

DEVOIR

VIVIER SAMUEL

CRÉER UN LOGICIEL DE GESTION DE MÉDIATHÈQUE

```
from django.db import models

class Book(models.Model):
    """Book model"""
    title = models.CharField(max_length=100)
    author = models.CharField(max_length=100)
    email = models.EmailField(max_length=100)
    isbn = models.CharField(max_length=10)
    is_available = models.BooleanField(default=True)

    def __str__(self):
        return f'{self.first_name}, {self.last_name}'
```

Projet Django

JUIN 2024

Créer un logiciel de gestion de médiathèque

Sommaire

- 1/ Présentation du projet
- 2/ Etudes et correctif du code fourni
- 3/ Mise en place des fonctionnalités demandées
- 4/ Stratégie de tests
- 5/ Base de données avec des données test




Créer un logiciel de gestion de médiathèque

1/ Présentation du projet

La médiathèque « Notre livre, notre média » nous a contacté afin de moderniser son système de gestion interne qui est pour l'instant géré à l'ancienne par des fiches papiers. L'objectif est d'avoir un système fonctionnel, sécurisé, utilisable par les bibliothécaires et également par le public.

La médiathèque propose à la consultation et à l'emprunt des livres, des Cds, des jeux de plateaux et des dvd. Tous ces supports sont ouverts à la consultation et à l'emprunt, sauf les jeux de plateau qui sont uniquement disponibles en consultation.

Il y aura deux applications à déployer :

- Une application principale qui ne sera accessible qu'aux bibliothécaires.
 - Une application de consultation accessible aux emprunteurs.
- 

Créer un logiciel de gestion de médiathèque

2/ Etudes et correctif du code fourni

Dans ce projet un développeur avait déjà commencé un code. Afin d'avoir un code propre j'ai analysé le code et corrigé ce qui devait être corrigé.

```
def menu():
    print("menu")

if __name__ == '__main__':
    menu()

class livre():
    name = ""
    auteur = ""
    dateEmprunt = ""
    disponible = ""
    emprunteur = ""

class dvd():
    name = ""
    realisateur = ""
    dateEmprunt = ""
    disponible = ""
    emprunteur = ""

class cd():
    name = ""
    artiste = ""
    dateEmprunt = ""
    disponible = ""
    emprunteur = ""

class jeuDePlateau :
    name = ""
    createur = ""

class Emprunteur():
    name = ""
    bloque = ""

def menuBibliotheque() :
    print("c'est le menu de l'application des bibliothécaire")

def menuMembre():
    print("c'est le menu de l'application des membres")
    print("affiche tout")
```

Créer un logiciel de gestion de médiathèque

2/ Etudes et correctif du code fourni

Le script `if __name__ == '__main__':`
`menu()`

n'a pas besoin d'être déclaré car il est déjà déclaré dans le fichier **manage.py** lors de la création du projet. Je l'ai donc pas réutilisé.

Les fonctions `"def menu()"`, `"def menuBibliotheque()"` et `"def menuMembre()"`. Ne sont pas à conserver car ils vont être gérés par des views.

Les noms des classes doivent toutes commencer par une majuscule

Dans le langage python, l'indentation joue un rôle important mais dans la classe `"Dvd"`, les attributs `"realisateur"`, `"dateEmprunt"` et `"disponible"` ainsi que dans la classe `"Emprunteur"` pour l'attribut `"bloque"`, ont une indentation en trop.

Afin de raccourcir le code et de ne pas se répéter, j'ai regroupé tout les médias dans une liste média et modifier `"auteur"`, `"realisateur"`, `"artiste"` et `"createur"` par `"creator"`.

Créer un logiciel de gestion de médiathèque

3/ Mise en place des fonctionnalités demandées

J'ai commencer par la création du projet "mediatheque". Ensuite j'ai installé django et je l'ai importer dans le projet. J'ai ensuite créé une application "bibliotheque". J'ai utiliser la partie "mediatheque" pour la parti membre afin d'afficher uniquement la liste de tous les médias et la page de connection. Et la partie 'bibliotheque' pour afficher les autres pages demandés.

4/ Stratégie de tests

Pour les tests, j'ai utiliser pytest afin de vérifier mes fonctions. J'ai donc c'est des fichiers test_models.py et test_views.py. Dans ces fichiers toutes les fonctions test commence par "test_" afin que pytest fonctionne correctement. J'ai aussi créer un fichier "pytest.ini", contenant :

```
[pytest]
DJANGO_SETTINGS_MODULE = mediatheque.settings
```

afin de spécifier l'app à utiliser.

Créer un logiciel de gestion de médiathèque

5/ Base de données avec des données test

Pour la base de données j'ai créer plusieurs membres ainsi que plusieurs média afin de tester réellement mon site.

Nom	Prenom	Detail
vivier	samuel	Voir profil
Print	Jean	Voir profil
Sparrow	Jack	Voir profil
Doe	John	Voir profil
Ajouter un membre		Retour au menu

Liste des Médias

Nom	Créateur	Type	Disponible	Emprunteur
7 Wonders	Antoine Bauza	Jeu de plateau	Disponible	Pas emprunté
7 Wonders Duel	Antoine Bauza	Jeu de plateau	Disponible	Pas emprunté
Harry Potter	J.K Rowling	Livre	Indisponible	Jean Print
Harry Potter 2	J.K Rowling	Livre	Indisponible	Jack Sparrow
Harry Potter 3	J.K Rowling	Livre	Indisponible	Jean Print
Harry Potter 4	J.K Rowling	Livre	Indisponible	Jean Print
Harry Potter 5	J.K Rowling	Livre	Disponible	Pas emprunté
Harry Potter 6	J.K Rowling	Livre	Disponible	Pas emprunté
Harry Potter 7	J.K Rowling	Livre	Disponible	Pas emprunté
Hunger Games	suzanne collins	DVD	Disponible	Pas emprunté
Retour au menu				

Livre

- Harry Potter - J.K Rowling
Emprunté par : Jean Print
Date d'emprunt : 05/06/2024 [Retourner le Livre](#)
- Harry Potter 2 - J.K Rowling
Emprunté par : Jack Sparrow
Date d'emprunt : 05/06/2024 [Retourner le Livre](#)
- Harry Potter 3 - J.K Rowling
Emprunté par : Jean Print
Date d'emprunt : 05/06/2024 [Retourner le Livre](#)
- Harry Potter 4 - J.K Rowling
Emprunté par : Jean Print
Date d'emprunt : 05/06/2024 [Retourner le Livre](#)

CD

- Aucun emprunt pour ce type de média.

DVD

- Aucun emprunt pour ce type de média.

Jeu de plateau

- Aucun emprunt pour ce type de média.

[Ajouter un emprunt](#) [Retour au menu](#)

Créer un logiciel de gestion de médiathèque

6/ Instruction pour exécuter le programme

Afin de pouvoir exécuter le programme, vous devez suivre les étapes suivantes:

- Installez python sur votre ordinateur. (vous pouvez le télécharger a partir du site officiel de python <https://www.python.org/downloads/>)

- Télécharger le programme sur github : https://github.com/Samuel-Viv/mediatheque_python.git

- Une fois le fichier télécharger extraire le fichier du fichier .zip

- Ouvrir le fichier avec votre IDE (Environnement de développement)

- Vous devez créer un environnement virtuel, afin d'isoler les dépendances du projet. Pour cela allez dans le terminal de l'IDE et tapez la ligne de commande : `python -m venv env` . Puis pour activer l'environnement virtuel tapez: `env\Scripts\activate`

- Ouvrer un nouveau terminal allez dans le dossier `mediatheque_python-main` et ensuite tapez `python manage.py runserver`

- Cliquez sur le lien <http://127.0.0.1:8000/> pour ouvrir l'application

- Pour accéder à l'interface administrateur du site ainsi que la partie bibliothécaire un identifiant et un mot de passe vous sera demandé.

Lien admin: <http://127.0.0.1:8000/admin/>

Identifiant : `bibliothécaire`

Mot de passe : `mediatheque`