

Práctica 06. API Rest en Flask

Administracion y Diseño de Bases de Datos

Cheuk Kelly Ng Pante (alu0101364544@ull.edu.es)

8 de diciembre de 2023

Índice general

1. Introducción	1
2. Actividad 1	1
2.1. Instalación del framework <i>Flask</i> y la biblioteca <i>psycpg2-binary</i>	1
2.2. Despliegue de la aplicación web	2
2.3. Creación de la base de datos e inserción de datos	2
2.4. Personalizar la referencia <i>About</i>	2
2.5. Verificar funcionamiento de la operación de visualizar los registros	3
3. Bibliografía	4

1. Introducción

La API RESTful es una interfaz que dos sistemas de computación utilizan para intercambiar información de manera segura a través de Internet. La mayoría de las aplicaciones para empresas deben comunicarse con otras aplicaciones internas o de terceros para llevar a cabo varias tareas

Flask es un framework para desarrollo web escrito en Python. Se puede utilizar para diversos tipos de aplicación, entre ellas, desarrollo de APIs. Existen muchas maneras de implementar un API REST en Flask. Desde usar el framework con lo que ofrece de base, o con la ayuda de extensiones con diferentes configuraciones.

2. Actividad 1

2.1. Instalación del framework *Flask* y la biblioteca *psycpg2-binary*

Para la instalación de *Flask* y *psycpg2-binary*, así como el desarrollo de la práctica se va a crear un entorno virtual con *virtualenv*. Primero se crea un directorio para el entorno virtual y se accede a él.

```
$ mkdir practica_flask
$ cd practica_flask
```

Dentro del directorio se crea el entorno virtual con *virtualenv* y se instalará *Flask* y *psycpg2-binary* dentro de él. Para ello, se ejecutarán los siguientes comandos:

```
$ sudo apt install python3.10-venv
$ python3 -m venv venv
$ source venv/bin/activate
$ pip install Flask
$ pip install psycpg2-binary
```

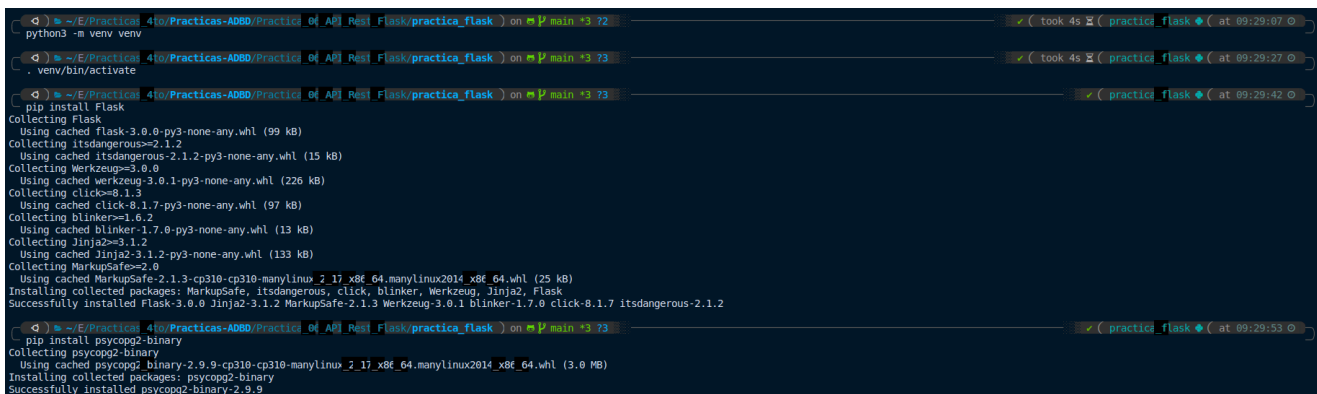
A screenshot of a terminal window showing the installation process. The terminal has a dark background with light-colored text. The prompt is a shell icon followed by the path ~/E/Practicas-ABDD/Practico_04_A01_R01_Flask/practica_flask. The commands and their outputs are as follows:
1. Command: python3 -m venv venv
Output: (no visible output, but the prompt changes to .venv/bin/activate)
2. Command: source venv/bin/activate
Output: (no visible output, but the prompt changes to .venv/bin/activate)
3. Command: pip install Flask
Output: Collecting Flask
Using cached flask-3.0.0-py3-none-any.whl (99 kB)
Collecting itsdangerous<=2.1.2
Using cached itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting Werkzeug<=3.0.0
Using cached werkzeug-3.0.1-py3-none-any.whl (226 kB)
Collecting click<=8.1.3
Using cached click-8.1.7-py3-none-any.whl (97 kB)
Collecting blinker<=1.6.2
Using cached blinker-1.7.0-py3-none-any.whl (13 kB)
Collecting Jinja2<=3.1.2
Using cached Jinja2-3.1.2-py3-none-any.whl (133 kB)
Collecting MarkupSafe<=2.0
Using cached MarkupSafe-2.1.3-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Installing collected packages: MarkupSafe, itsdangerous, click, blinker, Werkzeug, Jinja2, Flask
Successfully installed Flask-3.0.0 Jinja2-3.1.2 MarkupSafe-2.1.3 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 itsdangerous-2.1.2
4. Command: pip install psycpg2-binary
Output: Collecting psycpg2-binary
Using cached psycpg2_binary-2.9.9-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.0 MB)
Installing collected packages: psycpg2-binary
Successfully installed psycpg2-binary-2.9.9

Figura 2.1: Instalación de Flask y psycpg2-binary

2.2. Despliegue de la aplicación web

Para desplegar la aplicación web se va a crear una base de datos en *PostgreSQL* con el nombre de *flask_db*. Para ello, ejecutamos dentro de *psql* el siguiente comando:

```
postgres=# CREATE DATABASE flask_db;
```

Una vez creada la base de datos, sobre el directorio *practica_flask* se ponen los archivos que se van a utilizar para el desarrollo de la práctica. Estos archivos son: *app.py* e *init.py* y dentro de estos ficheros añadimos el usuario y contraseña de *postgres* para poder acceder a la base de datos.

Ya modificados los archivos, se ejecuta el siguiente comando para desplegar la aplicación web:

```
$ python3 app.py
```

También se puede ejecutar el siguiente comando para desplegar la aplicación web:

```
$ flask --app app.py run --host 0.0.0.0 --port=5000
```

Con los comandos anteriores desplegamos la aplicación en local en el puerto 5000, pero va a fallar ya que antes necesita la inicialización de la base de datos, lo cual se hace con el siguiente comando:

```
$ python3 init_db.py
```

2.3. Creación de la base de datos e inserción de datos

La creación de la base de datos se hizo en el apartado anterior y la inserción de datos se crean en el script *init_db.py*.

2.4. Personalizar la referencia *About*

Para personalizar la referencia del *about* creamos un nuevo archivo *html* en el directorio *templates* *about.html* en donde añadimos los nombres y apellidos de los integrantes del grupo. Luego modificamos *base.html* y el fichero *app.py* para poder acceder a la nueva sección.

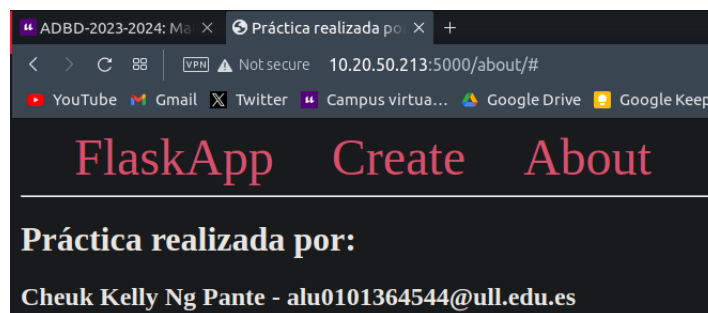


Figura 2.2: About personalizado

2.5. Verificar funcionamiento de la operación de visualizar los registros

3. Bibliografía

- 1.