

Contenido

<i>Instalación y configuración</i>	<i>2</i>
Python.....	2
Windows	2
MacOs	2
Versiones instaladas.....	3
Ambiente virtual	3
Librerías adicionales.....	4
Sqlite Browser	5
Github (opcional)	5
<i>Instrucciones Django - Terminal.....</i>	<i>5</i>
Crear un Proyecto	5
Iniciar el servidor web	5
Crear una aplicación.....	5
Mapear un modelo con la base de datos.....	6

Instalación y configuración

Python

Descargar el programa para instalar.

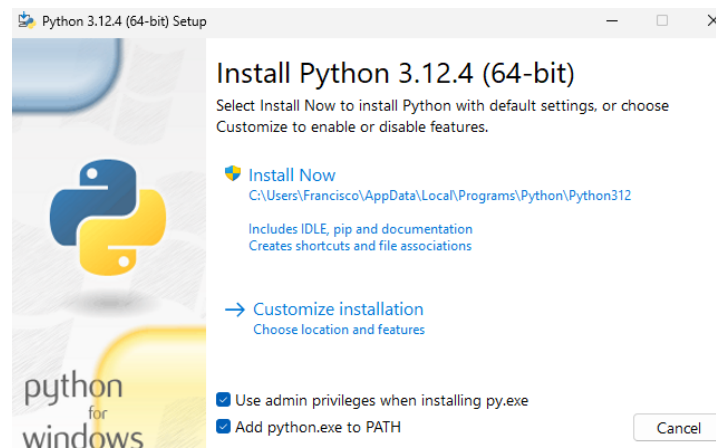
<https://www.python.org/downloads/>

Automáticamente detectará el sistema operativo que se tenga instalado, por ejemplo:



Windows

Es importante seleccionar “agregar python al PATH”



MacOs

Por default en Mac, ya se encuentra instalado python, sin embargo, se puede actualizar haciendo uso de la misma liga.

Recuerde que en MacOS se teclea **python3** en lugar de python

Versiones instaladas

```
Windows PowerShell
PS C:\Users\Francisco> py --list
-V:3.12 *      Python 3.12 (64-bit)
-V:3.11       Python 3.11 (64-bit)
-V:3.10       Python 3.10 (64-bit)
-V:3.8        Python 3.8 (64-bit)
PS C:\Users\Francisco> py -3.12 --version
Python 3.12.4
PS C:\Users\Francisco> py -3.11 --version
Python 3.11.4
PS C:\Users\Francisco>
```

Ambiente virtual

- Crear el ambiente virtual
python -m venv venv (MacOS python3)

```
PROBLEMS  PORTS  DEBUG CONSOLE  OUTPUT  TERMINAL  GITLENS
PS D:\Developer\Git\DAW_M5_Solution> py -3.12 -m venv venv
```

- Activar el ambiente virtual – Windows
.\venv\Scripts\activate

Activar el ambiente virtual – MacOS:
source venv/bin/activate

```
PS D:\Developer\Git\DAW_M5_Solution> py -3.12 -m venv venv
PS D:\Developer\Git\DAW_M5_Solution> .\venv\Scripts\activate
(venv) PS D:\Developer\Git\DAW_M5_Solution> python --version
Python 3.12.4
(venv) PS D:\Developer\Git\DAW_M5_Solution>
```

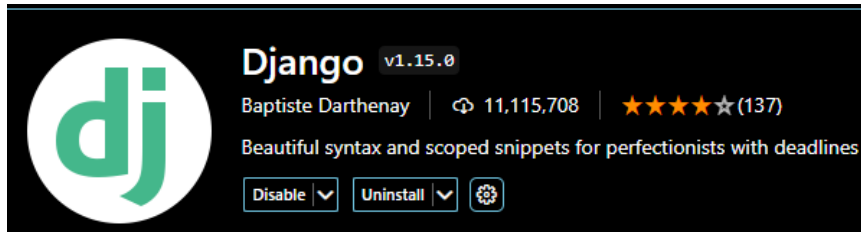
- (Windows*) En caso de tener problemas con la terminal, es necesario :
 - Ejecutar windows power shell (como administrador)
 - Ejecutar el siguiente comando :
 - Set-ExecutionPolicy Unrestricted
- Actualizar pip
python -m pip install --upgrade pip

- Instalar de django (versión 5)
pip install django

- Verificar la versión de django
python -m django --version

```
(venv) PS D:\Developer\Git\DAW_M5_Solution> python -m django --version
5.0.6
```

- En el editor VisualCode instalar la extensión “Django”



Librerías adicionales

- Pylint - Comprobador de sintaxis
pip install pylint
- Pillow – Manejo de Imágenes
pip install pillow
- Editor CKEDITOR – wysiwyg
pip install django-ckeditor
- Verificar librerías instaladas
pip list

```
(venv) PS D:\Developer\Git\DAW_M5_Solution> pip list
Package            Version
-----
asgiref            3.8.1
astroid            3.2.2
colorama           0.4.6
dill               0.3.8
Django             5.0.6
django-ckeditor    6.7.1
django-js-asset    2.2.0
isort              5.13.2
mccabe             0.7.0
pillow             10.3.0
pip               24.1.1
platformdirs       4.2.2
pylint            3.2.5
sqlparse           0.5.0
tomlkit            0.12.5
tzdata            2024.1
```

Sqlite Browser

Descargar e instalar sqlite browser

<https://sqlitebrowser.org/>

Github (opcional)

<https://github.com/>

Registrarse

Instalar github desktop (opcional)

<https://desktop.github.com/>

Instrucciones Django - Terminal

Crear un Proyecto

- Startproject

`django-admin startproject <Nombre del proyecto>`

Ejemplo: `django-admin startproject webpersonal`

Iniciar el servidor web

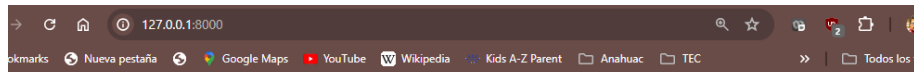
- Iniciar servidor en el puerto 8000 (Default)

`python manage.py runserver`

- Iniciar servidor en otro puerto

`python manage.py runserver <Puerto>`

Ejemplo: `python manage.py runserver 0.0.0.0:8080`



django

View [release notes](#) for Django 5.0



The install worked successfully! Congratulations!

You are seeing this page because `DEBUG=True` is in your settings file and you have not configured any URLs.

Crear una aplicación

- Startapp

`python manage.py startapp <Nombre de la aplicación>`

Ejemplo: `python manage.py startapp core`

Mapear un modelo con la base de datos

- Makemigrations & Migrate – para todas las aplicaciones

```
python manage.py makemigrations
```

```
python manage.py migrate
```

- Makemigrations & Migrate – para una aplicación particular

```
python manage.py makemigrations <Nombre de la aplicación>
```

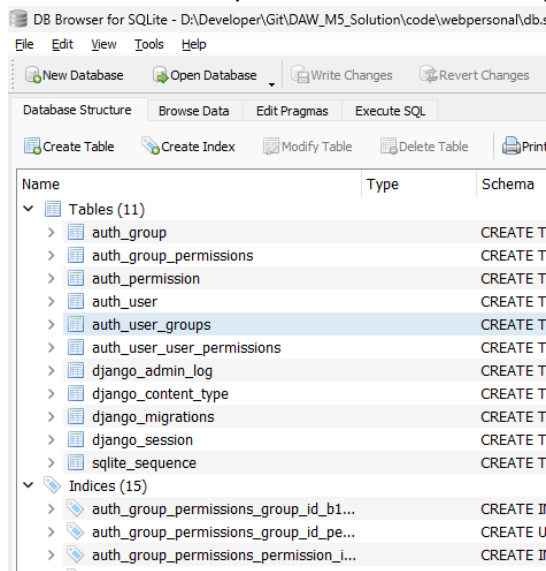
```
python manage.py migrate <Nombre de la aplicación>
```

Ejemplo:

```
python manage.py makemigrations portfolio
```

```
python manage.py migrate portfolio
```

- Verificar las tablas y columnas creadas con sqlite



- Crear un super usuario

```
python manage.py createsuperuser
```
- Verificar el funcionamiento de la aplicación “admin”
<http://127.0.0.1:8000/admin/>

