

TP : SGBD 2

Exercice 1 :

Une application de gestion des résultats des matchs de football de la saison 2011-2012 utilise la base de données suivante :

| Table Equipe | | |
|--------------------------|--------------|-------------------|
| Colonne | Type | Signification |
| <u>codeEquipe</u> | Entier | code d'une équipe |
| nomEquipe | Alphabétique | Nom d'une équipe |

| Table Stade | | |
|-------------------------|--------------|---------------|
| Colonne | Type | Signification |
| <u>codeStade</u> | Entier | code du stade |
| nomStade | Alphabétique | Nom du stade |

| Table Match | | |
|------------------------|-------------|------------------------------------------|
| Colonne | Type | Signification |
| <u>numMatch</u> | Entier | Numéro du match |
| dateMatch | Date | Date du match |
| nombreSpectateur | Entier long | Nombre de spectateurs du match |
| numJournee | Entier | Numéro de la journée |
| #codeStade | Entier | code du stade |
| #codeEquipeLocaux | Entier | code de l'équipe Locaux |
| #codeEquipeVisiteurs | Entier | code de l'équipe Visiteurs |
| nombreButLocaux | Entier | Nombre de buts marqués par les locaux |
| nombreButVisiteurs | Entier | Nombre de buts marqués par les visiteurs |

Les clés primaires sont en gras et soulignées ; les clés étrangères sont marquées par un #.

Un match se joue entre une équipe locale et une équipe visiteur dans un stade donné et pour une journée du championnat national (journée 1, 2 , ...). La table Match enregistre également le nombre de buts marqués par l'équipe des locaux et le nombre de buts marqués par l'équipe des visiteurs.

- 1) Ecrire une requête qui affiche le nombre de matchs joués dans la journée n°**12**. (0,5 pt)
- 2) Ecrire une requête qui affiche le nombre de matchs joués par journée. (0,5 pt)
- 3) Ecrire une requête qui affiche le match qui a compté le plus grand nombre de spectateurs. (1pt)
- 4) Ecrire une requête qui affiche le nombre de points de l'équipe de code **112** ; le nombre de points se calcule de la façon suivante : une victoire donne 3 points, une égalité donne 1 point et une défaite donne 0 point. (1 pt)

- 5) Ecrire une procédure stockée qui affiche les équipes qui ont gagné leur match dans une journée dont le numéro est donné comme paramètre. (1 pt)
- 6) Ecrire un trigger qui refuse l'ajout d'une ligne à la table Match pour laquelle la colonne codeEquipeLocaux est égale à la colonne codeEquipeVisiteurs. (1 pt)

Exercice 2 :

Pour gérer les élections des représentants des employés d'une entreprise, une application utilise une base de données composée des tables Electeurs, Candidats et Votes :

- **Electeurs** (idElecteur , nomElecteur, prenomElecteur, aVoté)
Le champ aVoté prend la valeur 0 quand l'électeur n'a pas encore voté et 1 quand il a voté.
- **Candidats** (idCandidat , nomCandidat, prenomCandidat, dateNaissance, nombreVoix)
nombreVoix est le nombre de voix obtenus par le candidat.
- **Votes** (#idElecteur , #idCandidat)
Cette table enregistre les votes des électeurs.

N.B : Les champs en **gras et soulignés** sont des clés **primaires**, les champs précédés par **#** sont des clés **étrangères**.

Un électeur ne peut voter qu'une seule fois pour choisir 1, 2 ou 3 candidats.

1. Ecrire une requête qui affiche la liste des électeurs qui ont choisi un seul candidat pendant le vote. (2 pts)
2. Ecrire une requête qui affiche les trois premiers candidats qui ont gagné les élections (ceux qui ont obtenu le plus grand nombre de voix). En cas d'égalité des nombres de voix, on retient le candidat le plus âgé. (3 pts)
3. Ecrire une fonction qui retourne dans une table la liste des électeurs ayant votés pour un candidat donné. (2 pts)
4. Ecrire un trigger qui permet d'incrémenter de 1, le champ nombreVoix d'un candidat à chaque ajout d'une ligne à la table **Votes** qui concerne ce candidat. Le trigger doit également mettre le champ aVoté à 1 pour l'électeur qui vient de voter. (3 pts)
5. Ecrire une procédure stockée qui permet d'enregistrer le vote d'un électeur ; elle a les paramètres : (3 pts)
 - @idElect : identifiant de l'électeur.
 - @idCandidat1, @idCandidat2 et @idCandidat3 : identifiants des 3 candidats choisis par l'électeur (si l'électeur choisit moins de 3 candidats, les valeurs non choisies restent NULL).
 - La procédure ajoute 1 à 3 lignes à la table Votes selon les valeurs non NULL des paramètres @idCandidat1, @idCandidat2 et @idCandidat3.

Si ces paramètres sont tous NULL, la procédure affiche un message d'erreur.