

## TP 1 : Les fonctions SQL/PSM

### Exercice 1 :

Soit le modèle relationnel suivant :

**Employé** (Matricule, Nom\_Emp, Prénom\_Emp, DateNaissance\_Emp, Salaire\_Emp, #Num\_Dep)

**Département** (Num\_Dep, Nom\_Dep, Ville\_Dep)

**Vente** (Num\_Vente, #Matricule, Date\_Vente, Montant)

1. Créer une fonction qui retourne le salaire moyen des employés dans un département donné.
2. Créer une fonction stockée qui calcule la différence de salaire entre deux employés spécifiés par leur matricule.
3. Créer une fonction qui retourne le nombre total d'employés dans une ville donnée.
4. Créer une fonction qui retourne le nom complet (prénom et nom) de l'employé ayant le salaire le plus élevé.
5. Créer une fonction qui retourne le nom complet de l'employé le plus âgé.
6. Créer une fonction qui retourne le montant total des ventes réalisées par un employé spécifié au cours de l'année spécifiée.

### Exercice 2 :

On considère le schéma relationnel suivant :

**CLIENT** (codeclt, nomclt, prenomclt, adresseclt, CPclt, villeclt)

**APPARTEMENT** (ref, superficie, pxvente, secteur, #coderep, #codeclt)

**REPRESENTANT** (coderep, nomrep, prenomrep)

1. Créer une fonction qui retourne le nombre d'appartements vendus par le représentant passé en paramètre (coderep).
2. Créer une fonction qui calcule la somme des superficies des appartements à la charge du représentant dont le nom est code en paramètre.
3. Créer une fonction qui calcule la recette (total vente) du représentant dont le est passé en paramètre.

4. Créer une fonction qui retourne le nombre de clients du représentant dont le nom complet est passé en paramètre.

**Exercice 3 :**

Soit le modèle relationnel suivant :

CLIENT (**codeclt**, nomclt, prenomclt, adresse, cp, ville)

PRODUIT (**référence**, désignation, prix)

TECHNICIEN (**codetec**, nomtec, prenomtec, tauxhoraire)

INTERVENTION (**numero**, date, raison, #codeclt, #référence, #codetec)

1. Créer une fonction qui prend une ville en paramètre et retourne le nombre de clients qui habitent cette ville.
2. Créer une fonction qui retourne le nombre d'interventions effectuées par le technicien dont le nom est passé en paramètre.
3. Créer une fonction qui retourne la somme des prix des produits qui ont subi une intervention entre deux dates passées en paramètre et qui sont à la propriété du client auquel le nom est passé en paramètre.
4. Créer une fonction qui retourne le nombre de villes dans lesquelles le technicien auquel le nom est passé en paramètre a intervenu.
5. Créer une fonction qui retourne le nombre de clients qui n'ont pas demandé d'interventions entre deux dates passées en paramètre.