau département que vous avez crée.

```
DEPTNO DNAME LOC
```

```
-----
```

```
50 EDUCATION
```

3. Créer un bloc PL/SQL qui mette à jour la localité (LOC) d'un département existant. Sauvegarder sous p18q3.sql.

- a. Utilisez une variable de substitution SQL*Plus pour le numéro de département.
- b. Utilisez une variable de substitution SQL*Plus pour la location de département.
- c. Testez le bloc PL/SQL.

```
Please enter the department number: 50
Please enter the department location: HOUSTON
PL/SQL procedure successfully completed.
```

- d. Affichez le numéro, nom et localité du département que vous venez de modifier.

```
DEPTNO DNAME LOC
-----
50 EDUCATION HOUSTON
```

4. Créer un bloc PL/SQL qui supprime le département que vous avez crée dans la section précédente. Sauvegarder sous p18q4.sql.

- a. Utilisez une variable de substitution SQL*Plus pour le numéro de département.
- b. Affichez à l'écran le numéro, de lignes affectées.
- c. Testez le bloc PL/SQL.

```
Please enter the department number: 50
PL/SQL procedure successfully completed.
G_RESULT
-----
1 row(s) deleted.
```

- d. Qu'est-ce qui se passe si vous entrez un numéro de département qui n'existe pas ?

```
Please enter the department number: 99
PL/SQL procedure successfully completed.
G_RESULT
-----
0 row(s) deleted.
```

Partie IV :

- 1. Créer un bloc PL/SQL qui sélectionne le nom du département dans la table DEPT et affichez le nom de chaque département à l'écran, en utilisant une table PL/SQL.
- a. Déclarer une table PL/SQL, MY_DEPT_TABLE, pour stocker temporairement le nom de ces départements.
- b. En utilisant une boucle, retrouvez le nom de tous les départements de la table DEPT et stockez-les dans la table PL/SQL table. Chaque département a un numéro multiple de 10.
- c. En utilisant une autre boucle, retrouvez les noms des départements dans la table PL/SQL et affichez les en écran en utilisant DBMS_OUTPUT.PUT_LINE.

```
SQL> START p20_1
ACCOUNTING
RESEARCH
SALES
OPERATIONS
PL/SQL procedure successfully completed.
```

- 2. Modifier le bloc PL/SQL que vous venez de créer pour retrouver toutes les informations relatives à chaque département de DEPT et affichez ces informations à l'écran en utilisant une table de Records PL/SQL.
- a. Déclarer une table PL/SQL, MY_DEPT_TABLE, pour stocker temporairement le nom de ces départements.
- b. En utilisant une boucle, retrouvez le nom de tous les départements de la table DEPT et stockez-les dans la table PL/SQL table. Chaque département a un numéro multiple de 10.
- c. En utilisant une autre boucle, retrouvez les noms des départements dans la table PL/SQL et affichez les en écran en utilisant DBMS_OUTPUT.PUT_LINE.

```
SQL> START p20_3
Dept. 10, ACCOUNTING is located in NEW YORK
Dept. 20, RESEARCH is located in DALLAS
Dept. 30, SALES is located in CHICAGO
Dept. 40, OPERATIONS is located in BOSTON
PL/SQL procedure successfully completed.
```

Partie V : Les triggers

Ecrivez un trigger permettant de garantir que les salaires des employés ne peuvent qu'être augmentées. Testez son utilisation.