

À : Vous
De : Charlie
Cc : Édouard
Objet : Sauvegarde/chargement dans le programme d'échecs

Bonjour, j'espère que tu vas bien !

Ed et moi venons de discuter de la meilleure façon de mettre en œuvre la sauvegarde et le chargement dans le programme. Tu te souviens certainement avoir lu les spécifications techniques à ce sujet, mais nous n'étions pas sûrs de ce que nous voulions au moment d'écrire ces lignes...

Après quelques lectures supplémentaires, je pense que la méthode la plus simple serait d'utiliser un paquet appelé **TinyDB**. Ne t'inquiète pas si tu n'as pas beaucoup d'expérience avec les bases de données : nous ne ferons aucune recherche ! Nous l'utiliserons uniquement pour exporter et importer l'état du programme dans un fichier.

Tu peux principalement utiliser TinyDB pour créer deux tables : les joueurs et les tournois. La table des joueurs stockera tous les joueurs dans la mémoire du programme, et de même pour les tournois et la table des tournois.

Pour stocker une instance de joueur, tu dois d'abord la convertir en dictionnaire (c'est-à-dire la sérialiser). Par exemple, supposons que j'aie ce modèle de joueur et que je crée une instance à partir de celui-ci :

```
class Player:
    def __init__(self, name, age):
        self.name = name
        self.age = age

player = Player(name='John', age=22)
```

Je peux sérialiser l'instance comme ceci :

```
serialized_player = {
    'name': player.name,
    'age': player.age
}
```

Une instance sérialisée peut être reconvertie en une instance comme suit :

```
name = serialized_player['name']
```

```
age = serialized_player['age']
player = Player(name=name, age=age)
```

Pour sauvegarder plusieurs joueurs en utilisant TinyDB, tu dois d'abord sérialiser toutes les instances de joueurs, puis les charger dans la table des joueurs comme ceci :

```
from tinydb import TinyDB

db = TinyDB('db.json')
players_table = db.table('players')
players_table.truncate() # clear the table first
players_table.insert_multiple(serialized_players)
```

Pour recharger les joueurs sérialisés, tu peux faire ceci :

```
serialized_players = players_table.all()
```

Une liste d'instances de lecteur désérialisées peut alors être créée à partir du contenu de `serialized_players`, après quoi le reste du programme devrait se dérouler normalement. Veille à bien préserver l'ordre de la liste lorsque tu sérialises/désérialises les joueurs !

Tu devrais suivre la même approche pour sauvegarder les tournois dans le tableau des tournois. Cela peut être un peu plus compliqué, car les tournois sont des objets plus grands que les joueurs, mais le même principe de sérialisation/désérialisation des instances s'applique. Si tu es bloqué, tu peux consulter [la documentation officielle de TinyDB](#) pour obtenir de l'aide supplémentaire.

Enfin, je te recommande d'implémenter cette fonctionnalité vers la fin du projet, une fois que tu auras le schéma clairement défini et que tu sauras que le programme de base fonctionne correctement.

Bon courage !

Meilleures salutations,
Charlie
Assistant informatique
Centre échecs