

Package

***** Remplacer *votrenom* dans le sujet par votre nom ou votre login !**

- Créer un package qui implémente l'interface suivante (interface = la partie spécification):
 - ⇒ Curseurs : emp_par_dep_**votrenom** (deptno). /* curseur de package pour le numéro et le nom des employés dans le département donné */
 - ⇒ Procédures : raise_salary_**votrenom**, /* réutiliser la procédure raise_salary_**votrenom** du TP précédant */
afficher_emp_**votrenom** (deptno) /* afficher le numéro et le nom des employés dans le département donnée, utiliser le curseur emp_par_dept_**votrenom** */.

Triggers

- Implémenter les contraintes suivantes :
 - 1) (Nom du trigger : raise_**votrenom**) Le salaire d'un employé ne diminue jamais.
 - 2) (Nom du trigger : numdept_**votrenom**) Le numéro de département doit être entre 61 - 69.
 - 3) (Nom du trigger : dept_**votrenom**) Si un employé est affecté à un département qui n'existe pas dans la base de données, ce département doit être rajouté avec pour valeur « A SAISIR » pour les attributs Dname et Loc.
 - 4) (Nom du trigger : nowweek_**votrenom**) Pour des raisons de sécurité, on souhaite interdire toute modification de la relation employé pendant le week-end (samedi et dimanche).
 - 5) Désactiver le trigger de la question précédente et tester le.
 - 6) Réactiver le trigger la question précédente.
 - 7) (Nom du trigger : stat_**votrenom**) On souhaite conserver des statistiques concernant les mises à jour (insertions, modifications, suppressions) sur la table EMP. Créer à l'aide de SQL la structure de la table STATS_**votrenom** et la peupler comme suit.

STATS_**votrenom** (TypeMaj, NbMaj, Date_derniere_Maj)

TypeMaj	NbMaj	Date_derniere_Maj
INSERT	0	NULL
UPDATE	0	NULL
DELETE	0	NULL

- A) Définir un déclencheur après insertion/modification/suppression de la table EMP permettant de mettre à jour automatiquement la table STATS_**votrenom**. Tester son utilisation en effectuant diverses mises à jour sur la table EMP.

B) Tester l'effet de la présence et de l'absence de la clause FOR EACH ROW sur le comportement du déclencheur en utilisant une requête qui modifie plusieurs n-uplets (ex. UPDATE EMP SET SAL = SAL * 1.05;)

- 8) (Nom du trigeur : checksal_*votrenom*) Chaque fois qu'un employé change de « job », on lui accorde une augmentation de salaire d'un montant de 100 euros. Attention ! le président (avant le changement) n'est pas concerné car il est déjà bien payé... Le nouveau salaire doit être dans l'intervalle de salaires (entre lsal et hsal dans la table SalIntervalle du TP précédant) pour le nouveau job. Si le nouveau salaire est supérieur à la valeur de la colonne hsal, dans ce cas, il faudrait lui affecter la valeur de hsal et si le nouveau salaire est inférieur à de la valeur lsal, affecter lui la valeur de lsal.

NB) A la fin du TP, effacer tous vos objets procéduraux (package, procédure, fonctions, trigger, ...) sur la machine.