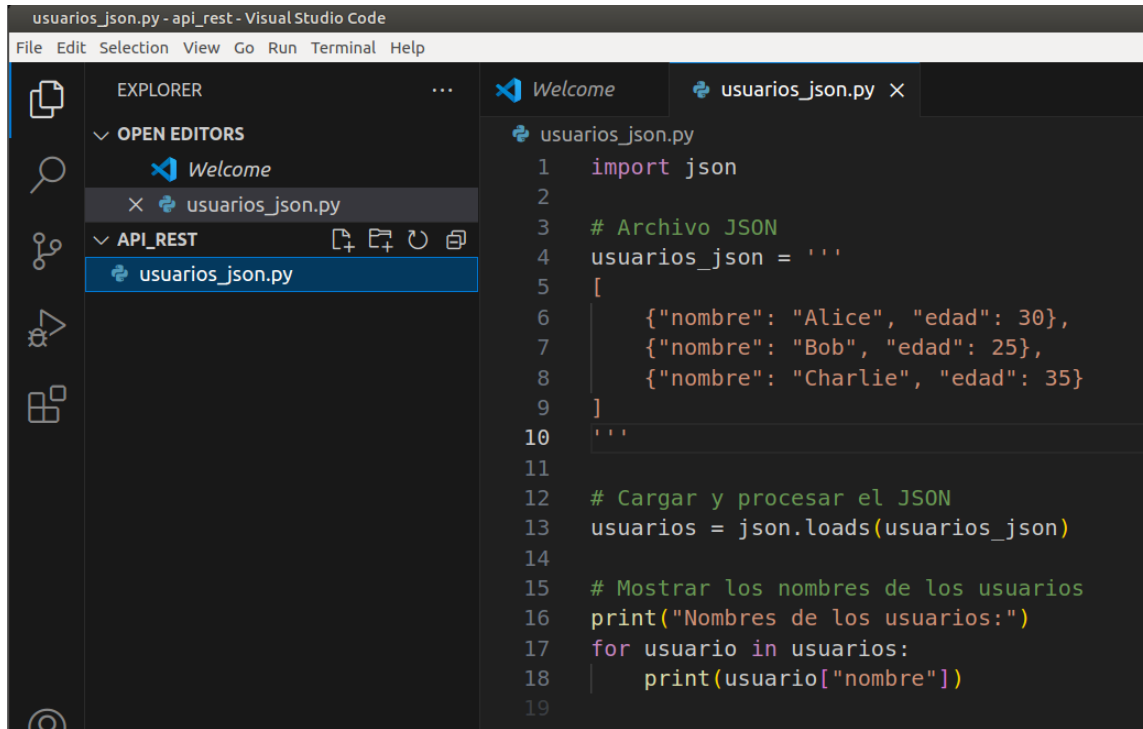


EXAMEN FINAL --- PROGRAMACION EN REDES

Nombre: Juan Daniel Flores Paco

PARTE 2

1.



```
usuarios_json.py - api_rest - Visual Studio Code
File Edit Selection View Go Run Terminal Help

EXPLORER
  OPEN EDITORS
    Welcome
    usuarios_json.py
  API_REST
    usuarios_json.py

usuarios_json.py
1  import json
2
3  # Archivo JSON
4  usuarios_json = '''
5  [
6      {"nombre": "Alice", "edad": 30},
7      {"nombre": "Bob", "edad": 25},
8      {"nombre": "Charlie", "edad": 35}
9  ]
10 '''
11
12 # Cargar y procesar el JSON
13 usuarios = json.loads(usuarios_json)
14
15 # Mostrar los nombres de los usuarios
16 print("Nombres de los usuarios:")
17 for usuario in usuarios:
18     print(usuario["nombre"])
19
```

```
devasc@labvm:~/labs/devnet-src/api_rest$ cat usuarios.json.py
import json

# Archivo JSON
usuarios_json = '''
[
    {"nombre": "Alice", "edad": 30},
    {"nombre": "Bob", "edad": 25},
    {"nombre": "Charlie", "edad": 35}
]

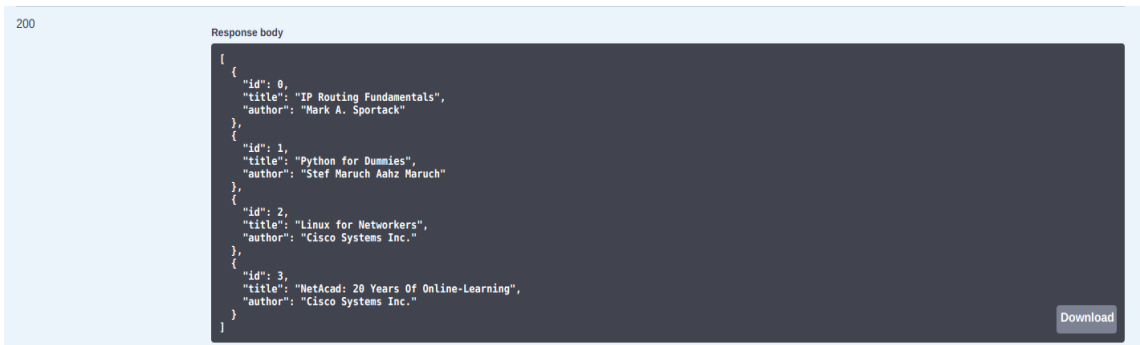
```

```
devasc@labvm:~/labs/devnet-src/api_rest$ python3 usuarios_json.py
Nombres de los usuarios:
Alice
Bob
Charlie

```

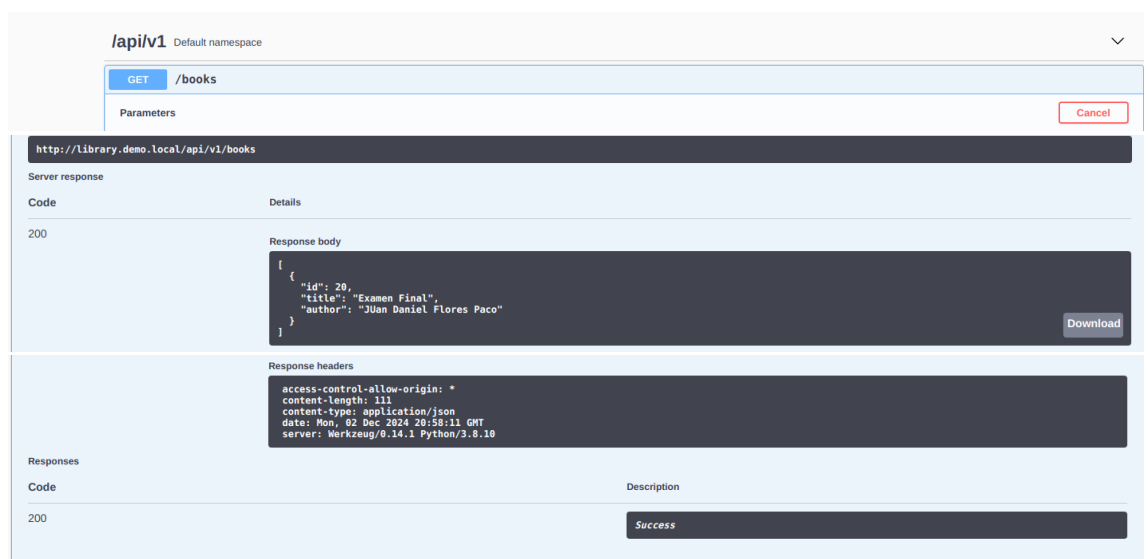
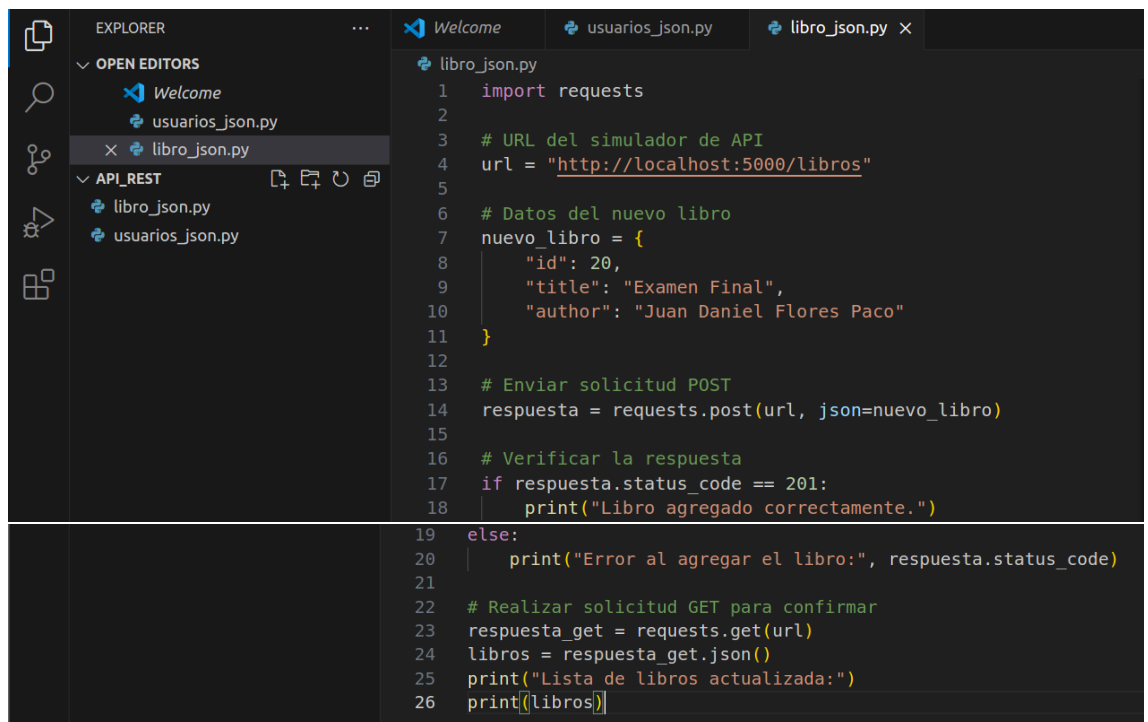
2.

GET	/books
Parameters	
<div>Cancel</div>	



R: Aparecen 4 libros.

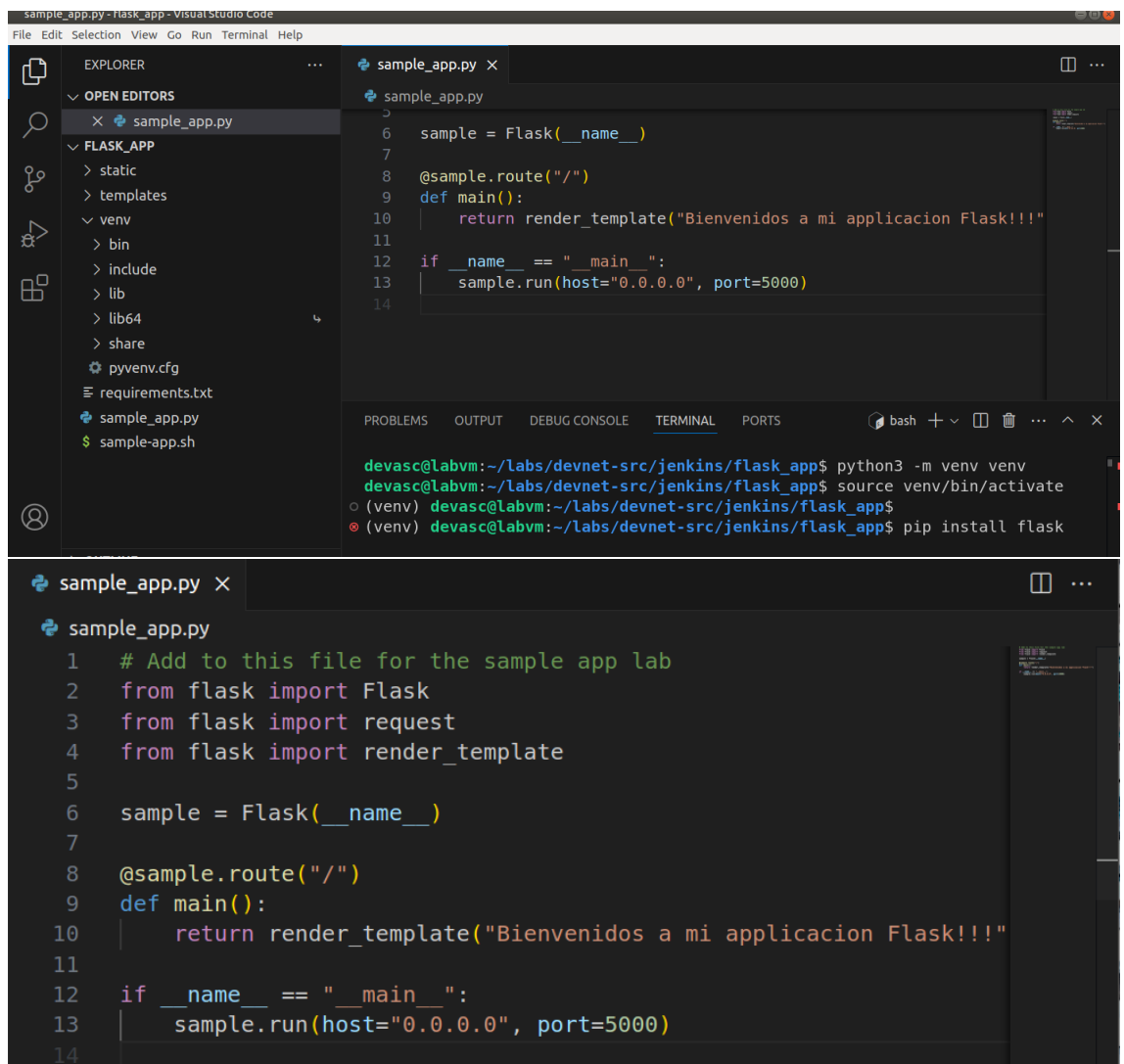
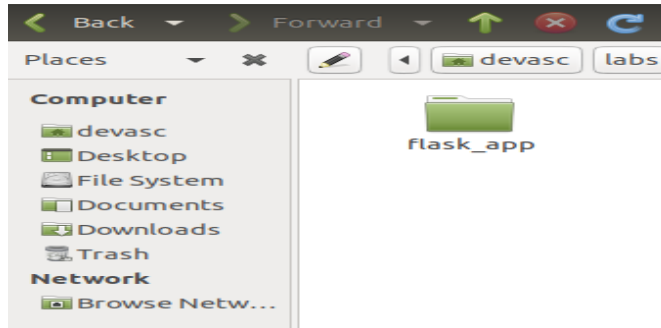
3.



4. Las pruebas están en orden a las capturas seleccionadas y las pruebas están de los resultados. Además, todo está su comprobación.

PARTE 3

- 1.



- 2.

This screenshot shows the Visual Studio Code interface with the `docker_script.sh` file open in the editor. The Explorer sidebar on the left shows the project structure with folders `FLASK_APP` and `static`, and files `index.html` and `style.css`. The editor displays the following script:

```
$ docker_script.sh
1  #!/bin/bash
2
3  echo "Construyendo la imagen de Docker..."
4  docker build -t flask_app .
5
6  echo "Iniciando el contenedor..."
7  docker run -d -p 5000:5000 flask_app
8
9  echo "La aplicación Flask está disponible en http://<ip_vm>:5000"
```

3.

4.

This screenshot shows the Visual Studio Code interface with the `run_docker.sh` file open in the editor. The Explorer sidebar on the left shows the project structure with folders `FLASK_APP` and `static`, and files `index.html` and `style.css`. The editor displays the following script:

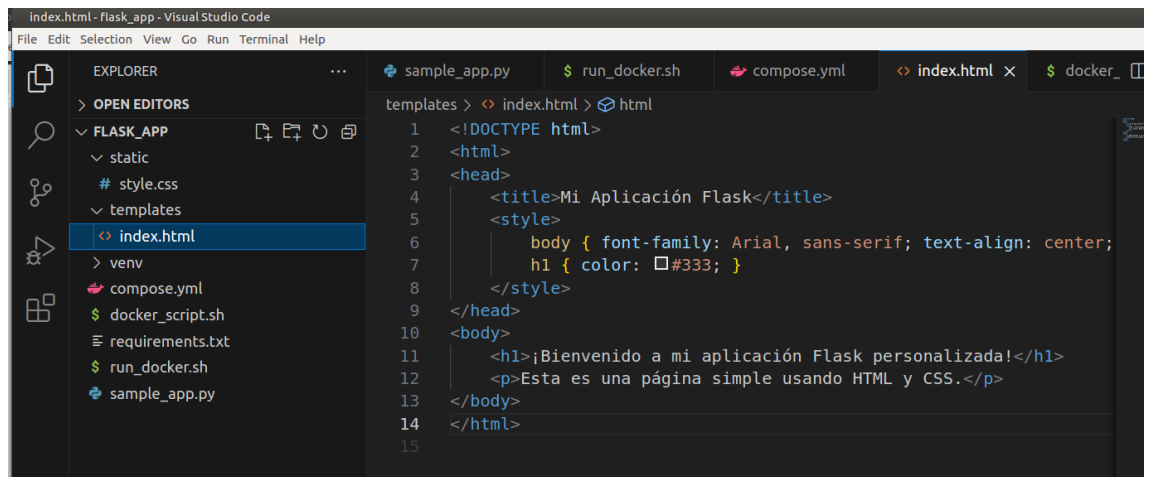
```
$ run_docker.sh
1  #!/bin/bash
2
3  mkdir tempdir
4  mkdir tempdir/templates
5  mkdir tempdir/static
6
7  cp sample_app.py tempdir/.
8  cp -r templates/* tempdir/templates/.
9  cp -r static/* tempdir/static/.
10
11 echo "FROM python" >> tempdir/Dockerfile
12 echo "RUN pip install flask" >> tempdir/Dockerfile
13 echo "COPY ./static /home/myapp/static/" >> tempdir/Dockerfile
14 echo "COPY ./templates /home/myapp/templates/" >> tempdir/Dockerfile
15 echo "COPY sample_app.py /home/myapp/" >> tempdir/Dockerfile
16 echo "EXPOSE 5000" >> tempdir/Dockerfile
17 echo "CMD python /home/myapp/sample_app.py" >> tempdir/Dockerfile
18
19 cd tempdir
20 docker build -t sampleapp .
21 docker run -t -d -p 5000:5000 --name samplerunning sampleapp
22 docker ps -a
```

5.

This screenshot shows the Visual Studio Code interface with the `compose.yml` file open in the editor. The Explorer sidebar on the left shows the project structure with folders `FLASK_APP` and `static`, and files `index.html` and `style.css`. The editor displays the following Docker Compose configuration:

```
compose.yml
1  version: "3.8"
2
3  services:
4    flask_app:
5      build:
6        context: .
7      ports:
8        - "5000:5000"
9      container_name: flask_container
10     restart: always
```

6.



PARTE 4

1. Git init
Git add
Git commit -m "Descripción del cambio realizado"
2. Git checkout -b feature-API
Git commit -m "Implementacion de las nuevas funciones en la API"
3. Git checkout main
Git merge feature-API
Git commit
4. Git remote add origin
https://github.com/JFlores100/examen_final_progrmacion_redes.git
Git push -u origin main
Git push -u origin feature-API