

Python - Flask

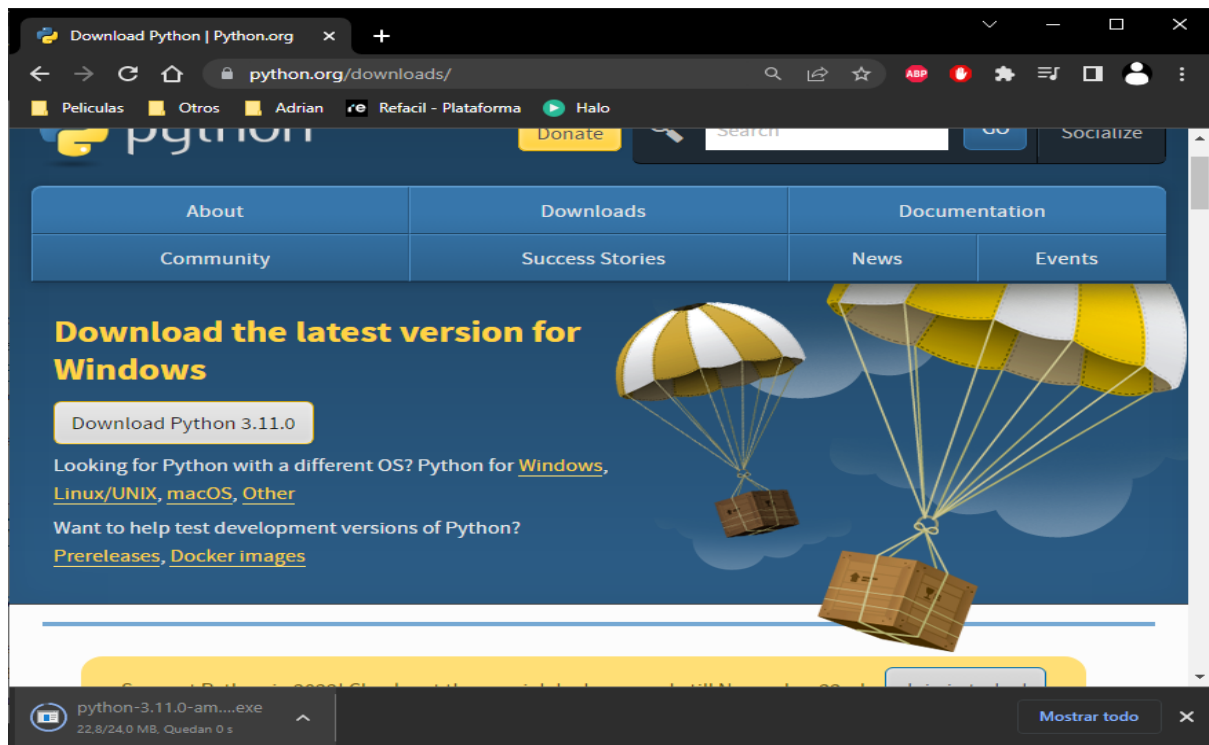
Jose Ivan Geovo Polo
Adrian Manuel Vidal Cardona

Electiva 2

Corte 3

Unicolombo Fundación Universitaria Colombo Internacional
Cartagena de Indias
Noviembre 2022

Python sitio de descarga: <https://www.python.org/downloads/>



Flask sitio oficial: <https://flask.palletsprojects.com/en/2.2.x/>

Es una libreria que debe ser descargada “ \$ pip install flask”



The screenshot shows a web browser window with the address bar displaying 'flask.palletsprojects.com/en/2...'. The page title is 'Installation — Flask Documentati'. The main heading is 'Install Flask'. Below it, the text says 'Within the activated environment, use the following command to install Flask:'. A code block shows the command: `$ pip install Flask`. Below the code block, the text says 'Flask is now installed. Check out the [Quickstart](#) or go to the [Documentation Overview](#).' There is an advertisement for Read the Docs with a code snippet:

```
# .readthedocs.yaml
build.tools:
  python: "3.11"
sphinx:
  configuration: conf.py
python.install:
  - requirements: pip.in
```

 and the text 'Supercharge your Sphinx docs with deployment on Read the Docs. Sign up today!'. At the bottom, there is a 'Contents' section with links to 'Installation', 'Python Version', 'Dependencies', 'Optional dependencies', and 'greenlet'. A version selector shows 'v: 2.2.x'.

Documentación de Python: <https://docs.python.org/3/>

Documentación de Flask: <https://flask.palletsprojects.com/en/2.2.x/>

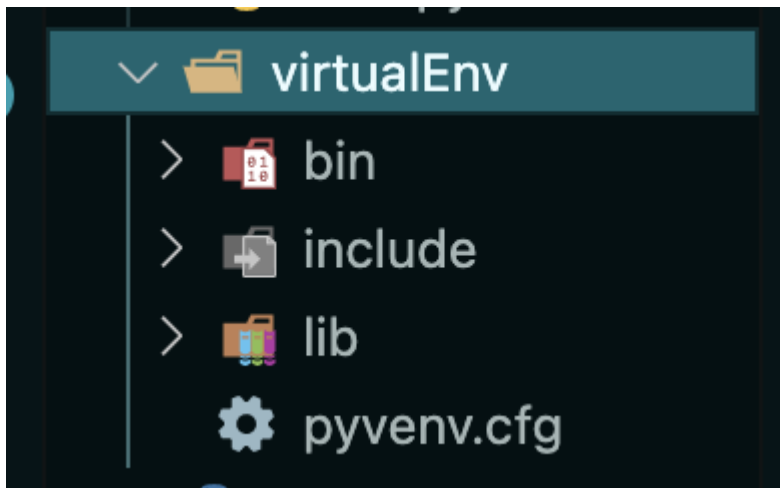
Comunidades de Python: <https://www.python.org/community/>

Comunidades de Flask: <https://flask.palletsprojects.com/en/2.2.x/>

Pasos para configurar y crear el proyecto

Lo primero es crear una carpeta donde queremos que este nuestro proyecto, y a continuación procederemos a crear nuestro entorno virtual en el cual ejecutaremos nuestro código, con el siguiente comando:

```
python3 -m venv /path/to/new/virtual/environment
```



Esto nos creará una carpeta como la presentada en la imagen, ahora lo siguiente es proceder a hacer la instalación de las librerías necesarias, para ello activaremos nuestro entorno virtual con el siguiente comando:

```
project_flask git:(master) x source virtualEnv/bin/activate
```

Y este nos mostrará lo siguiente en la terminal después de ser ejecutado:

```
^C  
(virtualEnv) → project_flask git:(master) x
```

Ahora procederemos a instalar las librerías, la principal librería es flask, con la cual ejecutaremos nuestros templates, entre otras:

```
pip install Flask==2.2.2 mysql-connector-python==8.0.31
```

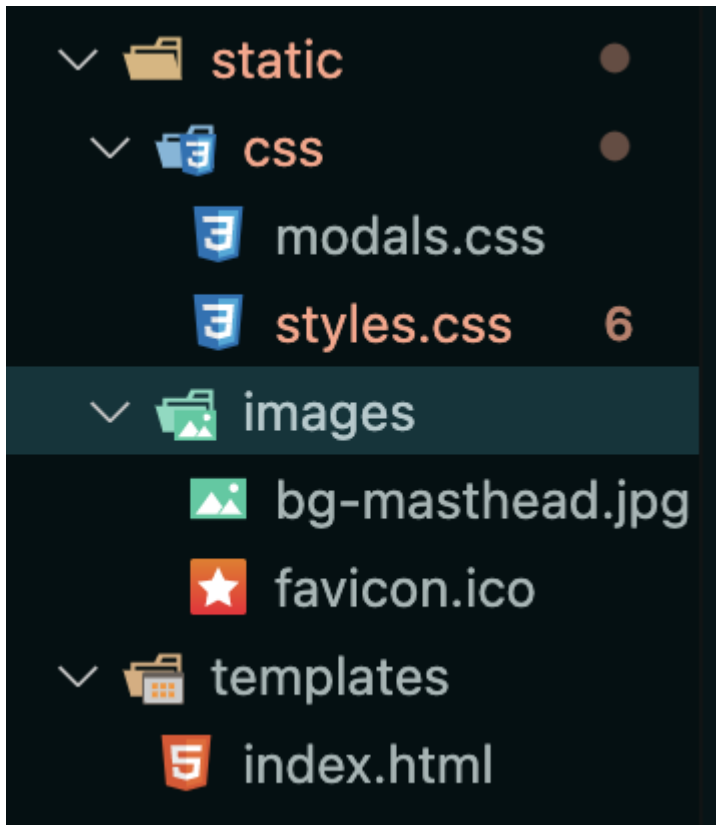
Para este proyecto usamos mysql como base o motor de base de datos, para la conexiona la base de datos creamos un archivo de utilidad encargado de hacer la conexión y de ejecutar los queries solicitados:

También definimos nuestro modelo dentro de una carpeta models:

```
1  class footballTeam:
2
3  def __init__(self):
4      self.id = ''
5      self.clave = ''
6      self.nombre = ''      "nombre": Unknown word.
7      self.eslogan = ''     "eslogan": Unknown word.
8      self.tecnico = ''     "tecnico": Unknown word.
9      self.pais = ''        "pais": Unknown word.
10     self.ciudad = ''       "ciudad": Unknown word.
11     self.categoria = ''    "categoria": Unknown word.
12     self.numGoles = 0      "Goles": Unknown word.
13     self.numPartidosJugados = 0      "Partidos": Unknown
14     self.numPartidosGanados = 0      "Partidos": Unknown
15     self.numCampeonatos = 0          "Campeonatos": Unknown v
16     self.numExpulsiones = 0          "Expulsiones": Unknown v
17     self.numEmpates = 0              "Empates": Unknown word.
```

Con un constructor, con unos valores por default, de ser necesario se reasignan los valores, y para que nuestro archivo modelo pueda ser importado sin problemas creamos un archivo llamado `__init__.py` en blanco.

Una vez tengamos el modelo, la utilidad para la conexión a la base de datos y el archivo principal para el arranque de flask, crearemos una carpeta llamada templates donde irán nuestros archivos html o plantillas, una carpeta llamada static donde irá una carpeta para nuestros archivos css y una carpeta para imágenes, flask por defecto busca la carpeta templates y static para cargar nuestros archivos.



Cabe resaltar que para nuestro proyecto también hace parte del scaffolding un archivo de requirements.txt el cual tiene todas las librerías y versiones instaladas de nuestro proyecto, este se puede generar con el siguiente comando:

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

Y un archivo llamado Makefile sin extensión, el cual contendrá nuestros scripts para ejecutar nuestro proyecto, los cuales se ejecutan con el comando make seguido de nuestro target o script definido, como por ejemplo:



Makefile

```
1  ∨ setup: requirements.txt
2      pip install -r requirements.txt
3
4  ∨ run:
5      flask run
6
7  ∨ debug:
8      python app.py
9
10 all: setup run
```