

 Ecole Supérieure Privée d'Ingénierie et de Technologies	Année Universitaire : 2013-2014 EXAMEN Session : Principale	
Module : Système de Gestion des Bases de Données	Documents autorisés : NON	
Enseignants : F.OUELBANI, H.BELLAKHDHAR I.BAKLOUTI, I.CHANNOUFI, I.SLIMENE, J.HKIMI, M.JEBARI, S.YEKKEN, W.SAIDI, Y.GASRI	Nombre de pages : 2	
	Classes : 3A	
Date : 26/05/2014	Heure : 9H00	Durée : 1h30

Soit le schéma relationnel suivant :

ENSEIGNANTSCHERCHEURS (Cin_ens, nom_ens, prenom_ens, spécialité, grade)
PROJETS (code_projet, nom_projet, date_lancement, duree)
INTERVENTIONS (cin_ens#, code_projet#, date_intervention)

1. Ecrire une procédure stockée nommée « proc_insertion » prenant comme paramètres les valeurs à insérer et permettant d'ajouter un enregistrement dans la table interventions. Cette procédure doit traiter les exceptions suivantes :
 - Vérifier si la CIN_ENS existe dans la table ENSEIGNANTSCHERCHEURS,
 - Vérifier si le CODE_PROJET existe dans la table PROJETS,
 - Vérifier si l'intervention existe déjà,
 - Si la table est vide afficher le message suivant : « pas d'interventions ». **(5pts)**
2. Ecrire une fonction stockée nommée « fn_nb_intervenants » prenant comme paramètre le nom d'un projet et retourne le nombre de ses intervenants (nb).
Si le projet n'existe pas : nb = -1. **(3pts)**
3. Ecrire une procédure stockée nommée « proc_nb_intervenants » permettant d'afficher pour chaque projet le NOM_PROJET et le nombre de ses intervenants en utilisant la fonction stockée « fn_nb_intervenants ». Cette procédure retourne le nombre de projets traités. **(3pts)**
4. Ecrire un bloc anonyme faisant appel à la procédure stockée « proc_nb_intervenants » définie dans (3). **(2pts)**

5. Ecrire une procédure stockée nommée « proc_liste_projets » permettant de lister les projets comportant au maximum 3 intervenants et d'afficher les détails de chacun de ses intervenants. Utilisez la fonction stockée « fn_nb_intervenants » définie dans (2).
(4pts)
6. Ecrire un trigger qui se déclenche avant chaque mise à jour et insertion sur la table interventions vérifiant que :
DATE_INTERVENTION >=DATE_LANCEMENT.
Sinon DATE_INTERVENTION=date système **(3pts)**
-

Annexe :

```
CREATE [OR REPLACE] FUNCTION nom_fonction
[(argument1,...argumentN)]
RETURN type_donnée IS
[Déclaration des variables locales]
BEGIN
<Instructions ;>
Return valeur ;
END ;
/
```

```
CREATE [OR REPLACE] PROCEDURE nom_procedure
[(argument1,...argumentN)] IS
[Déclaration des variables locales]
BEGIN
<Instructions ;>
END ;
/
```

```
CREATE [OR REPLACE] TRIGGER nom_trigger
{BEFORE|AFTER|INSTEAD OF}
{INSERT|UPDATE|DELETE} OR {INSERT|UPDATE|DELETE} OR
{INSERT|UPDATE|DELETE} OR UPDATE of COLUMN
ON nom_table
FOR EACH ROW
DECLARE <declarations>
BEGIN
<Instructions ;>
END;
/
```

BONNE CHANCE