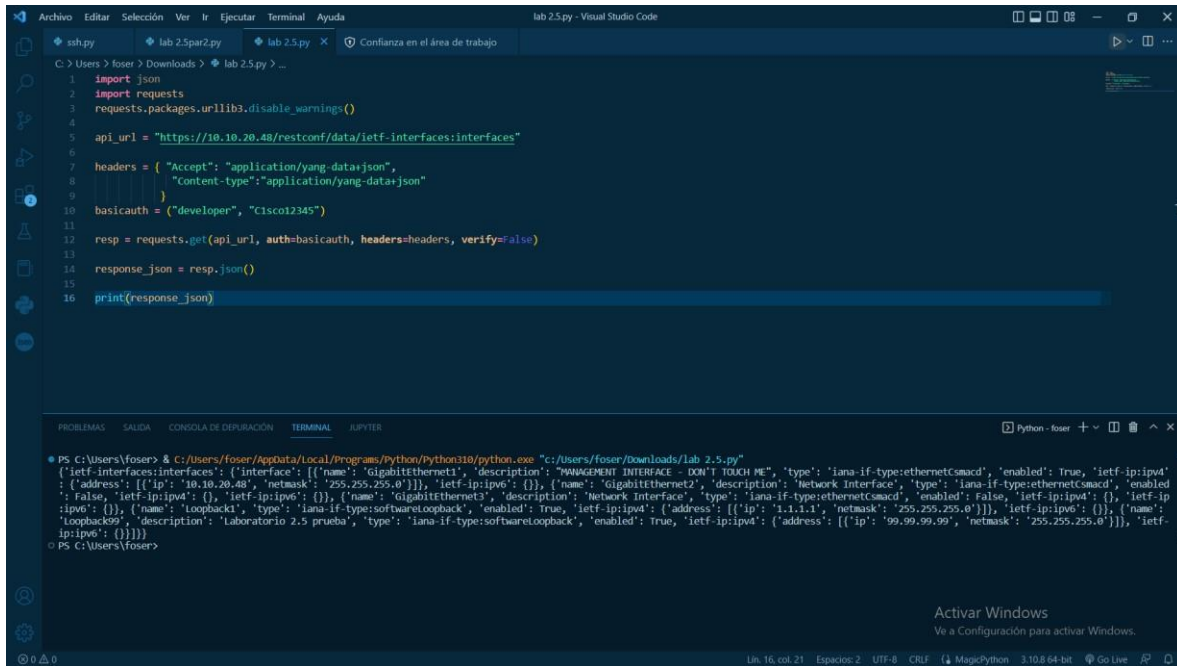


RESTCONF in Python, junto a la salida

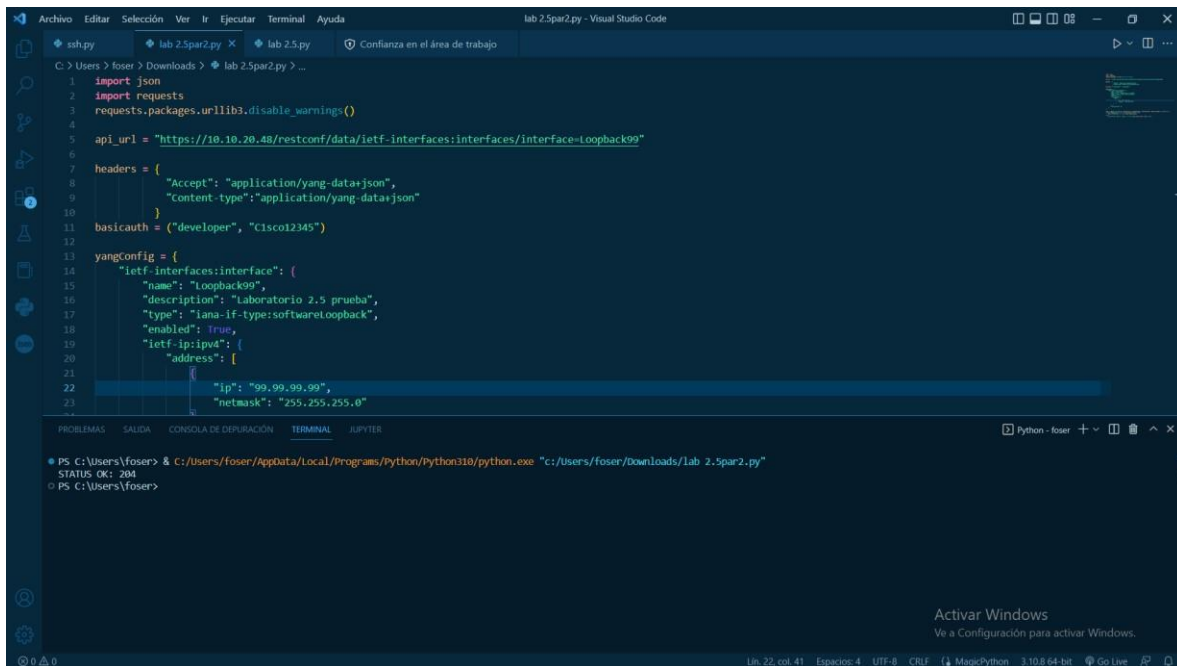


The screenshot shows a Visual Studio Code window with a Python script named `lab 2.5.py` and its terminal output. The script uses the `requests` library to perform a GET request to a RESTCONF endpoint. The terminal output shows the successful execution of the script, displaying the JSON response from the RESTCONF server.

```
1 import json
2 import requests
3 requests.packages.urllib3.disable_warnings()
4
5 api_url = "https://10.10.20.40/restconf/data/ietf-interfaces:interfaces"
6
7 headers = { "Accept": "application/yang-data+json",
8             "Content-type": "application/yang-data+json"
9           }
10 basicauth = ("developer", "Cisco12345")
11
12 resp = requests.get(api_url, auth=basicauth, headers=headers, verify=False)
13
14 response_json = resp.json()
15
16 print(response_json)
```

```
PS C:\Users\foser> & C:\Users\foser\AppData\Local\Programs\Python\python310\python.exe "c:/Users/foser/Downloads/lab 2.5.py"
{'ietf-interfaces:interfaces': {'interface': [{'name': 'GigabitEthernet1', 'description': 'MANAGEMENT INTERFACE - DON\'T TOUCH ME', 'type': 'iana-if-type:ethernetCsmacd', 'enabled': True, 'ietf-ip:ipv4': {'address': [{'ip': '10.10.20.48', 'netmask': '255.255.255.0'}]}, 'ietf-ip:ipv6': {}}, {'name': 'GigabitEthernet2', 'description': 'Network Interface', 'type': 'iana-if-type:ethernetCsmacd', 'enabled': False, 'ietf-ip:ipv4': {}}, {'name': 'Loopback1', 'type': 'iana-if-type:softwareLoopback', 'enabled': True, 'ietf-ip:ipv4': {'address': [{'ip': '1.1.1.1', 'netmask': '255.255.255.0'}]}, 'ietf-ip:ipv6': {}}, {'name': 'Loopback99', 'description': 'Laboratorio 2.5 prueba', 'type': 'iana-if-type:softwareLoopback', 'enabled': True, 'ietf-ip:ipv4': {'address': [{'ip': '99.99.99.99', 'netmask': '255.255.255.0'}]}, 'ietf-ip:ipv6': {}}]}}
```

Modificación de la configuración de la interfaz con RESTCONF en Python



The screenshot shows a Visual Studio Code window with a Python script named `lab 2.5par2.py` and its terminal output. The script uses the `requests` library to perform a PUT request to a RESTCONF endpoint, modifying the configuration of the `Loopback99` interface. The terminal output shows the successful execution of the script, displaying the status of the request.

```
1 import json
2 import requests
3 requests.packages.urllib3.disable_warnings()
4
5 api_url = "https://10.10.20.40/restconf/data/ietf-interfaces:interfaces/interface-Loopback99"
6
7 headers = { "Accept": "application/yang-data+json",
8             "Content-type": "application/yang-data+json"
9           }
10 basicauth = ("developer", "Cisco12345")
11
12 yangConfig = {
13     "ietf-interfaces:interface": {
14         "name": "Loopback99",
15         "description": "Laboratorio 2.5 prueba",
16         "type": "iana-if-type:softwareLoopback",
17         "enabled": True,
18         "ietf-ip:ipv4": {
19             "address": {
20                 "ip": "99.99.99.99",
21                 "netmask": "255.255.255.0"
22             }
23         }
24     }
25 }
```

```
PS C:\Users\foser> & C:\Users\foser\AppData\Local\Programs\Python\python310\python.exe "c:/Users/foser/Downloads/lab 2.5par2.py"
STATUS OK: 204
PS C:\Users\foser>
```

Los scripts se realizaron con éxito y lograron ejecutarse con éxito.

Conclusiones

¿Qué es RESCONF?

Es un protocolo definido por la IETF en el RFC 8040

¿Cuál es la manera de invocar los Scripts?

Invocación de un script dentro de un cmdlet

```
/bin/bash script_bash.sh
```

```
/bin/dash script_bash.sh
```

¿En qué casos utilizo RESCONF?

Utiliza al protocolo HTTP para transportar las operaciones que permite obtener los YANG