



TP1 : L'environnement Django-Python, le premier pas.

Définition: Django est un framework web gratuit et open-source basé sur *Python*, qui favorise le développement web2.0 rapide ainsi qu'une conception propre et pragmatique. Django offre aux développeurs les ressources prêtes à l'emploi nécessaires. Il fournit du code pour des opérations courantes telles que la gestion de bases de données, la création de modèles HTML, la gestion des routes d'URL, la gestion de sessions et la sécurité.

Installation de Python & Django:

- 1- Téléchargez et installez une version récente de **Python3** à partir du site officiel : <https://www.python.org/downloads/> (N'oubliez pas de cocher l'option « Add Python to PATH » lors de l'installation).
- 2- Créez et accédez ensuite au répertoire de votre projet, par exemple **TP1**.
- 3- À partir de votre répertoire de projet, ouvrez l'interpréteur de ligne de commande (**cmd**).
 - Les commandes suivantes seront exécutées à l'aide de cet interpréteur de ligne de commande.

- Créez un environnement virtuel pour isoler vos dépendances de packages localement

```
Python -m venv env_name
```

À la base, l'objectif principal des environnements virtuels Python est de créer un environnement isolé pour les projets Python. Cela signifie que chaque projet peut avoir ses propres dépendances.

- Activez l'environnement virtuel précédemment créé

```
Linux:      source env_name/bin/activate
Windows:   env_name\Scripts\activate
```

- Installez le framework Django dans l'environnement virtuel

```
pip install django
```

- Mettez en place un nouveau projet avec une seule application

```
django-admin startproject tp1      # créer un nouveau projet ayant comme nom : tp1
cd tp1                             # se placer à l'intérieur du répertoire créé
django-admin startapp app1         # créer la 1ère application de votre projet nommée : app1
```

- Synchronisez maintenant votre base de données pour la première fois :

```
Python manage.py migrate
```

Après avoir exécuté cette commande, un nouveau fichier de base de données (par défaut, au format **SQLite**) sera créé dans votre répertoire.

- Nous allons également créer un utilisateur initial nommé : **admin** avec le mot de passe : **pswd123**. Nous nous authentifierons en tant que cet utilisateur plus tard dans notre exemple.

```
Python manage.py createsuperuser --email admin@example.com --username admin
```

- Lancez le serveur Django sur une machine locale pour tester si l'installation est bien faite

```
Python manage.py runserver
```

Si vous aurez le message suivant : « *Starting development server at <http://ip:port/>* », alors l'installation s'est déroulée avec succès, et le serveur de développement est démarré : l'adresse localhost = **ip@** (exp : **127.0.0.1**) et numéro de port = **port** (exp : **8000**).

Pour vous assurer que votre utilisateur a été créé avec succès, vous pouvez vous authentifier en utilisant son **username** et son **password** pour accéder à l'administration Django, il suffit juste de rajouter **/admin** à la fin de l'url où votre serveur est démarré (par exemple : <http://127.0.0.1:8000/admin>).