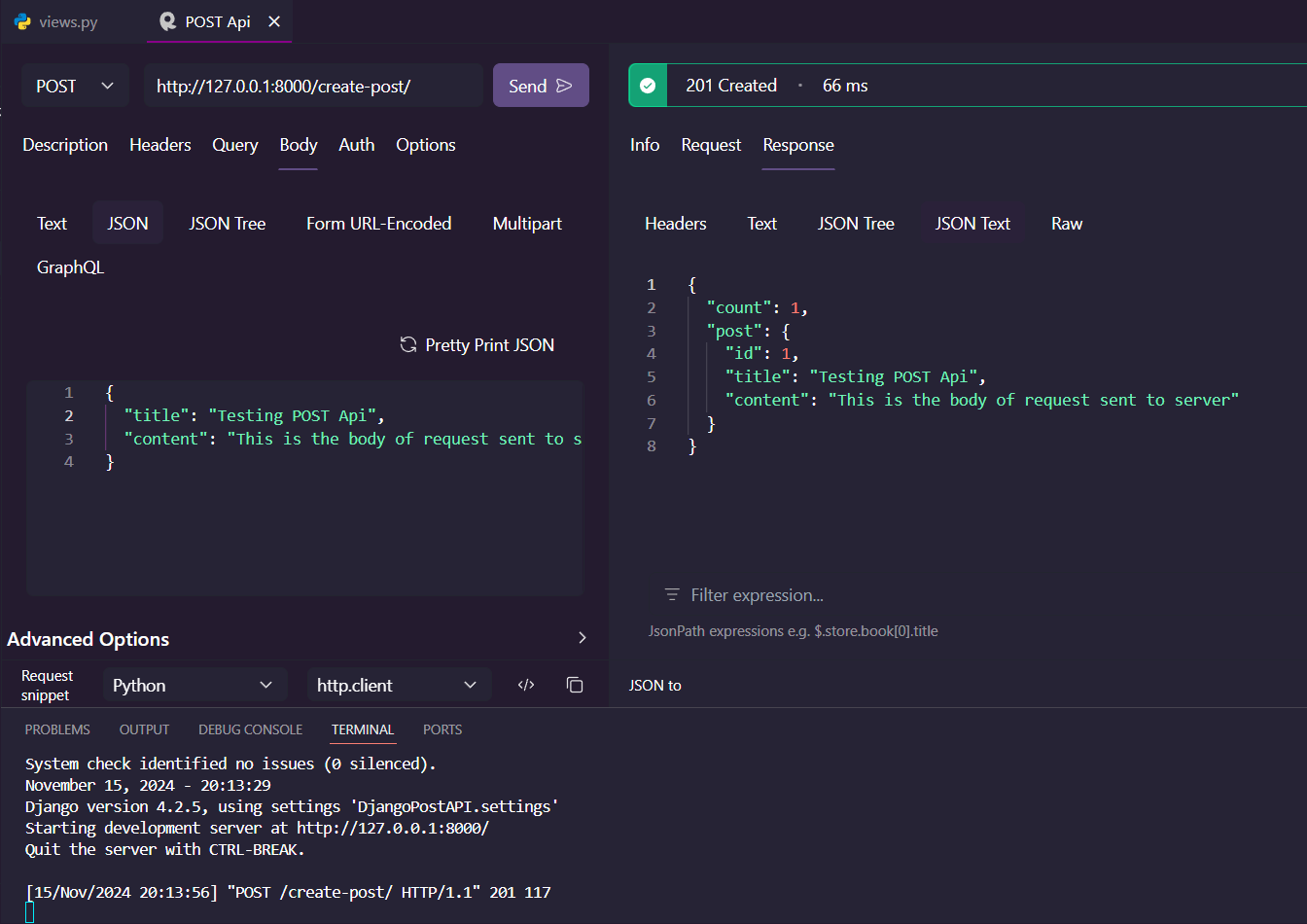
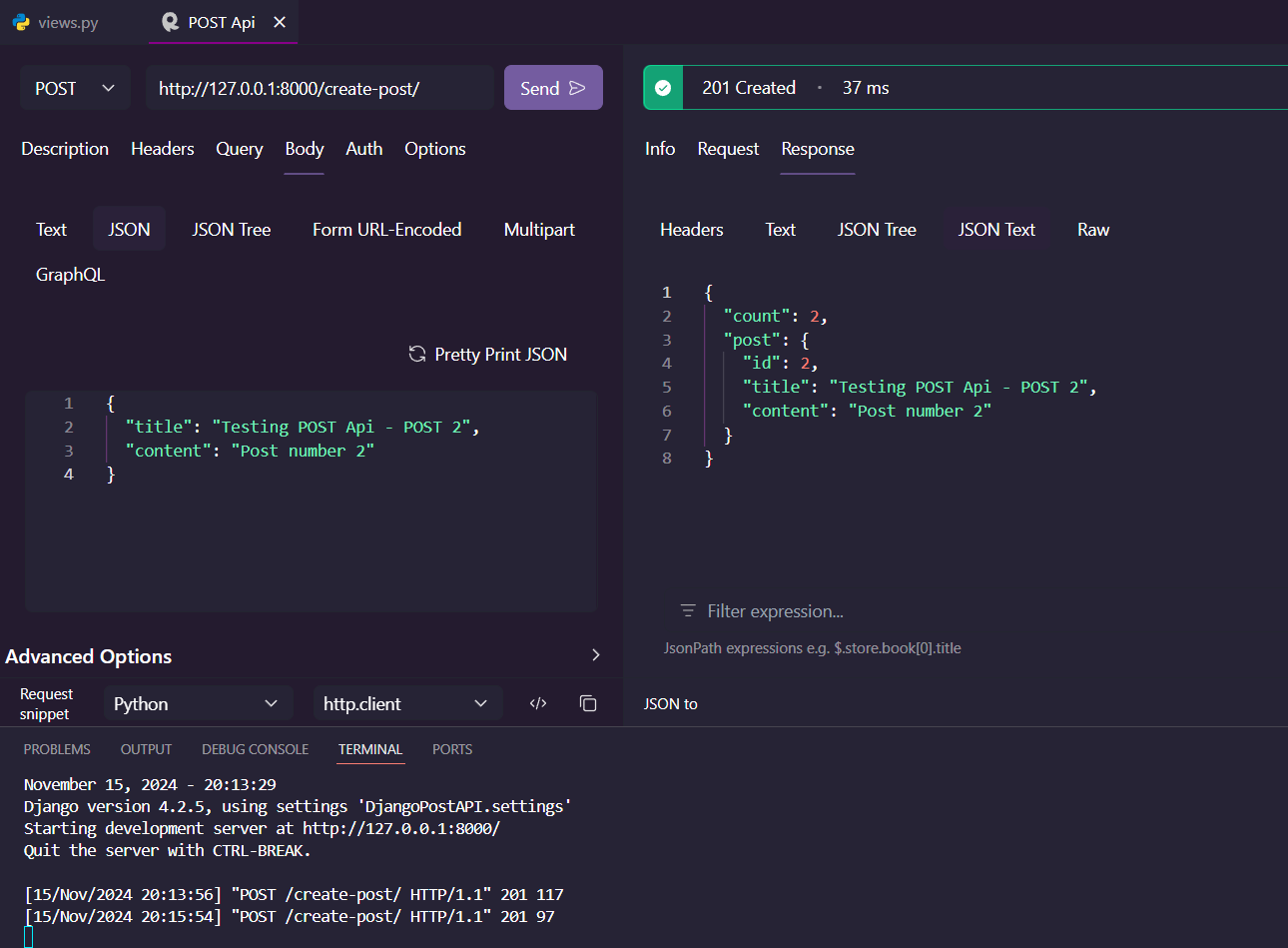
**Task 3**

1. **Task Description:**  
   Build a Post Api which also returns the serialized data with count of data saved.
2. **Task Output Screenshot:**





1. **Algorithm Used in Task:**

For this task, the algorithm used is a POST API endpoint in Django that creates a new blog post and returns the count of all posts along with the details of the newly created post.

1. The view function checks if the incoming request is a POST request and then tries to load body as JSON.
2. If data is valid, it extracts title and content field from the request.
3. If both title and content are present, the function creates a new Post object in the database using the provided data.
4. The function then retrieves the total count of Post objects in the database.
5. Finally, the function returns a JSON response containing the count of posts and the details of the newly created post, with a 201 Created status.
6. If the request body is not in valid JSON format, the function returns an error message with a 400 Bad Request status.
7. If the request method is not POST, the function returns an error message with a 405 Method Not Allowed status.

This algorithm provides a simple API endpoint for creating new blog posts, validating the input data, and returning the created post along with the updated post count.

**Note for tester:**

No front-end has been created for this API. Please use a tool like Postman or cURL to test the API endpoints.

The key API endpoint is create\_post, which accepts POST requests to create a new blog post. The request body should be in JSON format and contain the following fields:

* title: The title of the blog post
* content: The content of the blog post

The API will respond with a JSON object containing the following information:

* count: The total count of blog posts in the database
* post: An object containing the details of the newly created post, including its id, title, and content