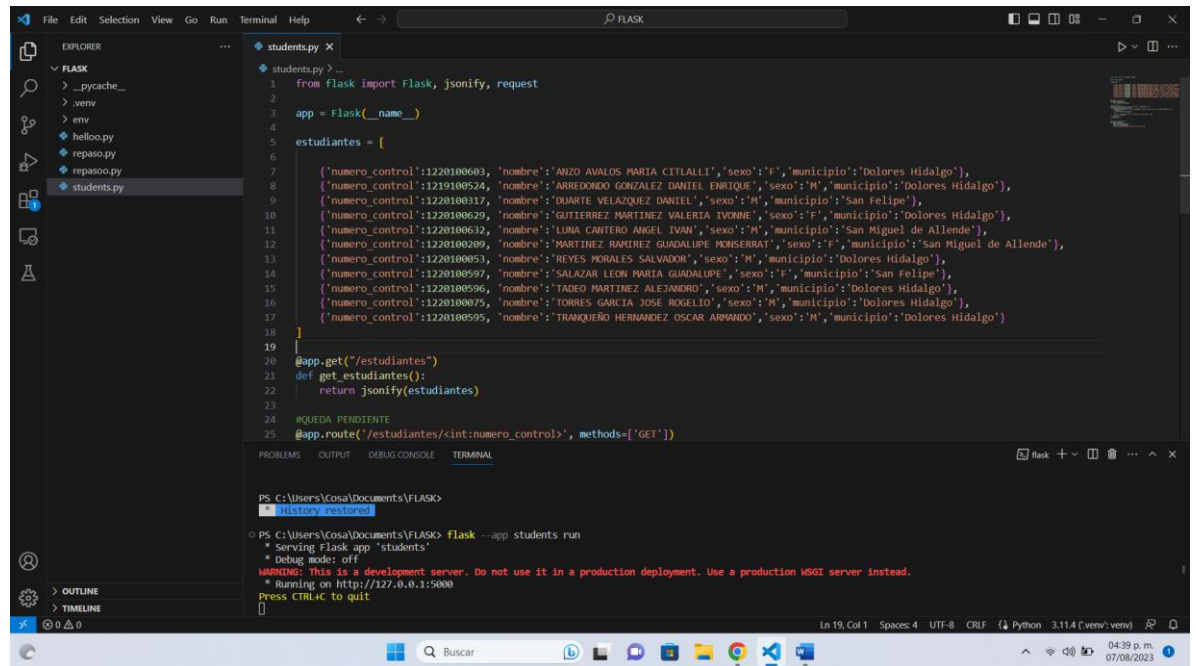


Práctica del ejercicio 4

- Agregar al menos dos datos más al estudiante, por ejemplo: edad, carrera, etc

GET



```
1 from flask import Flask, jsonify, request
2
3 app = Flask(__name__)
4
5 estudiantes = [
6
7     {'numero_control': '1220100603', 'nombre': 'ANZO AVALOS MARIA CITLALLI', 'sexo': 'F', 'municipio': 'Dolores Hidalgo'},
8     {'numero_control': '1219100524', 'nombre': 'ARREDONDO GONZALEZ DANIEL ENRIQUE', 'sexo': 'M', 'municipio': 'Dolores Hidalgo'},
9     {'numero_control': '1220100317', 'nombre': 'DUARTE VELAZQUEZ DANIEL', 'sexo': 'M', 'municipio': 'San Felipe'},
10    {'numero_control': '1220100629', 'nombre': 'GUTIERREZ MARTINEZ VALERIA IVONNE', 'sexo': 'F', 'municipio': 'Dolores Hidalgo'},
11    {'numero_control': '1220100632', 'nombre': 'LUNA CANTERO ANGEL IVAN', 'sexo': 'M', 'municipio': 'San Miguel de Allende'},
12    {'numero_control': '1220100509', 'nombre': 'MARTINEZ RAMIREZ GUADALUPE MONSERRAT', 'sexo': 'F', 'municipio': 'San Miguel de Allende'},
13    {'numero_control': '1220100653', 'nombre': 'REVES MORALES SALVADOR', 'sexo': 'M', 'municipio': 'Dolores Hidalgo'},
14    {'numero_control': '1220100597', 'nombre': 'SALAZAR LEON MARIA GUADALUPE', 'sexo': 'F', 'municipio': 'San Felipe'},
15    {'numero_control': '1220100596', 'nombre': 'TADEO MARTINEZ ALEJANDRO', 'sexo': 'M', 'municipio': 'Dolores Hidalgo'},
16    {'numero_control': '1220100075', 'nombre': 'TORRES GARCIA JOSE ROGELIO', 'sexo': 'M', 'municipio': 'Dolores Hidalgo'},
17    {'numero_control': '1220100595', 'nombre': 'TRANQUERO HERNANDEZ OSCAR ARMANDO', 'sexo': 'M', 'municipio': 'Dolores Hidalgo'}
18 ]
19
20 @app.get("/estudiantes")
21 def get_estudiantes():
22     return jsonify(estudiantes)
23
24 @app.route('/estudiantes/<int:numero_control>', methods=['GET'])
25
```

PS C:\Users\Cosa\Documents\FLASK> flask --app students run

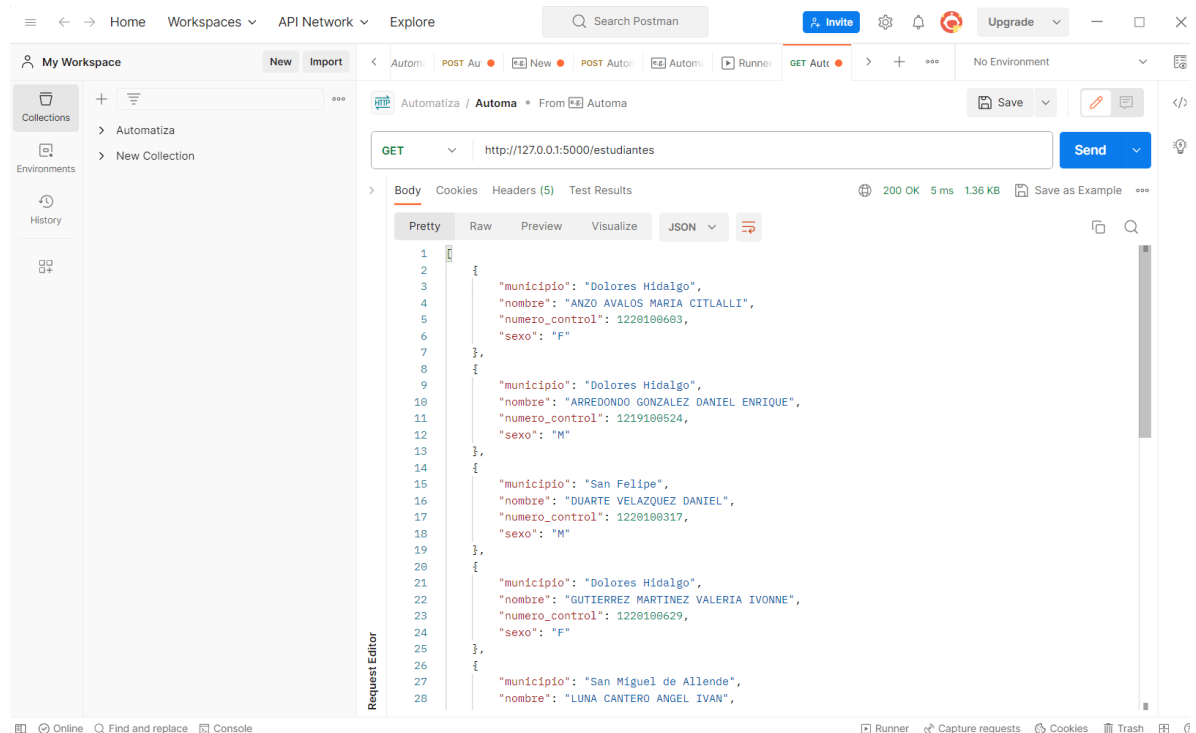
* Serving Flask app 'students'

* Debug mode: off

WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.

* Running on http://127.0.0.1:5000

Press CTRL+C to quit

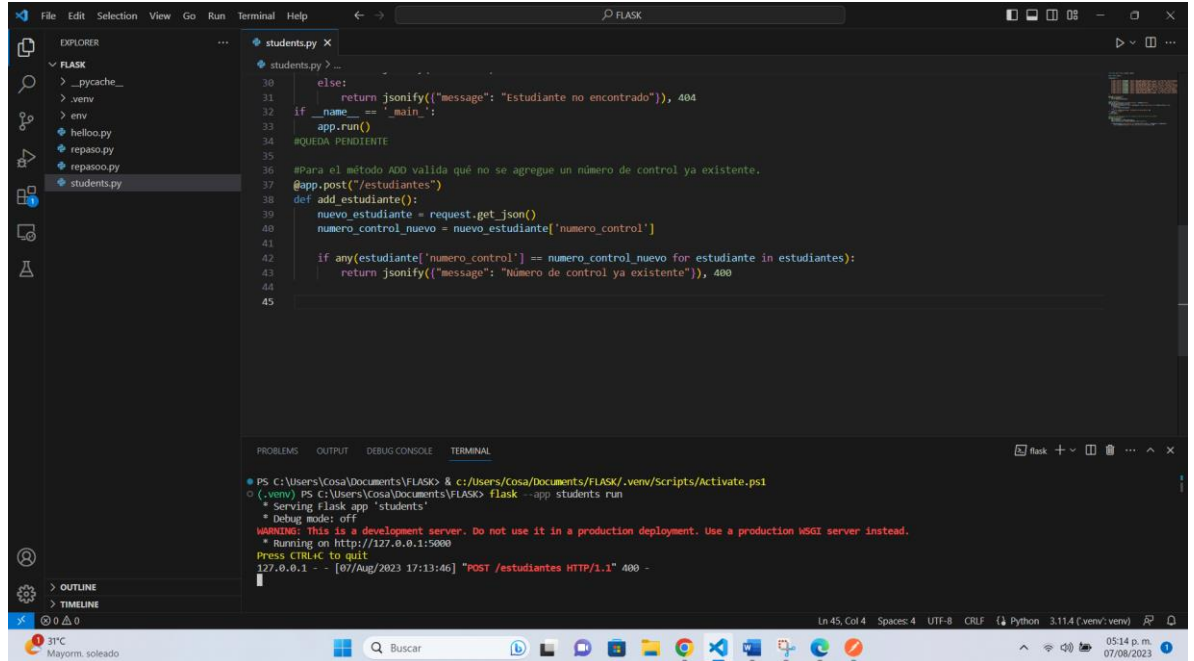


GET http://127.0.0.1:5000/estudiantes

200 OK 5 ms 1.36 KB

```
1 {
2   "municipio": "Dolores Hidalgo",
3   "nombre": "ANZO AVALOS MARIA CITLALLI",
4   "numero_control": 1220100603,
5   "sexo": "F"
6 },
7 {
8   "municipio": "Dolores Hidalgo",
9   "nombre": "ARREDONDO GONZALEZ DANIEL ENRIQUE",
10  "numero_control": 1219100524,
11  "sexo": "M"
12 },
13 {
14   "municipio": "San Felipe",
15   "nombre": "DUARTE VELAZQUEZ DANIEL",
16   "numero_control": 1220100317,
17   "sexo": "M"
18 },
19 {
20   "municipio": "Dolores Hidalgo",
21   "nombre": "GUTIERREZ MARTINEZ VALERIA IVONNE",
22   "numero_control": 1220100629,
23   "sexo": "F"
24 },
25 {
26   "municipio": "San Miguel de Allende",
27   "nombre": "LUNA CANTERO ANGEL IVAN",
28
```

- Para el método **ADD** valida qué no se agregue un número de control ya existente.
POST

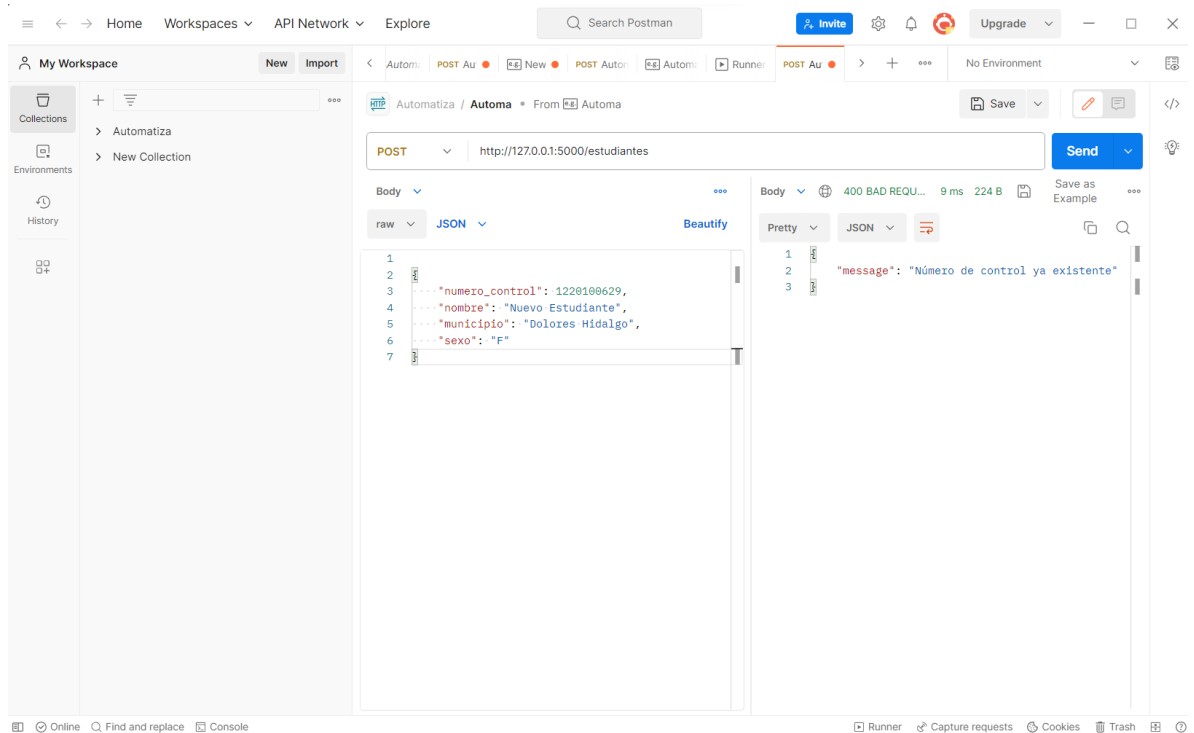


The screenshot shows a VS Code editor with a Python Flask application. The file explorer on the left shows a project named 'FLASK' with files like 'hello.py', 'repaso.py', and 'students.py'. The main editor shows the 'students.py' file with the following code:

```
30 else:
31     return jsonify({"message": "Estudiante no encontrado"}), 404
32 if __name__ == '__main__':
33     app.run()
34 #QUEDA PENDIENTE
35
36 #Para el método ADD valida qué no se agregue un número de control ya existente.
37 @app.post("/estudiantes")
38 def add_estudiante():
39     nuevo_estudiante = request.get_json()
40     numero_control_nuevo = nuevo_estudiante['numero_control']
41
42     if any(estudiante['numero_control'] == numero_control_nuevo for estudiante in estudiantes):
43         return jsonify({"message": "Número de control ya existente"}), 400
44
45
```

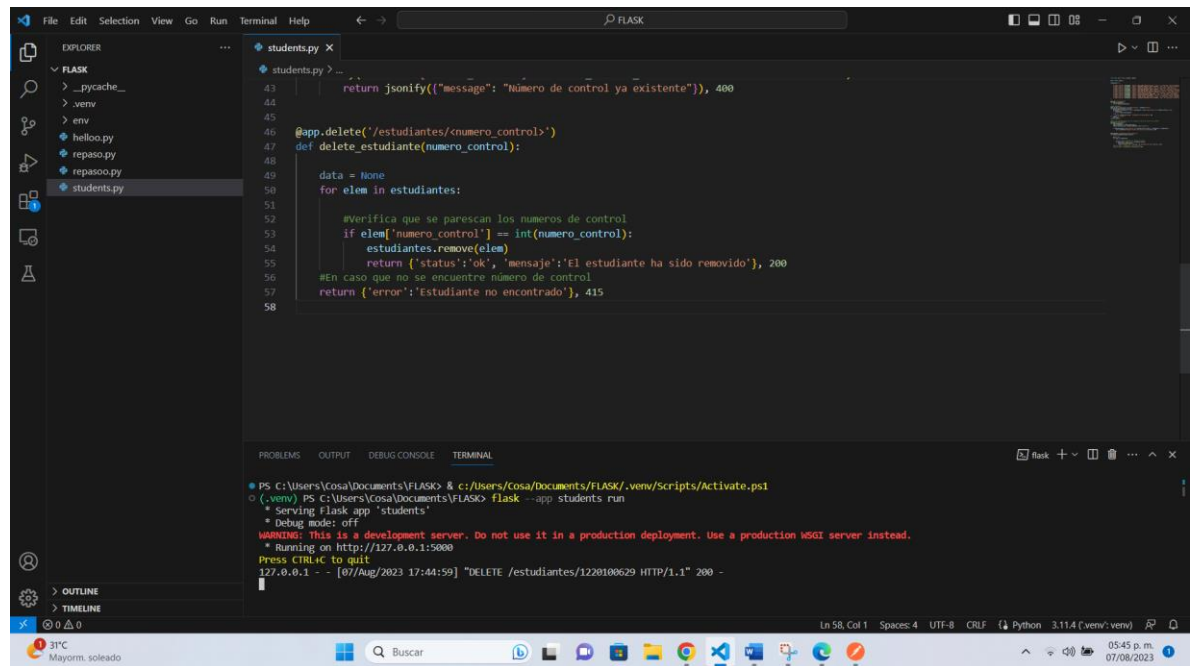
The terminal at the bottom shows the command to run the Flask application and the output:

```
PS C:\Users\Cosa\Documents\FLASK> .\venv\Scripts\Activate.ps1
(.venv) PS C:\Users\Cosa\Documents\FLASK> flask --app students run
 * Serving Flask app "students"
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [07/Aug/2023 17:13:46] "POST /estudiantes HTTP/1.1" 400 -
```



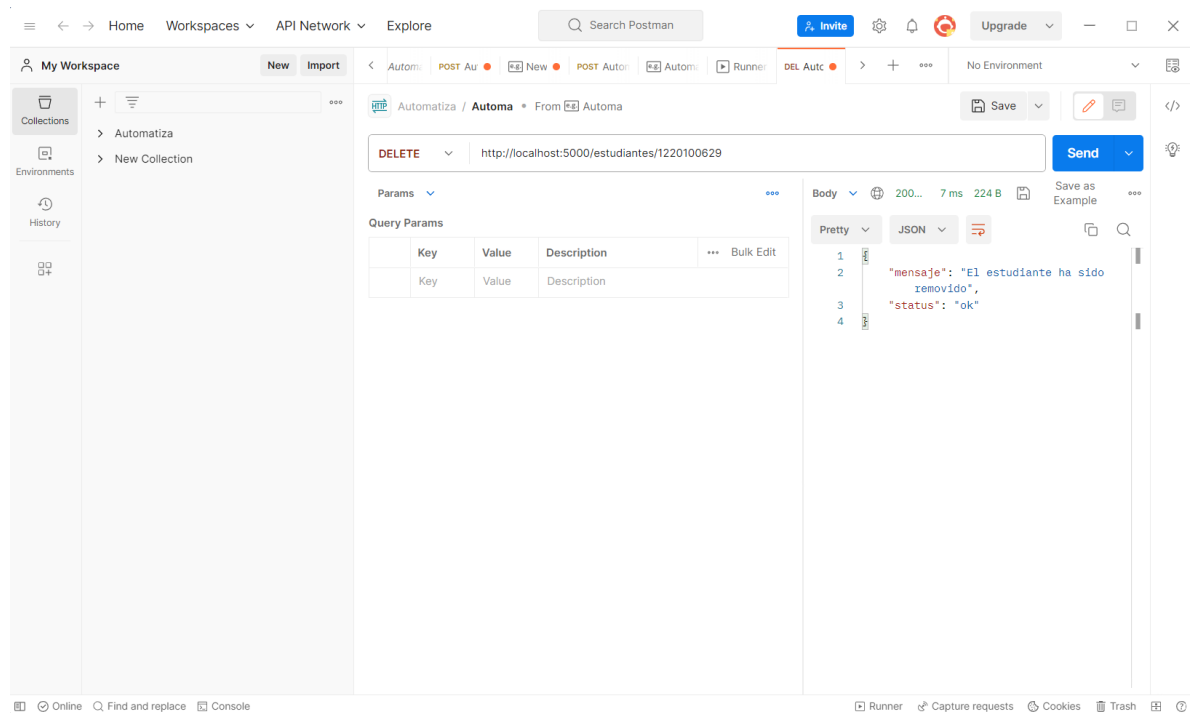
- Modificar todos los métodos para que los nuevos campos tengan efecto.

DELETE

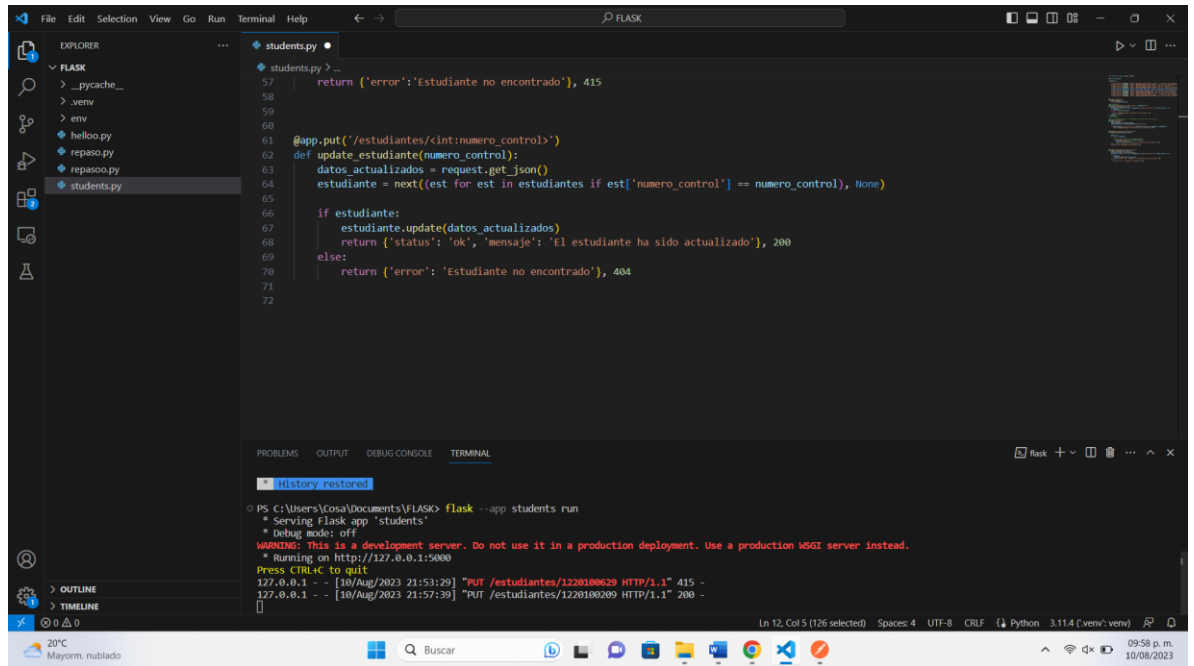


```
students.py
43     return jsonify({"message": "Número de control ya existente"}), 400
44
45
46 @app.delete('/estudiantes/<numero_control>')
47 def delete_estudiante(numero_control):
48
49     data = None
50     for elem in estudiantes:
51
52         #Verifica que se parezcan los numeros de control
53         if elem['numero_control'] == int(numero_control):
54             estudiantes.remove(elem)
55             return ('status':'ok', 'mensaje':'El estudiante ha sido removido'), 200
56         #En caso que no se encuentre número de control
57         return ('error':'Estudiante no encontrado'), 415
58
```

```
PS C:\Users\Cosa\Documents\FLASK> & c:\Users\Cosa\Documents\FLASK\.venv\Scripts\activate.ps1
(.venv) PS C:\Users\Cosa\Documents\FLASK> flask --app students run
 * Serving Flask app "students"
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [07/Aug/2023 17:44:59] "DELETE /estudiantes/1220100629 HTTP/1.1" 200 -
```



PUT



The screenshot shows a VS Code editor with a Flask application. The file explorer on the left shows a project named 'FLASK' with files like 'hello.py', 'repaso.py', and 'students.py'. The main editor shows the 'students.py' file with the following code:

```
57 | return ('error': 'Estudiante no encontrado'), 415
58 |
59 |
60 |
61 | @app.put('/estudiantes/<int:numero_control>')
62 | def update_estudiante(numero_control):
63 |     datos_actualizados = request.get_json()
64 |     estudiante = next((est for est in estudiantes if est['numero_control'] == numero_control), None)
65 |
66 |     if estudiante:
67 |         estudiante.update(datos_actualizados)
68 |         return {'status': 'ok', 'mensaje': 'El estudiante ha sido actualizado'}, 200
69 |     else:
70 |         return ('error': 'Estudiante no encontrado'), 404
71 |
72 |
```

The terminal at the bottom shows the command 'flask --app students run' and the output:

```
PS C:\Users\Cosa\Documents\FLASK> flask --app students run
* Serving Flask app "students"
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [10/Aug/2023 21:53:29] "PUT /estudiantes/1220100209 HTTP/1.1" 415 -
127.0.0.1 - - [10/Aug/2023 21:57:39] "PUT /estudiantes/1220100209 HTTP/1.1" 200 -
```

