



Epreuve d'informatique

2^{ème} semestre AU : 2024 – 2025

Section : MP2/PC2/PT2

Date : Février 2025

Durée : 1H

Nombre de page : 2

Contexte : Tournoi de ping pong

Dans le cadre de l'organisation de tournois de ping-pong, il est nécessaire de mettre en place une base de données pour gérer les informations relatives aux joueurs, arbitres, tournois, matchs. La base de données doit permettre un suivi détaillé et précis des participants, des résultats des matchs, des arbitres et de l'organisation des tournois.

Structure de la base de données :

1. Table établissement :

- **id_etablissement** : entier (Clé primaire) : identifiant unique de l'établissement.
- **nom** : chaîne de caractères : nom de l'établissement.
- **adresse** : chaîne de caractères : adresse de l'établissement.

2. Table joueur :

- **cin** : chaîne de caractères (Clé primaire) : identifiant unique du joueur (CIN).
- **nom** : chaîne de caractères : nom du joueur.
- **date_naissance** : DATE : date de naissance du joueur de la forme 'aaaa-mm-jj'
- **type_joueur** : chaîne de caractère ∈ ('étudiant', 'enseignant', 'personnel')
- **id_etablissement** : entier : clé étrangère référence à l'établissement

3. Table arbitre :

- **cin** : chaîne de caractères (Clé primaire) : identifiant unique de l'arbitre (CIN).
- **nom** : chaîne de caractères : nom de l'arbitre.
- **experience** : entier : nombre d'années d'expérience de l'arbitre.

4. Table tournoi :

- **nom** : chaîne de caractères (Clé primaire) : nom du tournoi.
- **année** : entier : année du déroulement du tournoi
- **id_etablissement** : entier : clé étrangère référence à l'établissement organisateur du tournoi.

5. Table match :

- **id_tournoi** : chaîne de caractères : référence au tournoi auquel appartient le match.
- **cin_joueur1** : chaîne de caractères : référence au premier joueur du match.
- **cin_joueur2** : chaîne de caractères : référence au second joueur du match.
- **cin_arbitre** : chaîne de caractères : référence à l'arbitre du match.
- **score** : chaîne de caractères : résultat du match de la forme 'points joueur1-points joueur2', par exemple "21-16".
- **Clé primaire** : (id_tournoi, cin_joueur1, cin_joueur2)

Règles de gestion :

1. Les joueurs sont affiliés à un seul établissement.
2. Un joueur peut arbitrer un match
3. Un arbitre peut être impliqué dans plusieurs matchs
4. Dans une même année, on peut organiser plusieurs tournois.
5. Le score d'un match est enregistré sous forme de texte et permet de déterminer le gagnant.

Travail demandé

On suppose que les tables de la base de données sont créées et remplies en respectant les contraintes et les règles de gestion.

Algèbre Relationnelle

Exprimer en algèbre relationnelle les requêtes permettant de :

- Q1.** Lister les cin des joueurs qui n'ont joué dans aucun match pour les tournois de l'année 2025.
- Q2.** Lister les cin et les noms des arbitres ayant arbitré dans les tournois de 2024 et 2025, en même temps (c'est-à-dire ayant arbitré à la fois dans les deux années).

SQL

Exprimer en sql les requêtes permettant de :

- Q3.** Calculer le nombre de matchs que chaque joueur a joué dans les tournois organisés dans l'année 2024.
- Q4.** Lister tous les matchs du tournoi 'ipeis_2025' en affichant les noms des joueurs, le nom de l'arbitre et le score.
- Q5.** Modifier le type des joueurs à la valeur 'enseignant' pour les joueurs affiliés à l'établissement d'identifiant 'IPEIS' et qui ont joué un match dans le tournoi 'ipeis_2025'.
- Q6.** Lister les identifiants des arbitres qui sont affectés à des matchs où les deux joueurs sont de même établissement
- Q7.** Lister toutes les informations de l'établissements dont le nombre de joueurs étudiants est maximum.

Sqlite

Dans la suite, les fonctions demandées doivent être écrites en Python en désignant par **cur** le curseur d'exécution des requêtes.

- Q8.** Ecrire la fonction **points_tournoi** qui prend les paramètres **cur**, **id_tournoi** : l'identifiant du tournoi, **cin_j** : l'identifiant du joueur. Cette fonction retourne le nombre total des points du joueur d'identifiant **cin_j** marqués dans le tournoi d'identifiant **id_tournoi**.
- Q9.** Ecrire la fonction **dict_points** qui prend en paramètre **cur** et **id_tournoi** l'identifiant du tournoi. Cette fonction retourne un dictionnaire où les clés sont les identifiants des joueurs et les valeurs sont le total des points de chaque joueur.
- Q10.** Ecrire la fonction **classement** qui prend en paramètre **cur** et **id_tournoi** l'identifiant du tournoi. Cette fonction retourne une liste de tuple de la forme (**cin_joueur**, **points**) des trois meilleurs joueurs du tournoi **id_tournoi**.