

DÉVELOPPEMENT WEB CÔTÉ SERVEUR FRAMEWORK PYTHON-DJANGO

TRAVAUX PRATIQUE N°1 (PROJET MYSITE) INSTALLATION ET CONFIGURATION

Objectif du TP :

1. Installer Python
2. Installer un environnement virtuel pour un projet Django
3. Installer Django
4. Créer un projet Django
5. Créer une base de données SQLite3 pour le projet Django
6. Lancer le serveur web de développement de Django

DJANGO

The web framework for perfectionists with deadlines

Django is used by many large sites, including :

Disqus (<https://disqus.com/>),

Youtub(<https://www.youtube.com/>)

Instagram (<https://www.instagram.com/>),

Mozilla (<https://www.mozilla.org/>),

Pinterest (<https://www.pinterest.com/>),

Open Stack (<https://www.openstack.org/>),

National Geographic (<http://www.nationalgeographic.com/>).

Udemy (<https://www.udemy.com/>)

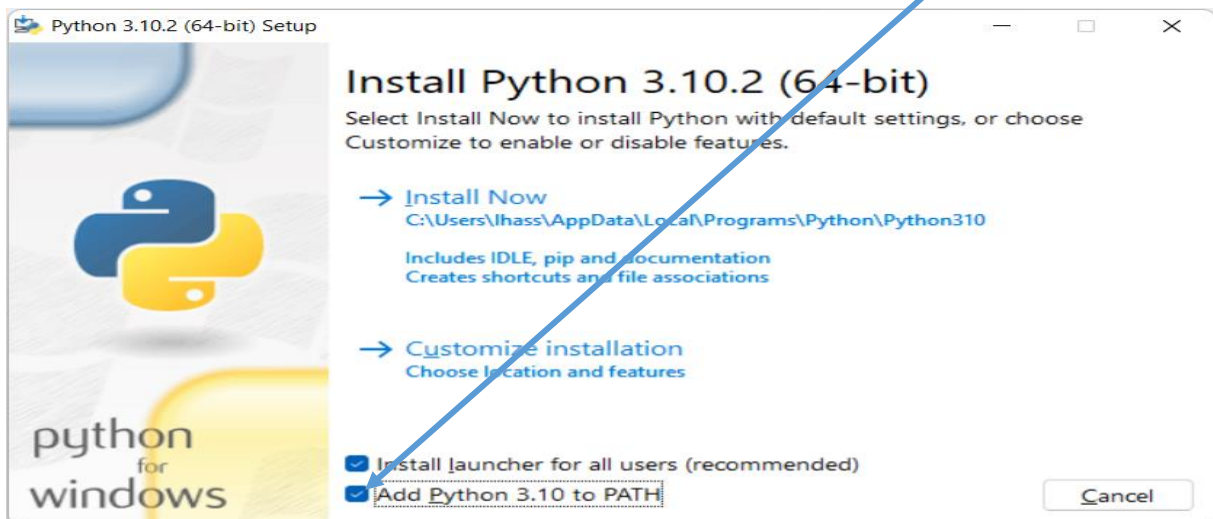
Etape 1 : Télécharger et installer python

Aller à l'URL : <https://www.python.org/downloads/>

Puis, cliquer sur le bouton



Exécuter le fichier et n'oublier pas de cocher la case "Add Python 3.10 to PATH".



Etape 2 : Création d'un environnement virtuel pour votre projet Django

Avant de créer notre projet Django, nous devons d'abord créer un dossier où nous allons sauvegarder les différents éléments du projet.

Appelons par exemple le dossier par mydjango

```
E:\>mkdir mydjango
```

Placez-vous ensuite dans le dossier du projet et Créez un environnement virtuel pour votre projet en exécutant la commande :

```
E:\mydjango>python -m venv venv
```

Exécution du module venv

Nom de l'environnement virtuel à créer

```

E:\mydjango>dir
Volume in drive E is Data
Volume Serial Number is 7877-820E

Directory of E:\mydjango

01/19/2024  06:12 PM    <DIR>          .
01/19/2024  06:12 PM    <DIR>          venv
               0 File(s)                0 bytes
               2 Dir(s)  402,047,348,736 bytes free

```

```

E:\mydjango>cd venv

E:\mydjango\venv>dir
Volume in drive E is Data
Volume Serial Number is 7877-820E

Directory of E:\mydjango\venv

01/19/2024  06:12 PM    <DIR>          .
01/19/2024  06:12 PM    <DIR>          ..
01/19/2024  06:12 PM    <DIR>          Include
01/19/2024  06:12 PM    <DIR>          Lib
01/19/2024  06:12 PM                175 pyvenv.cfg
01/19/2024  06:12 PM    <DIR>          Scripts
               1 File(s)                175 bytes
               5 Dir(s)  402,047,348,736 bytes free

```

Si vous regardez dans le dossier \Lib du dossier venv, vous verrez que virtualenv a créé une installation complète de Python distincte des installations existantes, de sorte que vous pouvez travailler sur votre projet sans affecter les autres logiciels sur votre système.

Pour utiliser ce nouvel environnement virtuel Python, nous devons l'activer. Pour ce faire, lancez le script d'activation « activate » qui se trouve dans le sous-dossier Scripts de votre environnement virtuel.

```

E:\mydjango\venv>cd scripts

E:\mydjango\venv\Scripts>dir
Volume in drive E is Data
Volume Serial Number is 7877-820E

Directory of E:\mydjango\venv\Scripts

01/19/2024  06:12 PM    <DIR>          .
01/19/2024  06:12 PM    <DIR>          ..
01/19/2024  06:12 PM                2,056 activate
01/19/2024  06:12 PM                985 activate.bat
01/19/2024  06:12 PM               26,199 Activate.ps1
01/19/2024  06:12 PM                393 deactivate.bat
01/19/2024  06:12 PM             108,394 pip.exe
01/19/2024  06:12 PM             108,394 pip3.11.exe
01/19/2024  06:12 PM             108,394 pip3.exe
01/19/2024  06:12 PM             270,608 python.exe
01/19/2024  06:12 PM             259,352 pythonw.exe
01/19/2024  06:12 PM             789,504 pythonw_d.exe
01/19/2024  06:12 PM             790,016 python_d.exe
                11 File(s)          2,464,295 bytes
                2 Dir(s)  402,047,348,736 bytes free

```

```
E:\mydjango\venv\Scripts>activate
```

Vous remarquerez que votre invite de commande a maintenant changé.

```
(venv) E:\mydjango\venv\Scripts>
```

Le message (venv) au début de l'invite de commande vous permet de savoir que votre environnement virtuel est actif.

Etape3 : Installation de Django

Placez-vous dans le répertoire de votre projet après avoir activé l'environnement virtuel de votre projet puis exécuter la commande:

```
(venv) E:\mydjango>pip install django
```

Cette commande pip installe la dernière version de Django dans votre environnement virtuel. Vous devez obtenir l'affichage ci-dessous.

```
Collecting django
  Using cached Django-3.2.9-py3-none-any.whl (7.9 MB)
Collecting asgiref<4,>=3.3.2
  Using cached asgiref-3.4.1-py3-none-any.whl (25 kB)
Collecting pytz
  Using cached pytz-2021.3-py2.py3-none-any.whl (503 kB)
Collecting sqlparse>=0.2.2
  Using cached sqlparse-0.4.2-py3-none-any.whl (42 kB)
Installing collected packages: sqlparse, pytz, asgiref, django
Successfully installed asgiref-3.4.1 django-3.2.9 pytz-2021.3 sqlparse-0.4.2
```

Pour tester si l'installation a fonctionné, démarrez l'interpréteur interactif Python, puis importer le module django.

Pour connaître la version de Django installé, exécuter la fonction `django.get_version()` comme le montre la figure ci-dessous.

```
(venv) E:\mydjango>python
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import django
>>> django.get_version()
'5.0.1'
```

N'oubliez pas de quitter l'interpréteur python.

```
>>> exit()
```

Il est possible de tester la version de Django directement à partir de la ligne de commande par la commande ci-dessous :

```
(venv) E:\mydjango>python -m django --version
```

Etape 4 : Création d'un projet Django

Une fois que vous avez installé Python et Django, vous pouvez faire le premier pas pour développer une application Django en créant un projet Django.

Un projet Django contient généralement plusieurs applications Django.

La commande ci-dessous crée un projet Django sous le nom **mysite**.

```
(venv) E:\mydjango>django-admin startproject mysite
```

Remarquez que vous devez vous trouver dans le dossier de votre projet et que l'environnement virtuel du projet doit être actif.

Afficher le contenu du répertoire de votre projet et remarquer que le système a créé un dossier ayant le nom de votre site comme le montre la figure ci-dessous.

```
(venv) E:\mydjango>dir
Volume in drive E is Data
Volume Serial Number is 7877-820E

Directory of E:\mydjango

01/19/2024  06:28 PM    <DIR>          .
01/19/2024  06:28 PM    <DIR>          mysite
01/19/2024  06:12 PM    <DIR>          venv
               0 File(s)                0 bytes
               3 Dir(s)  401,996,582,912 bytes free
```

Afficher le contenu du dossier de votre site comme le montre la figure ci-dessous.

```
(venv) E:\mydjango>cd mysite

(venv) E:\mydjango\mysite>dir
Volume in drive E is Data
Volume Serial Number is 7877-820E

Directory of E:\mydjango\mysite

01/19/2024  06:28 PM    <DIR>          .
01/19/2024  06:28 PM    <DIR>          ..
01/19/2024  06:28 PM                684 manage.py
01/19/2024  06:28 PM    <DIR>          mysite
               1 File(s)                684 bytes
               3 Dir(s)  401,994,084,352 bytes free
```

Le dossier contient un module python de nom **manage.py** qui a pour rôle de gérer votre site créé.

Le dossier contient un sous-dossier qui a aussi le nom de votre site (**mysite**). Le contenu de ce sous-dossier est présenté à la figure ci-dessous.

```
(venv) E:\mydjango\mysite>cd mysite

(venv) E:\mydjango\mysite\mysite>dir
Volume in drive E is Data
Volume Serial Number is 7877-820E

Directory of E:\mydjango\mysite\mysite

01/19/2024  06:28 PM    <DIR>          .
01/19/2024  06:28 PM    <DIR>          ..
01/19/2024  06:28 PM                405 asgi.py
01/19/2024  06:28 PM            3,344 settings.py
01/19/2024  06:28 PM                784 urls.py
01/19/2024  06:28 PM                405 wsgi.py
01/19/2024  06:28 PM                 0 __init__.py
               5 File(s)              4,938 bytes
               2 Dir(s)  401,994,072,064 bytes free
```

Nous verrons en détail la signification et le rôle de chaque fichier au fur et à mesure de l'avancement dans le cours.

Étape 5 : Création d'une base de données

Django inclut un certain nombre d'applications par défaut. Certaines de ces applications utilisent au moins une table de base de données, nous devons donc créer des tables dans une base de données avant de pouvoir les utiliser. Pour ce faire, placez-vous dans le dossier externe du site (là où se trouve le module **manage.py**) créé à la dernière étape et exécutez le module **manage.py** avec la commande **migrate**

```
(venv) E:\mydjango\mysite>python manage.py migrate
```

Le résultat de l'exécution de la commande est fourni ci-dessous.

```
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
```


Comme le montre la figure ci-dessous, le système crée une base de données SQLite3

```
(venv) E:\mydjango\mysite>dir
Volume in drive E is Data
Volume Serial Number is 7877-820E

Directory of E:\mydjango\mysite

01/19/2024  06:52 PM    <DIR>          .
01/19/2024  06:28 PM    <DIR>          ..
01/19/2024  06:52 PM                131,072 db.sqlite3
01/19/2024  06:28 PM                684 manage.py
01/19/2024  06:52 PM    <DIR>          mysite
                2 File(s)        131,756 bytes
                3 Dir(s)    401,993,916,416 bytes free
```

Etape 6 : Lancement du serveur de développement

Les développeurs de Django ont créé un serveur de développement afin que vous puissiez développer vos sites rapidement, sans avoir à gérer la configuration d'un serveur de production (tel qu'Apache, glassfish, payara, Gunicorn(Django)...etc) tant que vous n'êtes pas prêt pour le déploiement. C'est un serveur Web léger écrit uniquement en Python. Pour déployer une application web développée avec Django en production, il faut utiliser le serveur d'application GUNICORN avec WSGI(Web Server Gate Interface)

Pour le lancer, commencer par vous placer dans le dossier externe de votre site (là où se trouve le module **manage.py**) puis exécuter la commande :

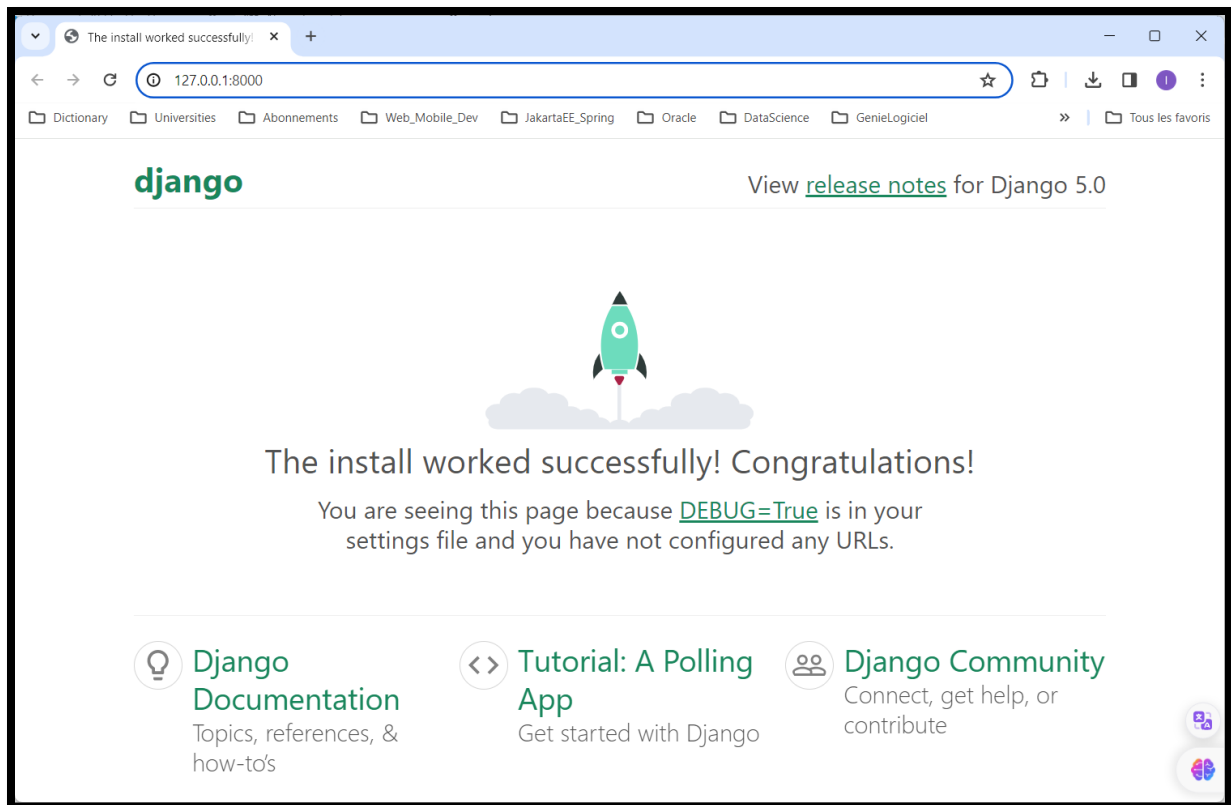
```
(venv) E:\mydjango\mysite>python manage.py runserver
```

Le serveur web de développement est lancé et vous pouvez lui envoyer des requêtes http avec l'url <http://127.0.0.1:8000>. Comme le montre la figure ci-dessous.

```
(venv) E:\mydjango\mysite>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
January 19, 2024 - 18:55:52
Django version 5.0.1, using settings 'mysite.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Maintenant que votre serveur web est lancé vous pouvez utiliser votre browser pour le visiter avec l'url <http://127.0.0.1:8000/>
Le serveur répond par la page web suivante:



Pour arrêter le serveur taper CTRL C.

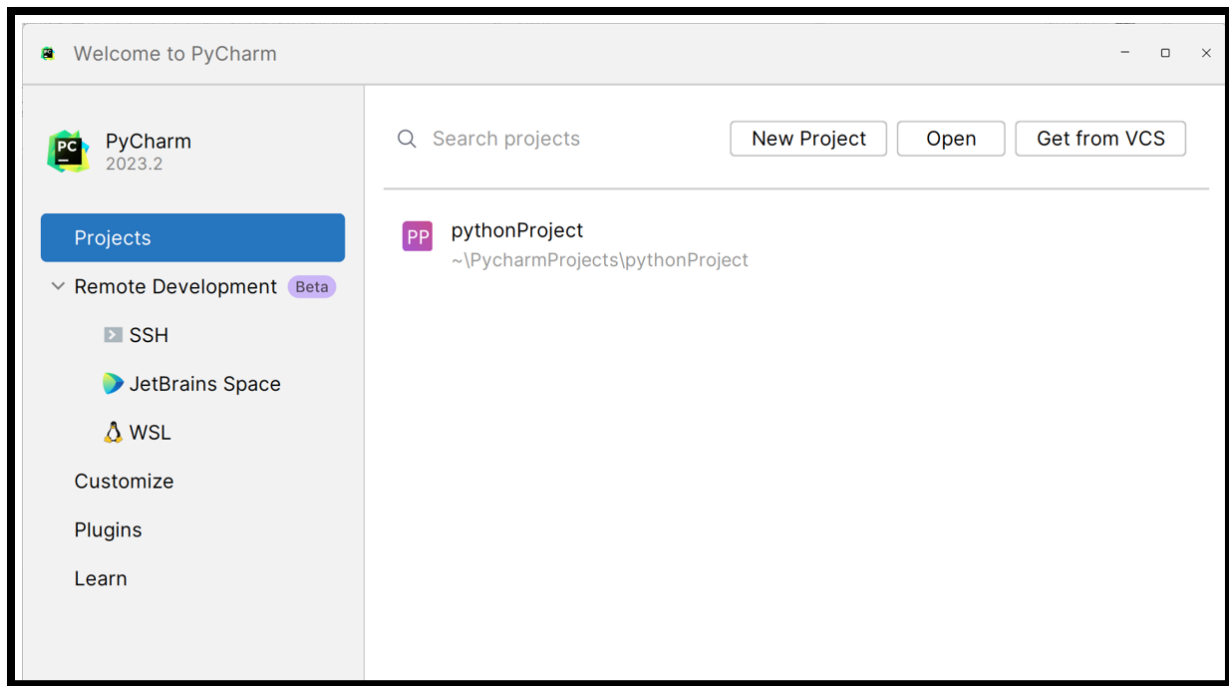
Etape 7 : Ouverture du projet dans l'IDE PyCharm

1-Après avoir lancé Pycharm exécuter la commande :

File -> Close Project

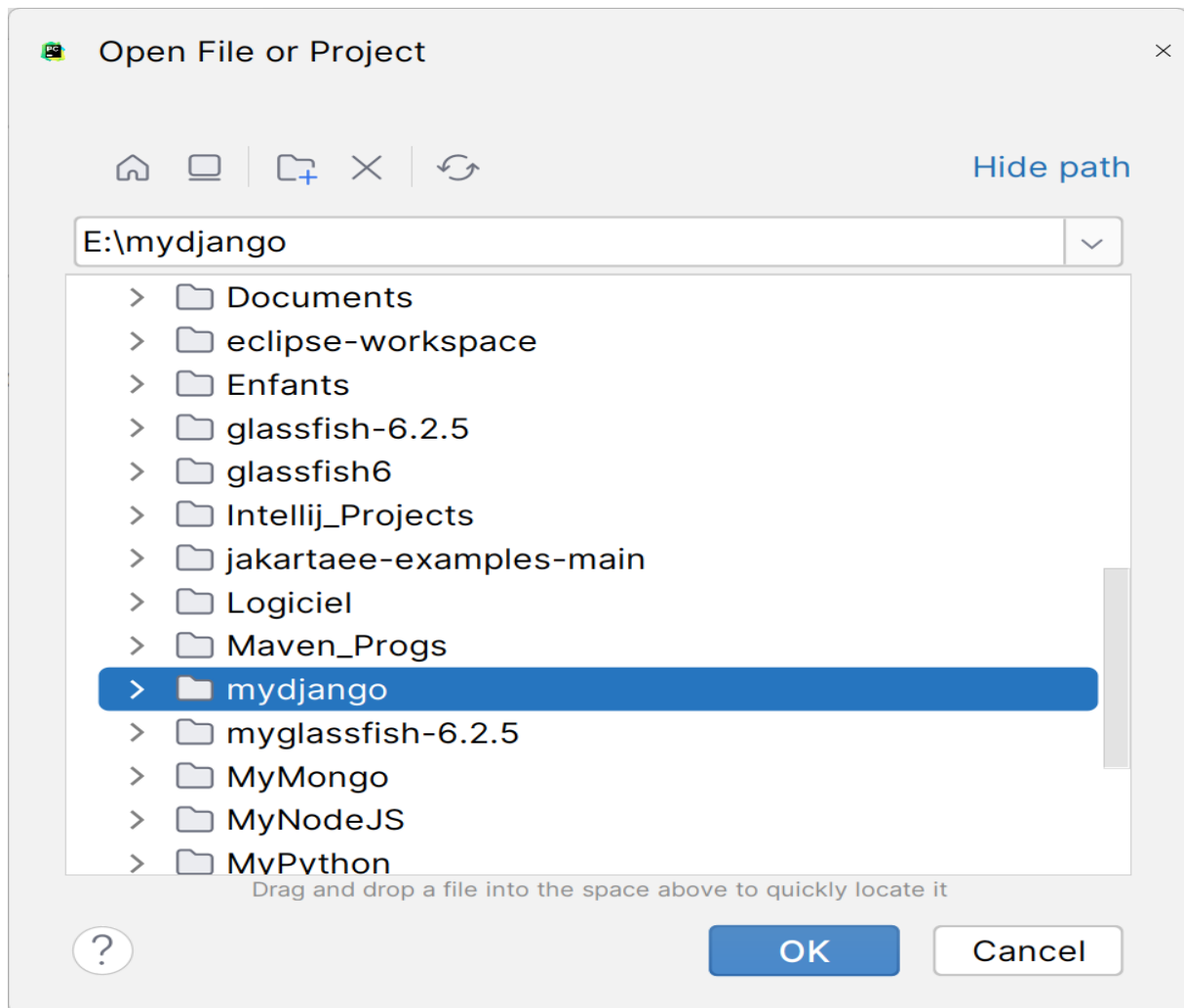
Pour fermer le projet en cours.

Vous obtenez la fenêtre ci-dessous :

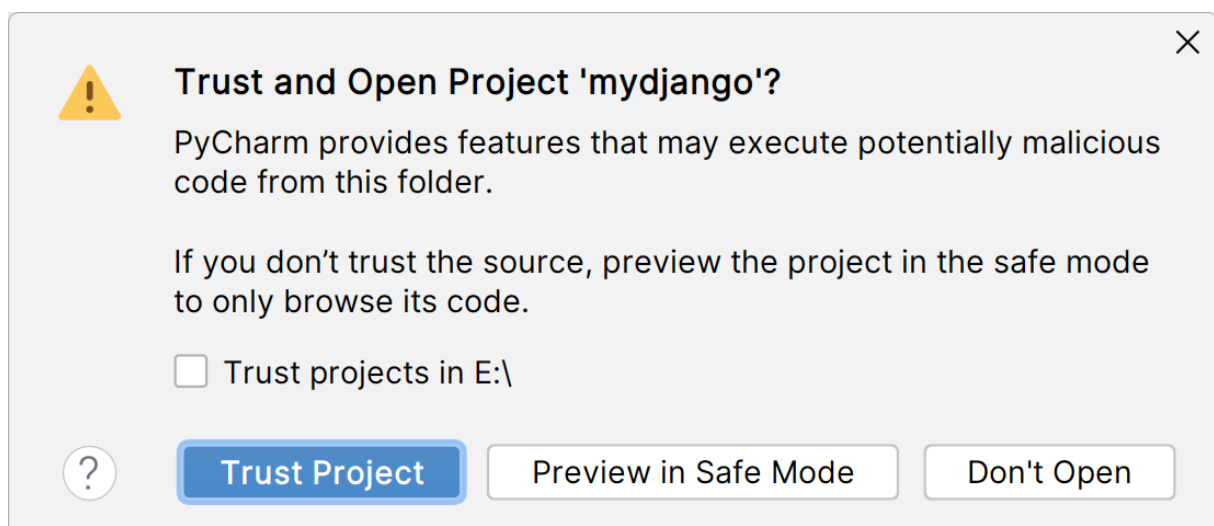


2-Cliquer sur le bouton **open**

3-Dans la boîte de dialogue qui s’affiche sélectionner le dossier mydjango qui contient le dossier externe de votre projet mysite

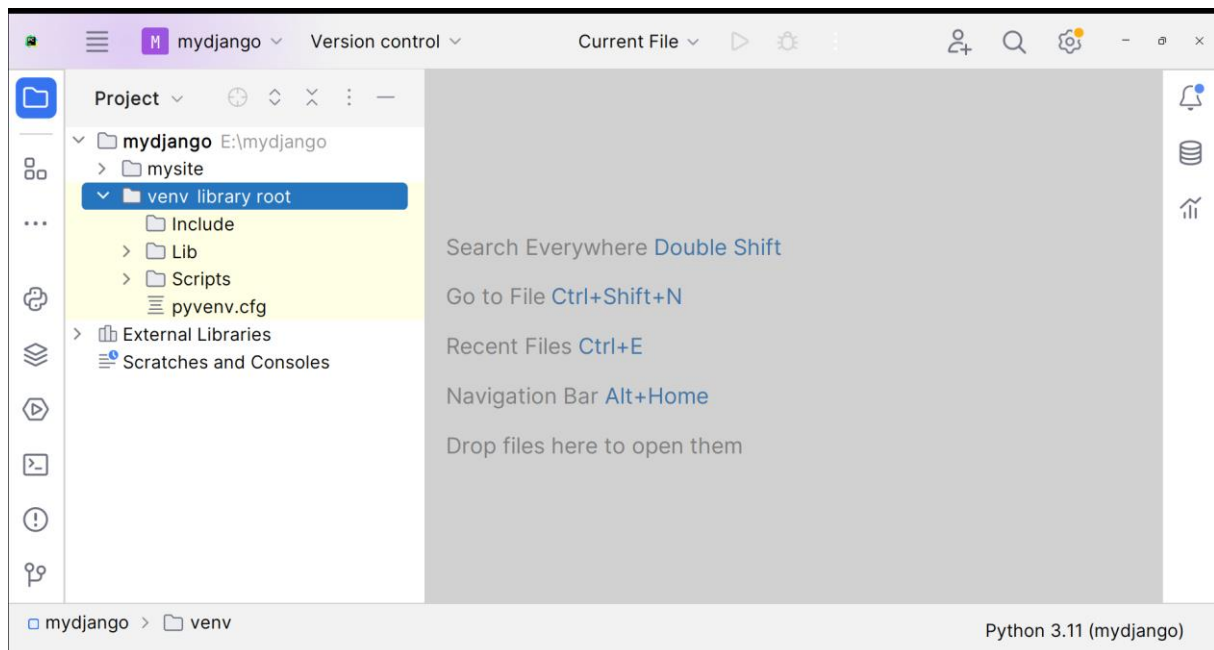


Nous obtenons la fenêtre ci-dessous



Cliquer sur le bouton **TrustProject**.

Nous obtenons ensuite la fenêtre ci-dessous :

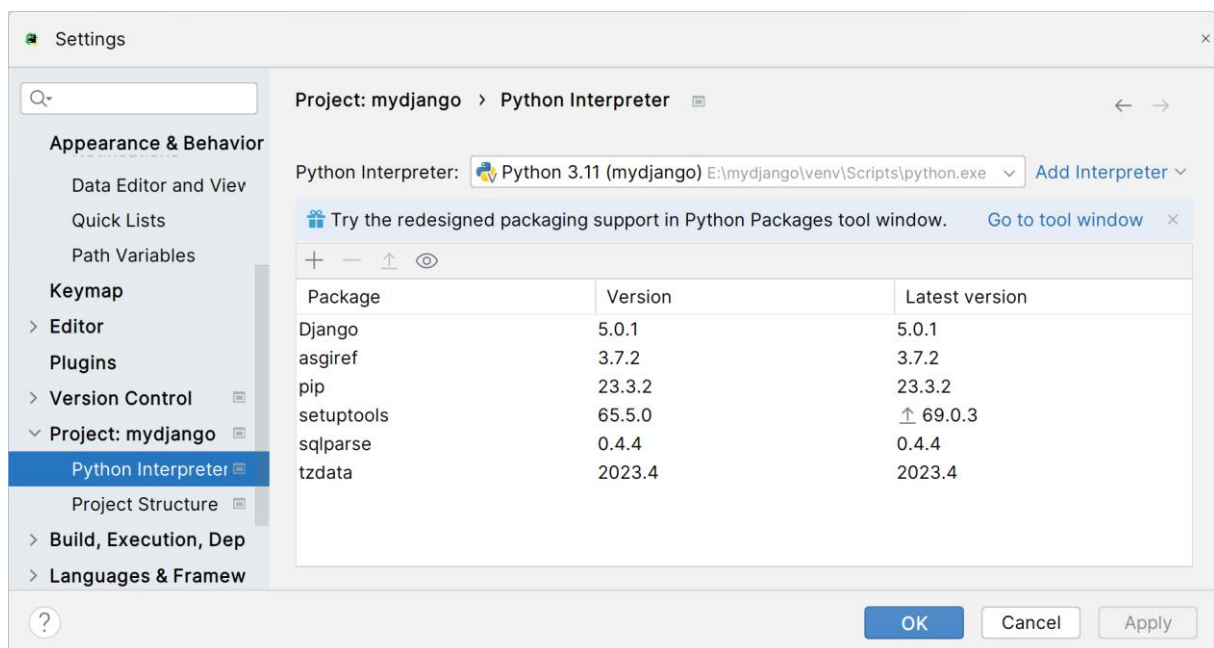


4- S'assurer que l'environnement virtuel est bien détecté par PyCharm

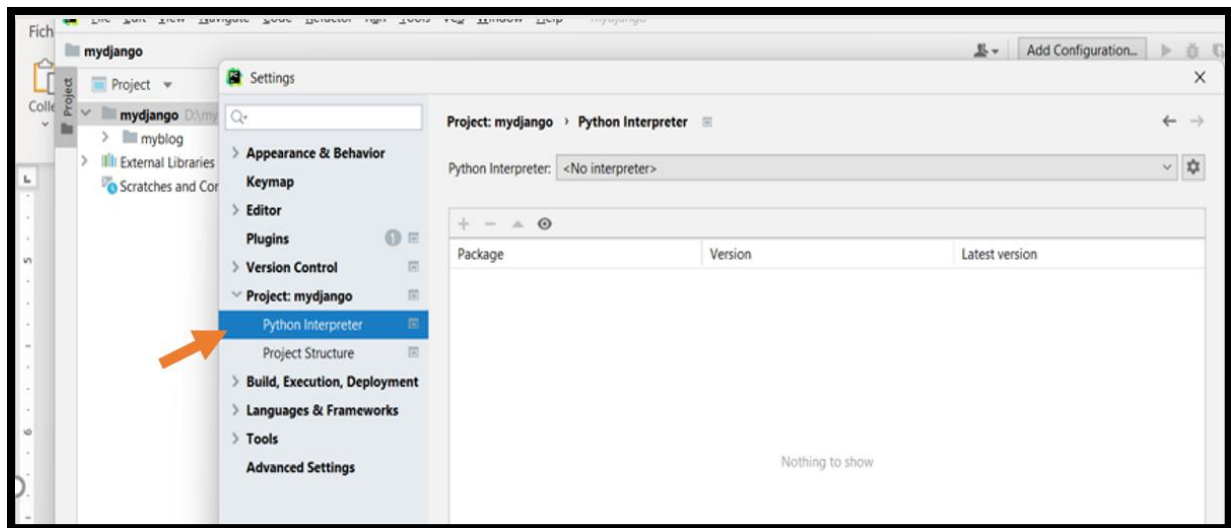
Exécuter la commande : **File->settings**

Selectionner **Project :mydjango** puis **python Interpreter**

La fenêtre ci-dessous s'affiche



Nous remarquons que pycharm a bien détecté notre **venv**

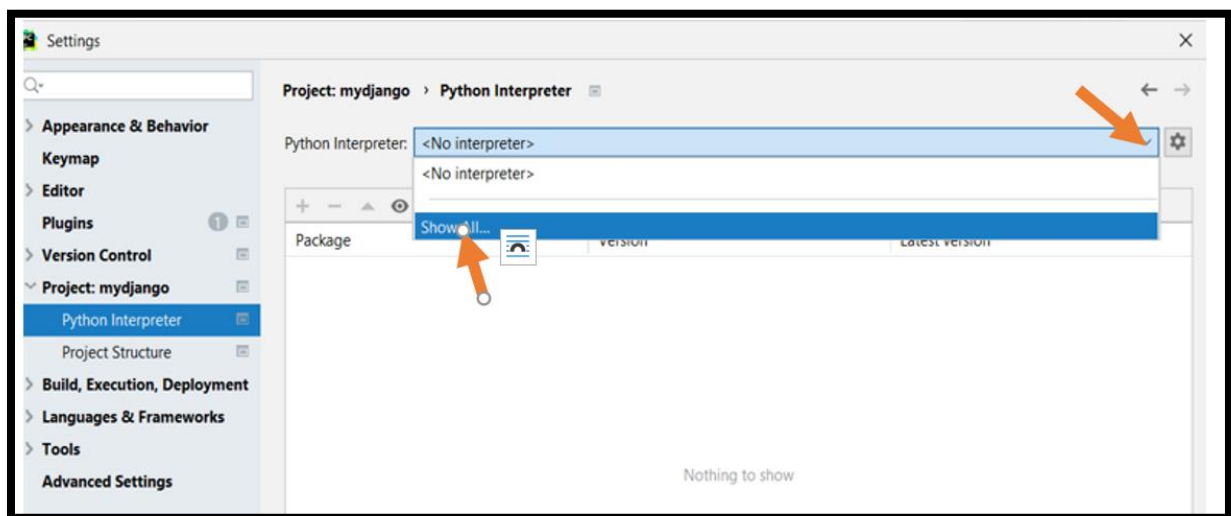


Sélectionner l'option : **Project :mysite**, puis l'option **Python Interpréter**

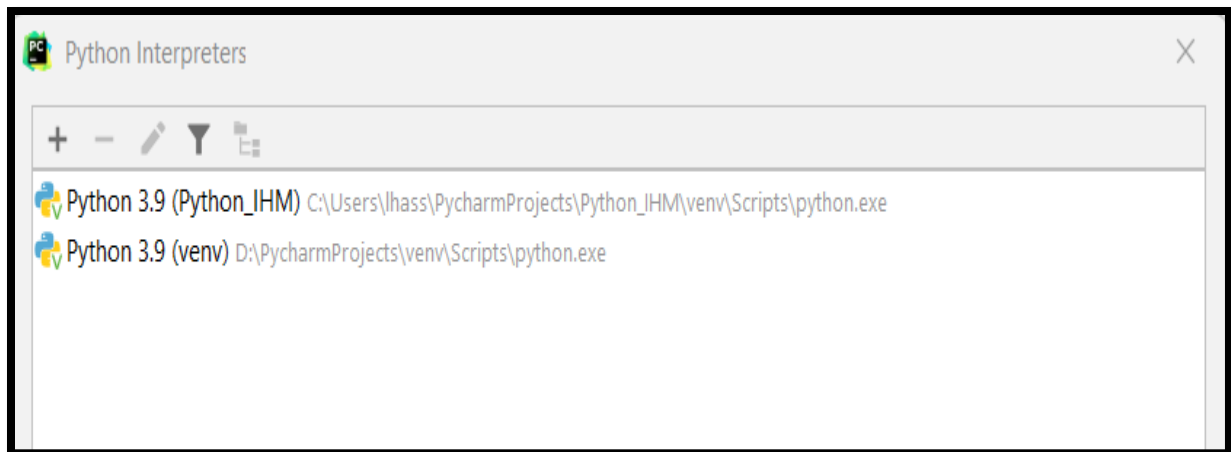
Nous remarquons que Pycharm n'a pas détecté venv.

Pour le sélectionner faire :

Cliquer sur la flèche de la zone déroulante puis sélectionner : Show All



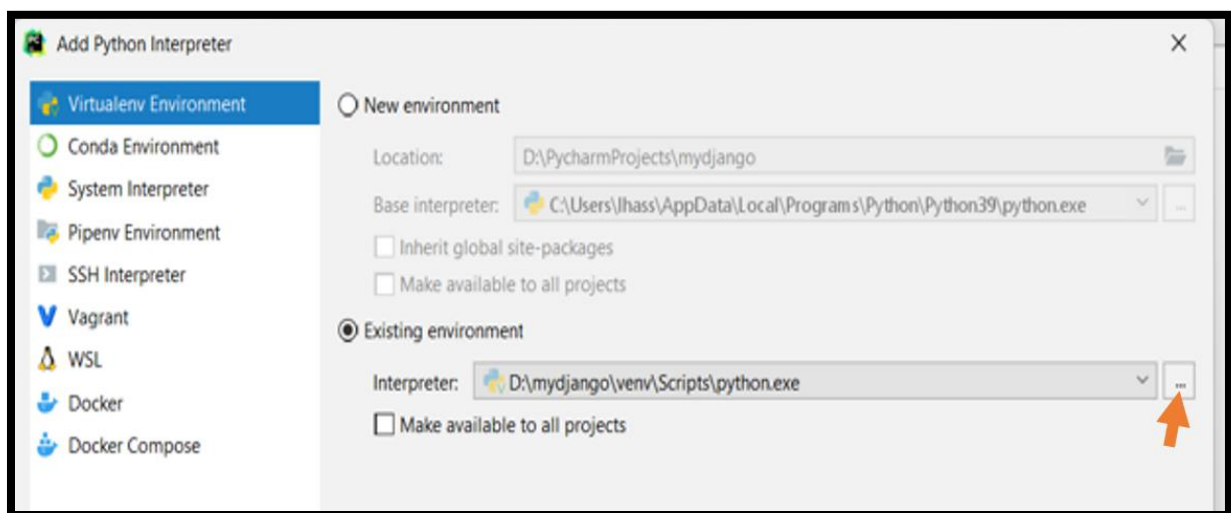
La liste des interpréteurs s'affiche dans la fenêtre ci-dessous.



Si votre environnement n'apparaît pas comme il est le cas ici, cliquer sur le bouton +

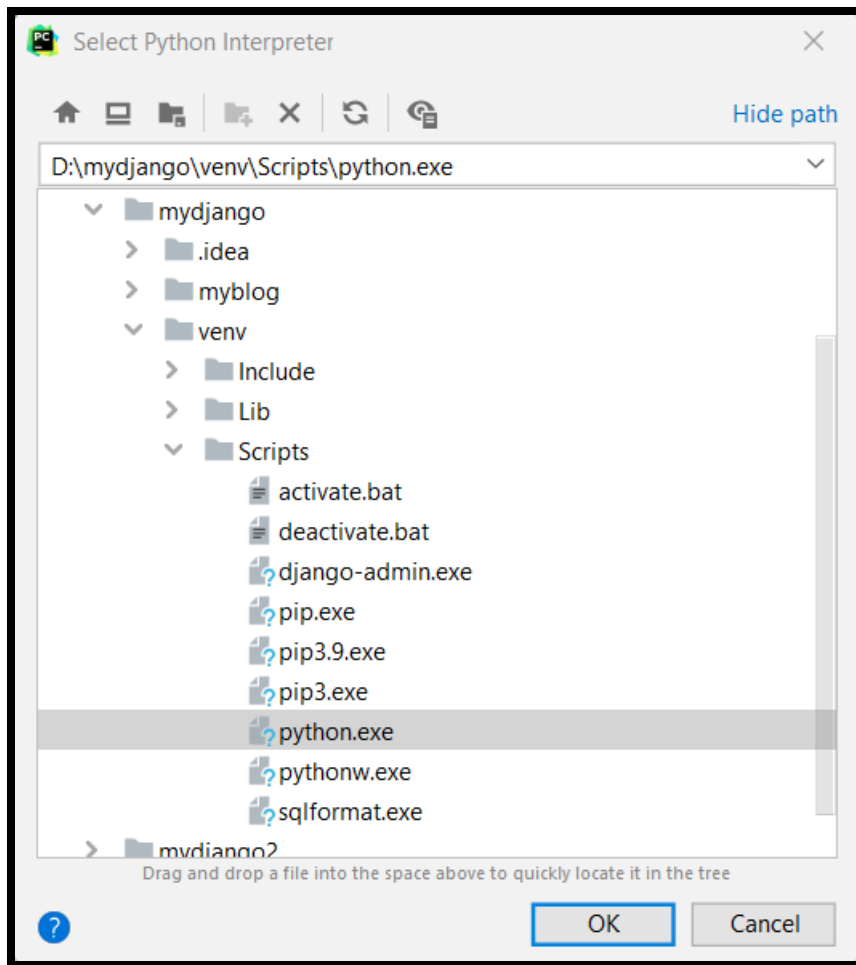


Ce qui affiche la fenêtre ci-dessous :



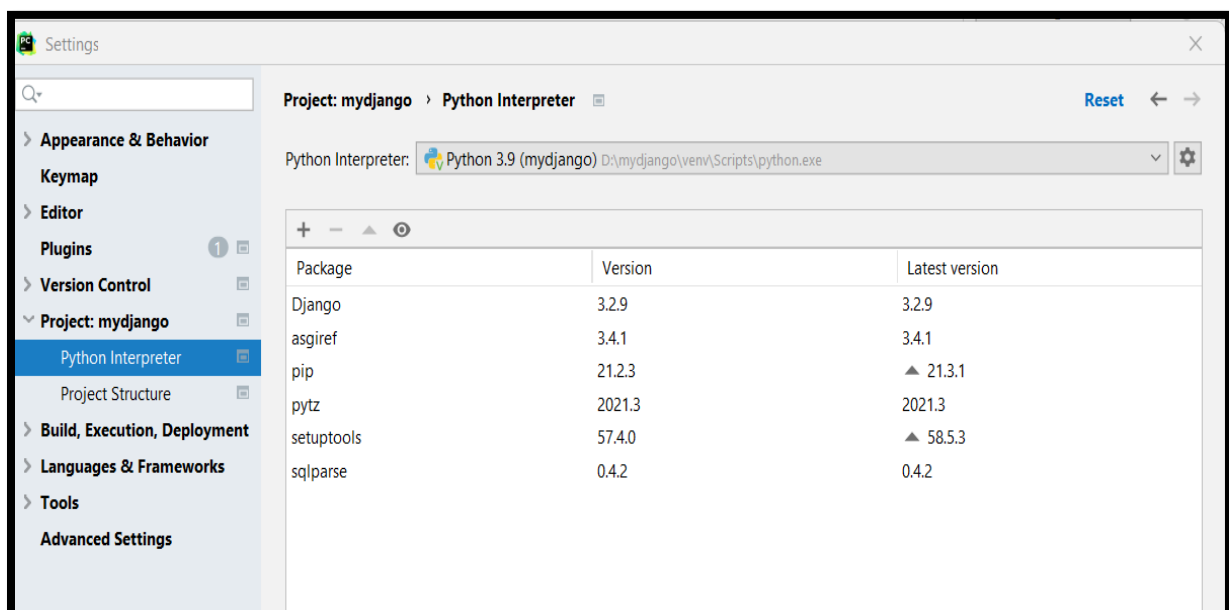
Nous remarquons ici qu'il a détecté notre environnement **venv**.

Au cas où il ne le trouve pas cliquer sur les trois points en face de la zone déroulante **Interpreter** de l'option **Existing environnement** ce qui affiche la fenêtre ci-dessous.



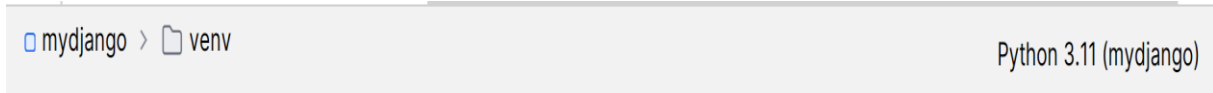
Sélectionner le fichier **python.exe** du sous-dossier **Scripts** du votre dossier **venv**.

Cliquer successivement sur les boutons jusqu'à obtenir la fenêtre ci-dessous



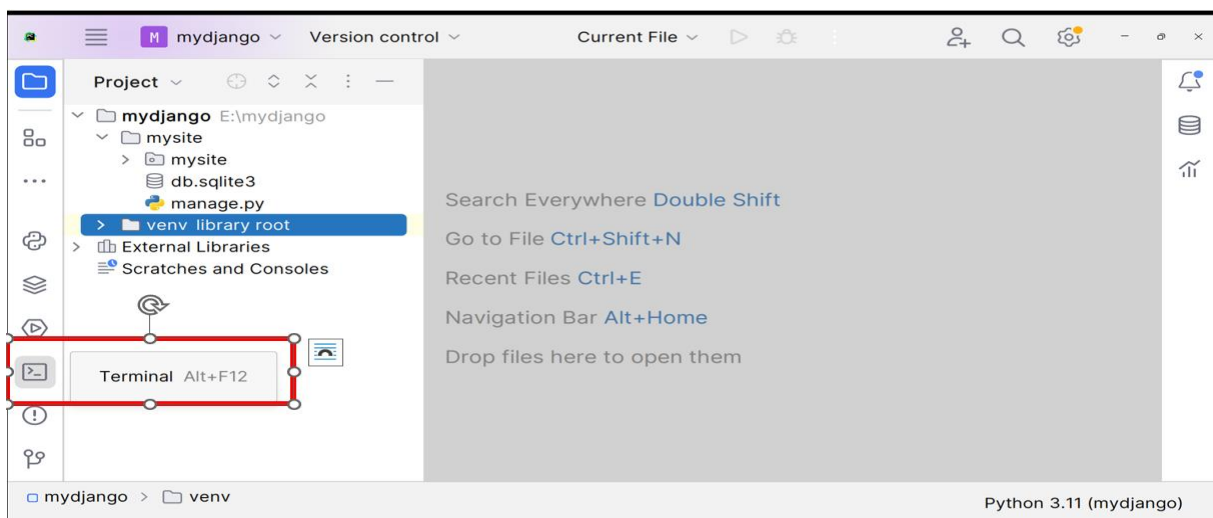
Nous trouvons dans cette fenêtre les packages que nous avons installés parmi lesquels se trouve Django.

On peut le voir tout en bas à droite de la fenêtre de pycharm avec le nom de notre projet entre parenthèses.

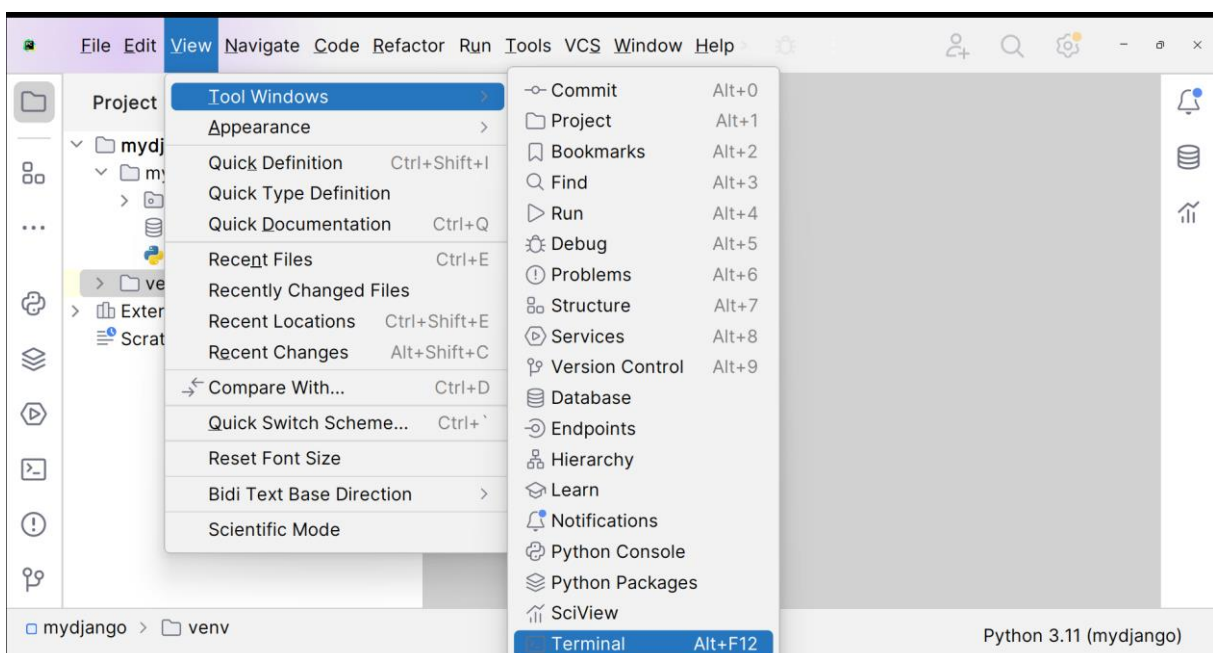


Changement du chemin du départ du terminal

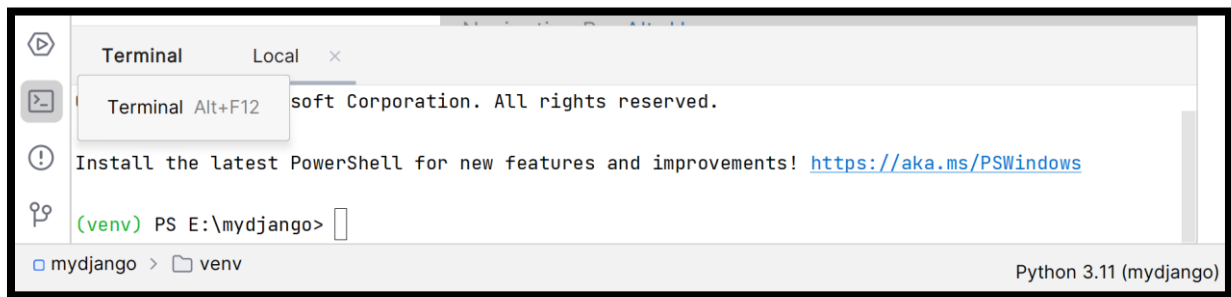
Pour ouvrir le terminal à partir du Pycharm cliquer sur le bouton Terminal qui se trouve à gauche



ou sélectionner la commande : **View -> Tool Windows -> Terminal** comme le montre la figure ci-dessous.

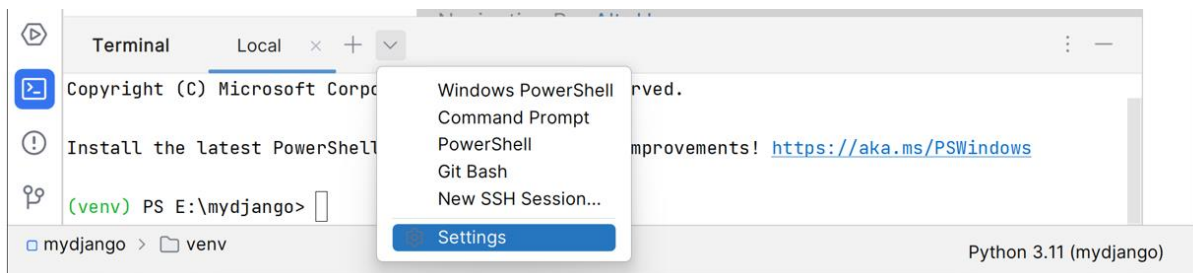


La fenêtre du terminal s'affiche en bas.

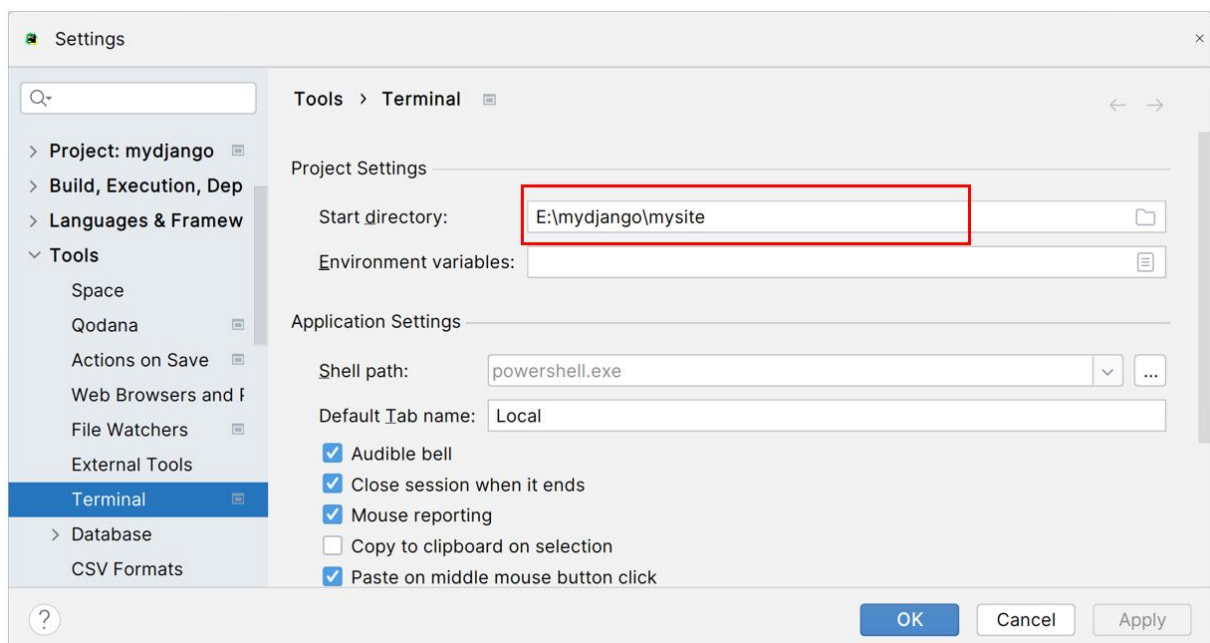


Puisque nous avons souvent besoin d'exécuter le fichier **manage.py** de notre projet django, nous allons configurer **PyCharm** pour que le **terminal** s'ouvre directement dans le dossier **mysite** qui contient le fichier **manage.py**.

Cliquer sur la petite flèche pour sélectionner l'interpréteur de commande. La fenêtre ci-dessus montre que c'est l'interpréteur **PowerShell 7.1.5** qui a été sélectionné. Vous pouvez sélectionner le **Command Prompt**. **Cliquer sur settings...**



Dans la fenêtre qui s'affiche, saisir **E:\mydjango\mysite** comme le montre la figure ci-dessous.



Quelques commandes supplémentaires utiles :

1-Vérifier que le module django est bien installé :

Méthode 1 : exécuter la commande :

```
(venv)E:..>Python -m django --version
```

Méthode 2 : lancer python, importer django, puis appeler la méthode get-version de django

```
(venv)E :..>Python
>>>import django
>>>django.get_version()
```

2- Geler l'environnement virtuel dans un fichier texte : requirements.txt

Pour garder en mémoire les bibliothèques que nous avons installées ainsi que leur version précise on va geler notre environnement virtuel dans un fichier texte que l'on appelle généralement requirements.txt . pour cela on va utiliser la commande pip freeze.

```
(venv)E:..>pip freeze > requirements.txt
```

3- Recréer l'environnement à partir du fichier requirements.txt

Désinstaller django :

```
pip uninstall django
```

Installer à partir du fichier requirement

```
pip install -r requirements.txt
```

4-Demander l'aide sur django

```
(venv) E:\mydjango>django-admin --help
```

5- Demander l'aide sur une commande spécifique

```
(venv) E:\mydjango>django-admin help shell
```

6-Erreurs courantes

- 1- Oubli d'activer l'environnement virtuel
- 2- Si vous rencontrez des problèmes pour exécuter : **django-admin**
Effectuer les opérations suivantes :
 - a. Désactiver l'environnement virtuel (commande deactivate)
 - b. Supprimer le dossier de l'environnement virtuel
 - c. Recréer l'environnement virtuel
 - d. Réinstaller Django