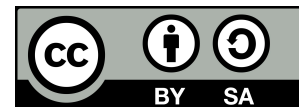


CURSO PYTHON

APLICACIONES WEB CON DJANGO



Autor: Jon Vadillo
www.jonvadillo.com



django

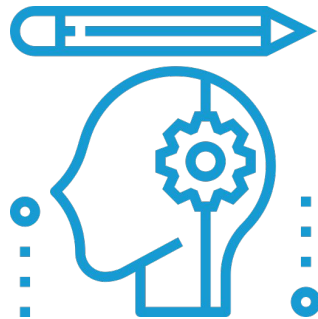
Contenidos

1. **Introducción e instalación**
2. Crea tu primer [proyecto](#) en Django
3. Crea tu primera [aplicación](#) en django
4. El [modelo](#) en Django, [acceso a datos](#) y la aplicación de administrador
5. [Vistas](#) y [plantillas](#) en Django
6. [Vistas basadas en clases](#) (Class Based Views)
7. [Formularios](#) en Django

1. Introducción e instalación

Metodología

- Aprendizaje continuo
- Learn by doing
- Paso a paso
- Código en Github
- Construye aplicaciones reales
- Fundamentos básicos y buenas prácticas



¿Qué es Django?

Django es un **framework web** de alto nivel que permite el **desarrollo rápido de sitios web** seguros y mantenibles

¿Por qué Django?

- Incluye **de serie un conjunto de componentes** para gestionar la mayoría de las tareas de desarrollo web
- Se **adapta** a cualquier tipo de aplicación web
- Permite la construcción de aplicaciones de forma **segura**
- Gratuito y de **código abierto**
- Tiene una **comunidad** próspera y activa
- Una gran **documentación**

Utilizan Django



Instagram

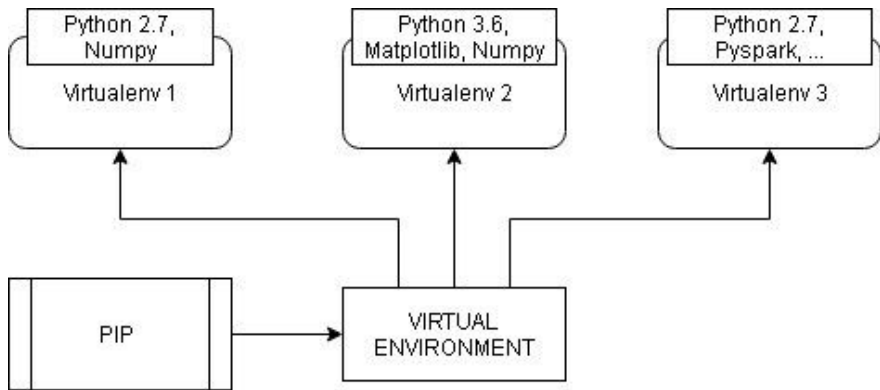
The
Washington
Post



moz://a

¿Qué es un entorno virtual?

- ❑ Un entorno virtual permite tener uno o más **espacios aislados** donde poder instalar distintas versiones de programas y paquetes.
 - ❑ Está formado por un intérprete de Python y los paquetes instalados.
- ❑ **Evitamos conflictos** de versiones de librerías, etc.



¿Qué es *venv*?

- ❑ Es una herramienta de Python para crear entornos virtuales (espacios aislado que permite gestionar dependencias de un proyecto de Python sin interferir con otras aplicaciones o proyectos en el sistema).
- ❑ Lo único que hacemos es decirle a Python: “carga los módulos instalados en una zona en lugar de utilizar los instalados en el sistema”.
- ❑ Ventajas
 - ❑ Realizar pruebas con nuevos paquetes sin necesidad de comprometer los paquetes del sistema.
 - ❑ Crear **un entorno con paquetes aislados** de los del sistema para un proyecto o programa.
 - ❑ **Aislar las dependencias de nuestros proyectos** de las del sistema, para poder importarlas y exportarlas en otros entornos.

Pasos en Windows

- ❏ Para crear un entorno, abrir CMD, navegar a tu proyecto y escribir:

```
python -m venv nombre_del_entorno
```

- ❏ Esto creará una carpeta llamada `nombre_del_entorno` en tu proyecto, donde se instalarán los paquetes necesarios para ese entorno.
- ❏ Para activar el entorno en CMD: `nombre_del_entorno\Scripts\activate`
- ❏ Para comprobar, escribir: `where python`
- ❏ Para desactivar el entorno: `deactivate`

Solventar error en Powershell

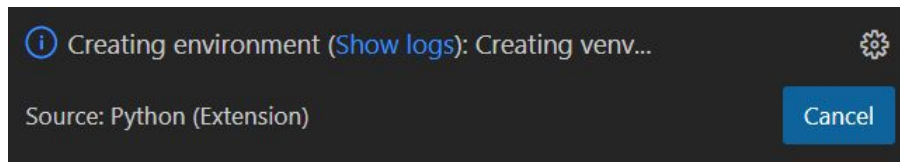
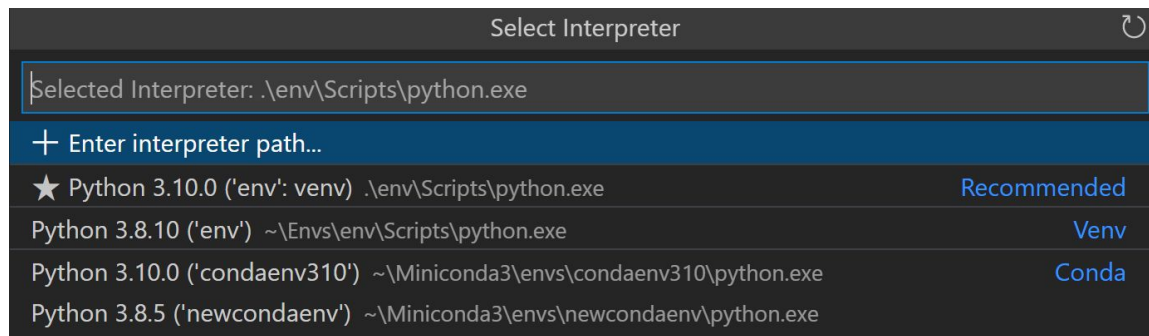
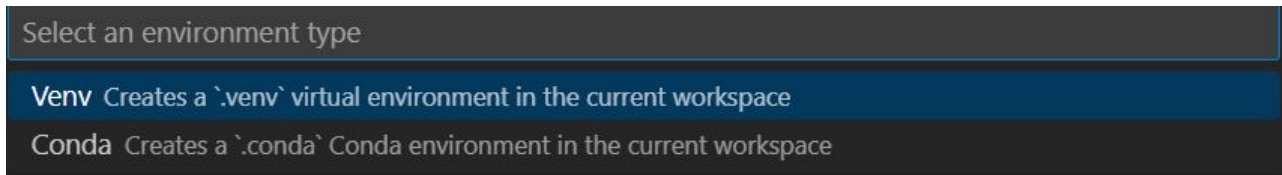


Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser

```
PS C:\Users\developer\Documents\dev\iw-2025\iw-2025-django> .\.venv\Scripts\Activate
_.venv\Scripts\Activate : No se puede cargar el archivo C:\Users\developer\Documents\dev\iw-2025\iw-2025-django\.venv\Scripts\Activate.ps1 porque la ejecución de scripts está deshabilitada en este sistema. Para obtener más información, consulta el tema about_Execution_Policies en https://go.microsoft.com/fwlink/?LinkID=135170.
En línea: 1 Carácter: 1
+ .\.venv\Scripts\Activate
+ ~~~~~
+ CategoryInfo          : SecurityError: (:) [], PSSecurityException
+ FullyQualifiedErrorId : UnauthorizedAccess

PS C:\Users\developer\Documents\dev\iw-2025\iw-2025-django> Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser
PS C:\Users\developer\Documents\dev\iw-2025\iw-2025-django> .\.venv\Scripts\Activate
(.venv) PS C:\Users\developer\Documents\dev\iw-2025\iw-2025-django> |
```

VS Code



Instalar Django

1. **Instalar Python** (comprobarlo con “python --version”)
2. Instalar el **gestor de paquetes PIP**

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
```

```
python get-pip.py
```

3. **Ir al directorio** donde quieres trabajar
4. **Crear un entorno** para virtua y activarlo
5. **Instalar Django:** `pip install django`

Verificar instalación de Django

```
>>> import django  
>>> print(django.get_version())  
2.2.2
```

```
pip freeze
```

Sources

- Documentación oficial: <https://www.djangoproject.com/>
- Mozilla MDN Web Docs: <https://developer.mozilla.org/>