\*) Api-> Application programing interface

app programs communicate each other

\*) Why API?

As I have read in one blog when you send data directly to the server

it will become heavy and server might crash with heavy data

but with API data will go smothly

\*) Multiple sources can communicate with one central server

Ex: you have built one rest api on one side

on the other side you have website with react and flutter with

mobile app, but you have one central api that all frontend can

get data from

This time api is like central service

\*) In normal website like whenyou visit wikipedia, it's going to

render the html css all these going to make up the web page

now in this case response you get from the server, is text

html, javascript

\*) So api will return only the data you want to render, that

data is typically returned json response, it's kind of like

dictionaries which has keys and values

\*) API will only return you the data:

send and receive

\*) When you pass dictionaries that time:

def test\_view(request):

data = {

'name':'john',

'age':23

}

return JsonResponse(data)

\*) But when you pass lists

def test\_view(request):

data = {

'name':'john',

'age':23

}

return JsonResponse(data, safe=False)

\*) Link this to url

http://127.0.0.1:8000/

When you press on this you are on root url

{"name": "john", "age": 23}

Called as json payload

Adding rest\_framework

from rest\_framework.response import  Response

from rest\_framework.views import  APIView

class TestView(APIView):

    def get(self, request, \*args, \*\*kwargs):

        data ={

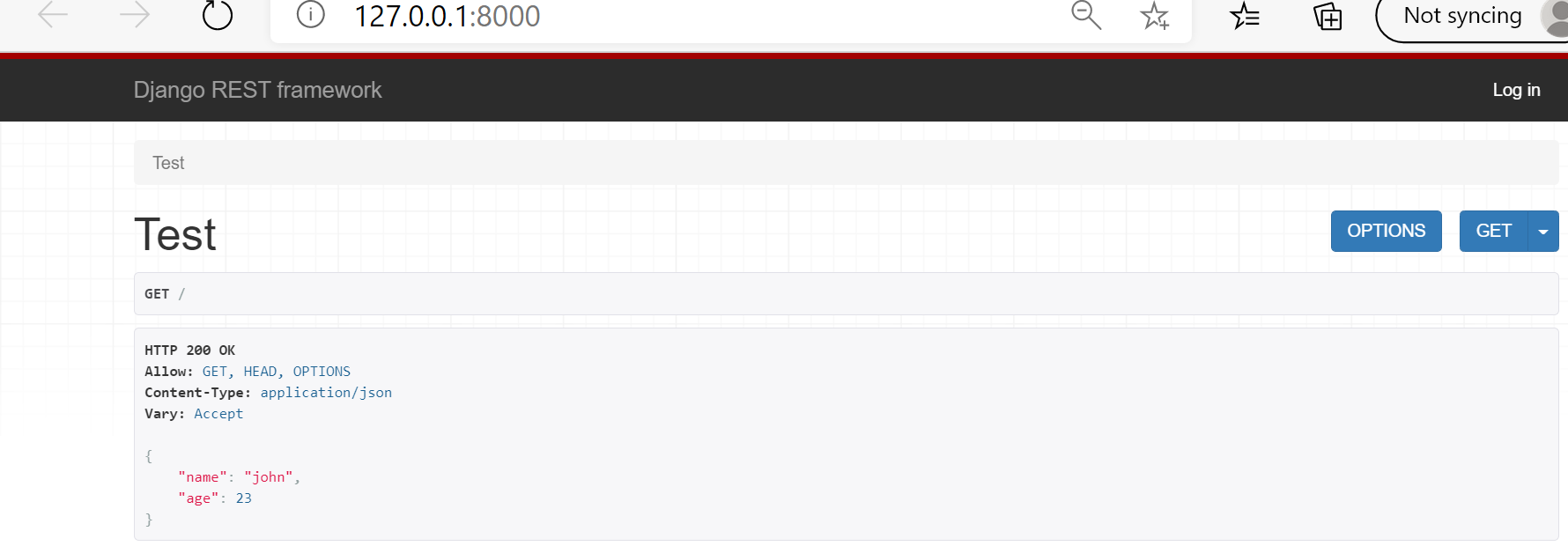
            'name':'john',

            'age':23

        }

        return Response(data)

After using above code and adding it in urls.py



When you press on get-> json->

You’ll get this: {"name":"john","age":23}