

Mini projet : Python (Semestre 1)

Objectifs :

- Savoir utiliser les structures de contrôle
- Comprendre le fonctionnement des structures de données
- Savoir manipuler les fonctions en python

1. Description des attendus de l'application

L'objet de cette application consiste au suivi du déroulement d'un tournoi de foot, regroupant huit équipes et se déroulant sur une période de trois semaines.

L'application doit permettre grâce à un menu qui s'affiche au lancement de l'application, à l'utilisateur d'avoir accès aux fonctions suivantes :

- Saisie des différentes équipes
- Affichage de la liste des équipes participantes
- Saisie des arbitres invités
- Planification des matchs par date
- Saisie des résultats des matchs de la journée
- Affichage des résultats des matchs de la journée
- Classement des équipes
- Pronostics
- Quitter

2. Utilisation des structures de données

Dans un premier temps, on utilisera des structures de données temporaire du type tableau, liste ou dictionnaire. Vous devez argumenter vos choix par rapport à cette question.

Pour chaque équipe, on doit avoir le nom de l'équipe, le nombre de matchs joués, le nombre de matchs gagnés, le nombre de matchs nuls et le nombre de matchs perdus.

Le tournoi se déroule sur une période de 20 jours, à raison de 4 matchs par jour dans des stades différents. A la fin de chaque journée, les résultats doivent être saisis.

A l'issue de chaque match joué, l'équipe récupère 3 pts en cas de victoire, un point en cas de match nul et 0 point en cas de perte.

La fonctionnalité "Planification des matchs" consiste à afficher pour une date donnée, les matchs planifiés dans la journée. Dans le cas où c'est une journée repos, un message d'erreur sera affiché.

Pour chaque match, seront affectés un arbitre, un stade et deux équipes. En fin de journée, la saisie des résultats de la journée doit s'effectuer, en recalculant le nouveau nombre de points des équipes ayant joué.

Une fois la mise à jour effectuée, on doit permettre l'affichage des résultats de la journée, avec un classement temporaire des équipes.

L'option "Pronostics" permet de proposer aux utilisateurs de miser sur les trois équipes qui occuperont le podium et de vérifier à la fin, si les pronostics étaient bons ou non.

Enfin, l'option "Quitter" permet de quitter l'application.

3. Utilisation d'une base de données

Dans cette deuxième partie, on s'intéressera à l'utilisation d'une base de données avec Python.

Cette fonctionnalité sera mise en place via l'installation d'un connecteur (pilote), avec la commande :

```
python -m pip install mysql-connector-python
```

La base de données à mettre en place sera créée via l'utilitaire **phpmyadmin**. On pourra l'installer via xampp, qui est une solution intéressante pour avoir accès à un serveur de bases de données.

Xampp est disponible sur tous les environnements Windows, Linux, Mac.

Nous procéderons à la création d'une base, nommée "tournoi", grâce au script suivant :

```
import mysql.connector

mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="root"
)

cursor = mydb.cursor()

cursor.execute("CREATE DATABASE tournoi")
```

Pour créer une table dans une base, on utilise le script suivant :

```
mydb = mysql.connector.connect(
    host="localhost",
    user="yourusername",
    password="yourpassword",
    database="mydatabase"
)

mycursor = mydb.cursor()

mycursor.execute("CREATE TABLE customers (name VARCHAR(255), address VARCHAR(255))")
```

Pour réaliser des requêtes SQL sur la base, on utilisera ce modèle de script :

```
import mysql.connector

mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="root",
    database="immodb"
)

mycursor = mydb.cursor()

# Requête 1
mycursor.execute("SELECT * FROM immeuble")
myresult1 = mycursor.fetchall()

for x in myresult1:
    print(x)
```

Structure de la base de données à utiliser

Nom de la base de données : **tournoi**

Tables de la base :

equipe (codeEqp(int), nom(varchar), j_matches(int), g_mactch (int), n_match(int), p_matches(int))

match (codeMatch(int), #codeEqp1, #codeEqp2, #codeArbitre, score1(int), score2(int))

arbitre(codeArbitre(int), nom(varchar), tel(varchar), mail(varchar))

...