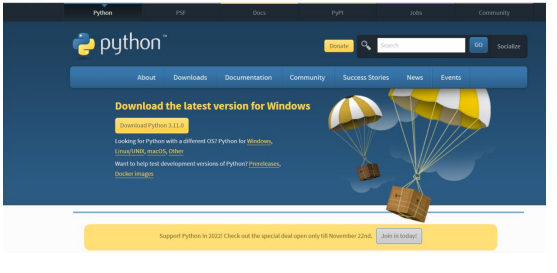
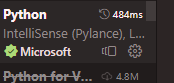
PRIMERA APLICACIÓN WEB CON PYTHON

1. Instalación de Python:

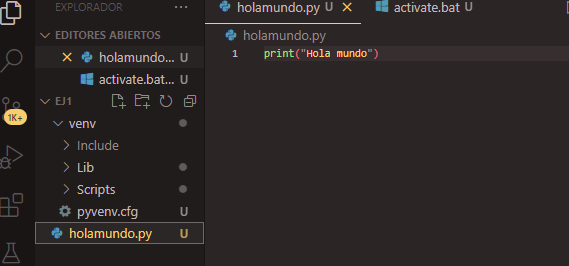


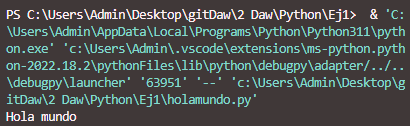
1. Instalación de la extensión en visual studio code:



1. Primera aplicación en phyton:

Primero se crea un archivo con la terminación .py. Luego pondremos echo (“hola mundo”) y cuando ejecutamos el termina se printea



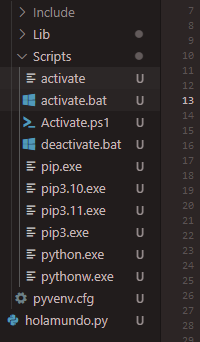


1. Crear el entorno virtual y activarlo

Ejecutar en ese mismo terminal :

Py -m venv c:/desktop/gitdaw/2Daw/EJ1

Se crea una carpeta en el directorio donde se pueden ver varios subdirectorios:



Ejecutamos el archivo actívate.bat, y nos sale un promt:

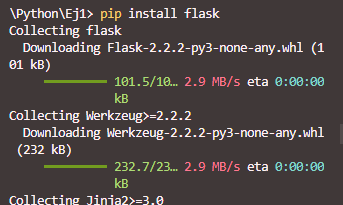


1. Codigo primer app-web:

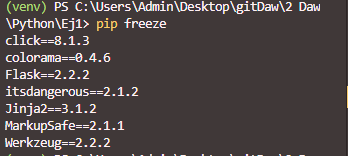
Copiamos el código del ejerccio:



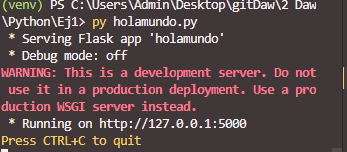
Y luego ejecutamos el comando **pip freeze** en el terminal. Dara un error asi que se tiene que instalar con el comando **pip install flask**



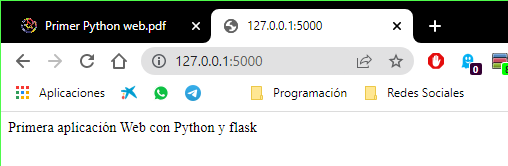
Luego se ejecuta de nuevo el comando **pip freeze** y esta vez obtenemos el siguiente resutlado:



Ejecutamos py holamundo.py



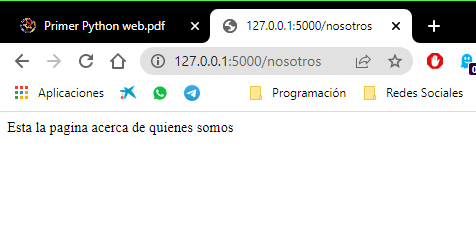
Cuando accedemos a esa url comprobamos que esta cargando la app bien:



* Creando más rutas:

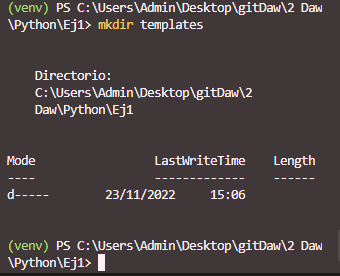
Ruta nosotros:



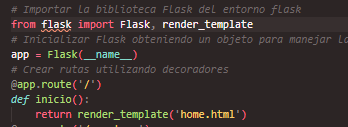


1. Añadiendo la V (vistas) al modelo MVC

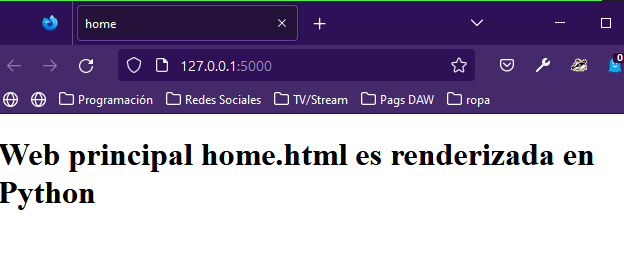
Primero creamos el nuevo directorio, y lo llamaremos “templates”



Luego modificamos el archivo holamundo.py a appy.py, y le cambiamos tambien un par de líneas tal y como viene en la guía.

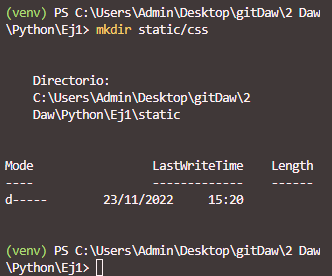


Esto produce que ahora lo que se carge el es html que hemos creado:



1. Enriqueciendo las vistas

Primero se crean la carpeta de static, y dentro de ella la de css.

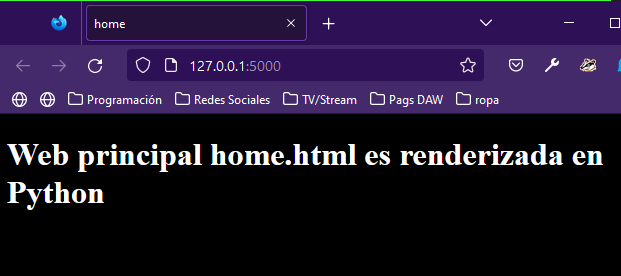


Dentro se crea el archivo css.css y se pone un fondo negro con la letra blanca.

Después se linkea al archivo html a través e una ulr.

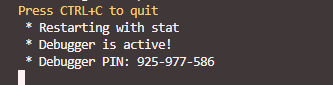


Y comprobamos que funciona:

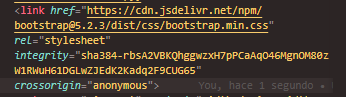


1. Cambiar el modo de depuración a desarrollo.

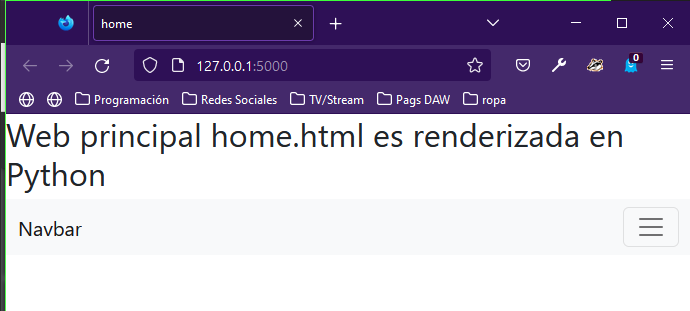
Solo tendremos que añadir dentro del app.run() del archivo appy.py un debug=true. Despues reinciamos el servidor y comprobamos que esta funcionando correctamente



1. Añadir boostrap:

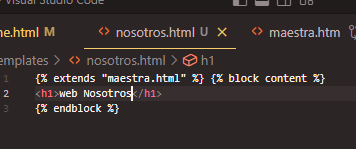


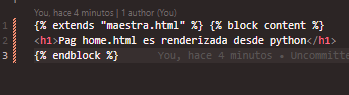
1. Implementamos la navbar de boostrap:



1. Herencia de plantillas

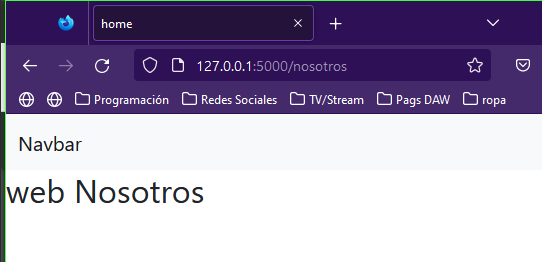
Se modifican home, y se crea maestra.html y nosotros html con el siguiente código:



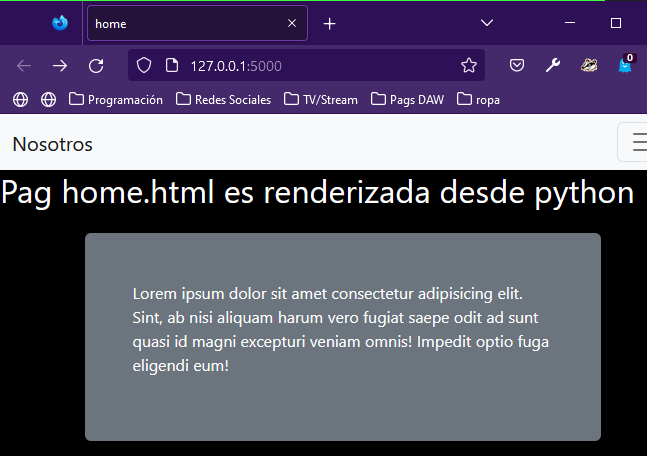


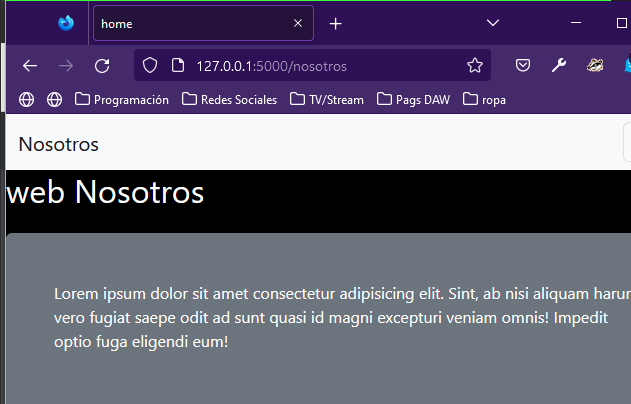
Maestra.html tiene el código original de home.html

Se modifica el appy.py para que la ruta de nosotros sea el render\_template(‘nosotros.html’), de esta forma, cuando ejecutamos appy.py observamos:

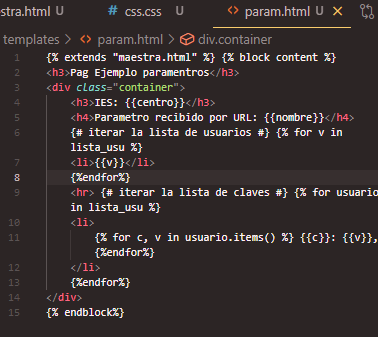


Luego se añade un poco de texto a cada una de las webs:





Creacion de la web param.html:

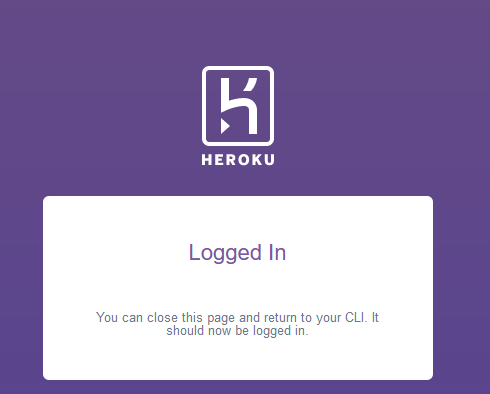


Se añade al appy.py

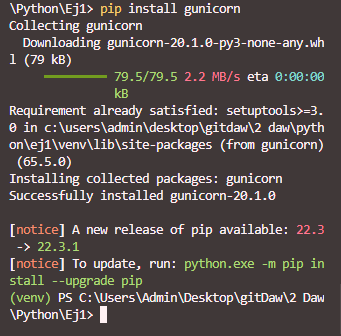


Despues nos creamos la cuenta de heroku, le pondremos una contraseña a través del link que hemos recibido al correo.

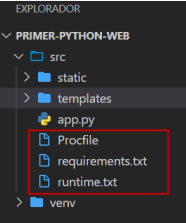
Seguimos los pasos descritos en el guion (ejecución de Loguin de heroku en cmd) y llegaremos a esta pantalla:



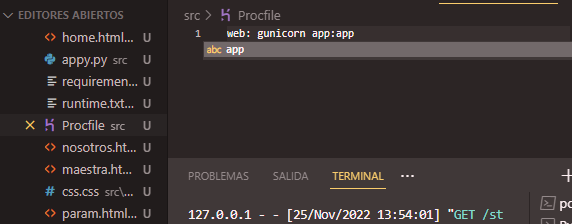
Luego instalaremos con el comando pipi instal gunicorn



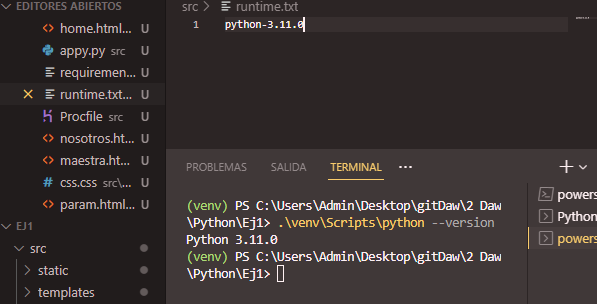
Después creamos los archivos requirements.txt, runtime.txt y procfile.



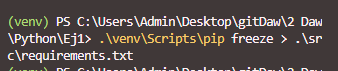
Luego modificamos el Procfile a:



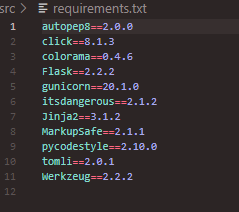
Modificamos el archivo runtime.txt



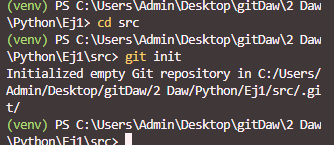
Luego se ejecuta la lista de comandos en la terminal:



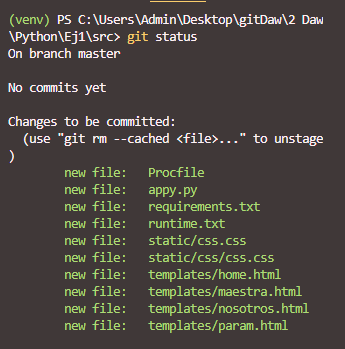
Y como resultado nos dará en el requirements.txt lo siguiente:



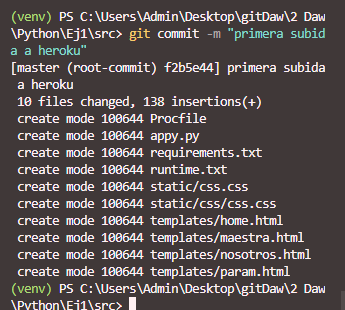
Creamos un repositorio Git y lo iniciamos



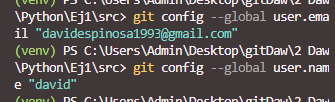
Luego hacemos un git add . y un git status y recibimos por consola lo siguiente:



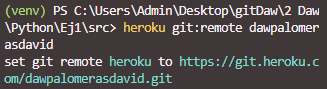
Y luego realizamos la subida:



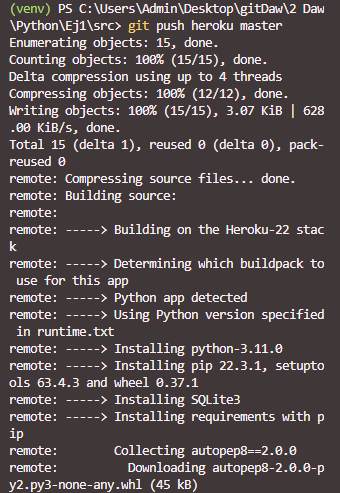
Luego creamos el nombre de la aplicación:



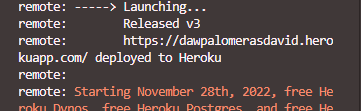
Después de configurar con el mail, y con el nombre, se sincronizara mediante el comando heroku git:remote nombreaplicacion



Despues se desplegara la aplicación con el comando git push heroku rama\_git, que en este caso al ser master será asi: git push heroku master



Al final delt todo, se podrá ver la url donde se ha desplegado la app, en nuestro caso es:



Finalmente para guardar los cambios que se han realizado se hará con los comandos git add . NOTA IMPORTANTE, EL PUNTO AL FINAL ES IMPORTANTE

Para poder ir viendo los cambios se van a realizar se podrá con el comando git status

Para poder lanzar la app se realizara con el comando heroku open, y este nos podrá automáticamente la url del siguiente modo:

<https://dawpalomerasdavid.herokuapp.com>