

202010406 – Dennis Mauricio Corado Muñóz

Petición GET:

The screenshot displays two windows. On the left is Visual Studio Code with a Python file named `app.py`. The code defines a Flask application with a `metodo_get` endpoint. The terminal shows the command `flask run` and the output indicating the server is running on `http://127.0.0.1:5000/`. A GET request to `/metodo_get` is shown in the terminal output, returning a 200 status.

```
app.py
1 from flask import Flask, request
2 from flask_cors import CORS
3
4 app = Flask(__name__)
5 CORS(app)
6 api = Api(app)
7
8 # GET, POST, PUT y DELETE
9
10 @app.route('/metodo_get', methods=['GET'])
11 def metodo_get():
12     print("Usando metodo GET.")
13     return {"message": "Petición hecha con Get."}
14
15
```

On the right is Postman, showing a GET request to `http://127.0.0.1:5000/metodo_get`. The response is a JSON object: `{"message": "Petición hecha con Get."}`.

Petición POST:

The screenshot displays two windows. On the left is Visual Studio Code with the same `app.py` file. The code now includes a `metodo_post` endpoint. The terminal shows the command `flask run` and the output indicating the server is running on `http://127.0.0.1:5000/`. A POST request to `/metodo_post` is shown in the terminal output, returning a 200 status.

```
app.py
1 from flask import Flask, request
2 from flask_cors import CORS
3
4 app = Flask(__name__)
5 CORS(app)
6 api = Api(app)
7
8 # GET, POST, PUT y DELETE
9
10 @app.route('/metodo_get', methods=['GET'])
11 def metodo_get():
12     print("Usando metodo GET.")
13     return {"message": "Petición hecha con Get."}
14
15 @app.route('/metodo_post', methods=['POST'])
16 def metodo_post():
17     contenido = request.get_json()
18     print(contenido['message'])
19     return {"message": "Petición hecha con POST."}
20
21 @app.route('/metodo_delete', methods=['DELETE'])
22 def metodo_delete():
23     print("Usando metodo DELETE.")
24
```

On the right is Postman, showing a POST request to `http://127.0.0.1:5000/metodo_post`. The request body is a JSON object: `{"message": "Esta es una petición con POST"}`. The response is a JSON object: `{"message": "Petición hecha con POST."}`.

Petición DELETE:

The screenshot shows a Python Flask application in Visual Studio Code and a corresponding DELETE request in Postman.

Visual Studio Code:

- File Explorer:** Shows the project structure with files like `app.py`, `__pycache__`, and `app.py`.
- Code Editor:** Displays the `app.py` file with the following code:

```
1 print(contenido['message'])
2 return {"message": "Petición hecha con POST."}
3
4 @app.route("/metodo_delete", methods=['DELETE'])
5 def metodo_delete():
6     print("Usando metodo DELETE.")
7     return {"message": "Petición hecha con DELETE."}
8
9 @app.route("/metodo_put", methods=['PUT'])
10 def metodo_put():
11     contenido = request.get_data()
12     print("Usando metodo PUT, parametro: " + contenido)
13     return {"message": "Petición hecha con PUT."}
14
```
- Terminal:** Shows the command `python app.py` and the output: `127.0.0.1 - - [04/Nov/2021 12:57:27] "DELETE /metodo_delete HTTP/1.1" 200 -`

Postman:

- Request:** Method `DELETE`, URL `http://127.0.0.1:5000/metodo_delete`.
- Response:** Status `200`, Body `{"message": "Petición hecha con DELETE."}`.

Petición PUT:

The screenshot shows a Python Flask application in Visual Studio Code and a corresponding PUT request in Postman.

Visual Studio Code:

- File Explorer:** Shows the project structure with files like `app.py`, `__pycache__`, and `app.py`.
- Code Editor:** Displays the `app.py` file with the following code:

```
1 print(contenido['message'])
2 return {"message": "Petición hecha con DELETE."}
3
4 @app.route("/metodo_delete", methods=['DELETE'])
5 def metodo_delete():
6     print("Usando metodo DELETE.")
7     return {"message": "Petición hecha con DELETE."}
8
9 @app.route("/metodo_put", methods=['PUT'])
10 def metodo_put():
11     contenido = request.get_data()
12     print("Usando metodo PUT, parametro: " + str(contenido))
13     return {"message": "Petición hecha con PUT."}
14
15 if __name__ == '__main__':
16     app.run()
```
- Terminal:** Shows the command `python app.py` and the output: `127.0.0.1 - - [04/Nov/2021 13:01:17] "PUT /metodo_put HTTP/1.1" 200 -`

Postman:

- Request:** Method `PUT`, URL `http://127.0.0.1:5000/metodo_put`, Body `parametro`.
- Response:** Status `200`, Body `{"message": "Petición hecha con PUT."}`.