

CYCLE DE FORMATION D'INGENIEURS - TROISIEME ANNEE

TP N° 5

GESTION DES EXCEPTIONS DANS PL/SQL

Dans la suite, nous considérons le schéma relationnel suivant d'une base de données :

Emp(NumEmp, NomEmp, PrenomEmp, Fonction, Adresse, Sexe, AgeE, Salaire, Commission, #NumChef, #NumDept)

Dept(NumDept, NomDept, Adresse, #NumDirecteur)

Grade(NumGrade, NomGrade, SalaireMin, SalaireMax)

Projet(NumProjet, NomProjet, Local, #NumDept)

TravaillerSur(#NumEmp, #NumProjet, NbreH)

Candidat(NumDos, NomC, PrenomC, Poste, Adresse, Sexe, AgeC, NbrAnneeExp, DateD, Accep)

Remarque : Les clés primaires sont soulignées et les clés étrangères sont marquées par #.

Le grade d'un employé est déterminé en comparant son salaire au salaire minimal et au salaire maximal associés aux différents grades. Seul un grade correspond à tout employé.

La table « Candidat » sert à enregistrer les demandes d'emploi. Les attributs de cette table ont la signification suivante :

- NumDos : c'est le numéro du dossier déposé par un candidat. On garde ce même numéro pour l'employé s'il est accepté.
- NomC, PrenomC, AgeC, Adresse, Sexe : informations personnelles concernant le candidat,
- Poste : c'est la fonction à laquelle le candidat postule,
- NbrAnneeExp : nombre d'années d'expérience du candidat,
- DateD : date du dépôt du dossier de candidature sous la forme d'une date (par exemple '16/07/1965'),
- Accep : ce champ vaut 0 si le candidat n'est pas accepté, et 1 s'il est accepté.

Exercice 1 : Ecrire un bloc anonyme PL/SQL permettant d'ajouter un montant T au salaire d'un employé dont le nom est nameE et dont le prénom est firstNameE.

Les valeurs de T, de nameE et de firstNameE seront saisies moyennant des variables de substitution avec la commande :

ACCEPT nomVariable PROMPT ' : '

La gestion des exceptions dans le bloc PL/SQL est à effectuer versus :

- 1- Le montant T qui doit être strictement positif.
- 2- Une exception est à lever si aucun employé de nom nameE et de prénom firstNameE n'existe. Un message d'erreur est alors à afficher indiquant l'inexistence de tel employé.
- 3- Une exception est à lever si plus qu'un employé existe. Dans ce cas, l'augmentation doit être effectuée pour tous les employés portant le nom nameE et le prénom firstNameE.
- 4- Si un seul employé de nom nameE et de prénom firstNameE existe, son salaire initial ne doit pas être nul, sinon une exception sera lancée.

Exercice 2 : Ecrire une fonction stockée verifProjet qui prend en paramètre un numéro de projet numP et un nombre d'heures nbH et qui retourne moyennant la gestion des exceptions :

- 0 si le numéro ne représente pas un numéro de projet valide
- 1 si aucun employé n'a été affecté à ce projet
- 2 si la moyenne du nombre d'heures de travail sur le projet dépasse de nbH la moyenne générale des moyennes du nombre d'heures de travail de tous les projets
- 3 sinon.