

TP02 Python Django – Modèles et base de données

fait par :

KOUTEMA Ditoma

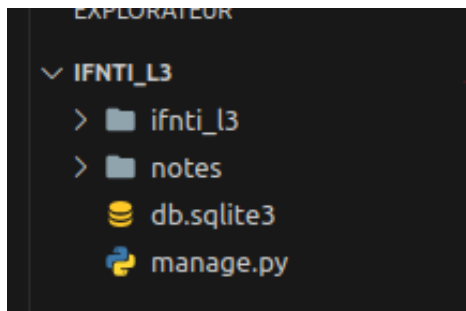
Table des matières

1	Mise en route	3
2	Configuration de la base de données	3
3	Création des modèles	4
4	Activation des modèles	5

1 Mise en route

1.
 - La commande pour activer l'environnement virtuel :
[source projetnotes/bin/activate](#)
 - Version de python : [Python 3.10.12](#)
 - Version de django : [4.2.6](#)

2.
 - Arborescence de mon projet :



- L'application **Notes** se trouve à la racine du projet IFNTI_L3
- L'application notes n'est pas dans le dossier ifnti_l3 parce que, le dossier ifnti_l3 est le dossier de configuration de toutes applications créées dans le projet.
- Si l'application notes s'y trouvait alors, l'application notes ne va pas fonctionner tout simplement.

2 Configuration de la base de données

1.
 - Le moteur de base de donnée utilisé par défaut est : [sqlite3](#)
 - Les caractéristiques des bases de donnée de ce type la sont :
 - C'est un système de gestion de base de données relationnelle (SGBDR) open source.
 - Il est très rapide.
 - Il est multi-plateformes
 - Il prend en charge les bases de données qui peuvent être très grandes ; Actuellement, la limite est de 2 To
 - On peut le stocker dans les fichiers.
 - Je ne pourrai pas l'utiliser en production car il a une limite de stockage.

2. Valeur du Time-Zone de lomé est : `TIME_ZONE = 'UTC'`
3.
 - Le type de `INSTALLED_APPS` est une `: list`.
 - Cette constante contient actuellement toutes les applications créées pendant la création du projet.
4.
 - La commande pour lancer un serveur : `py manage.py runserver`.
 - Cette constante contient actuellement toutes les applications créées pendant la création du projet.
 - Le texte en **rouge** signifie que les migrations ne sont pas encore exécutées pour l'application `ifnti_l3`.
 - Pour régler le problème il suffit de lancer la commande donnée juste dans le texte en rouge : `py manage.py migrate`

3 Création des modèles

Un modèle est la source unique et définitive d'informations sur nos données.

1. Le fichier `models.py` contient une importation et un commentaire.
Ce code veut dire qu'on importe les modèles issus de la classe `django.db`.
2. Code :

```
from django.db import models

# Create your models here.
class Personne(models.Model):
    nom = models.CharField(max_length=50)
    prenom = models.CharField(max_length=50)
    sexe = models.TextChoices = [("M", "Masculin"), ("F", "Femini"),]
    date_naissance = models.DateField()

    class Meta:
        abstract = True

class Niveau(models.Model):
    nom = models.CharField(max_length=2, unique=True)
    pass

class Enseignant(Personne):
```

pass

```
class Matiere(models.Model):
    nom = models.CharField(max_length=50, unique=True)
    enseignant = models.ForeignKey(Enseignant, on_delete=models.CASCADE)
    niveau = models.ForeignKey(Niveau, on_delete=models.CASCADE)
```

```
class Eleve(Personne):
    id=models.BigAutoField(primary_key=True)
    nom = models.CharField(max_length=30)
    niveau = models.ForeignKey(Niveau, on_delete=models.CASCADE)
    matiere = models.ManyToManyField(Matiere)
```

```
class Note(models.Model):
    eleve = models.ForeignKey(Eleve, on_delete=models.CASCADE)
    valeur = models.FloatField(max_length=30, null=True)
```

4 Activation des modèles

1.

- Contenu de `INSTALL_APPS` :

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'notes.apps.NotesConfig',
]
```

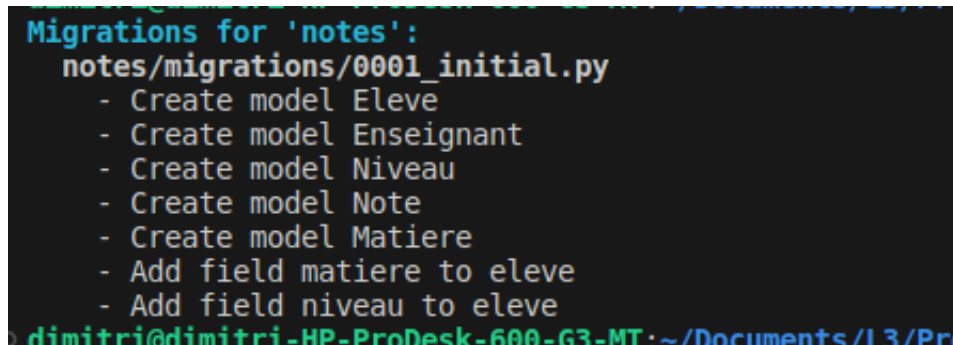
- `NotesConfig` est une classe qui hérite de `AppConfig` qui contient le nom de l'application.
- Cette classe se trouve dans le fichier [apps](#)

2.

- La commande pour lancer les migrations :

`py manage.py makemigrations`

Cette commande permet de construire les migrations.

A terminal window with a black background and green text. It shows the output of the 'py manage.py makemigrations' command. The text is as follows:

```
Migrations for 'notes':
  notes/migrations/0001_initial.py
    - Create model Eleve
    - Create model Enseignant
    - Create model Niveau
    - Create model Note
    - Create model Matiere
    - Add field matiere to eleve
    - Add field niveau to eleve
```

The prompt at the bottom is `dimitri@dimitri-HP-ProDesk-600-G3-MT: ~/Documents/L3/Pro`.

Sur le terminal je vois que les models : Eleve, Enseignant, Niveau, Note, Matiere, et les relations établient entre matiere et eleve ; niveau et eleve

3.

- Résultat de la commande : `py manage.py sqlmigrate notes 0001`
- Il s'agit des requettes sql en dure.

4.

- Commande que j'ai lancée : `py manage.py migrate`
- Le texte qui s'affiche en **rouge** demande de relancer la migration.