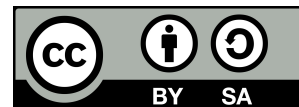


CURSO PYTHON

APLICACIONES WEB CON DJANGO



Autor: Jon Vadillo
www.jonvadillo.com



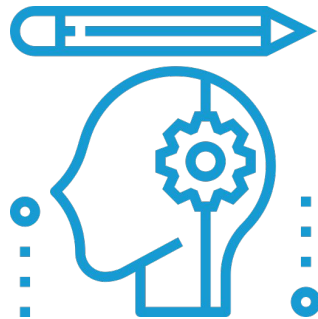
Contenidos

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



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. Instalar **virtualenvwrapper** (provee un entorno virtual dedicado a cada proyecto desarrollado con Django):

```
pip install virtualenvwrapper-win
```

Instalar Django

4. Establecer el directorio de trabajo en donde los entornos virtuales serán almacenados. Para ello añadir la variable del sistema **WORKON_HOME** (en Linux: `export WORKON_HOME = .virtualenvs`).
5. Establecer el directorio de trabajo en donde se crearán los proyectos. Para ello añadir la variable del sistema **PROJECT_HOME** (en Linux: `export PROJECT_HOME=./projects`).
6. **Crear un entorno** para tu primer proyecto: `mkvirtualenv env1`
 - a. Es posible crear el proyecto y el entorno al mismo tiempo con `mkproject`
7. **Instalar Django:** `pip install django`

Variables de entorno

Variables de usuario para Jon

Variable	Valor
CATALINA_HOME	C:\apache-tomcat-9.0.17
IntelliJ IDEA Community Edi...	C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2019...
OneDrive	C:\Users\Jon\OneDrive
Path	C:\Users\Jon\AppData\Local\Programs\Python\Python37-32\Scr...
TEMP	C:\Users\Jon\AppData\Local\Temp
TMP	C:\Users\Jon\AppData\Local\Temp
WebStorm	C:\Program Files\JetBrains\WebStorm 2019.1\bin;

Nueva...

Editar...

Eliminar

Variables del sistema

Variable	Valor
PROJECT_HOME	C:\Users\Jon\Documents\dev\python\projects\
PSModulePath	%ProgramFiles%\WindowsPowerShell\Modules;C:\WINDOWS\sy...
TEMP	C:\WINDOWS\TEMP
TMP	C:\WINDOWS\TEMP
USERNAME	SYSTEM
windir	C:\WINDOWS
WORKON_HOME	C:\Users\Jon\Documents\dev\python\projects\virtualenvs

Nueva...

Editar...

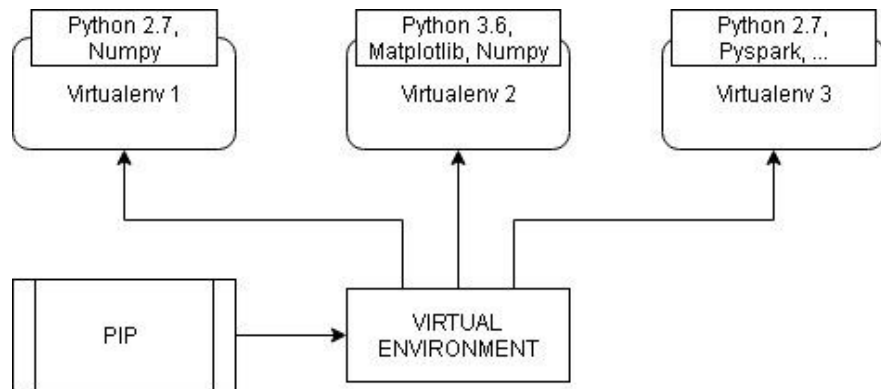
Eliminar

Aceptar

Cancelar

¿Qué es un entorno virtual?

- ❑ Un entorno virtual permite tener uno o más **espacios aislados** donde poder instalar distintas versiones de programas y paquetes
- ❑ **Evitamos conflictos** de versiones de librerías, etc.



Comandos de *virtualenvwrapper*

- `mkvirtualenv env1` — Crear un entorno virtual llamado “env1”
- `deactivate` — Salir del entorno virtual actual
- `workon` — Listar los entornos virtuales disponibles
- `workon name_of_environment` — Activar el entorno virtual
- `rmvirtualenv name_of_environment` — Borrar el entorno especificado..

* Comando `pip freeze`: Permite ver las librerías instaladas

* Más información en:

<https://code.tutsplus.com/es/tutorials/understanding-virtual-environments-in-python--cms-28272>

Verificar instalación de Django

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

Sources

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