**Token Based Authentication for Django Rest Framework**

We gonna follow these steps

1. Installing Django
2. Making the Django Project
3. Installing Django Rest Framework
4. Setup the Login Function and api route
5. Testing the Login api route
6. Using the Token to access authenticated Api
7. Installing Django :-

pip install mssql-django

2. Installing Django Rest Framework :-

pip install djangorestframework

1. Making the Django Project :-

django-admin startproject MyProject

now, in **MyProject**

**Settings.py :-**

|  |  |
| --- | --- |
|  | INSTALLED\_APPS = [ |
|  | 'django.contrib.admin', |
|  | 'django.contrib.auth', |
|  | 'django.contrib.contenttypes', |
|  | 'django.contrib.sessions', |
|  | 'django.contrib.messages', |
|  | 'django.contrib.staticfiles', |
|  | 'rest\_framework', |
|  | 'rest\_framework.authtoken' |
|  | ] |

Please also the following configuration for the rest\_framework in the settings.py file

|  |  |
| --- | --- |
|  | REST\_FRAMEWORK = { |
|  | 'DEFAULT\_AUTHENTICATION\_CLASSES': ( |
|  | 'rest\_framework.authentication.TokenAuthentication', |
|  | ), |
|  | 'DEFAULT\_PERMISSION\_CLASSES': ( |
|  | 'rest\_framework.permissions.IsAuthenticated', ) |
|  | } |

The configuration is meant to change the default authentication and permission class of the rest framework.

# Setup the Login Function and api route

Now we write login function which returns the token associated with the user and make api route for the same using.

In the MyProject folder create a views.py

**MyProject**

This Views.py file to be create

**views.py :-**

Write code in views.py

from django.contrib.auth import authenticate  
from django.views.decorators.csrf import csrf\_exempt  
from rest\_framework.authtoken.models import Token  
from rest\_framework.decorators import api\_view, permission\_classes  
from rest\_framework.permissions import AllowAny  
from rest\_framework.status import (  
 HTTP\_400\_BAD\_REQUEST,  
 HTTP\_404\_NOT\_FOUND,  
 HTTP\_200\_OK  
)  
from rest\_framework.response import Response  
  
  
@csrf\_exempt  
@api\_view(["POST"])  
@permission\_classes((AllowAny,))  
def login(request):  
 username = request.data.get("username")  
 password = request.data.get("password")  
 if username is None or password is None:  
 return Response({'error': 'Please provide both username and password'},  
 status=HTTP\_400\_BAD\_REQUEST)  
 user = authenticate(username=username, password=password)  
 if not user:  
 return Response({'error': 'Invalid Credentials'},  
 status=HTTP\_404\_NOT\_FOUND)  
 token, \_ = Token.objects.get\_or\_create(user=user)  
 print(token.key)  
 return Response({'token': token.key},  
 status=HTTP\_200\_OK)

Now make api route for the same in the urls.py

**MyProject**

**urls.py**

from django.contrib import admin  
from django.urls import path  
from .views import login,sample\_api  
  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('api/login', login),

]

map the /api/login route with our function.

In Terminal

python manage.py makemigrations

python manage.py migrtae

**Create SuperUser**

Python manage.py createsuperuser

Now run the python app using runserver and let’s test the api using the postman. Postman is really great tool to test your rest api’s.

Python manage.py runserver

Now in URL

[Sample Api – Django REST framework](http://127.0.0.1:8000/api/sampleapi) ( <http://127.0.0.1:8000/api/sampleapi> )

Now see how we test the api in postman by using postman tool

We are getting the auth token in the response, this token is used in the headers for calling the authenticated api’s.

Let’s make a sample api which is authenticated and then try to access the api using the token.

Add the following things in the views.py file

@csrf\_exempt  
@api\_view(["GET"])  
def sample\_api(request):  
 data = {'Id': 1,'Name':'Roshan','DOB':'17/10/1993','City':'Pune'}  
 return Response(data, status=HTTP\_200\_OK)

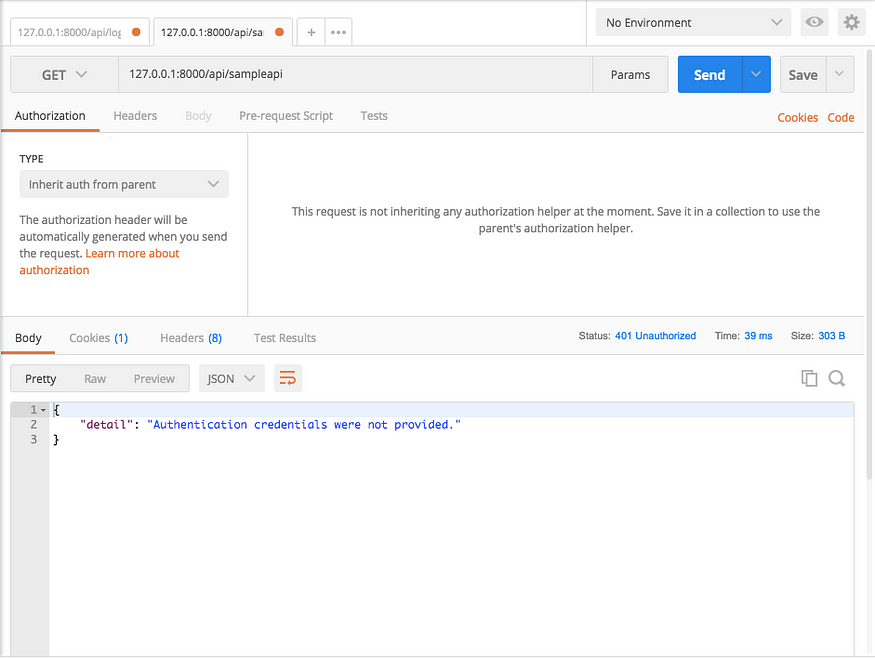
add the route for the same in the urls.py

from django.contrib import admin  
from django.urls import path  
from .views import login,sample\_api  
  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('api/login', login),  
 path('api/sampleapi', sample\_api)

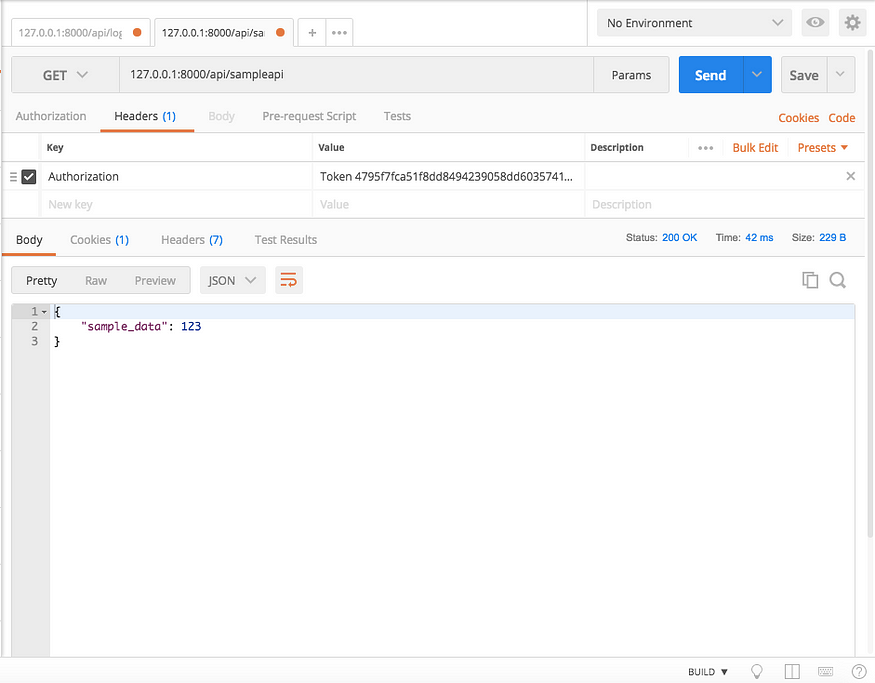
]

# Using the Token to access authenticated Api

Now we try to access the sample api without or with the token and see the response, first without token



We are blocked, now let’s do using the auth token



{

    "Id": 1,

    "Name": "Roshan",

    "DOB": "17/10/1993",

    "City": "Pune"

}

Yay, It works we are able to access the authenticated api.

So, you finally learned how to implement the token based authentication in the Django.