**User Creation POST API and User Get API**

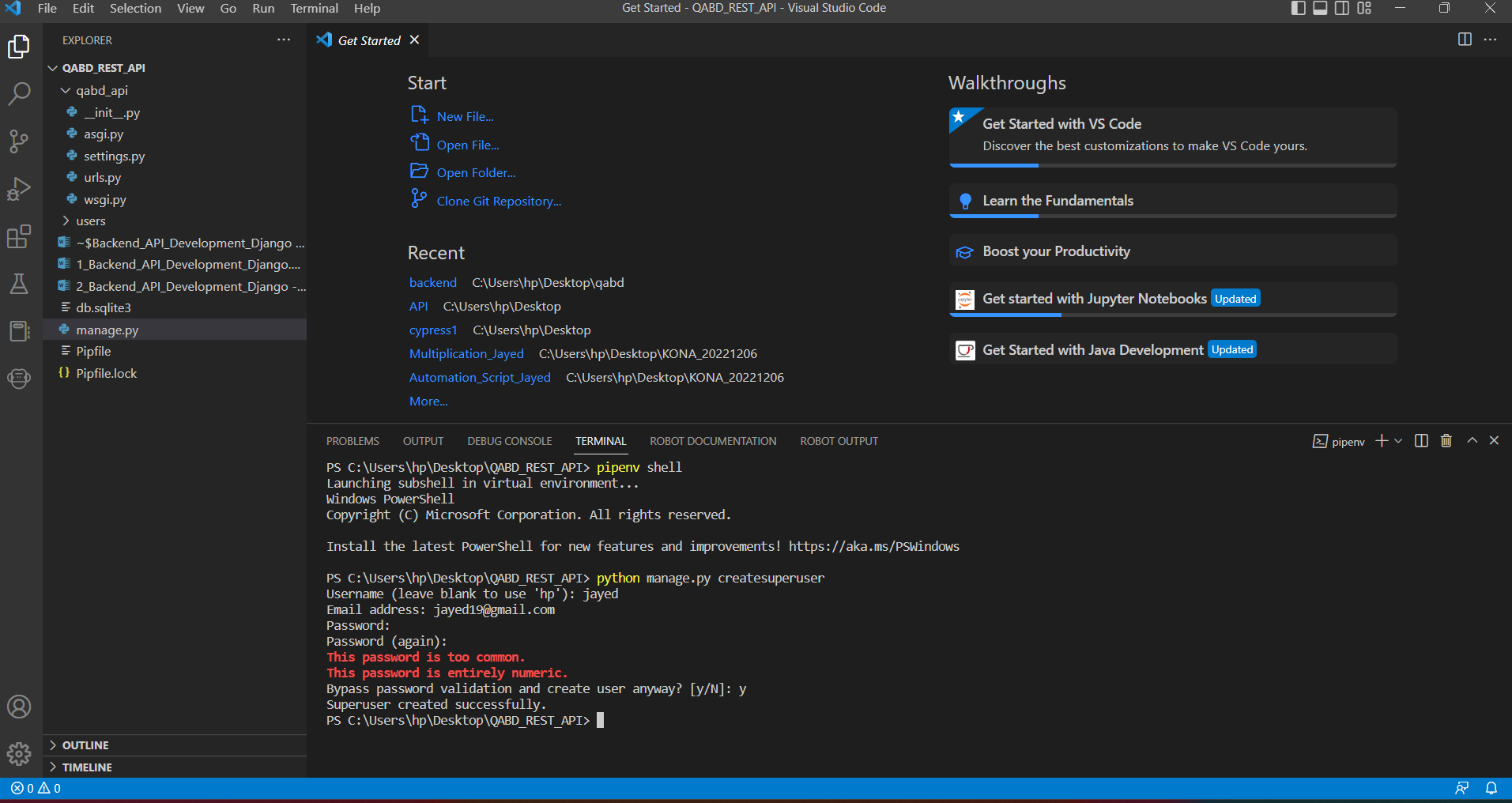
**Step0:**

Step0: Verify the virtual environment activate or not. CMD: pipenv shell

**Step1:**

Go to qabd\_api directory and run CMD:

python manage.py createsuperuser



**Step2:** create a file serializers.py inside the 'users' app directory. Write Code:

'''

Serialization is used to convert the data in your Django models into a format(JSON) that can be sent over the internet

and used by other applications.

'''

from rest\_framework import serializers

from django.contrib.auth.models import User

class UserSerializers(serializers.ModelSerializer): #inherit functions from ModelSerializer class

password=serializers.CharField(min\_length=8, write\_only=True)

def create(self, validated\_data): # This is for making password encrypted

password = validated\_data.pop('password', None)

instance = self.Meta.model(\*\*validated\_data)

if password is not None:

instance.set\_password(password)

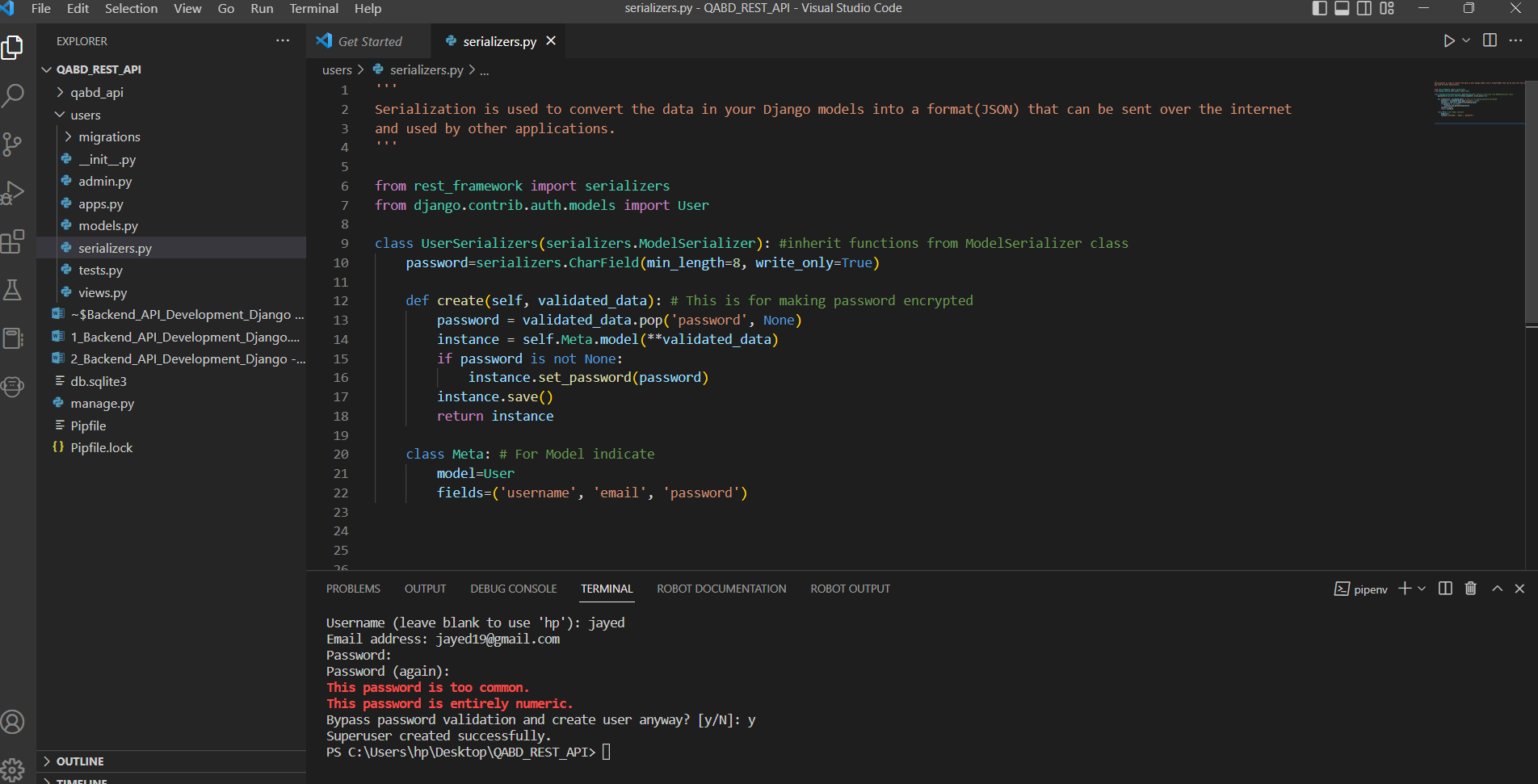
instance.save()

return instance

class Meta: # For Model indicate

model=User

fields=('username', 'email', 'password')



**Step3:** \*\*\*\*\* views.py\*\*\*\*\*\*\*

from django.shortcuts import render

# Create your views here.

from rest\_framework.views import APIView # Handle HTTP Requests

from rest\_framework.response import Response

from django.contrib.auth.models import User

from .serializers import UserSerializers #import UserSerializer class from serializers.py file

class UsersAPIView(APIView): # Inherit all functions from APIView Class

def get(self, request):

users = User.objects.all()

serializer = UserSerializers(users, many = True)

return Response(serializer.data) # Serializer used for converting to JSON format

def post(self, request):

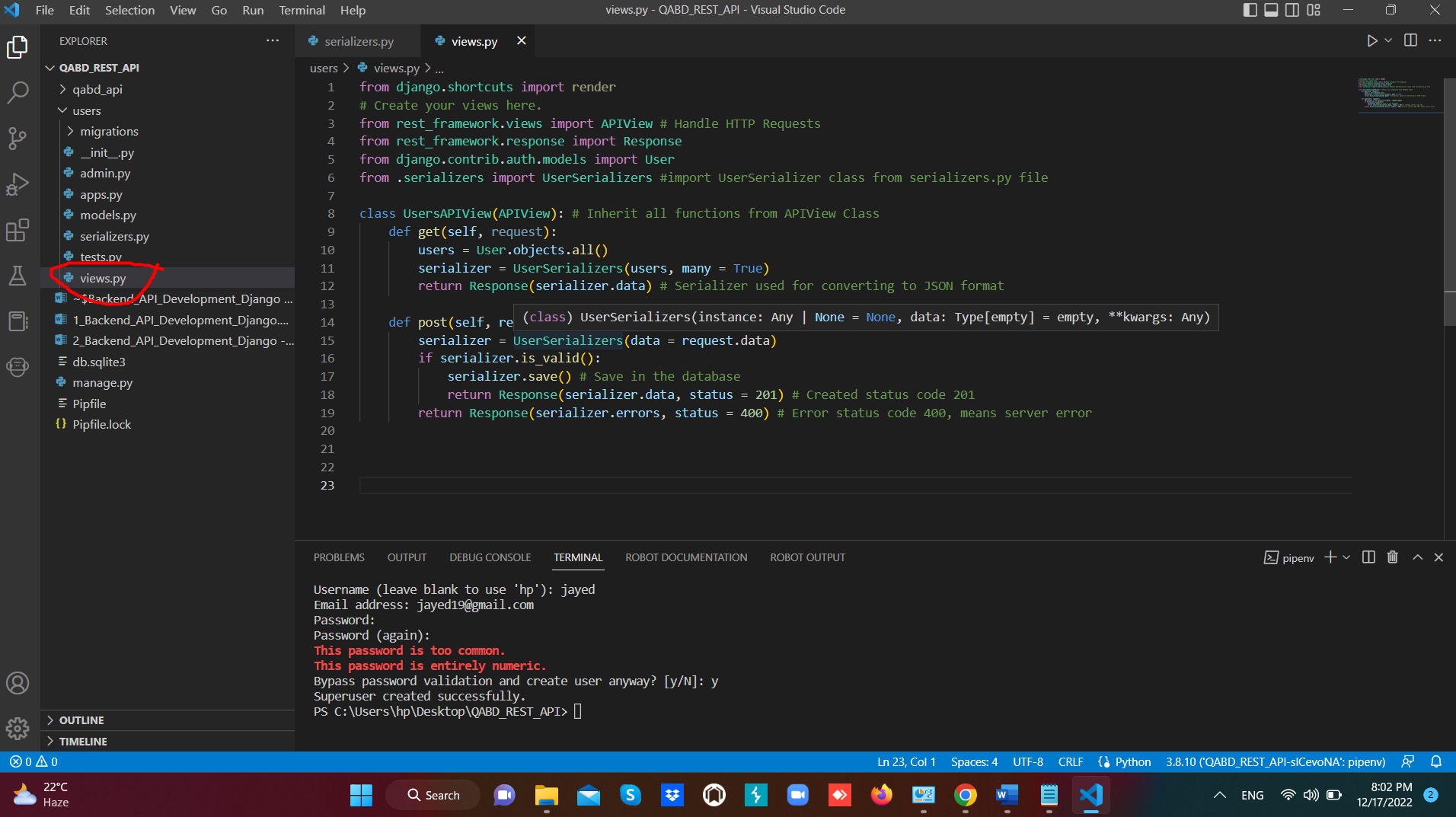
serializer = UserSerializers(data = request.data)

if serializer.is\_valid():

serializer.save() # Save in the database

return Response(serializer.data, status = 201) # Created status code 201

return Response(serializer.errors, status = 400) # Error status code 400, means server error



**Step4:** create urls.py file in QABD\_REST\_API\users\ and content of \*\*\*\*\*\*urls.py\*\*\*\*\*\*:

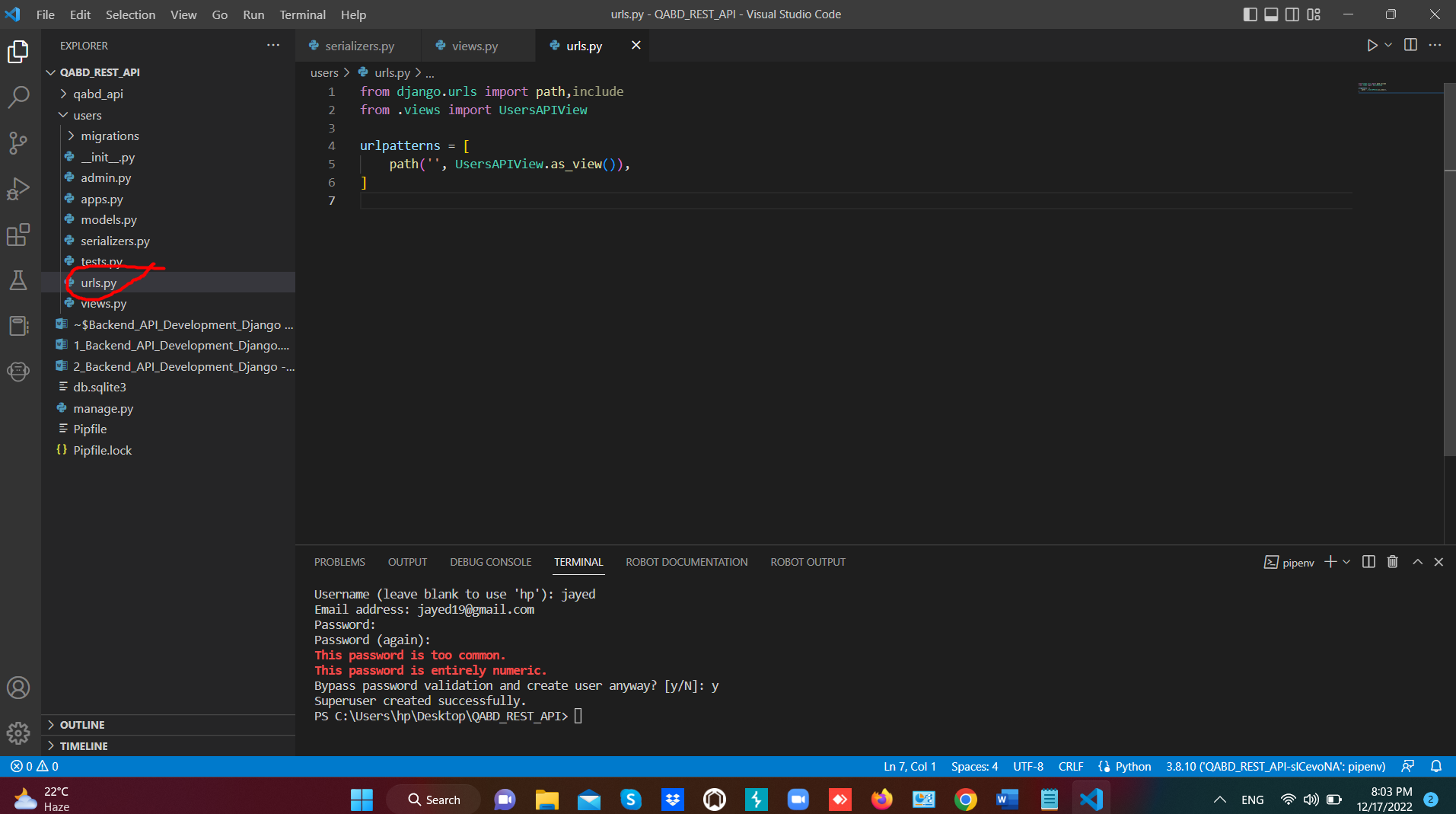
from django.urls import path,include

from .views import UsersAPIView

urlpatterns = [

path('', UsersAPIView.as\_view()),

]



**Step5:** Now Edit file \*\*\*\*\*\* QABD\_REST\_API\qabd\_api \urls.py\*\*\*\*\*\*\*\*\*\*\*\*

from django.contrib import admin

from django.urls import path,include

from users import urls as users\_urls

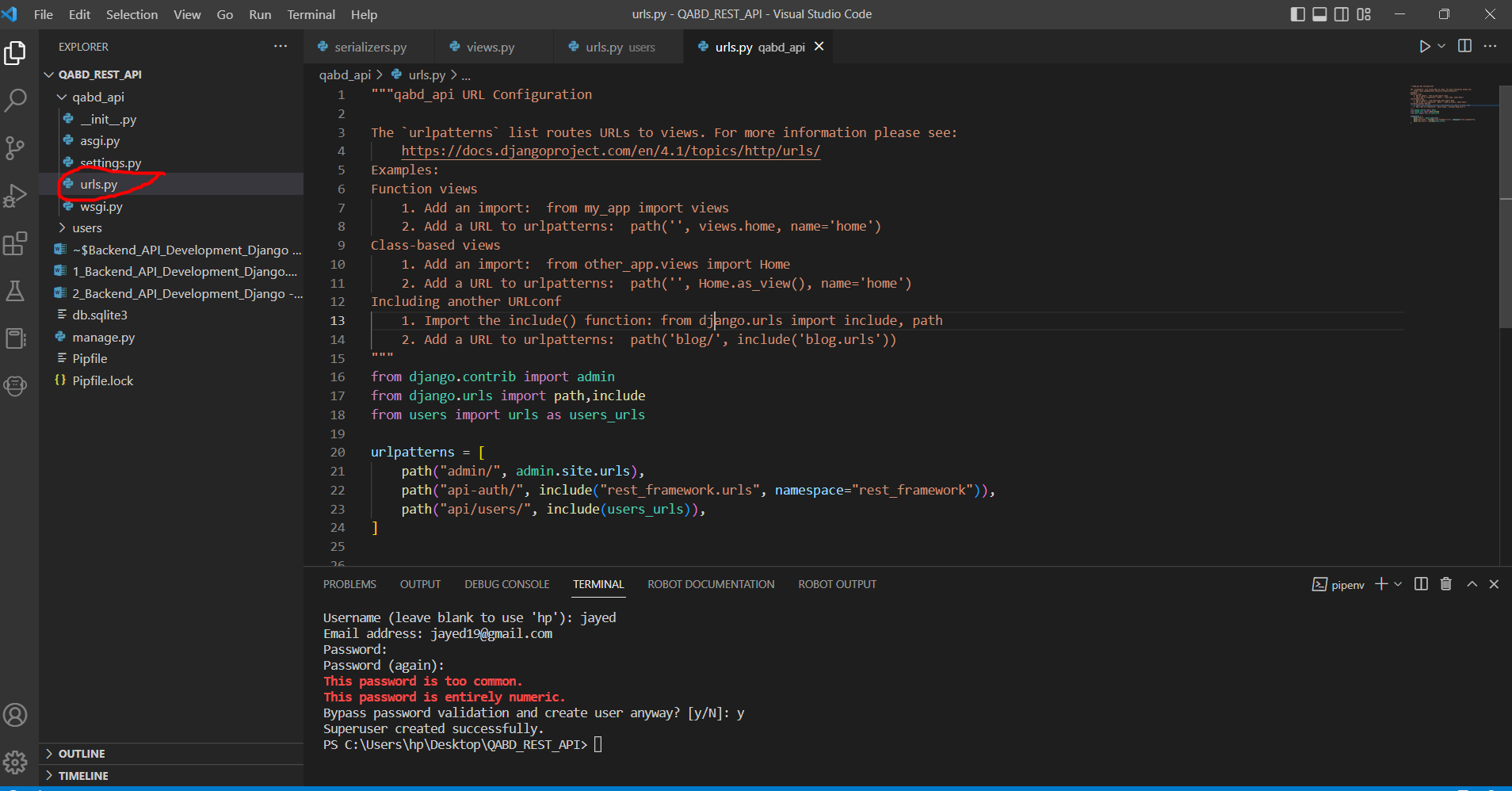
urlpatterns = [

path("admin/", admin.site.urls),

path("api-auth/", include("rest\_framework.urls", namespace="rest\_framework")),

path("api/users/", include(users\_urls)),

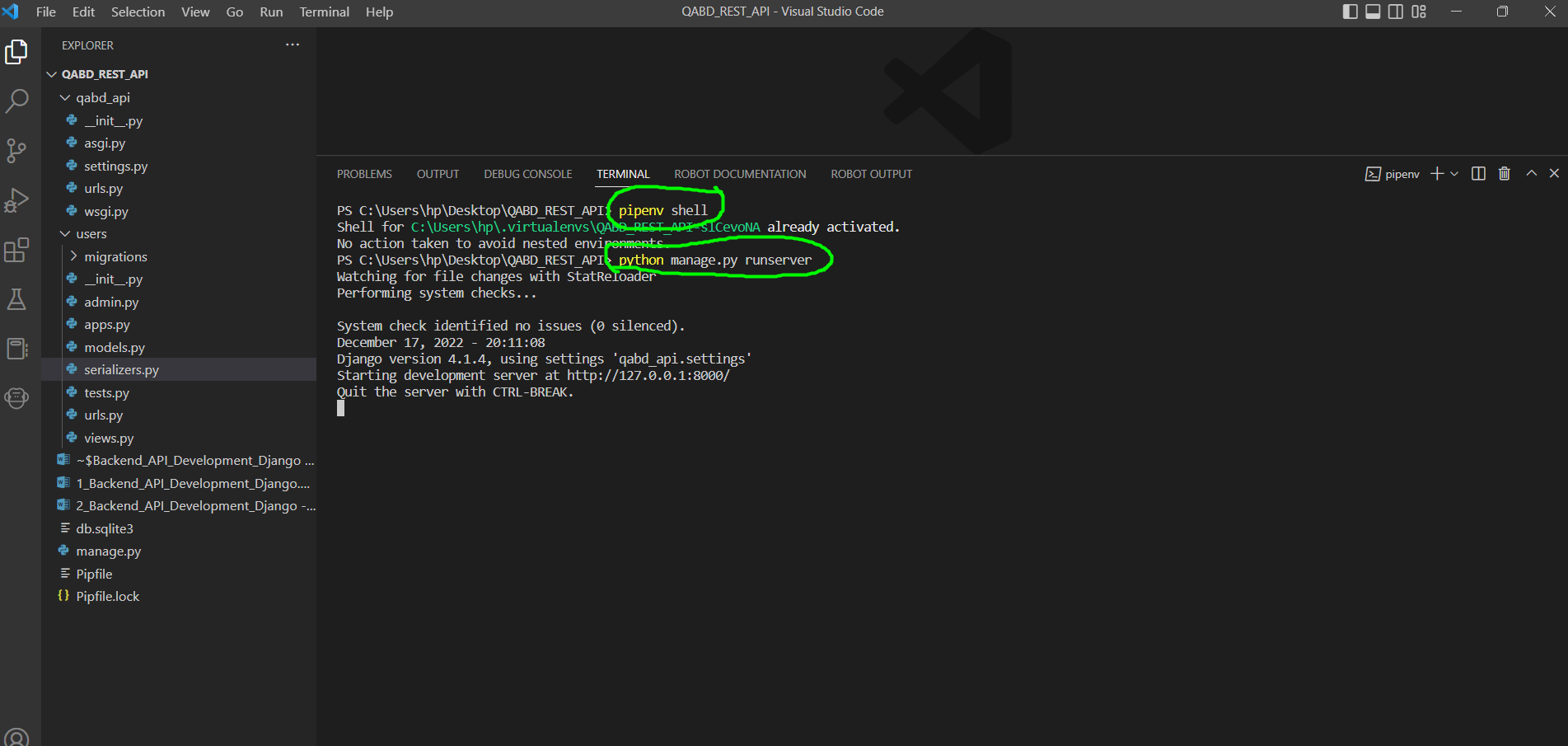
]



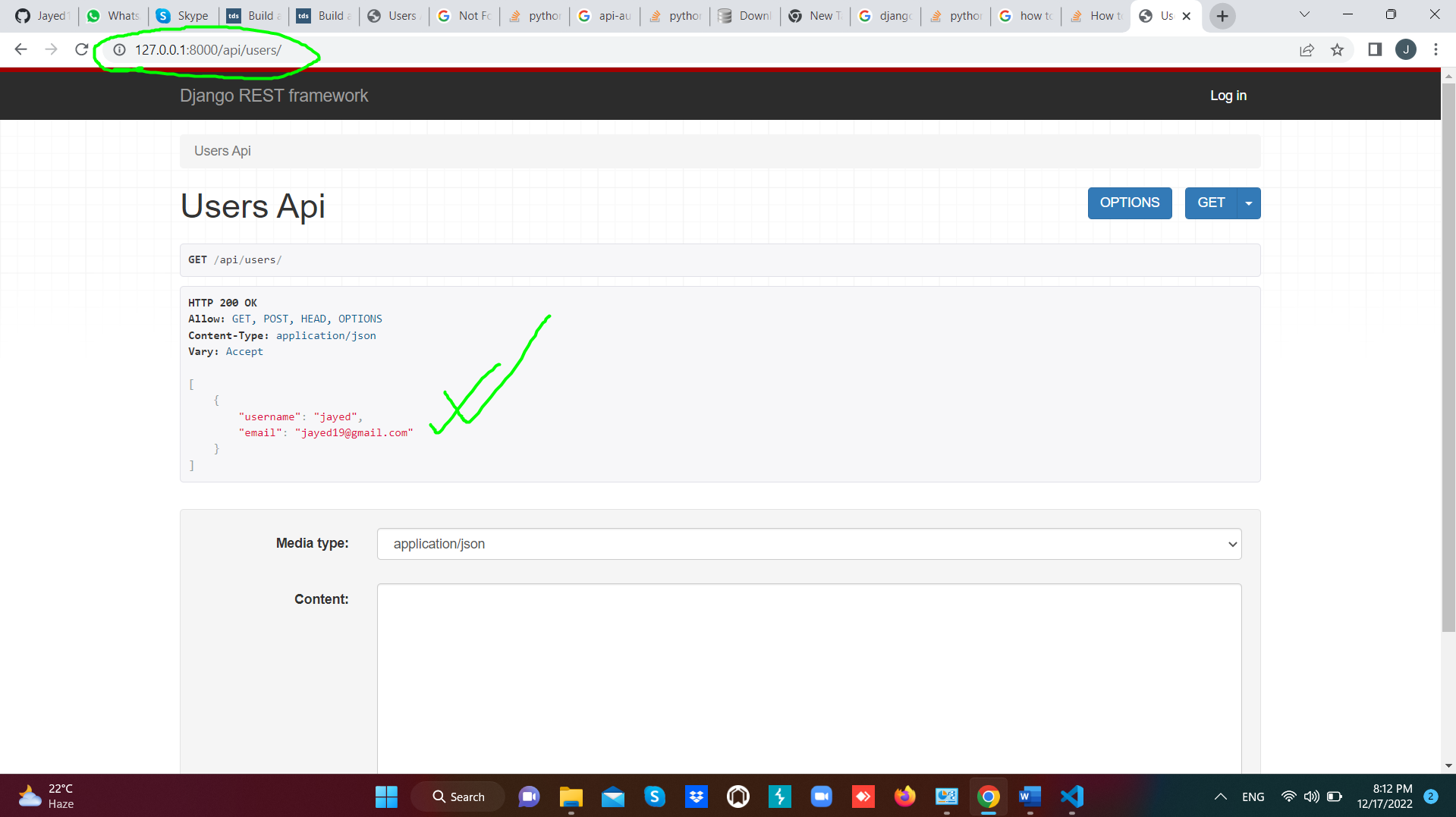
**Step6:** Test the API: CMD:

pipenv shell

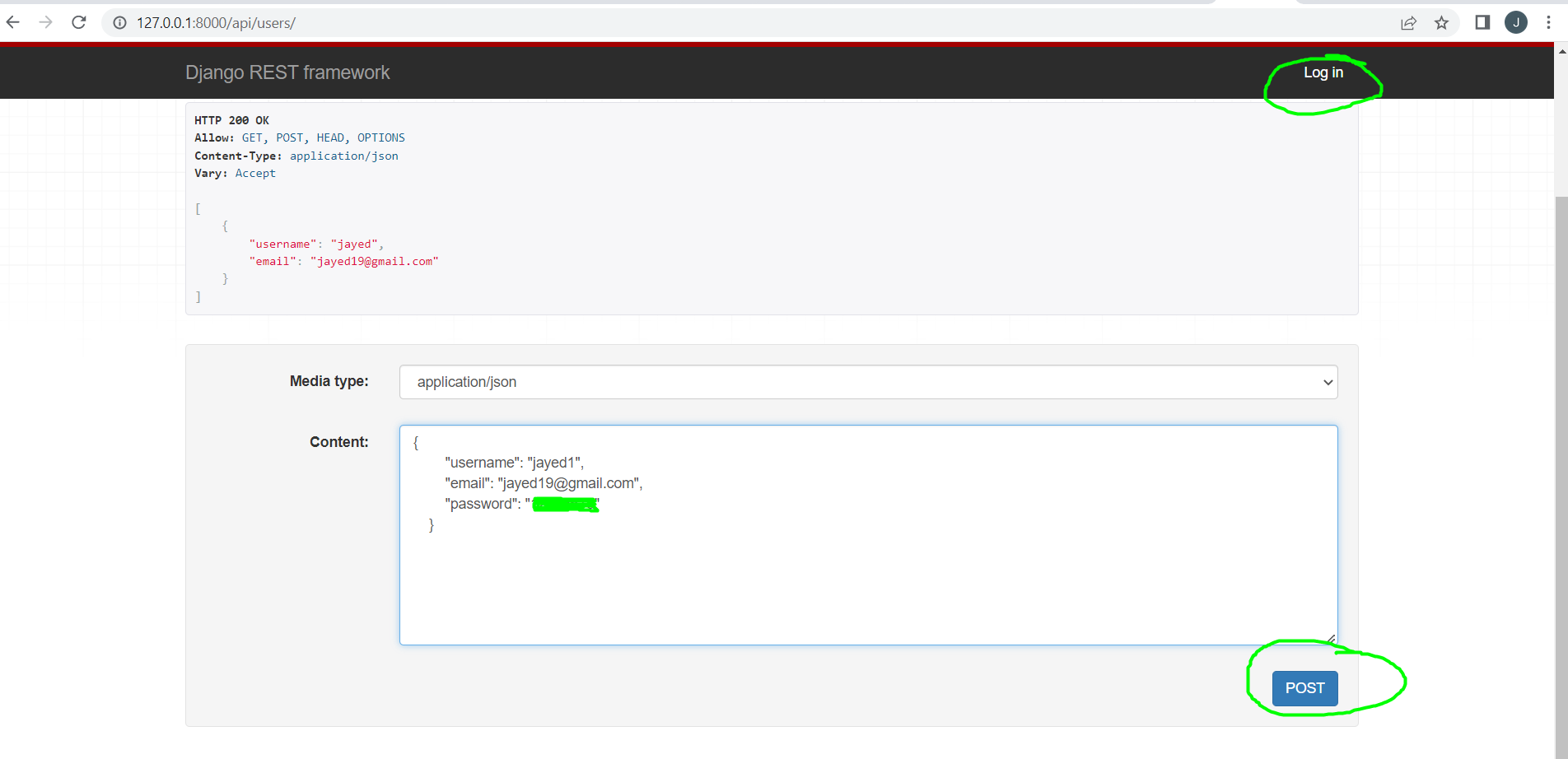
python manage.py runserver

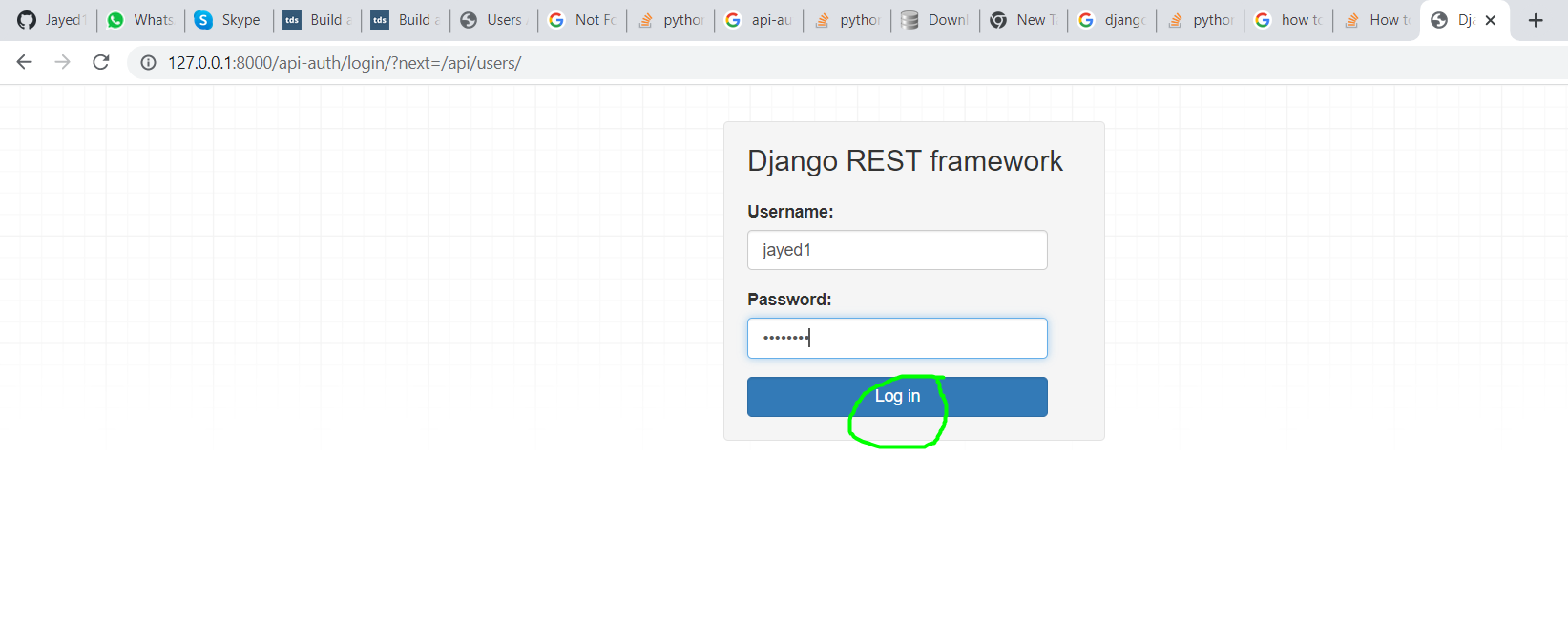


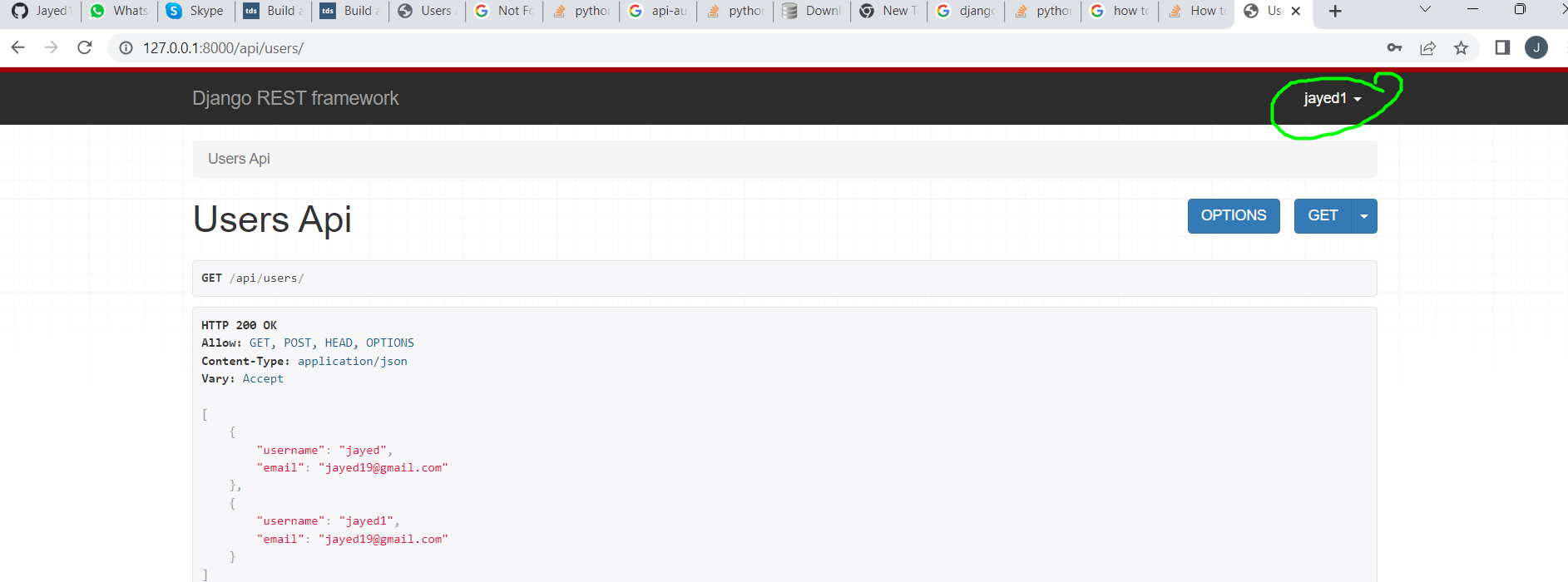
Step7: Check the API now: http://127.0.0.1:8000/api/users/



Step8: Create a User and check login with click on the 'Login' link of this page. Login will be failed if plain password saved in the database. To make the password encrypted sha256 we have already used a create function in the \*\*\*\*\*\*\*serializers.py\*\*\*\*\*\*\*\* file.







DB browse it from SQLite DB Browsers:

