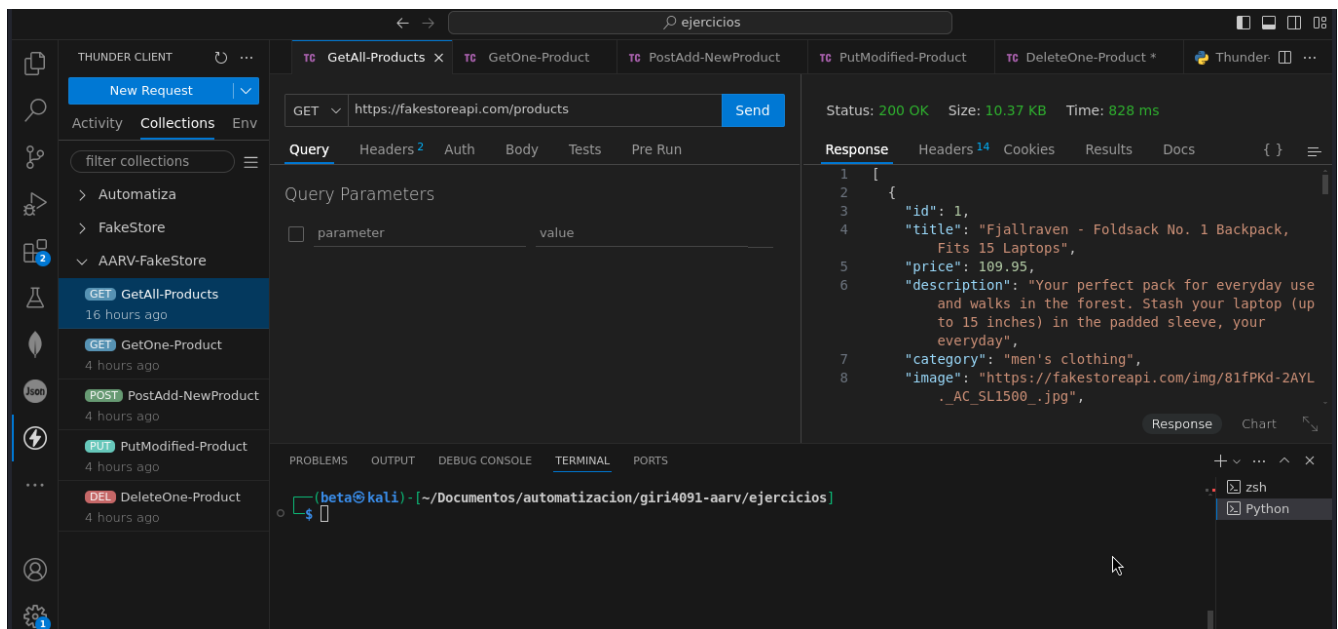
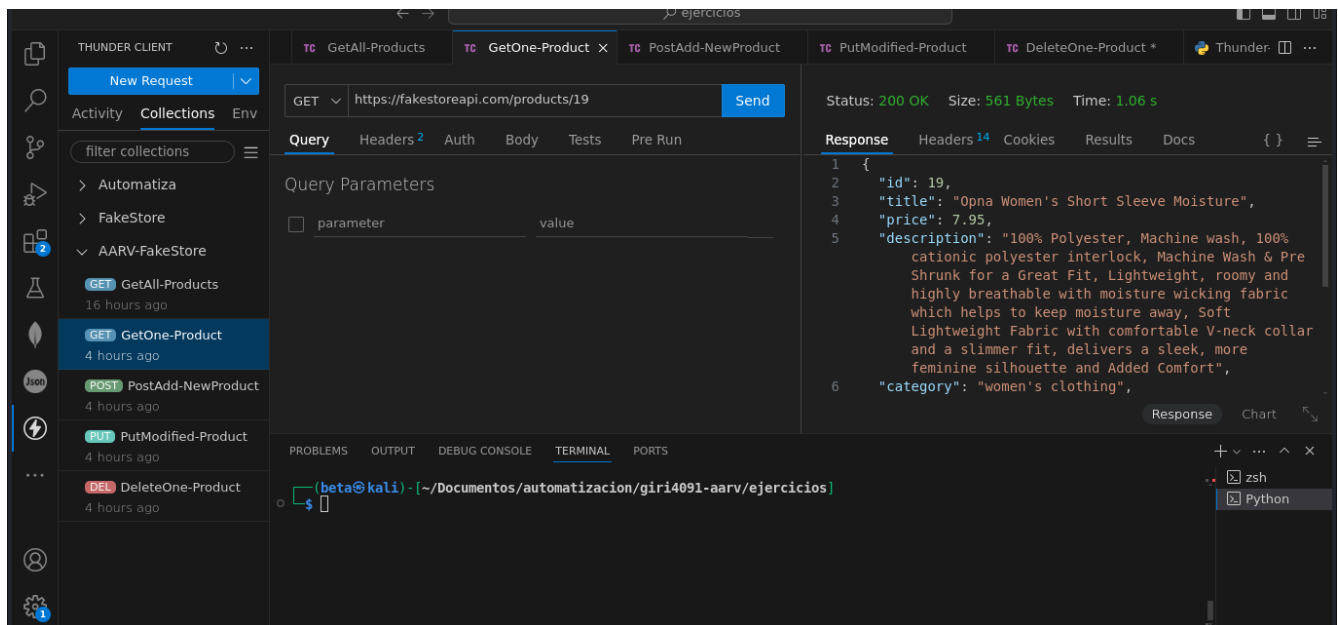


Práctica 01 Práctica con REST API

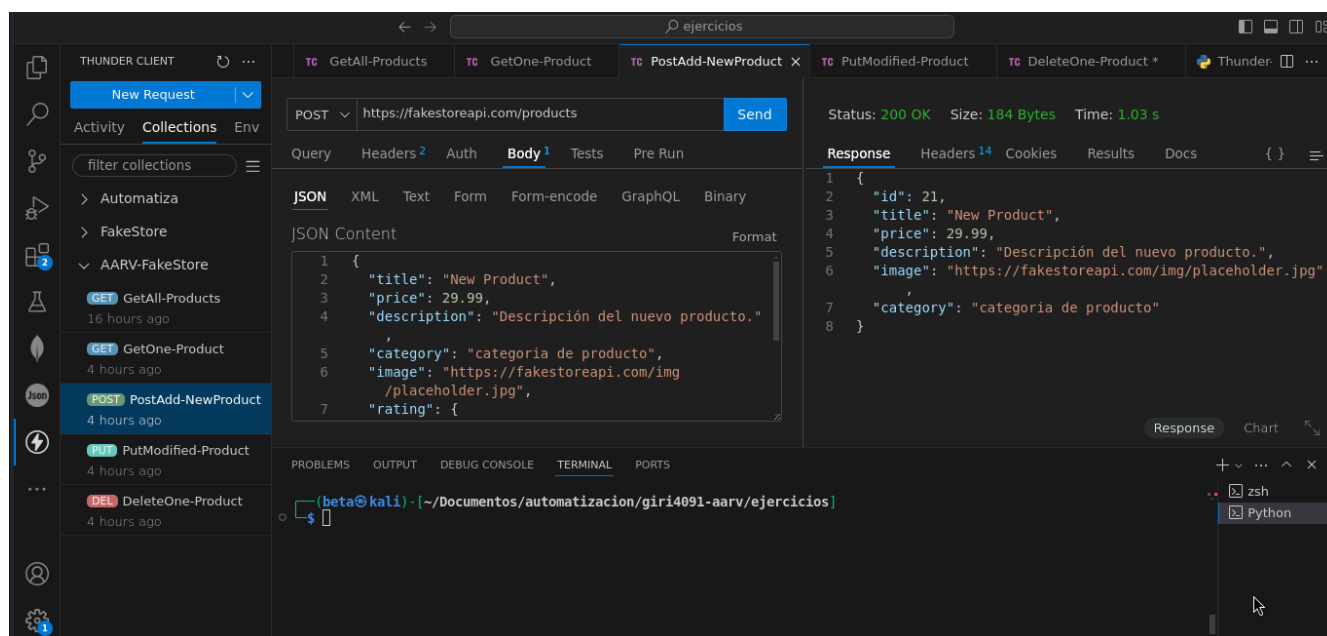
1.- Agregar una nueva petición llamada GetAll-Products con el método GET



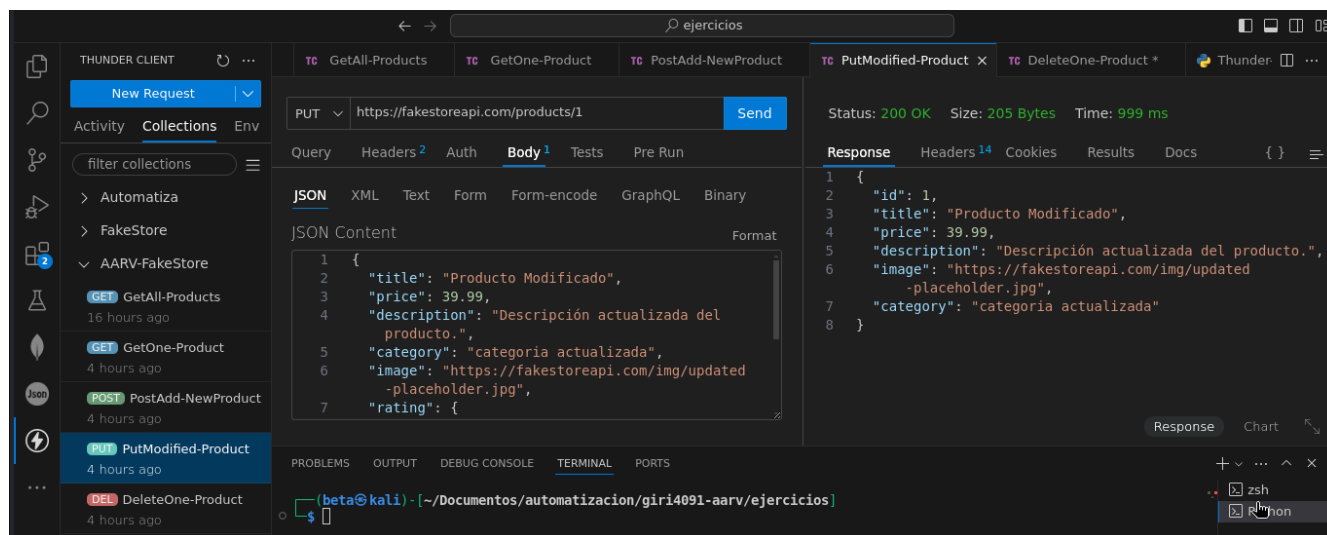
2.- Crear una nueva petición en la colección buscando un producto en particular.



3.- Crear una nueva petición para agregar un nuevo producto



4.- Modificando un nuevo producto



5.- Eliminando un Producto.

The screenshot displays the Thunder Client interface with a collection named 'ejercicios'. The 'DeleteOne-Product' request is selected, showing a DELETE method to the URL `https://fakestoreapi.com/products/1`. The response is a 200 OK status with a JSON body containing product details.

Request Details:

- Method: DELETE
- URL: `https://fakestoreapi.com/products/1`
- Body: 1

Response Details:

- Status: 200 OK
- Size: 364 Bytes
- Time: 853 ms
- Response Body (JSON):

```
{  "id": 1,  "title": "Fjallraven - Foldsack No. 1 Backpack, Fits 15 Laptops",  "price": 109.95,  "description": "Your perfect pack for everyday use and walks in the forest. Stash your laptop (up to 15 inches) in the padded sleeve, your everyday",  "category": "men's clothing",  "image": "https://fakestoreapi.com/img/81fPKd-2AYL . AC_SL1500_.jpg",  "rating": {    "rate": 3.9,  }}
```

Terminal:

```
(beta@kali) - [~/Documentos/automatizacion/giri4091-aarv/ejercicios]  
$
```

6.- Programa python de api requests.

```
import json
```

```
import requests
```

```
print("\nAdministración de Productos:")
print("1. Consultar todos los productos")
print("2. Consultar un producto en específico")
print("3. Agregar un nuevo producto")
print("4. Modificar producto en específico")
print("5. Eliminar un producto")
print("6. Salir")
```

```
def GetAllProducts():
```

```
url = "https://fakestoreapi.com/products"
```

```
response = requests.get(url)
```

```
json_formateado = json.dumps(response.json(), indent=4, ensure_ascii=False)
```

```
print("\n" + "Listado de productos: " + "\n")
```

```
print(json_formateado)
```

```
def GetProduct():
```

```
noProduct = (input("Ingresa el valor del número del producto: "))
```

```
url = "https://fakestoreapi.com/products/" + noProduct
```

```
print(url)
```

```
response = requests.get(url)
```

```
json_formateado = json.dumps(response.json(), indent=4, ensure_ascii=False)
```

```
print("\n" + "Listado de productos" + "\n")
```

```
print(json_formateado)
```

```
if response.status_code == 200:
```

```
print("\n" + "Producto consultado exitosamente :D")
```

```
print(response.json())
```

```
else:
```

```
print("\n" + "Error al consultar el producto :(")
```

```
print(response.status_code)
```

```
print(response.text)
```

```
def AddProduct():
```

```
print("\n" + "Agregar producto" + "\n")
```

```
url = "https://fakestoreapi.com/products"
```

```
titleProduct = (input("Ingresa el título del producto: " + "\n"))
```

```
priceProduct = (input("Ingresa el precio del producto: " + "\n"))
```

```
descriptionProduct = (input("Ingresa la descripción del producto: " + "\n"))
```

```
categoryProduct = (input("Ingresa la categoría del producto: " + "\n"))
```

```
payload = {
```

```
"title": " " + titleProduct,
```

```
"price": " " + priceProduct,
```

```
"description": " "+descriptionProduct,
"category": " "+categoryProduct,
"image": "https://fakestoreapi.com/img/placeholder.jpg",
"rating": {
"rate": 4.5,
"count": 10
}
}
```

```
headers = {
"Content-Type": "application/json"
}
```

```
response = requests.post(url, json=payload, headers=headers)
```

```
if response.status_code == 200:
print("\n" + "Producto creado exitosamente :D")
print(response.json())
else:
print("\n" + "Error al crear el producto ;(")
print(response.status_code)
print(response.text)
```

```
def UpdateProduct():
noProduct = (input("\n" + "Ingrese el número de producto a cambiar: " + "\n"))
url = "https://fakestoreapi.com/products/" + noProduct
```

```
url = "https://fakestoreapi.com/products"
titleProduct = (input("Ingresa el título del producto: " + "\n"))
priceProduct = (input("Ingresa el precio del producto: " + "\n"))
descriptionProduct = (input("Ingresa la descripción del producto: " + "\n"))
categoryProduct = (input("Ingresa la categoría del producto: " + "\n"))
```

```
payload = {
"title": " "+titleProduct,
"price": " "+priceProduct,
"description": " "+descriptionProduct,
"category": " "+categoryProduct,
"image": "https://fakestoreapi.com/img/placeholder.jpg",
"rating": {
"rate": 4.5,
"count": 10
}
}
```

```
headers = {
"Content-Type": "application/json"
```

```
}
```

```
response = requests.put(url, json=payload, headers=headers)
```

```
if response.status_code == 200:
```

```
print("Producto actualizado exitosamente.")
```

```
print(response.json())
```

```
else:
```

```
print("Error al actualizar el producto.")
```

```
print(response.status_code)
```

```
print(response.text)
```

```
print("Modificar producto")
```

```
def DeleteProduct():
```

```
noProduct = (input("Ingresa el valor del número del producto a eliminar: "))
```

```
url = "https://fakestoreapi.com/products/" + noProduct
```

```
response = requests.delete(url)
```

```
if response.status_code == 200:
```

```
print("Producto eliminado exitosamente :D")
```

```
print(response.json())
```

```
else:
```

```
print("Error al eliminar el producto ;(())")
```

```
print(response.status_code)
```

```
print(response.text)
```

```
while True:
```

```
opcion = input("Selecciona una opción (1-5): ")
```

```
if opcion == '1':
```

```
GetAllProducts()
```

```
elif opcion == '2':
```

```
GetProduct()
```

```
elif opcion == '3':
```

```
AddProduct()
```

```
elif opcion == '4':
```

```
AddProduct()
```

```
elif opcion == '5':
```

```
DeleteProduct()
```

```
elif opcion == '6':
```

```
print("Saliendo del programa...")
```

```
break
```

```
else:
```

```
print("Opción no válida, por favor intenta de nuevo.")
```