

MongoDB et PyMongo

Travaux pratiques notés

Travaux pratiques

Objectif : Sur un set de données contenant une liste de films.
Appliquer les notions MongoDB et Python vues en cours

Description du TP

Étape 1: Importer les données

- ▶ Créer une base de données MongoDB appelée "cinema" et une collection nommée "movies"
- ▶ Créer un schéma (validator)
- ▶ Utiliser python et pymongo pour lire le .csv et l'importer dans la collection movies
 - ▶ Validez et convertissez la donnée si nécessaire, ce n'est pas obligatoire mais ça pourrait vous aider pour la suite :)

Étape 2: Déclaration des classes

- ▶ Créer une class Director
 - ▶ L'ajout d'un nouveau réalisateur passe forcément par la classe, qui contrôle l'intégrité des données
- ▶ Créer une class Movie
 - ▶ L'ajout d'un nouveau film passe forcément par la classe, qui contrôle l'intégrité des données

Description du TP

Étape 3: Création d'une collection Director

- ▶ À l'aide de PyMongo, réaliser les opérations suivantes :
 - ▶ Enregistrer dans une nouvelle collection (ou vue, comme vous voulez) la liste des réalisateurs avec la liste de leurs films
- ▶ Implémenter la méthode permettant de lister les films d'un réalisateur (dans sa classe)
- ▶ Implémenter la méthode permettant de connaître la note moyenne d'un réalisateur (dans sa classe)
- ▶ Implémenter la méthode d'ajout de film par interaction utilisateur

Étape 4 : Agrégation (dans un fichier à part, utilisez la collection de votre choix)

- ? Lister les 5 réalisateurs les mieux notés
- ? Les 5 réalisateurs dont les films ont la durée moyenne la plus importante
- ? Les 5 réalisateurs ayant le plus de films

Description du TP

```
_id: "Nicolas Cage"
count: 34
▼ movies: Array
  0: "The Humanity Bureau"
  1: "Matchstick Men"
  2: "World Trade Center"
  3: "Windtalkers"
  4: "The Weather Man"
  5: "The Sorcerer's Apprentice"
```

```
_id: "Ryan Reynolds"
count: 28
► movies: Array
```

Résultat agrégation

- ❓ Requête d'agrégation : Le résultat de cette requête doit me donner la liste et le nombre de films des 15 acteurs les plus présents (avec leurs films, cf screenshot ci-dessus)

Bonus : Utilisation avancées POO + Mongo

- ❓ Empêcher la duplication d'un film
- ❓ Créer une classe commune à Director et Movie permettant la gestion de la propriété '_id' ainsi que la définition d'une méthode 'get_by_id' générique
- ❓ Externaliser la connexion à la BDD dans un singleton

Consignes générales

L'archive du rendu doit contenir :

Si Jupyter, séparez bien tout par cellule

Si pas jupyter

- ▶ Un fichier python par classe
- ▶ Un fichier python gérant l'import des données (à n'exécuter qu'une seule fois)
- ▶ Un fichier main.py bien commenté ou X fichiers répondant aux questions du cours