Icon

Description automatically generated

**Lab Submission 4**

[**REST API LAB**](https://ashesi.instructure.com/courses/1752/assignments/26356/submissions/2126)

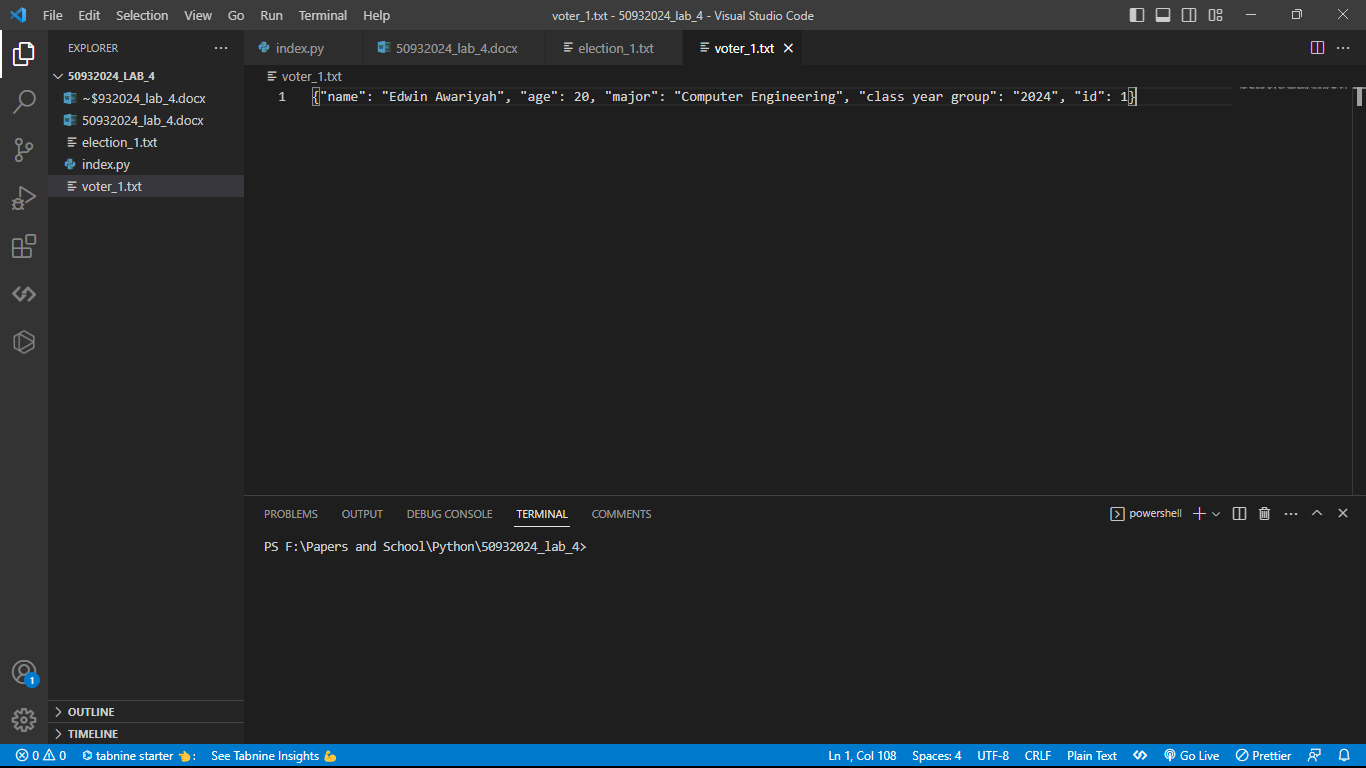
**DEPARTMENT OF COMPUTER SCIENCE**

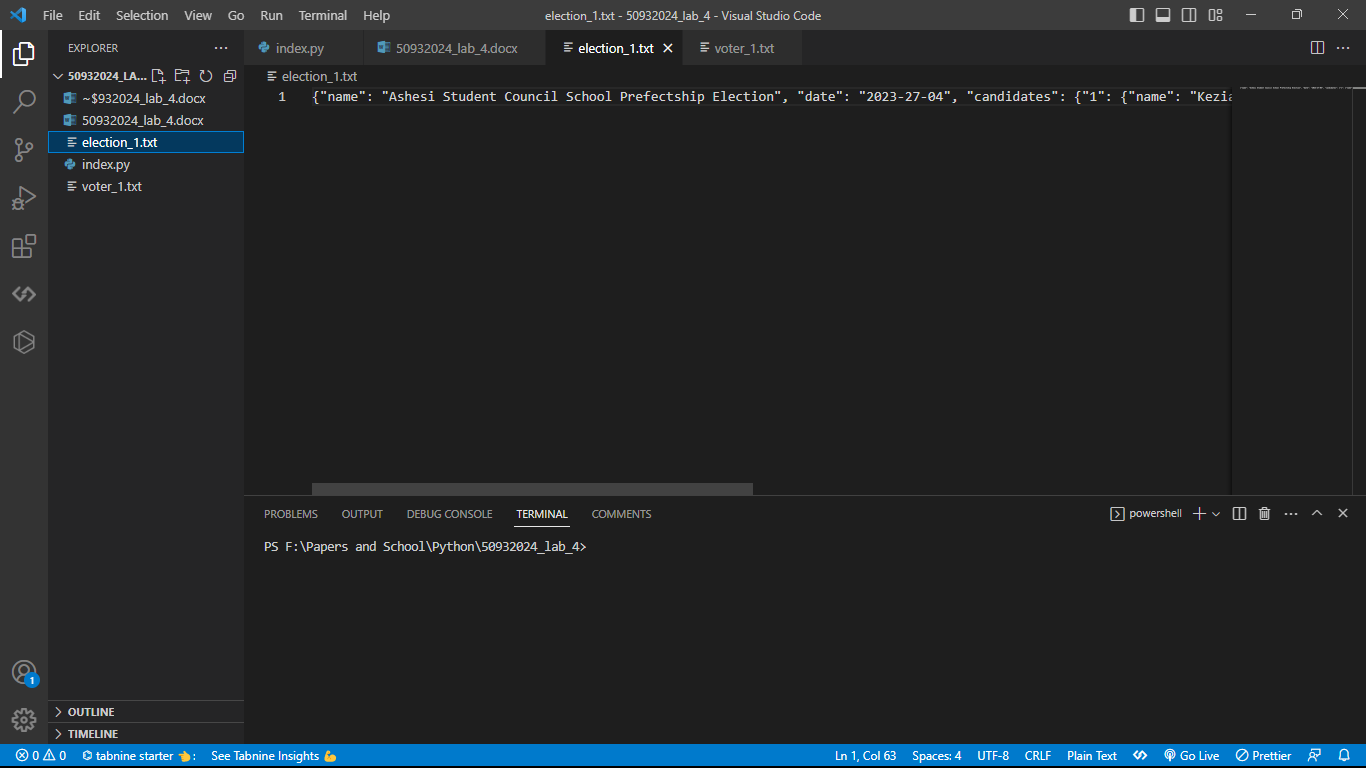
**Section B, CS - Dennis Owusu Asamoah, Akwasi Asante-Krobea**

**DUE DATE: 4th April 2023**

**GitHub link:**

JSON format text files for reference





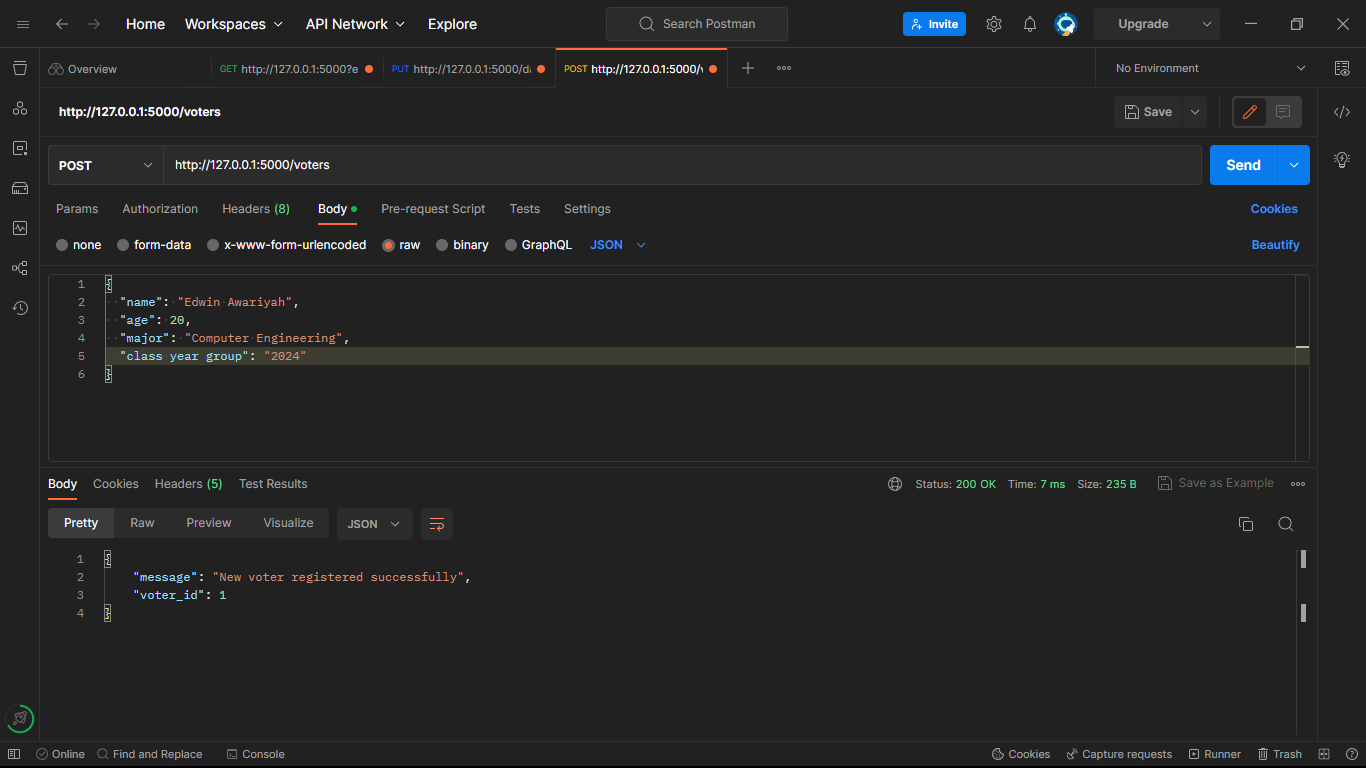
1. Registering a student as a voter.  
 a. It will be necessary for new students to be registered to vote.

- from my code if you are not registered you cannot vote

Request

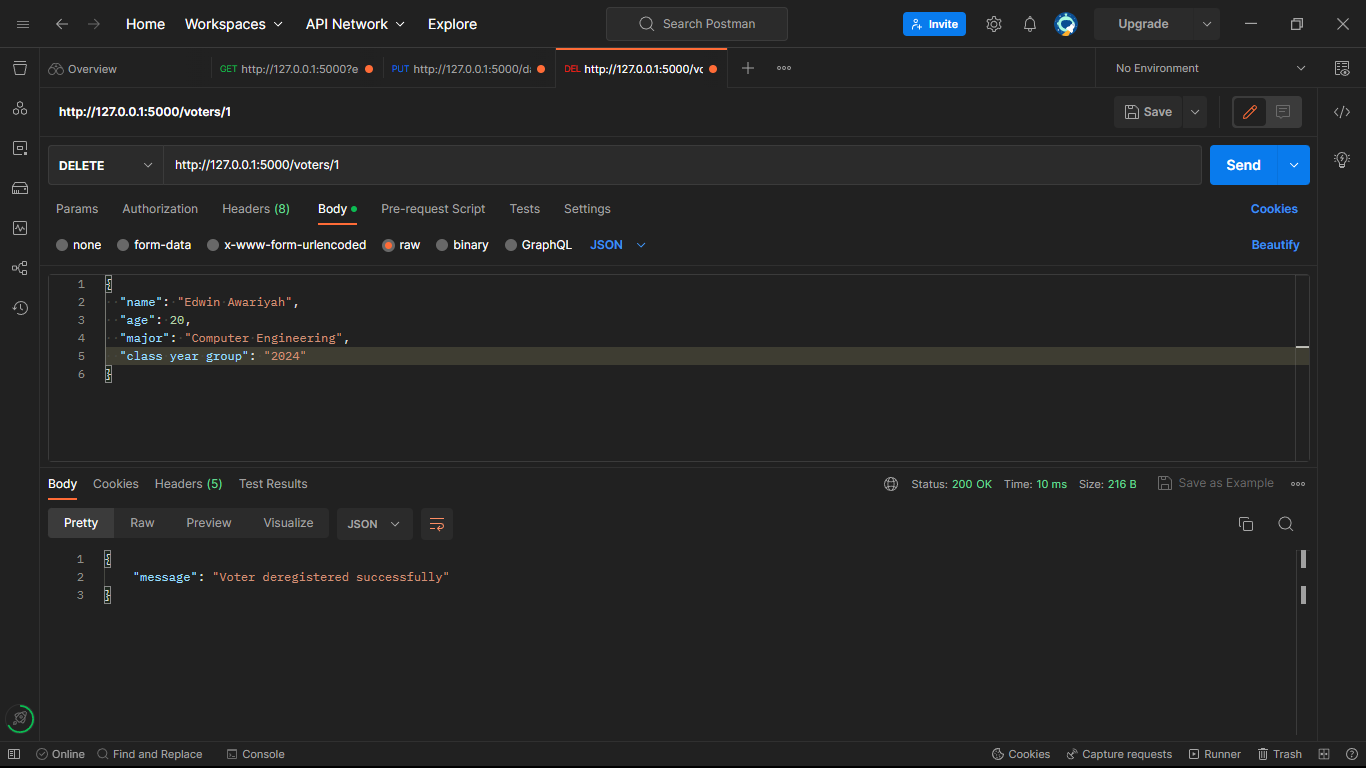
<http://127.0.0.1:5000/voters>

POST   
{"name": "Edwin Awariyah", "age": 20, "major": "Computer Engineering", "class year group": "2024", "id": 1}  
Response  
200 (Ok)  
{'message': 'New voter registered successfully', 'voter\_id': 1}

  
2. De-registering a student as a voter.  
 a. A student may need to be de-registered once they leave campus.

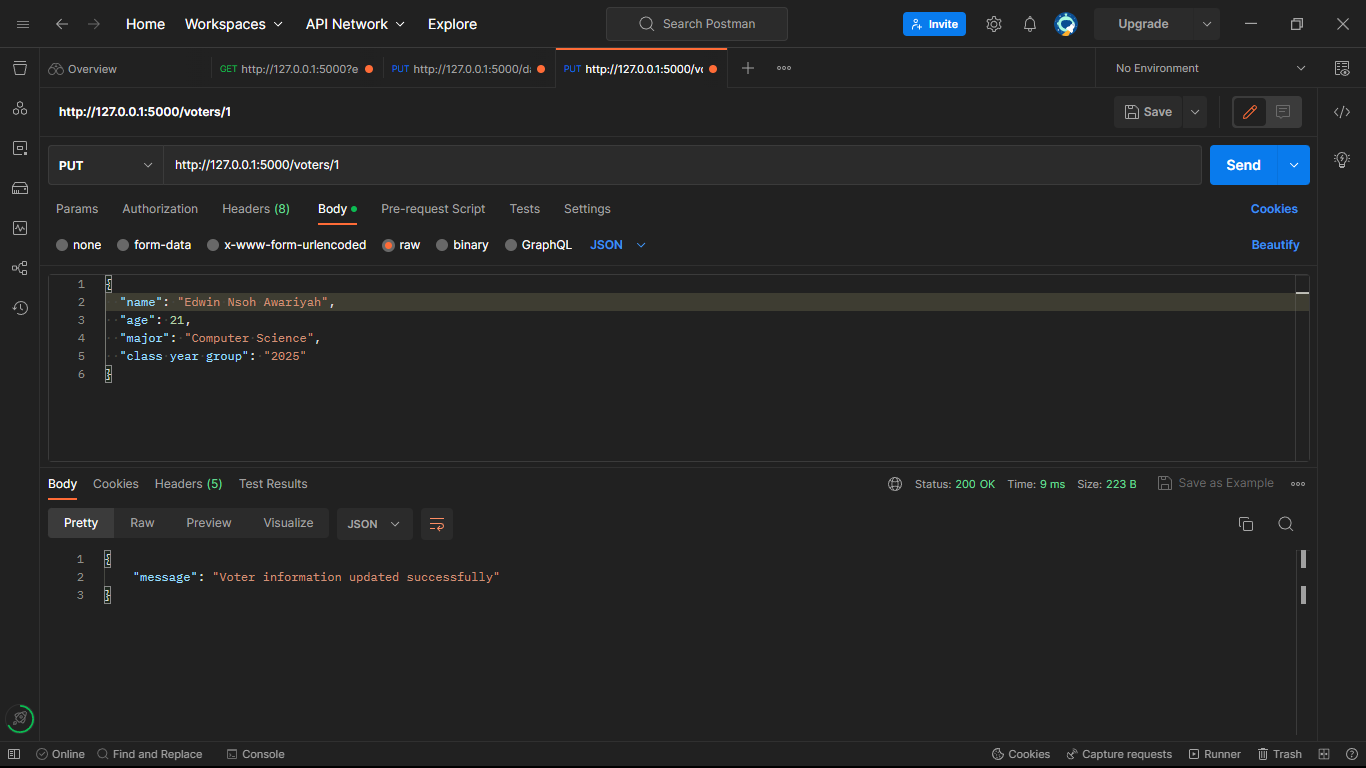
Request  
<http://127.0.0.1:5000/voters/1>

DELETE   
{"name": "Edwin Awariyah", "age": 20, "major": "Computer Engineering", "class year group": "2024", "id": 1}  
Response  
200 (Ok)  
{ 'message': 'Voter deregistered successfully'}

  
3. Updating a registered voter’s information.  
 a. A student’s year group, major or other information might change.

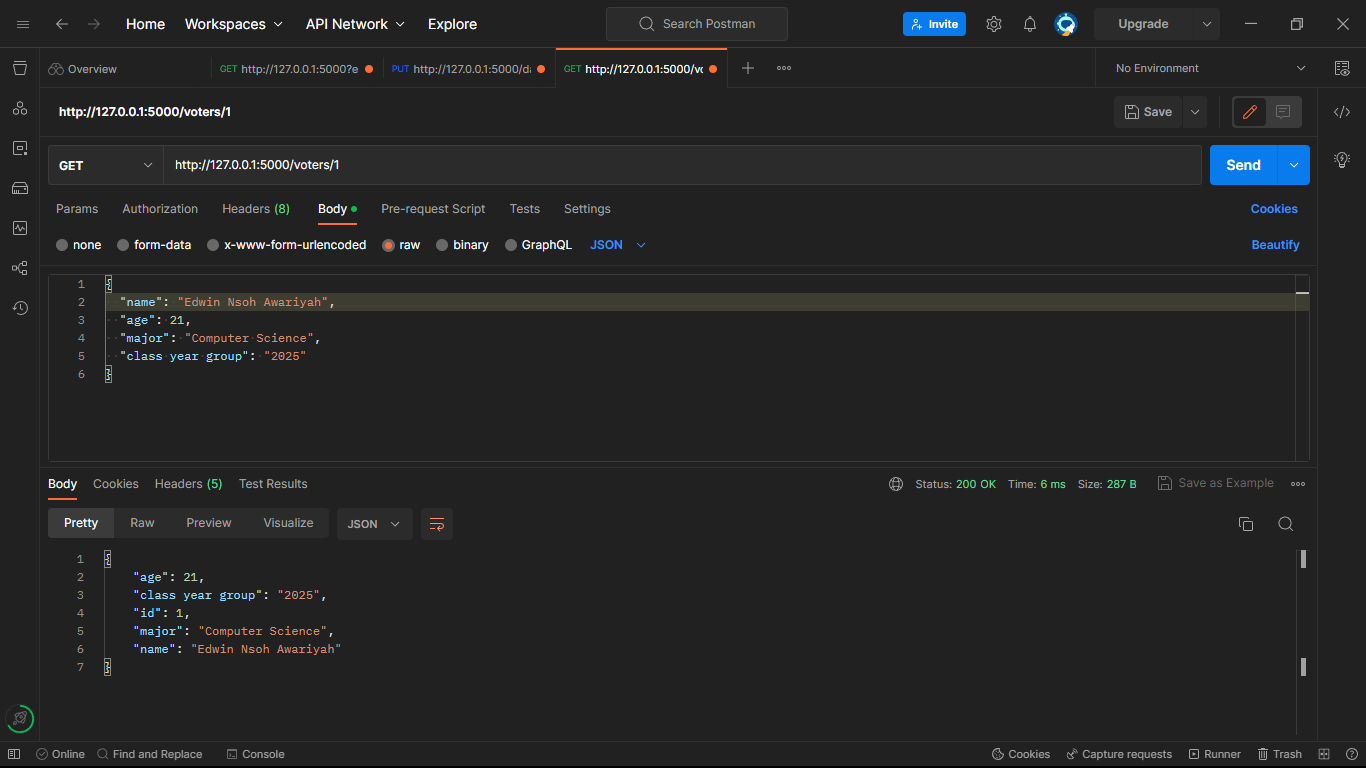
Request  
<http://127.0.0.1:5000/voters/1>

PUT   
{"name": "Edwin Nsoh Awariyah", "age": 21, "major": "Computer Science", "class year group": "2025", "id": 1}  
Response  
200 (Ok)  
{ 'message': 'Voter information updated successfully'}

  
4. Retrieving a registered voter.

Request  
<http://127.0.0.1:5000/voters/1>

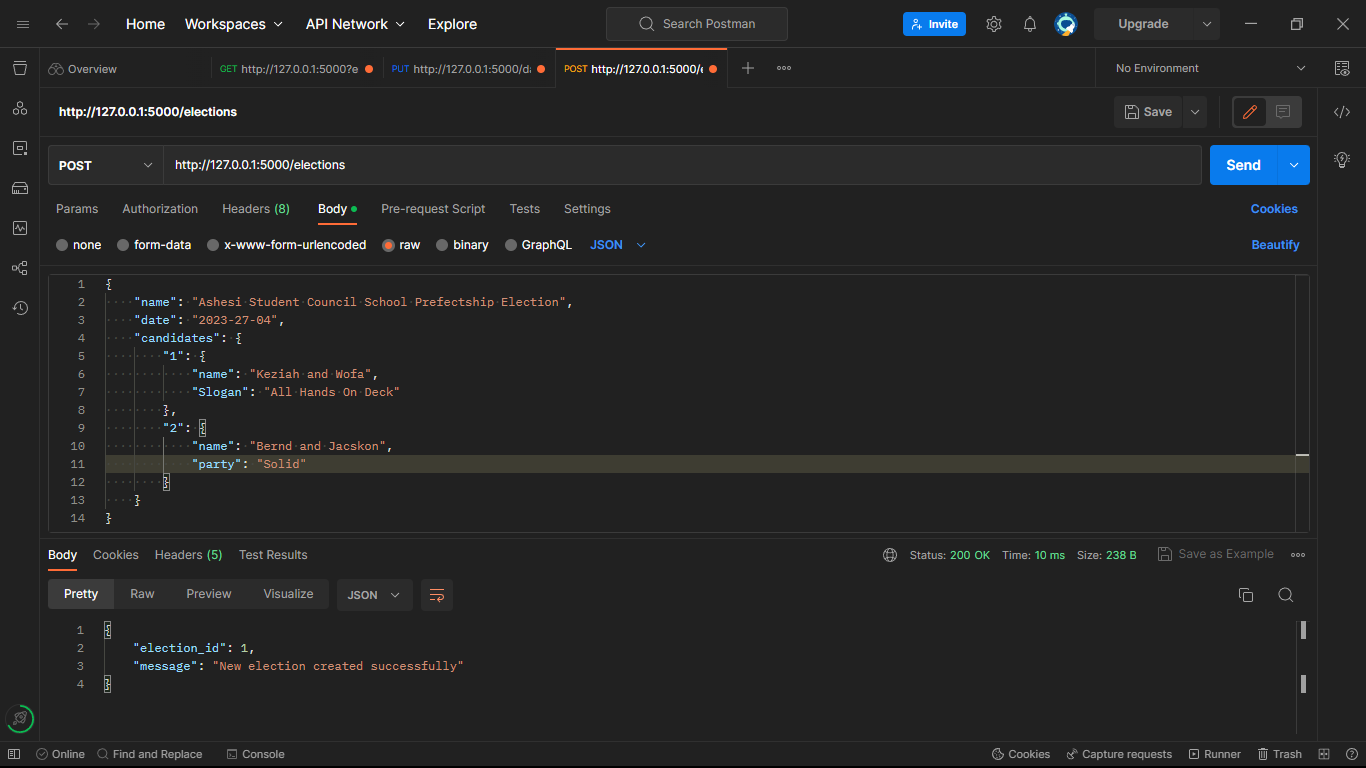
GET   
{"name": "Edwin Nsoh Awariyah", "age": 21, "major": "Computer Science", "class year group": "2025", "id": 1}  
Response  
200 (Ok)  
{"name": "Edwin Nsoh Awariyah", "age": 21, "major": "Computer Science", "class year group": "2025", "id": 1}

  
5. Creating an election.

Request  
[http://127.0.0.1:5000/election](http://127.0.0.1:5000/election/1)s

POST   
{"name": "Ashesi Student Council School Prefectship Election", "date": "2023-27-04", "candidates": {"1": {"name": "Keziah and Wofa", "Slogan": "All Hands On Deck"}, "2": {"name": "Bernd and Jacskon", "party": "Solid"}}

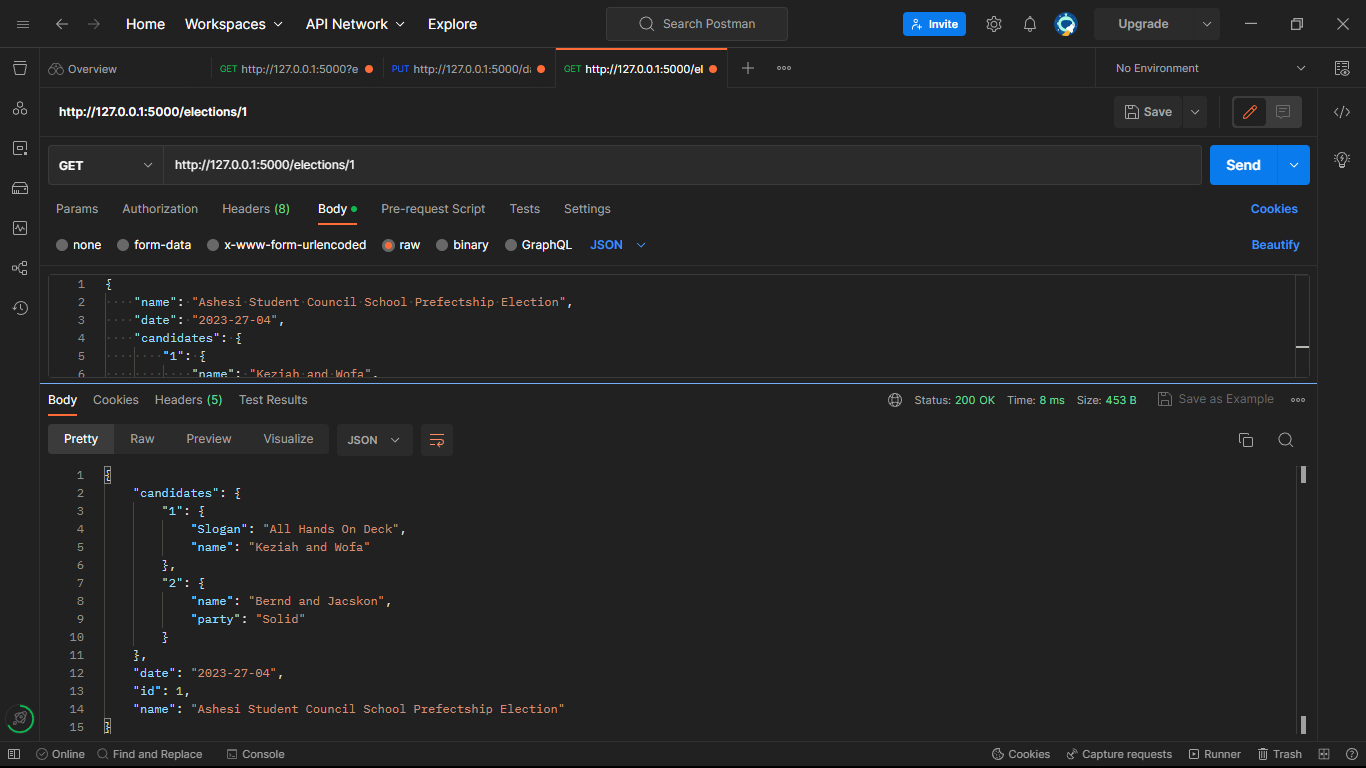
Response  
200 (Ok)  
{ 'message': 'New election created successfully'}

  
6. Retrieving an election (with its details).

Request  
<http://127.0.0.1:5000/elections/1>

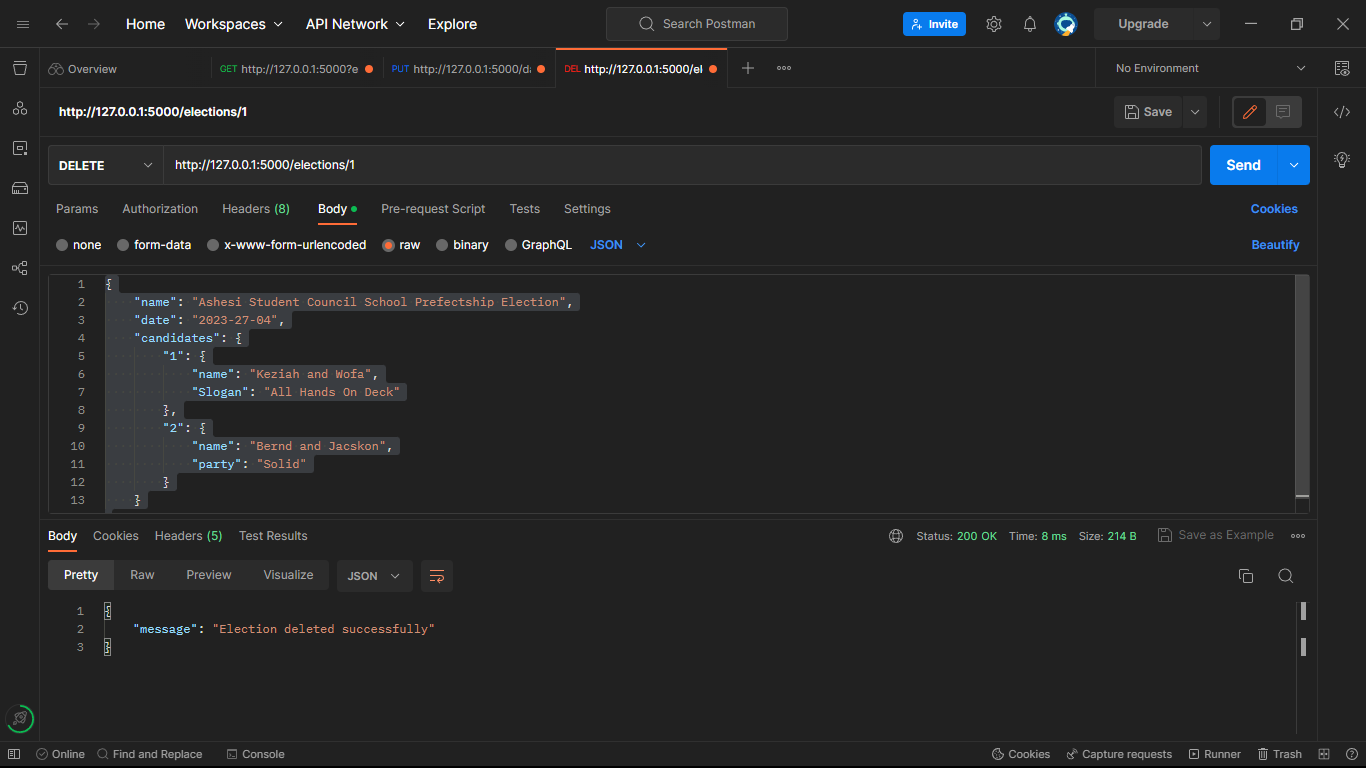
GET   
{"name": "Ashesi Student Council School Prefectship Election", "date": "2023-27-04", "candidates": {"1": {"name": "Keziah and Wofa", "Slogan": "All Hands On Deck"}, "2": {"name": "Bernd and Jacskon", "party": "Solid"}}

Response  
200 (Ok)  
{"name": "Ashesi Student Council School Prefectship Election", "date": "2023-27-04", "candidates": {"1": {"name": "Keziah and Wofa", "Slogan": "All Hands On Deck"}, "2": {"name": "Bernd and Jacskon", "party": "Solid"}}

  
7. Deleting an election.

Request  
<http://127.0.0.1:5000/elections/1>

DELETE   
{"name": "Ashesi Student Council School Prefectship Election", "date": "2023-27-04", "candidates": {"1": {"name": "Keziah and Wofa", "Slogan": "All Hands On Deck"}, "2": {"name": "Bernd and Jacskon", "party": "Solid"}}

Response  
200 (Ok)  
{ 'message': 'Election deleted successfully'}  
8. Voting in an election.

Request  
<http://127.0.0.1:5000/elections/vote/1>

PUT   
{"voter\_id": "1", "candidate\_id": "1"}

Response  
200 (Ok)  
{ 'message': Vote cast successfully'}