

## - Oracle & Langage PL/SQL -

### TP 1 : Fonctions, Procédures & Exceptions

#### Exercice 1 :

Soit la table suivante: **Pilote**(Matricule, Nom, Ville, Age, Salaire).

1. Écrivez un programme PL/SQL qui calcule la moyenne des trois plus grands salaires de pilote.
2. Ecrire un programme PL/SQL qui calcule la moyenne des salaires des pilotes dont l'âge est entre 30 et 40 ans.

#### Exercice 2 :

Écrire une fonction stockée qui prend en entrée un nombre entier N et qui retourne la somme de tous les nombres divisibles par 3 ou par 5 et inférieur ou égale à N.

#### Exercice 3 :

Écrire une fonction PL/SQL stockée (en version récursive et itérative) qui retourne le n-ème terme de Fibonacci :

$$F_0 = 0, \quad F_1 = 1, \text{ et } F_n = F_{n-1} + F_{n-2}$$

Tester les deux fonctions dans un programme PL/SQL en demandant à l'utilisateur de saisir le rang du terme calculé.

#### Exercice 4 :

À la fin de chaque année, le fournisseur des produits demande à l'association un inventaire sur les produits vendus. L'information demandée est la suivante : donner pour chaque produit, sa désignation, son prix actuel, et une mention sur la demande :

- demande "*forte*" si le nombre d'articles vendus dépasse 15,
- demande "*moyenne*" si le nombre d'articles vendus est compris entre 11 et 15,
- et demande "*faible*" sinon.

1. Créer les tables suivantes et remplites les par un minimum de données.

**Client** (#num:integer, nom:string, prenom:string, adresse:string, date\_nais:date, tel:string, sexe:char={'m','f'})

**Produit** (#num:integer, designation:string, prix:float, stock:integer)

**Facture** (#num:integer, date\_etabli:date, client=>Client(num))

**Ligne\_Fact** (#facture=>Facture(num), #produit=>Produit(num), qte:integer)

1. Écrire une fonction stockée qui retourne la mention de demande en fonction d'un numéro de produit.
2. Écrire un programme PL/SQL qui utilise la fonction précédente pour afficher la liste des produits : *num*, *designation* et *mention* sur la demande.
3. Écrire un code PL/SQL permettant d'obtenir le résultat précédent sous la forme ci-dessous et le sauvegardé dans un fichier txt.

*Num\_produit (nom\_produit) : mention\_produit*

### Exercice 5 :

Afin d'affiner l'analyse du profil de chaque client, le service marketing souhaite avoir un état pour chaque client avec le nombre de ses factures et le total d'achat.

1. Écrire deux fonctions stockées qui prendront en argument un numéro de client, et retourneront le nombre de factures et le total d'achat, respectivement.
2. Écrire un programme PL/SQL qui affiche pour chaque client son nombre de factures et son total d'achat.
3. Transformez le bloc anonyme en procédure. Testez cette procédure.
4. Écrire une procédure qui permet de supprimer les clients qui n'ont pas réalisé un total d'achat supérieur ou égale à 1500 dh. Testez cette procédure.
5. Écrire une procédure qui stocke les noms des clients ayant un nombre de facture supérieur à 2, dans un tableau indexé par des entiers. Pensez à gérer l'exception des données inexistantes ainsi que l'exception de clients infidèles (ie. Tous les clients ont un nombre de factures inférieur à 2).