

Instruction to Run the Application

- ❖ Access the running web application via Hosted Link

<https://flask-pymongo.anshkushwaha.repl.co>

OR

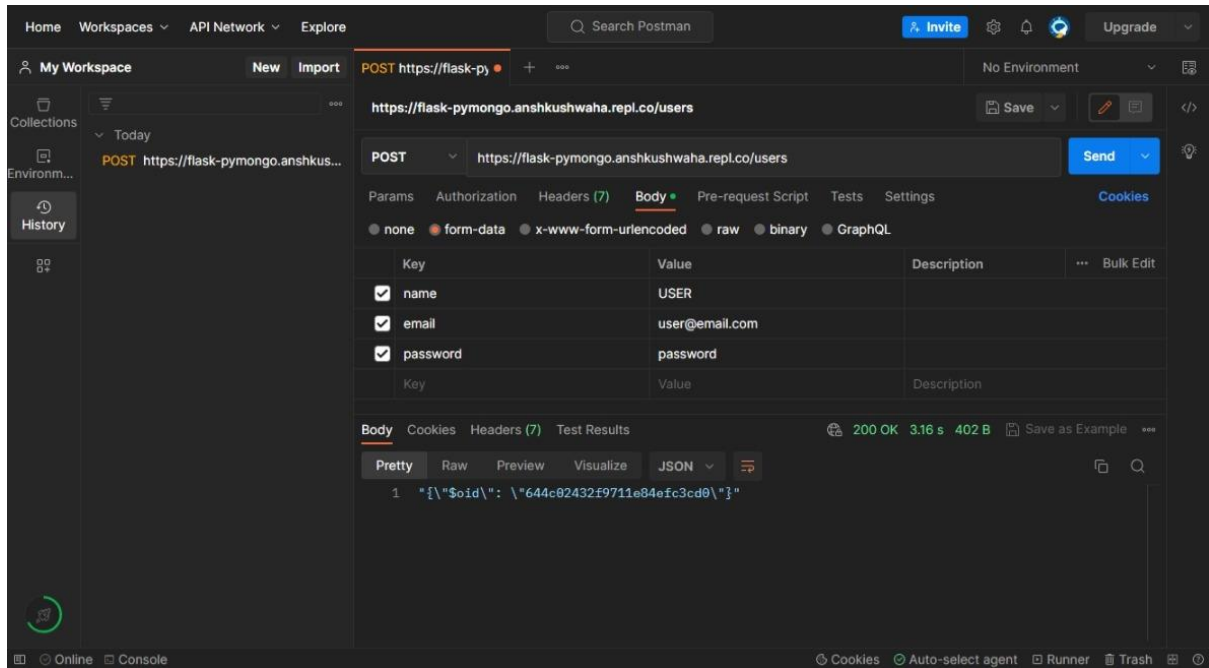
- ❖ Run on your Local System:
 - Download the Code & Extract it
 - Open the parent folder of “main.py” file in a Linux Terminal
 - Create a Virtual Environment using command:
`python -m venv ./`
 - Activate Virtual Environment using command:
`source ./bin/activate`
 - Install required packages using command:
`pip install -r requirements.txt`
 - Run the Application using command:
`python main.py`
 - Access the running web application of any JavaScript enabled browser at:
<https://127.0.0.1/5000>
 - To stop the running application, press keys:
`Ctrl + C`
 - To shut down the Virtual Environment, use command:
`deactivate`

API Endpoints

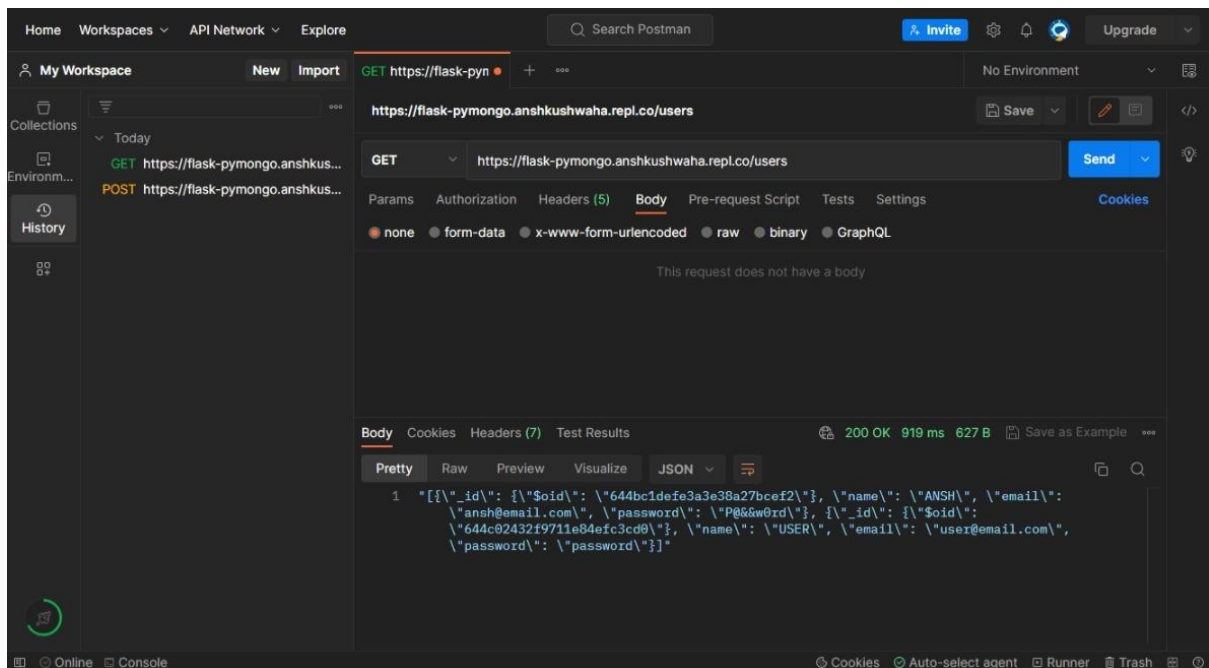
Endpoint	Method	Body	Body-Type
/users	POST	required	form-data
/users	GET	-	-
/users/<id>	GET	-	-
/users/<id>	PUT	required	form-data
/users/<id>	DELETE	-	-

API Testing (using Postman)

POST (/users)



GET (/users) ALL



GET (/users/<id>)

The screenshot shows a GET request in Postman to the endpoint `https://flask-pymongo.anshkushwaha.repl.co/users/644c02432f9711e84efc3cd0`. The request is successful, returning a 200 OK status with a response time of 1045 ms and a body size of 493 B. The response body is a JSON object:

```
1 { "_id": {"$oid": "644c02432f9711e84efc3cd0"}, "name": "USER", "email": "user@email.com", "password": "password" }
```

PUT (/users/<id>)

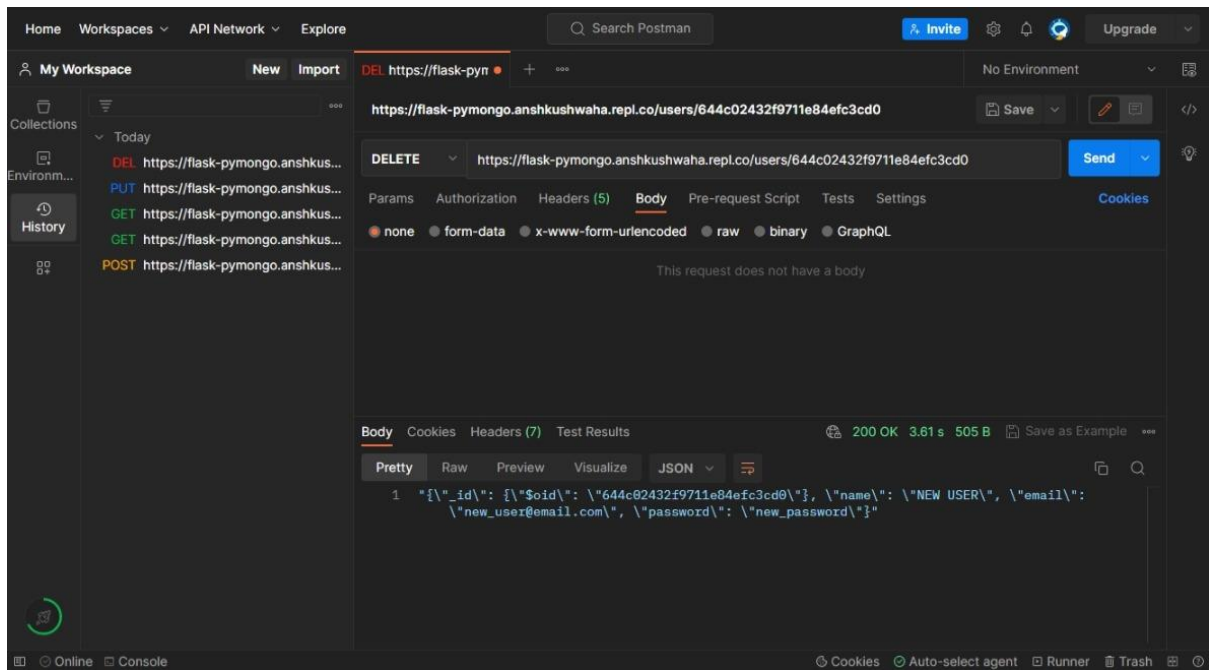
The screenshot shows a PUT request in Postman to the endpoint `https://flask-pymongo.anshkushwaha.repl.co/users/644c02432f9711e84efc3cd0`. The request is successful, returning a 200 OK status with a response time of 1951 ms and a body size of 493 B. The request body is form data with the following fields:

Key	Value	Description
<input checked="" type="checkbox"/> name	NEW USER	
<input checked="" type="checkbox"/> email	new_user@email.com	
<input checked="" type="checkbox"/> password	new_password	

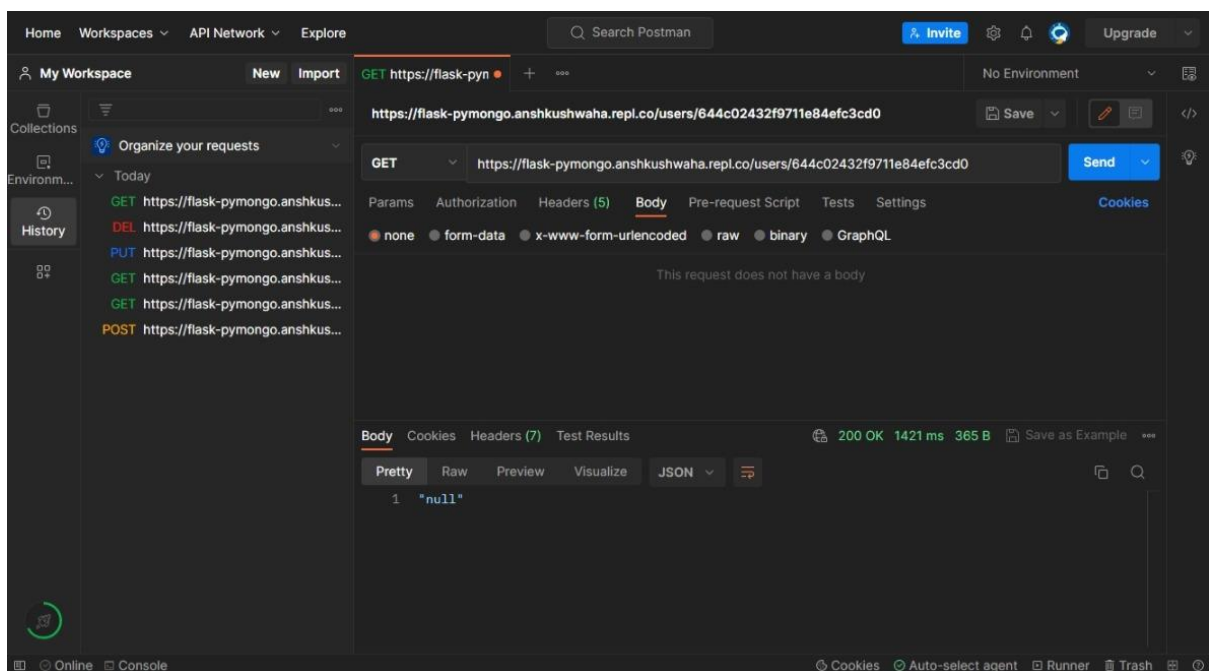
The response body is a JSON object:

```
1 { "_id": {"$oid": "644c02432f9711e84efc3cd0"}, "name": "USER", "email": "user@email.com", "password": "password" }
```

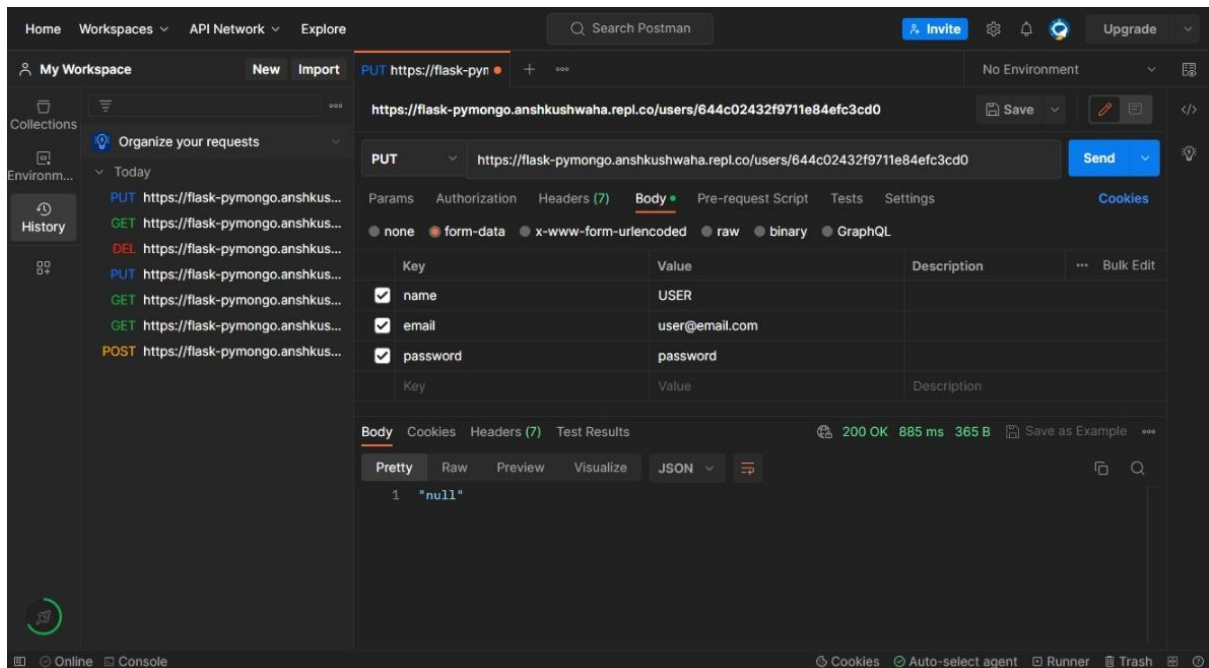
DELETE (/users/<id>)



GET on deleted “id”



PUT on deleted “id”



DELETE on deleted “id”

