

Examen Parcial

Curso: Administración de Redes
Semestre: 2024-II
Ciencias de la Computación - UNI

Apellidos y Nombres: Pacheco Taboada André Joaquín

Código: 20222189G

Pregunta 1

1.1 ¿Cómo añadiría con el comando ip, el IP 192.168.1.50/20 a la interface eth0?

```
ip addr add 192.168.1.50/20 dev eth0
```

1.2 ¿Cómo deshabilitaría con el comando ip, la interface de red enp3s0?

```
ip link set enp3s0 down
```

1.3 El comando neighbour se utiliza para ver la dirección MAC de los dispositivos conectados a su sistema. ¿Cómo añadiría con el comando ip, la entrada ARP con el IP 192.168.1.50/24 a la interface enp3s0?

```
ip neigh add 192.168.1.50 lladdr aa:bb:cc:dd:ee:ff dev enp3s0
```

Nota: Reemplazar **aa:bb:cc:dd:ee:ff** con la dirección MAC real.

1.4 Si tenemos la ruta 192.168.4.0/20 que apunta a la interfaz enp3s0 ¿cómo la eliminamos?

```
ip route del 192.168.4.0/20 dev enp3s0
```

1.5 Explique qué resultado obtendrá con:

```
$ sudo nmcli connection add type ethernet ifname enp0s5
```

Este comando creará una nueva conexión Ethernet en NetworkManager:

- Asociada a la interface enp0s5
- Con un nombre automático (e.g., "Ethernet connection 1")
- Configurada para DHCP por defecto

- Guardada en la configuración de NetworkManager, pero no activada automáticamente

Se mostrará un mensaje de confirmación con detalles como el nombre de la conexión y su UUID.

Pregunta 2

En el archivo `ejercicio-1_examen-parcial_2024-2.ipynb`

Pregunta 3

3.1 Instalando FastAPI

El comando para instalar FastAPI y Uvicorn es:

```
pip install fastapi uvicorn
```

En mi caso, ya tengo las bibliotecas instaladas en un entorno de Conda.

3.2 Script

El código se encuentra en el archivo `main.py`. De todas maneras adjunto el código a continuación:

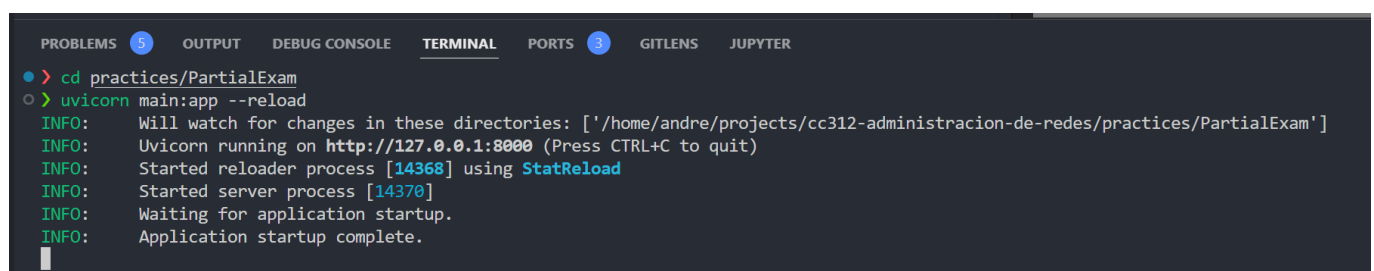
```
from typing import Union
from fastapi import FastAPI

app = FastAPI()

@app.get("/")
def read_root():
    return {"Hola": "CC 312"}

@app.get("/items/{item_id}")
def read_item(item_id: int, q: Union[str, None] = None):
    return {"item_id": item_id, "q": q}
```

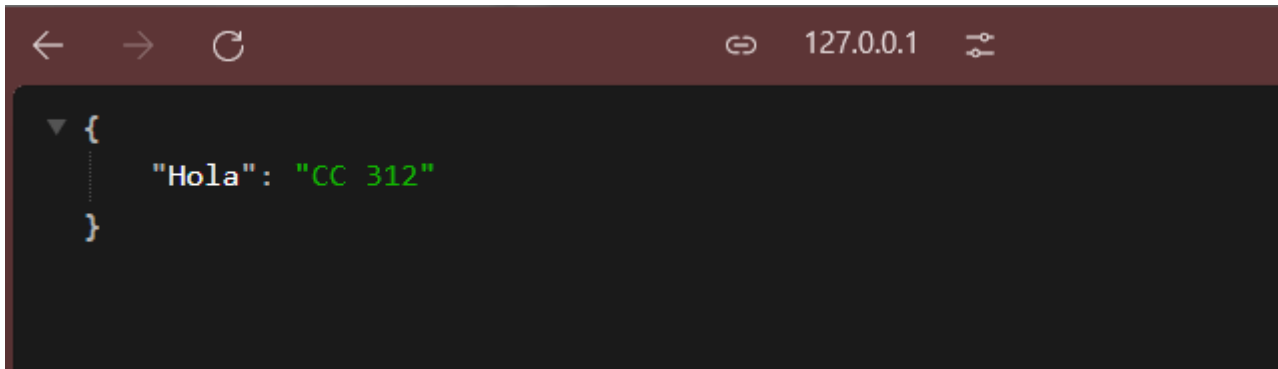
3.3 Ejecutando la aplicación



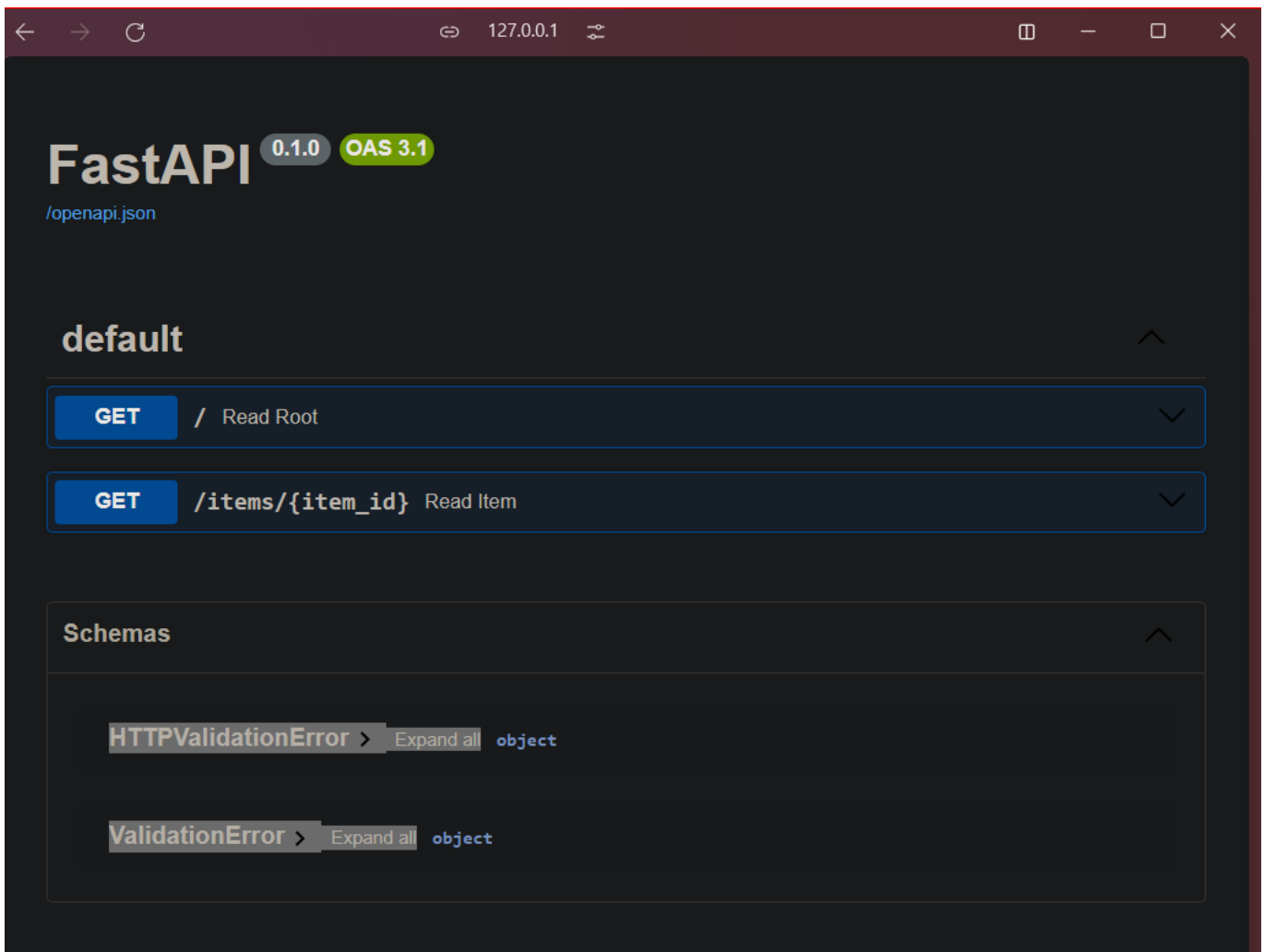
```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS 3 GITLENS JUPYTER
➤ cd practices/PartialExam
➤ uvicorn main:app --reload
INFO: Will watch for changes in these directories: ['/home/andre/projects/cc312-administracion-de-redes/practices/PartialExam']
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO: Started reloader process [14368] using StatReload
INFO: Started server process [14370]
INFO: Waiting for application startup.
INFO: Application startup complete.
```

3.4 Comprobando en el navegador

Por defecto, el servidor se ejecuta en el puerto 8000 y ejecuta el endpoint raíz (`read_root`).



3.5 Revisando la documentación



3.6 Realizando una petición a la API

127.0.0.1

GET /items/{item_id} Read Item

Parameters

Cancel

Name	Description
item_id * required integer (path)	<input type="text" value="1"/>
q string (query)	<input type="text" value="a"/>

Servers

These operation-level options override the global server options.

/

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'http://127.0.0.1:8000/items/1?q=a' \
  -H 'accept: application/json'
```

Request URL

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "item_id": 1, "q": "a" }</pre></div><div><div>Download</div></div></div> <div><div>Response headers</div><div><pre>content-length: 21</pre></div></div>