

## ■ Starter Pack Django (Windows / cmd)

Étape 1 : Ouvrir un terminal

- Dans VS Code ou dans cmd.exe.

Étape 2 : Créer un dossier pour tes projets

```
cd Desktop
```

```
mkdir projects
```

```
cd projects
```

Étape 3 : Créer un dossier pour l'app Django

```
mkdir django-web-app
```

```
cd django-web-app
```

Initialiser un repository Git (optionnel mais conseillé) :

```
git init
```

Étape 4 : Créer un environnement virtuel

Vérifier la version de Python :

```
python --version
```

Créer l'environnement virtuel :

```
python -m venv env
```

Vérifier que env/ a bien été créé :

```
dir
```

## ■ Ajouter env/ dans ton .gitignore

Étape 5 : Activer l'environnement virtuel

```
env\Scripts\activate
```

➡■ Ton terminal doit afficher (env) au début.

Pour le désactiver :

```
deactivate
```

Étape 6 : Installer Django

```
pip install django
```

Vérifier la version installée :

```
python -m django --version
```

Étape 7 : Sauvegarder les dépendances

```
pip freeze > requirements.txt
```

■ Le fichier requirements.txt contiendra :

- Django
- asgiref
- sqlparse
- tzdata (si inclus)

■ Voilà, tu as ton environnement Django prêt à l'emploi sous Windows !

■ Suite Starter Pack Django (Windows / cmd)

■ Projet créé : merchex

■ Migrations effectuées

■ Serveur lancé et accessible sur <http://127.0.0.1:8000/>

Tu es officiellement prêt à coder ton app Django ■

---

■ Prochaines étapes conseillées :

Étape 8 : Créer une application (exemple : portfolio)

`python manage.py startapp portfolio`

Étape 9 : Ajouter l'app dans merchex/settings.py :

```
INSTALLED_APPS = [  
    "django.contrib.admin",  
    "django.contrib.auth",  
    "django.contrib.contenttypes",  
    "django.contrib.sessions",  
    "django.contrib.messages",  
    "django.contrib.staticfiles",  
    "portfolio", # ■ ajouté  
]
```

Étape 10 : Créer une vue simple dans portfolio/views.py :

```
from django.http import HttpResponse  
  
def home(request):  
    return HttpResponse("Bienvenue dans mon portfolio Django ■")
```

Étape 11 : Ajouter une URL dans merchex/urls.py :

```
from django.contrib import admin  
from django.urls import path  
from portfolio.views import home  
  
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path("", home, name='home'),  
]
```

Étape 12 : Relancer le serveur

`python manage.py runserver`

➡■ Résultat : sur <http://127.0.0.1:8000/> tu verras ton premier message personnalisé.

---

■ Tu as maintenant une base Django fonctionnelle (projet + app).

Tu peux ajouter des templates, des fichiers statiques (CSS, JS, images) et commencer ton portfolio.