

## Exercices PL-SQL (Série 3 : Exceptions et Procédures)

### Exercice 1

*Dans cet exercice, vous devez créer des gestionnaires d'exceptions pour des situations particulières.*

1. Ecrivez un bloc PL/SQL permettant de sélectionner le nom d'un employé à partir du montant de son salaire.

a. Utilisez la commande `DEFINE` pour indiquer le salaire.

b. Transmettez cette valeur au bloc PL/SQL à l'aide d'une variable de substitution `SQL*Plus`. Si le salaire entré renvoie plusieurs lignes, traitez l'exception à l'aide d'un gestionnaire approprié et insérez dans la table `MESSAGES` le message "Plusieurs employés ont ce salaire : <salaire>."

c. Si le salaire entré ne renvoie aucune ligne, traitez l'exception à l'aide d'un gestionnaire approprié et insérez dans la table `MESSAGES` le message "Pas d'employé avec ce salaire : <salaire>."

d. Si le salaire entré ne renvoie qu'une seule ligne, insérez dans la table `MESSAGES` le nom de l'employé et le montant de son salaire.

e. Traitez toutes les autres exceptions à l'aide d'un gestionnaire approprié et insérez dans la table `MESSAGES` le message "Autre erreur rencontrée."

f. Testez le bloc selon différents cas. Affichez les lignes de la table `MESSAGES` pour vérifier si le bloc PL/SQL s'est correctement exécuté.

2. Modifiez le code PL/SQL de la question 3 exercice 3 pour ajouter un gestionnaire d'exceptions.

a. Utilisez la commande `DEFINE` pour fournir le numéro et l'emplacement du service. Transmettez ces valeurs au bloc PL/SQL via des variables de substitution `SQL*Plus`.

b. Ecrivez un gestionnaire d'exceptions pour transmettre un message à l'utilisateur indiquant que le service spécifié n'existe pas. Utilisez une variable attachée (bind variable) pour transmettre le message à l'utilisateur.

c. Exécutez le bloc PL/SQL en entrant un service qui n'existe pas.

3. Ecrivez un programme PL/SQL permettant d'afficher le nombre d'employés qui gagnent 100 \$ de plus ou de moins que le montant du salaire défini pour une variable de substitution `SQL*Plus`.

Utilisez la commande `DEFINE` pour indiquer le salaire. Transmettez cette valeur au bloc PL/SQL à l'aide d'une variable de substitution `SQL*Plus`.

a. S'il n'existe pas d'employés dans cette tranche de salaire, affichez un message à l'utilisateur à l'aide d'une exception.

b. S'il existe au moins un employé dans cette tranche de salaire, le message doit indiquer le nombre d'employés.

c. Traitez toutes les autres exceptions à l'aide d'un gestionnaire approprié. Le message doit indiquer qu'un autre type d'erreur est survenu.

```
DEFINE p_sal = 7000
```

```
DEFINE p_sal = 2500
```

```
DEFINE p_sal = 6500
```

### Exercice 2

*Le bloc PL/SQL d'un programme client est transmis au serveur pour exécution. Une procédure cataloguée est stockée en permanence sur le serveur, le poste client envoie l'ordre*

*d'exécution avec les valeurs des paramètres. Dans cet exercice vous allez écrire des fonctions et procédures PL/SQL.*

1. Ecrire une fonction `MAJORER(N NUMBER, X NUMBER)` qui permet de majorer un nombre `n` donné par un autre nombre `x` donné. Exemple : Si `n=10` et `x=0,1` la fonction retourne `10 * 1,1 = 11`

2. Ecrire un programme PL/SQL qui permet de majorer de 10% les salaires des employés du département numéro 10 et les salaires des employés du département numéro 20 de 15%. Utiliser la fonction `MAJORER`.

3. Ecrire une fonction `TOTAL_SAL` qui permet de calculer le total des salaires des employés d'un département donné. 2) Ecrire un programme PL/SQL qui permet d'afficher le total des salaires des employés pour chaque département. Utiliser la fonction `TOTAL_SAL`. L'affichage doit être comme suit :

```
NUM DEPT: 10 TOTAL SAL: 9735  
NUM DEPT: 20 TOTAL SAL: 12506,25  
NUM DEPT: 30 TOTAL SAL: 9400 ...
```

4. Ecrire une procédure `INFO_EMP` qui affiche le nom, le salaire et la fonction d'un employé donné. 2) Ecrire une procédure `PERSO_DEP` qui prend entrée le nom d'un département donné et affiche tout les personnels (nom, salaire et fonction) de cet département. Utiliser la procédure `INFO_EMP`.