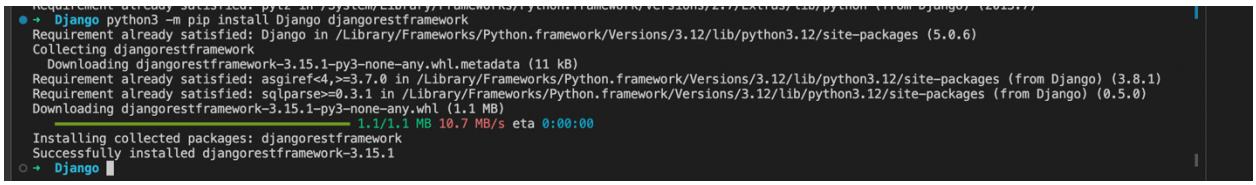
CS548
Web Services Techniques and REST Technologies

Module 5

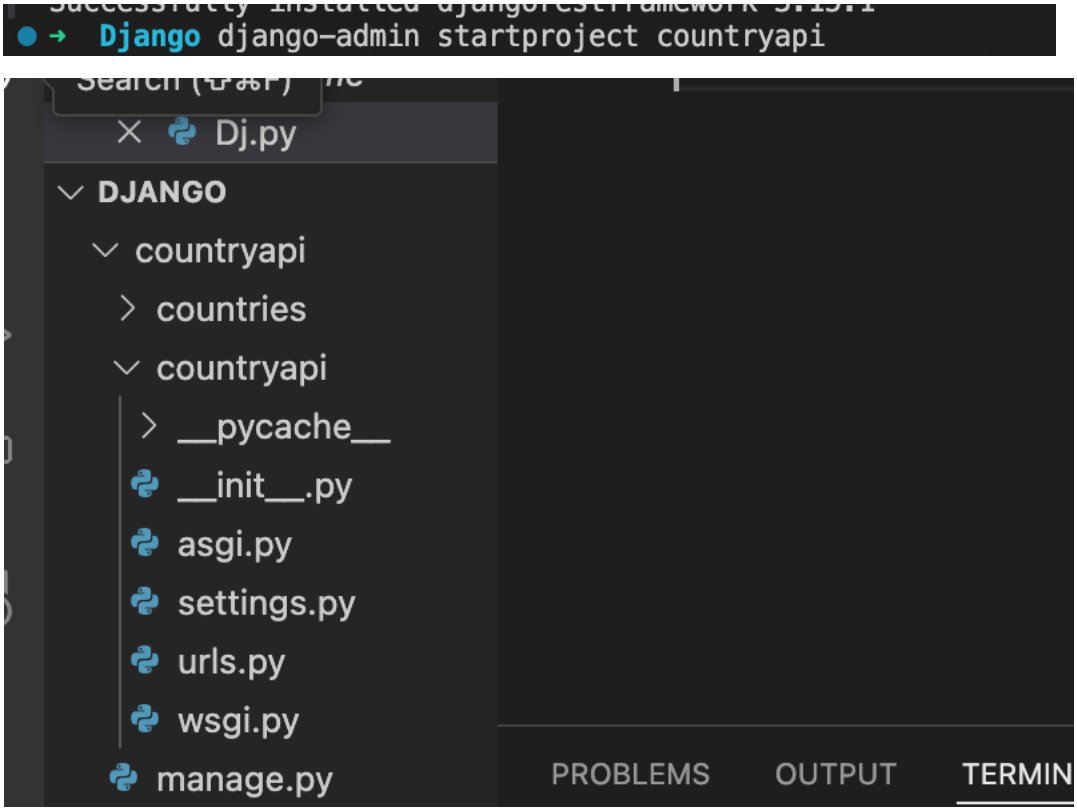
Belsabel Woldemichael

ID - 20052

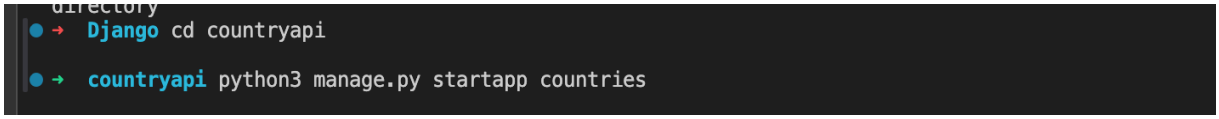
1. Build a Django project and add in Django REST framework.

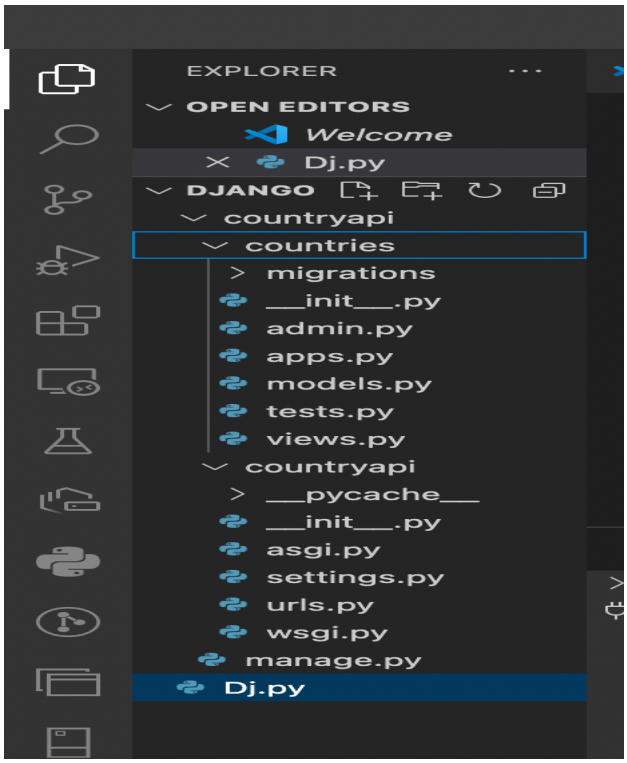


2. **Create the Django Project:** Make sure you are in the desired directory where you want to create your Django project. Run:

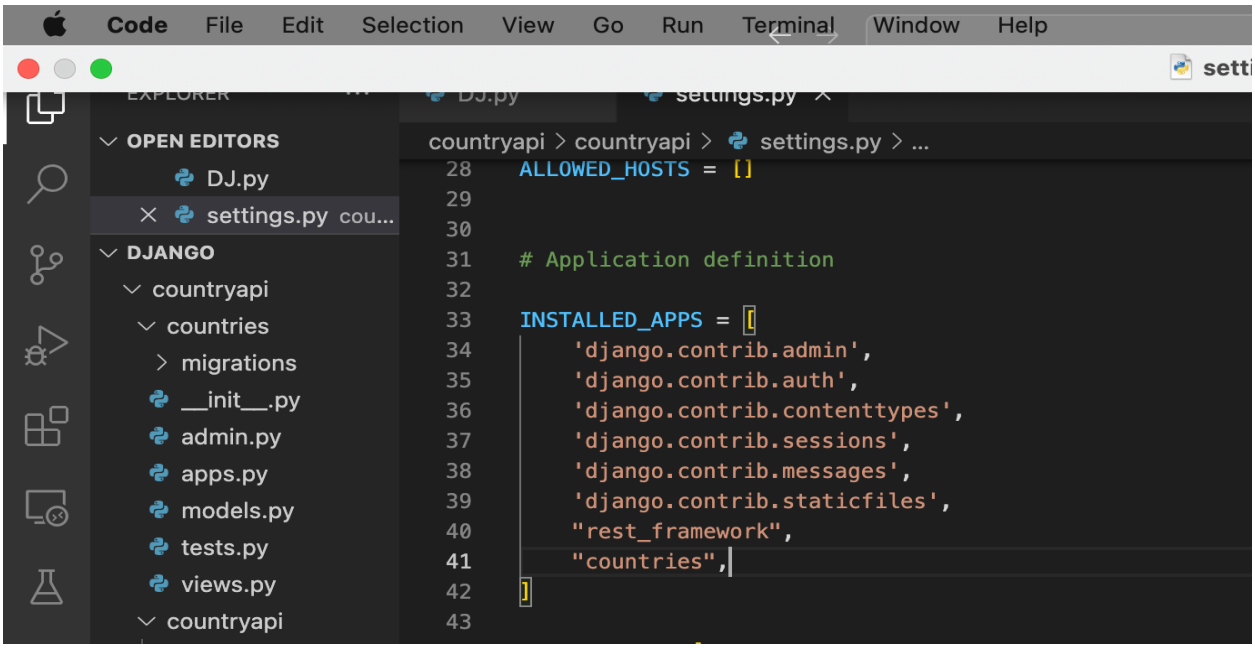


3. **Navigate to the Project Directory:** Change to the directory where manage.py is located:
4. **Create the Django App:** Run the following command within the countryapi directory:

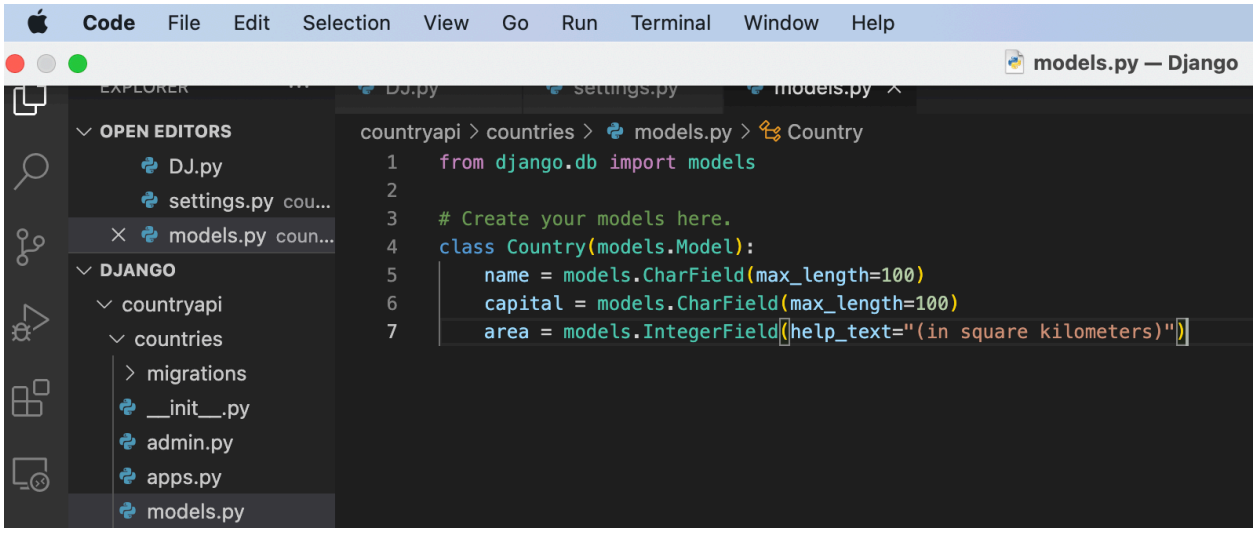




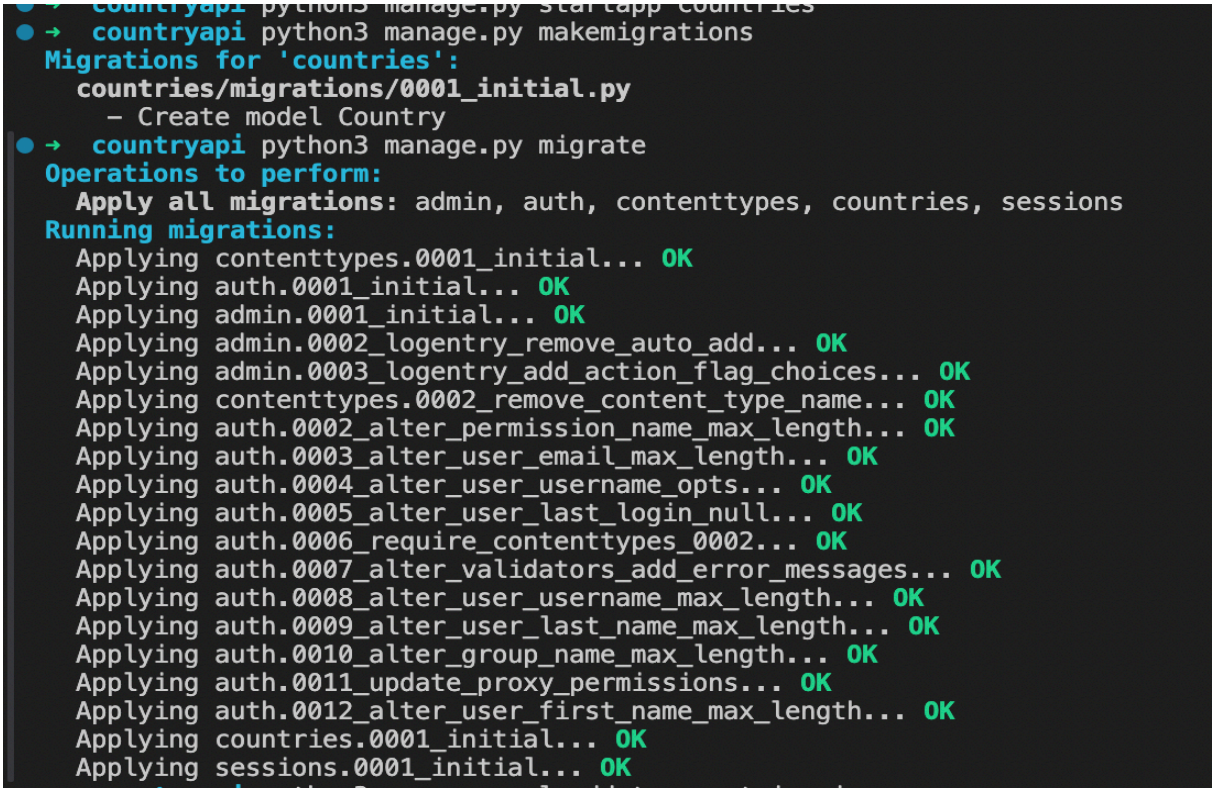
5. Open up the settings.py file that's inside the countryapi folder. Add the last two lines to INSTALLED_APPS to tell Django about the countries application and Django REST framework:



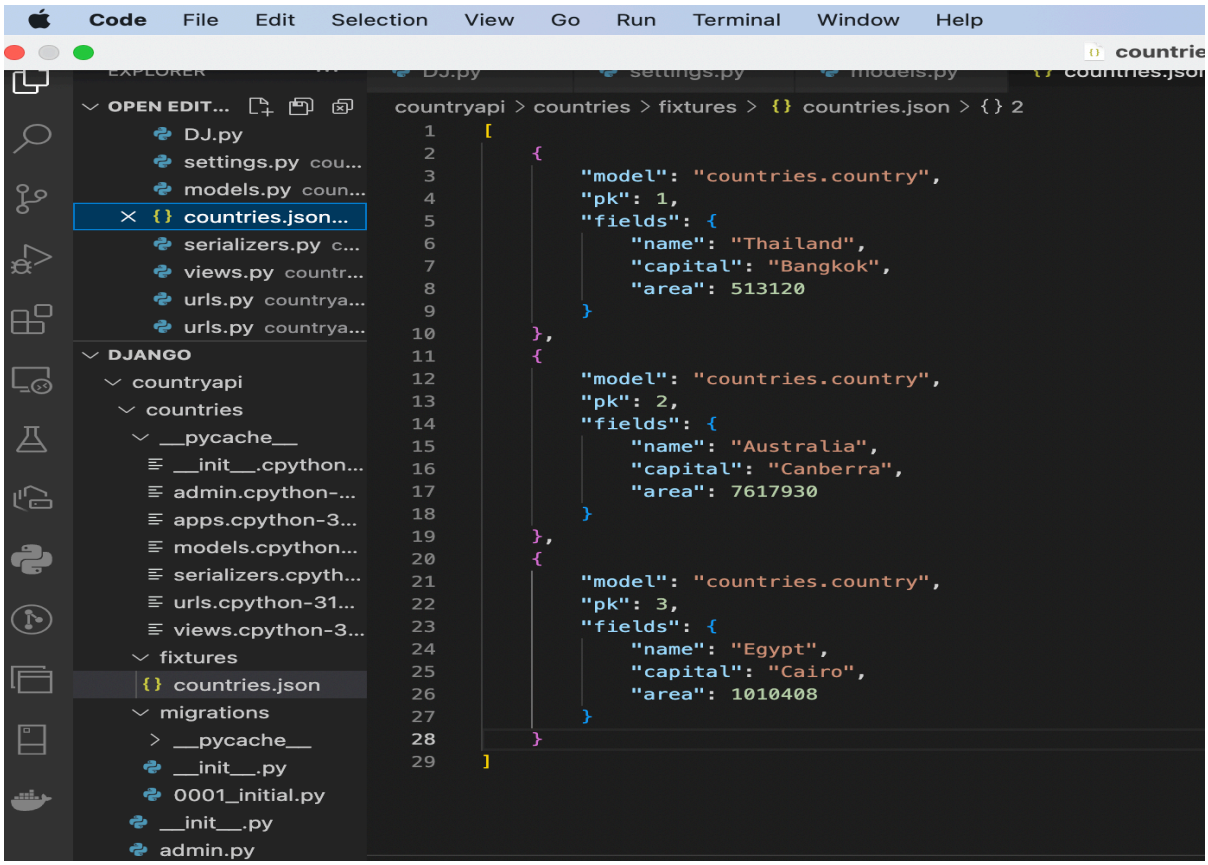
6. Inside of the countries application, update models.py with the following code:



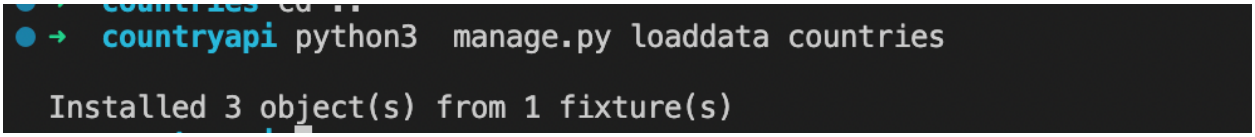
7. Run the following commands to have Django update the database based on this model:



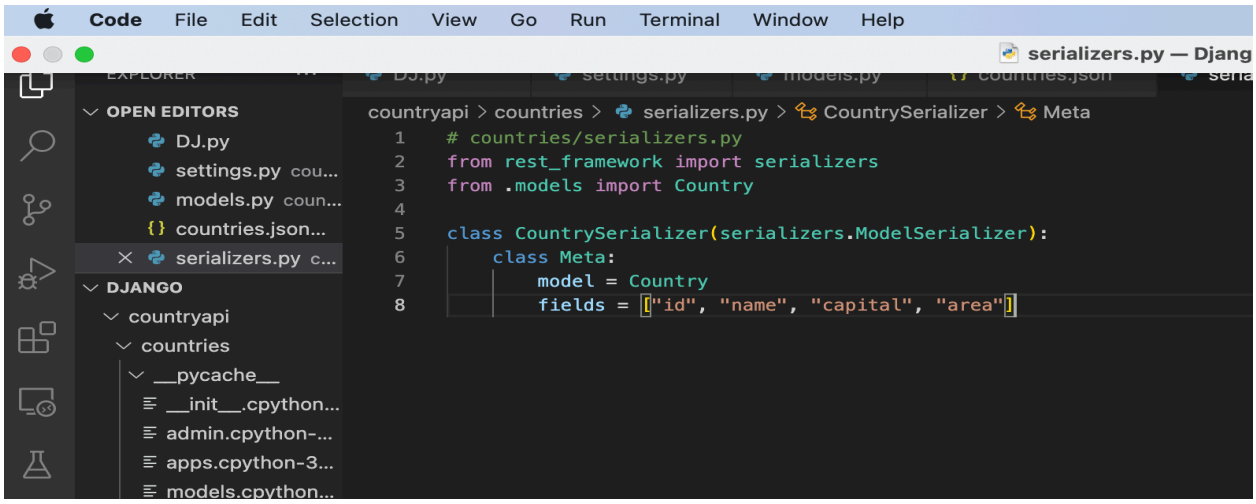
8. Copy and save the following JSON data into a file called countries.json and save it inside the countries directory:



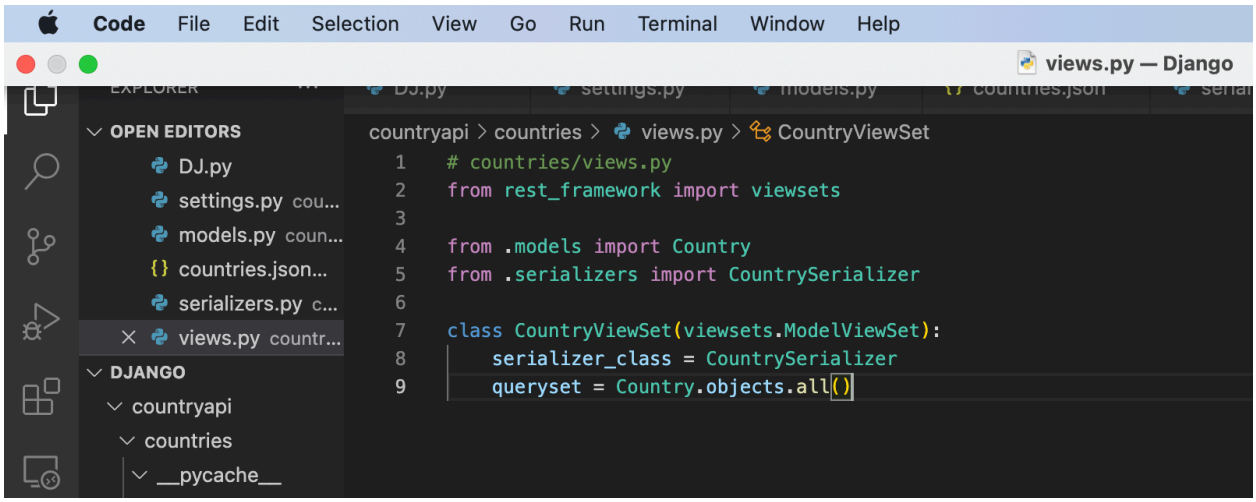
9. Call the following command to load this data in the database:



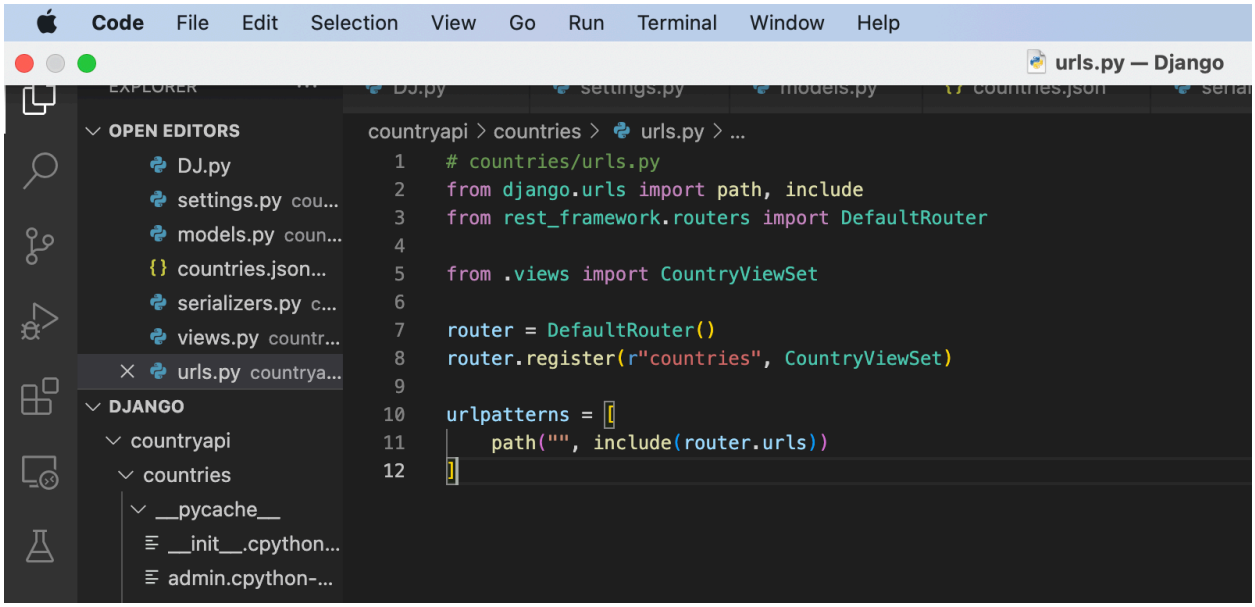
10. Start by creating a file called serializers.py inside of the countries application



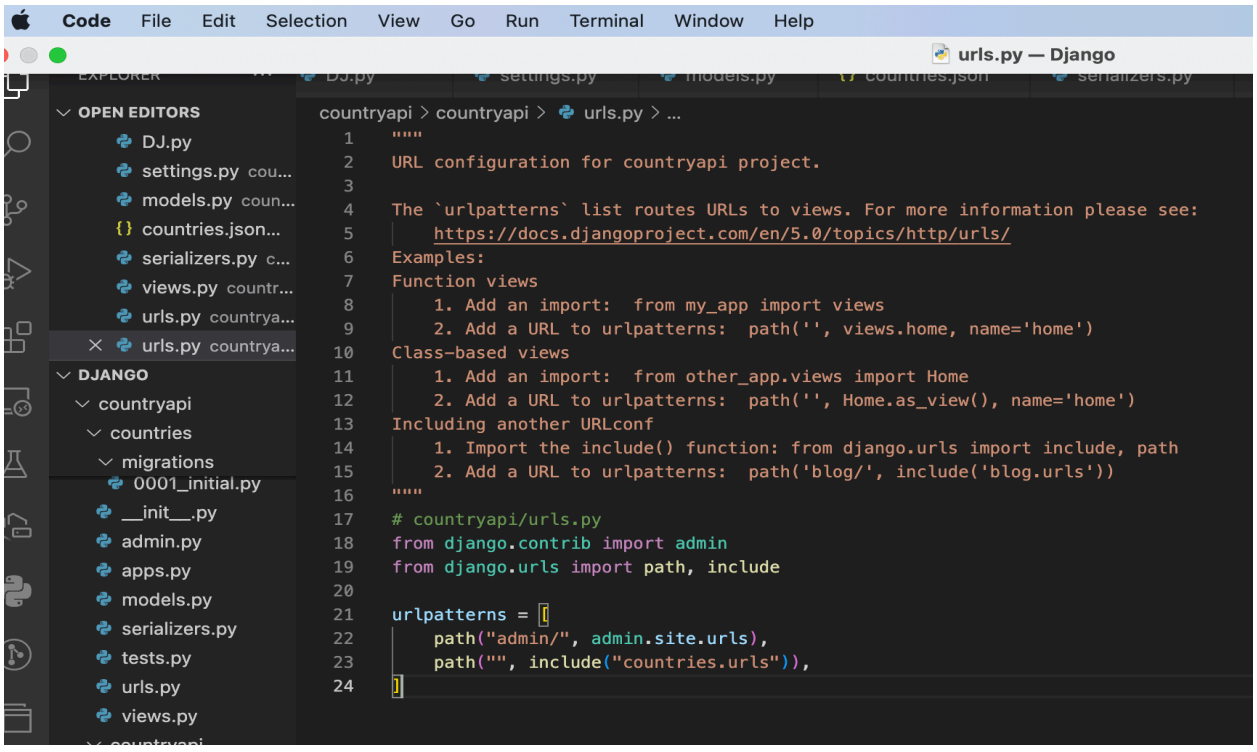
11. Below is the code for a ModelViewSet subclass called CountryViewSet. This class will generate the views needed to manage Country data. Add the following code to views.py inside the countries application:



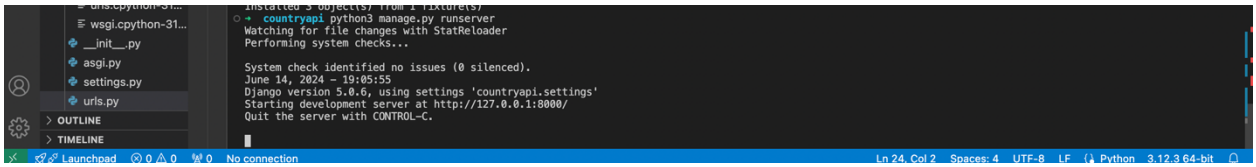
12. Create a urls.py file in the countries application and add the following code to the file:



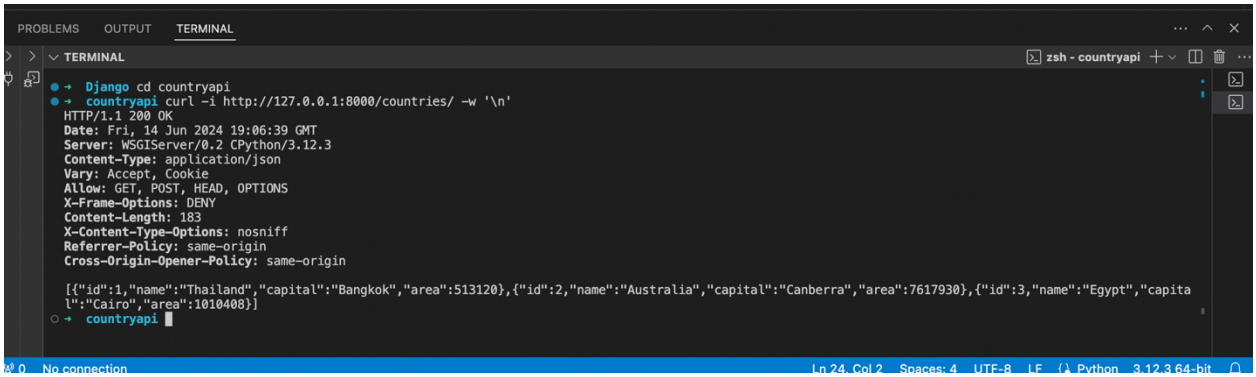
13. Finally, you need to update the project's base urls.py file to include all the countries URLs in the project. Update the urls.py file inside of the countryapi folder with the following code:



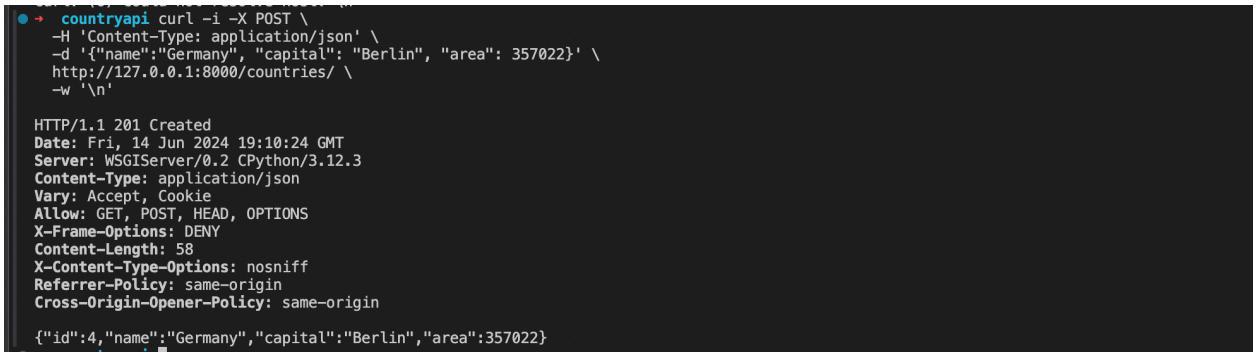
14. Run the following command in the root countryapi directory to start the Django development server:



15. The development server is now running. Go ahead and send a GET request to /countries/ to get a list of all the countries in your Django project:



16. Send a POST request to /countries/ to a create a new Country in your Django project:

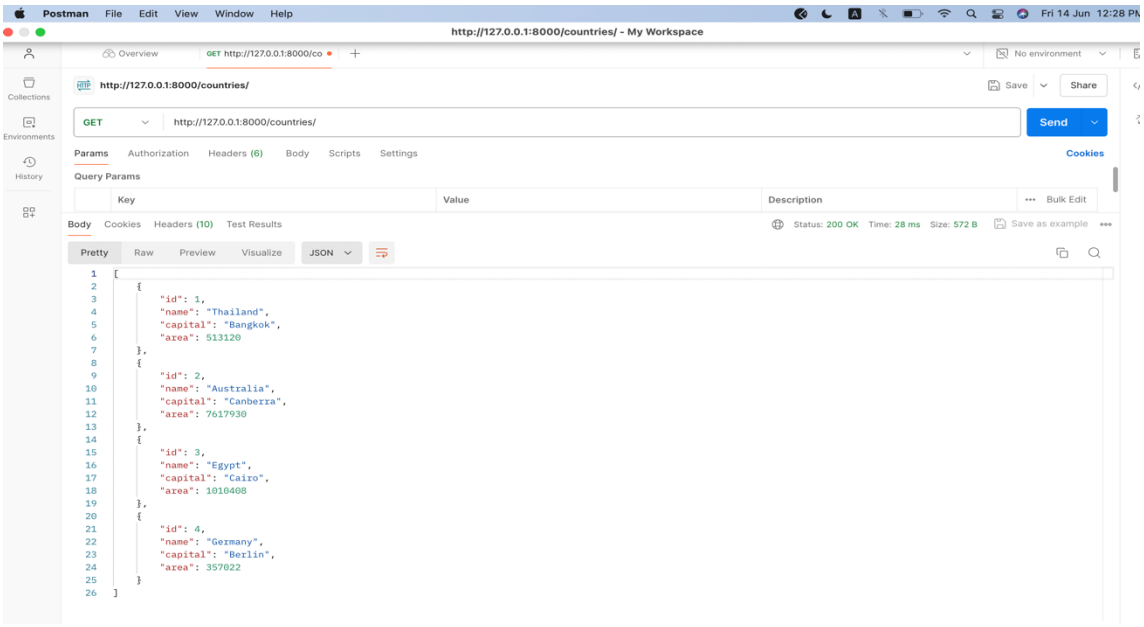


17. Run the following command to get the first Country:

```
→  countryapi curl -i http://127.0.0.1:8000/countries/1/ -w '\n'
HTTP/1.1 200 OK
Date: Fri, 14 Jun 2024 19:09:25 GMT
Server: WSGIServer/0.2 CPython/3.12.3
Content-Type: application/json
Vary: Accept, Cookie
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
X-Frame-Options: DENY
Content-Length: 60
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin

{"id":1,"name":"Thailand","capital":"Bangkok","area":513120}
→  countryapi
```

18. Get Request by using Postman.



PART 1.B

Create client calls to the server you created, as we did in class.

1. POST a New Country

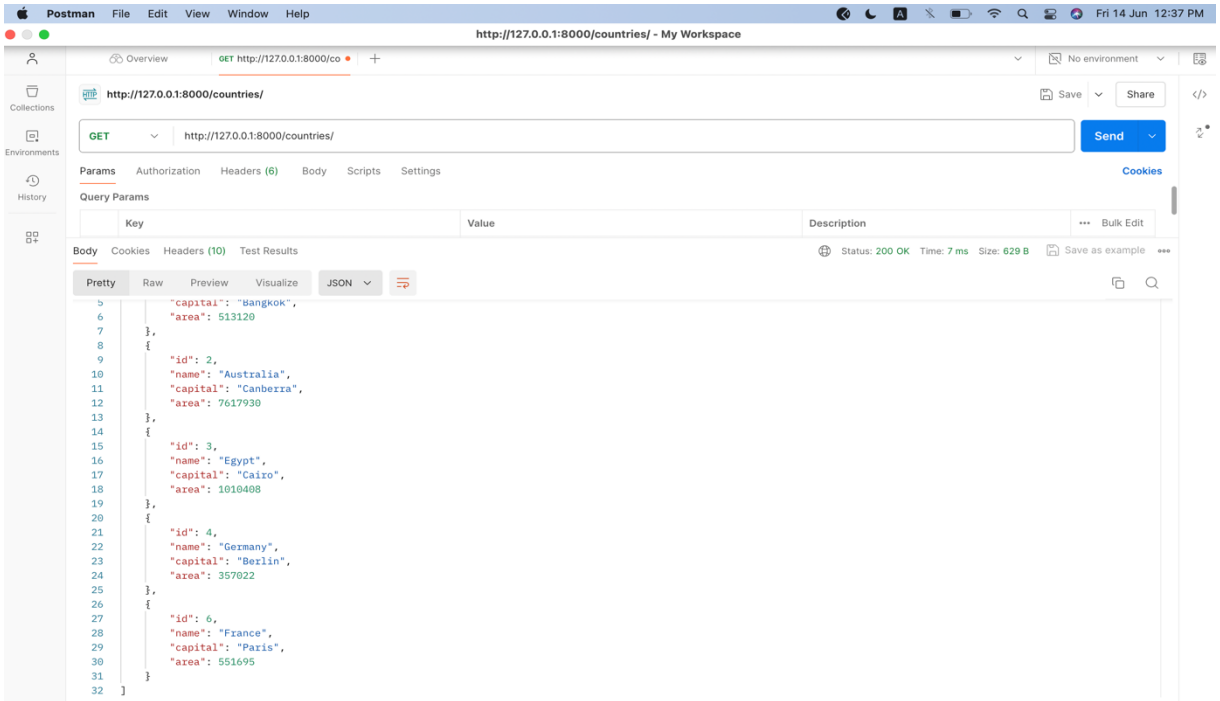
```
{"id":4,"name":"Germany","capital":"Berlin","area":357022}
countryapi curl -i -X POST \
  -H 'Content-Type: application/json' \
  -d '{"name":"France", "capital": "Paris", "area": 551695}' \
  http://127.0.0.1:8000/countries/ \
  -w '\n'

HTTP/1.1 201 Created
Date: Fri, 14 Jun 2024 19:34:32 GMT
Server: WSGIServer/0.2 CPython/3.12.3
Content-Type: application/json
Vary: Accept, Cookie
Allow: GET, POST, HEAD, OPTIONS
X-Frame-Options: DENY
Content-Length: 56
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin

{"id":5,"name":"France","capital":"Paris","area":551695}
countryapi
```

2. Get Request

- When I checked in post man, I have now France been added.



3. PUT Update a Country

```
countryapi curl -i -X PUT \
  -H 'Content-Type: application/json' \
  -d '{"name":"France", "capital": "Nice", "area": 550000}' \
  http://127.0.0.1:8000/countries/5/ \
  -w '\n'

HTTP/1.1 200 OK
Date: Fri, 14 Jun 2024 19:35:28 GMT
Server: WSGIServer/0.2 CPython/3.12.3
Content-Type: application/json
Vary: Accept, Cookie
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
X-Frame-Options: DENY
Content-Length: 55
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin

{"id":5,"name":"France","capital":"Nice","area":550000}
countryapi
```

4. PATCH Update Specific Fields of a Country

```
{"id":5,"name":"France","capital":"Nice","area":550000}
countryapi curl -i -X PATCH \
  -H 'Content-Type: application/json' \
  -d '{"capital": "Lyon"}' \
  http://127.0.0.1:8000/countries/5/ \
  -w '\n'

HTTP/1.1 200 OK
Date: Fri, 14 Jun 2024 19:36:07 GMT
Server: WSGIServer/0.2 CPython/3.12.3
Content-Type: application/json
Vary: Accept, Cookie
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
X-Frame-Options: DENY
Content-Length: 55
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin

{"id":5,"name":"France","capital":"Lyon","area":550000}
countryapi
```

5. DELETE a Country

```
countryapi curl -i -X DELETE \
  http://127.0.0.1:8000/countries/5/ \
  -w '\n'

HTTP/1.1 204 No Content
Date: Fri, 14 Jun 2024 19:36:37 GMT
Server: WSGIServer/0.2 CPython/3.12.3
Vary: Accept, Cookie
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
X-Frame-Options: DENY
Content-Length: 0
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin


countryapi
```

Explanation of Work Completed

First, I created a new Django project and integrated Django REST framework. Within this project, I created a new Django app, added it to INSTALLED_APPS, and set up the initial model in models.py. After running migration commands to create database tables based on this model, I loaded initial data into the database from a JSON file. I then developed the API by creating serializers, views, and URL configurations to handle requests for the Country model. Following this, I started the Django development server to handle these requests. Finally, I performed client calls (Post, Get, Put, Patch and Delete) to the server I created.

- **"I have neither given nor received unauthorized aid in completing this work, nor have I presented someone else's work as my own."**