

PROJET 4: Développez un programme logiciel en Python

1

Réalisé par : **Sabah EL-AOUNI**

Mentor : **Idriss Ben Geloune**

PLAN DE LA PRÉSENTATION

- Présentation globale du projet
- Architecture Model, Vue, Contrôleur
- Base de donnée
- Vérification PEP8 et FLAK8
- Conclusion

Présentation globale du projet 1/2

- Il s'agit d'un programme qui permet de gérer des tournois d'échecs
- Le programme est lancé hors ligne depuis la console
- Le déroulement de base d'un tournoi s'effectue comme suit:
 1. Création d'un nouveau tournoi.
 2. Ajout de huit joueurs.
 3. Génération des paires de joueurs pour le premier tour.
 4. saisie les résultats une fois le tour est terminé.
 5. Répétition des étapes 3 et 4 pour les tours suivants jusqu'à ce que tous les tours soient joués, et que le tournoi soit terminé.

Présentation globale du projet 2/2

Technologies utilisées:

- Le programme est développé en Python et utilise la programmation orientée objet
- L'architecture utilisée est le modèle MVC (Modèle, Vue, Contrôleur)
- Le module Tinydb a été utilisé pour gérer la base de données
- La Flake8 a été utilisée pour le peluchage du code
- Le guide de style de la PEP8 a été appliqué

Architecture Model, Vue, Contrôleur 1/2

➤ Modèle :

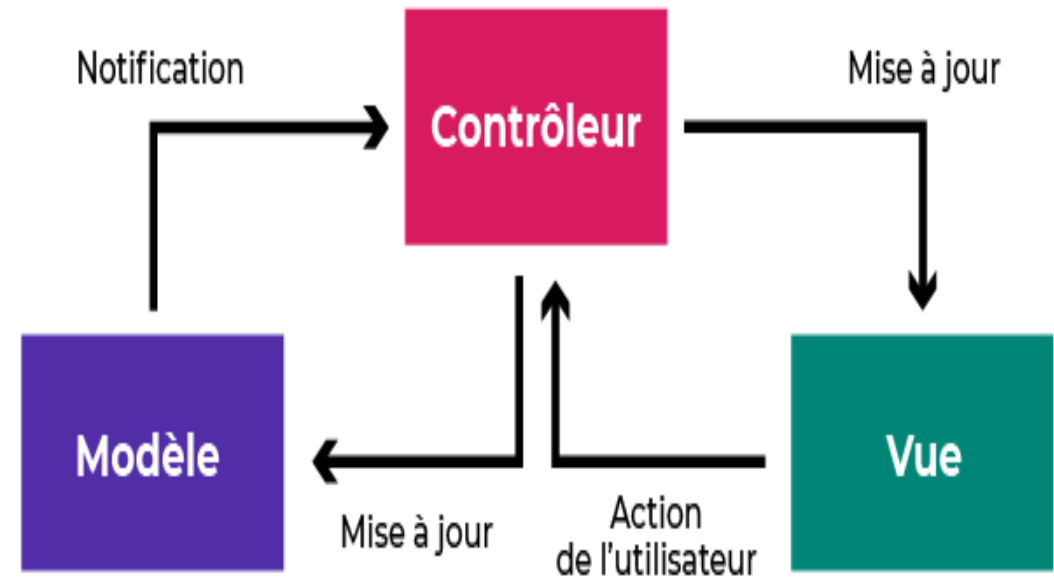
- Joueur
- Match
- Tour
- Tournoi

➤ Vue

- Menu

➤ Contrôleur

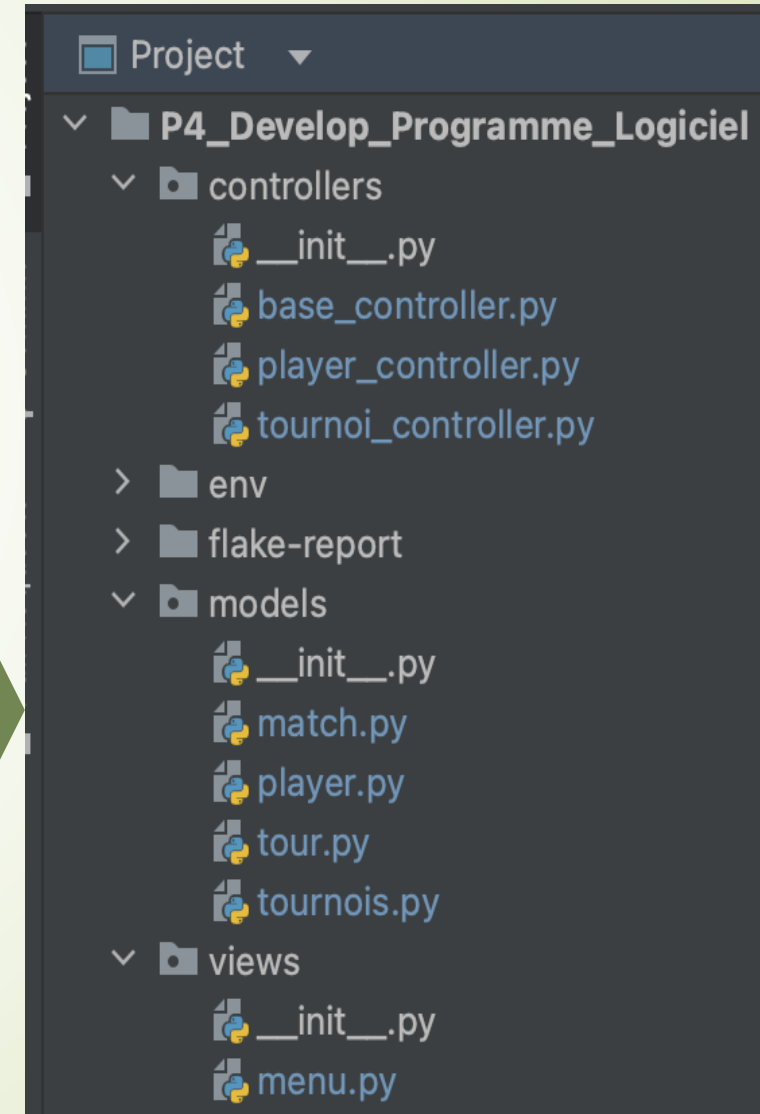
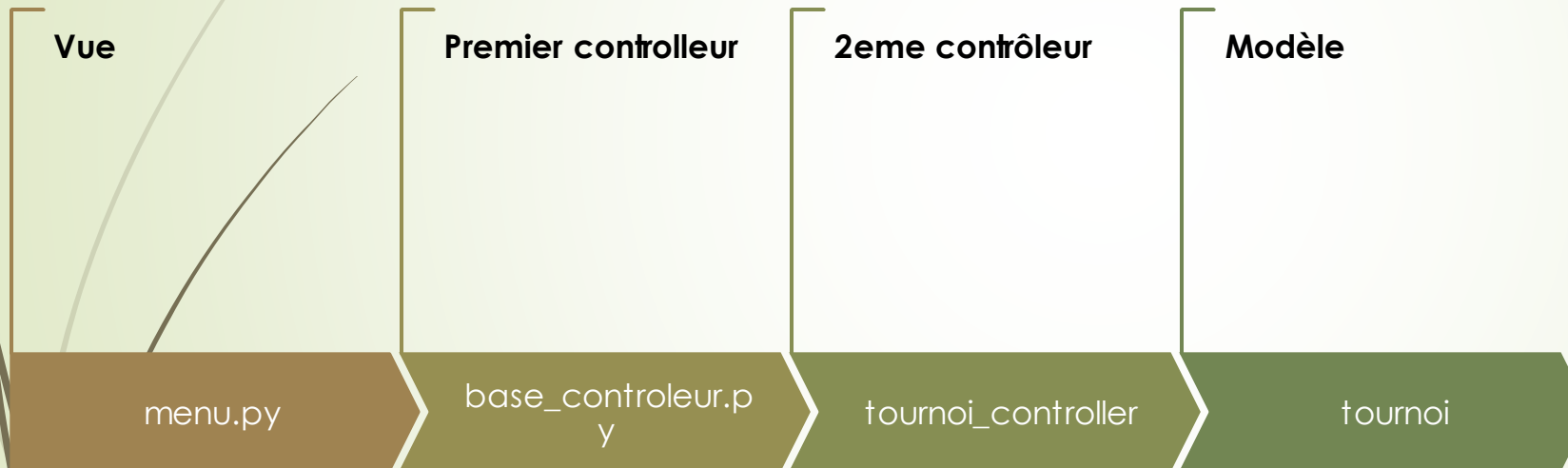
- Contrôleur de base
- Contrôleur Joueur
- Contrôleur Tournoi



Source: <https://openclassrooms.com/>



Architecture Model, Vue, Contrôleur 2/2

- Exécution du programme :
Exp : création d'un tournoi



Base de donnée

- Tinydb : deux tables JSON ont été créé pour ce projet :
- Composition des tables:

`DB_players.json`
`DB_tournoi.json`

DB_Players

- Prenom
- Nom
- Date de naissance
- Genre
- Classemen
- Score
- Id

DB_tournoi

- nom du tournoi
- Place
- Date
- Nombre tours
- liste joueurs
- Time control
- Remarques
- liste des Tours

Vérification PEP8 et FLAK8

- PEP8 (Python Enhancement Proposals): permet de rendre le code plus facile à lire et comprendre pour tout le monde.
- FLAK8 : une vérification Flake8 a été effectuée pour ce projet via les étapes suivantes:

1- Installation du module flake8 avec la commande:

```
>> python -m pip install flake8
```

2- Installation flake8-html avec la commande:

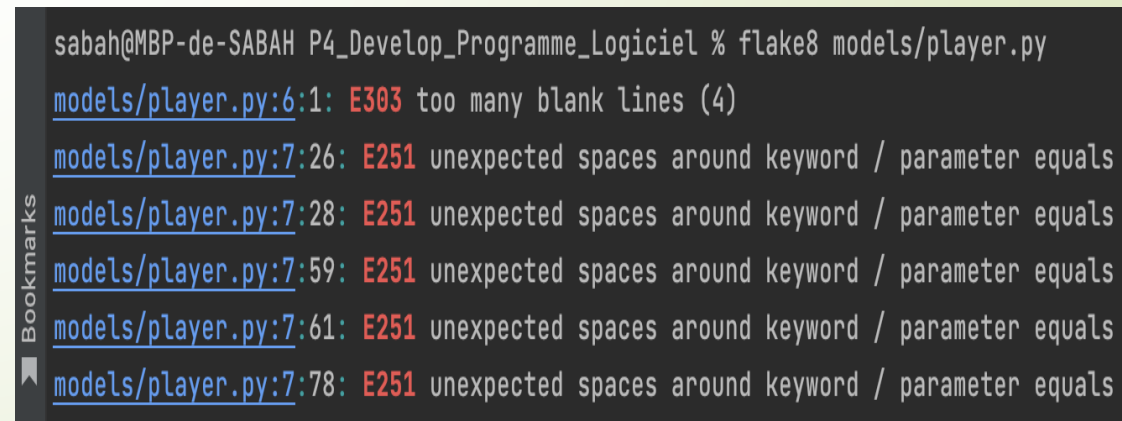
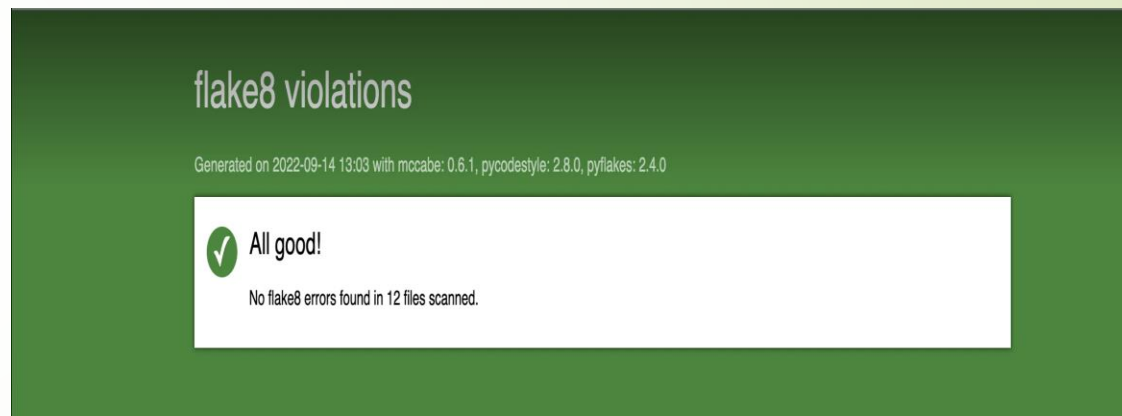
```
>> python pip install flake8-html
```

3- Génération du rapport avec:

```
>> flake8 --format=html --htmldir=flake-report
```

- ✓ Génération du rapport dans le terminal avec la commande:

```
>> flake8 fichier.py
```



Exécution du programme



Conclusion

- La programmation POO:
 - Classes , Méthodes
 - Instanciation
- Architecture MVC:
 - Structuration de son code
 - Fonctionnement de chaque composant
 - Relation entre chaque composant
 - Principe de la responsabilité unique
- Base de donnée TyniDB:
 - Sérialisation et désérialisation
 - Manipulation de la base donnée (requetés)
 - Chargement de la base de donnée
- La vérification Flake8:
 - Moyen de pelucher votre code et d'appliquer la conformité PEP8

