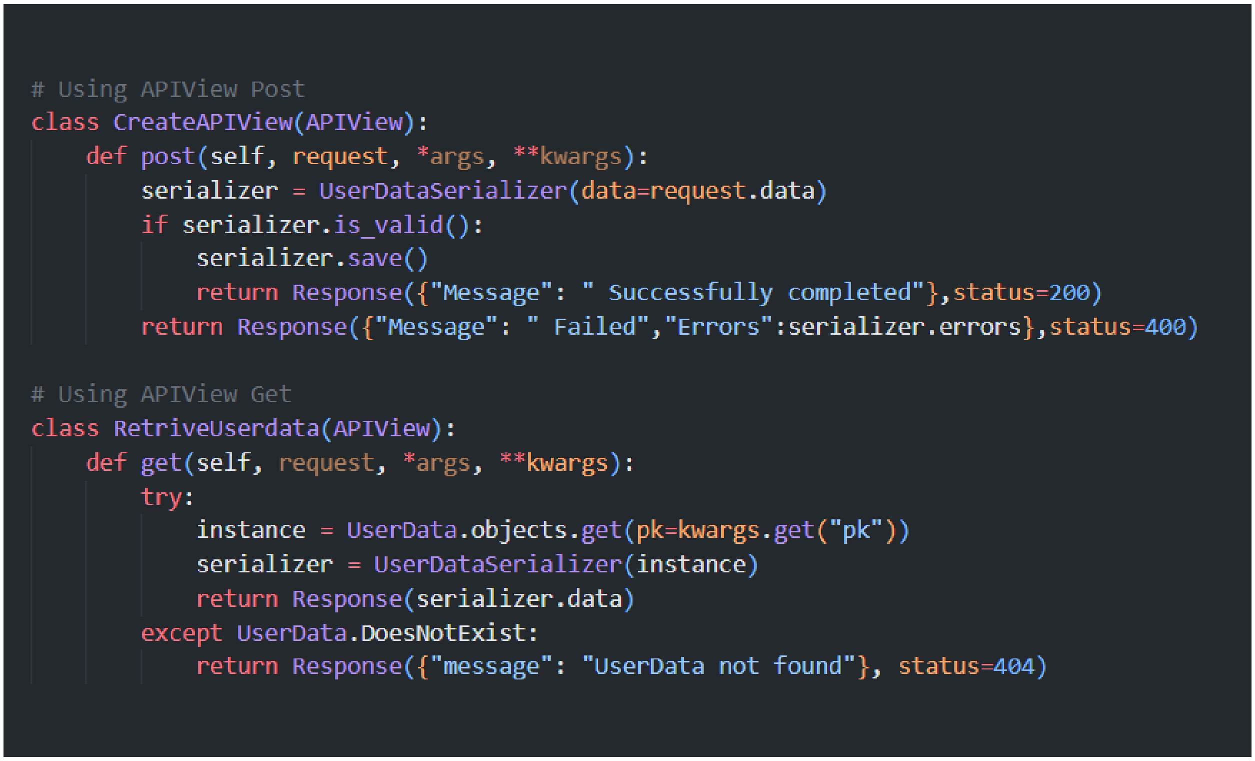
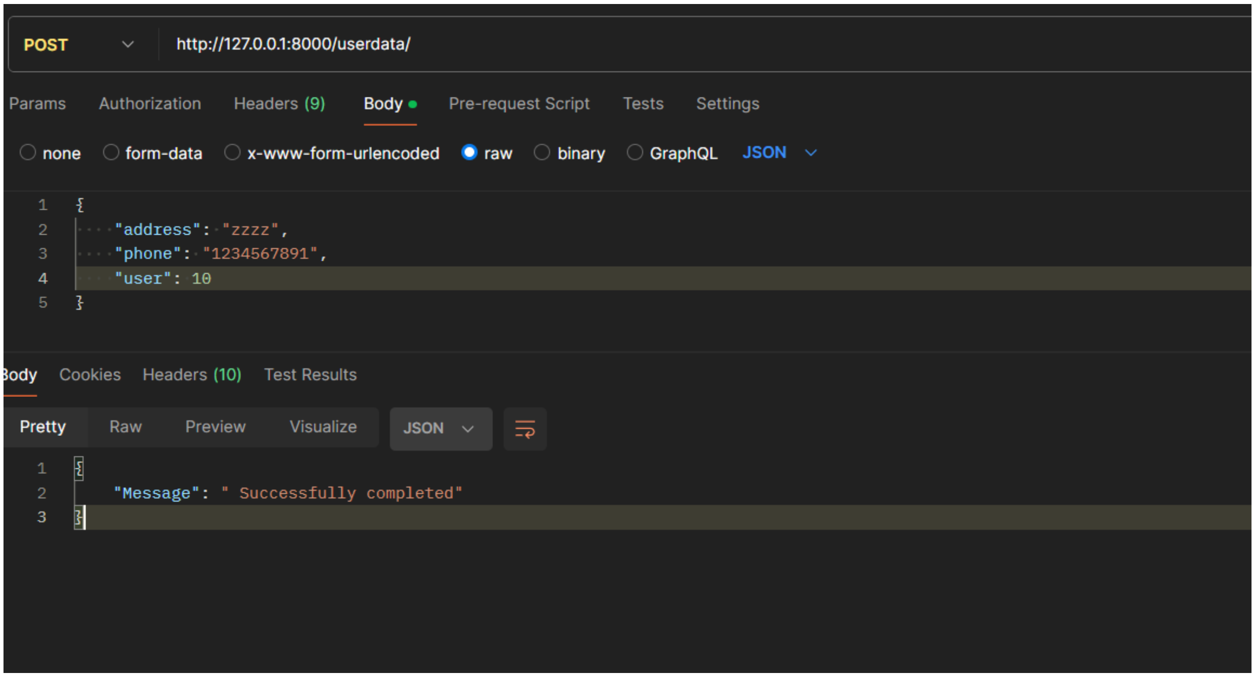
APIView



```python
# Using APIView Post
class CreateAPIView(APIView):
    def post(self, request, *args, **kwargs):
        serializer = UserDataSerializer(data=request.data)
        if serializer.is_valid():
            serializer.save()
            return Response({"Message": " Successfully completed"},status=200)
        return Response({"Message": " Failed","Errors":serializer.errors},status=400)

# Using APIView Get
class RetriveUserdata(APIView):
    def get(self, request, *args, **kwargs):
        try:
            instance = UserData.objects.get(pk=kwargs.get("pk"))
            serializer = UserDataSerializer(instance)
            return Response(serializer.data)
        except UserData.DoesNotExist:
            return Response({"message": "UserData not found"}, status=404)
```

APIView Post



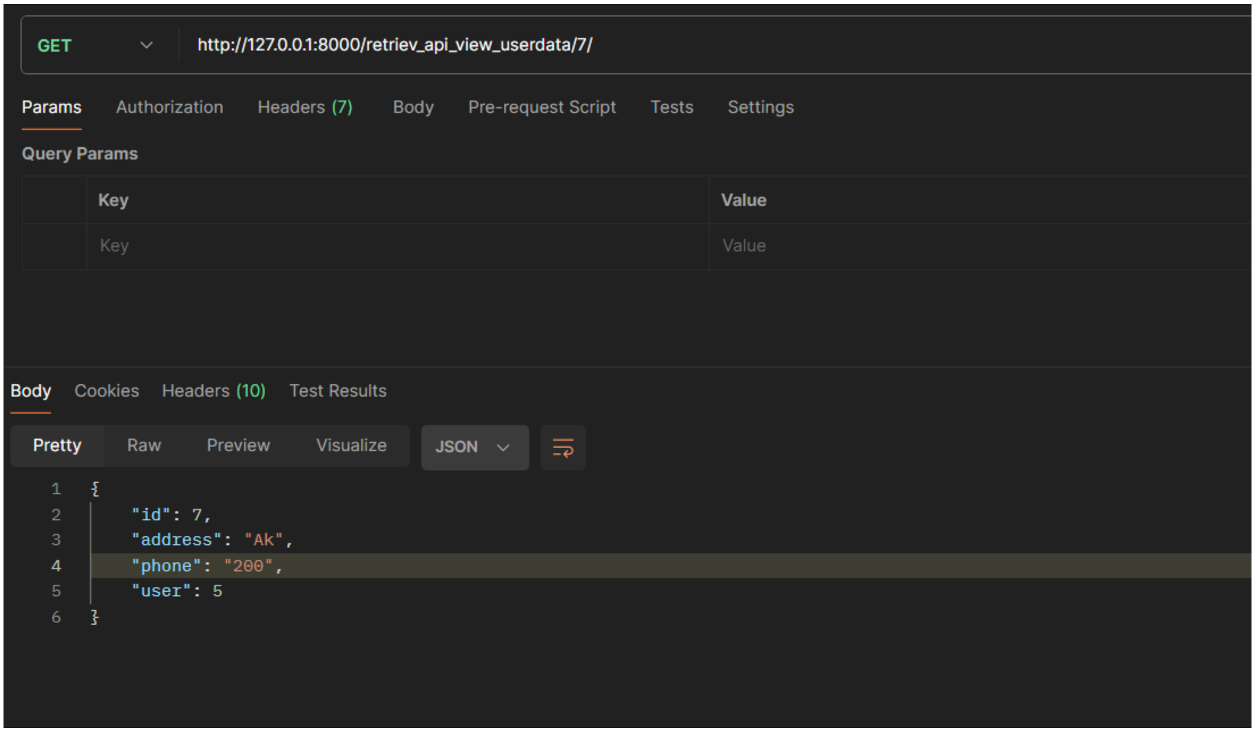
POST http://127.0.0.1:8000/userdata/

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```json
{
    "address": "zzzz",
    "phone": "1234567891",
    "user": 10
}
```

```json
{
    "Message": " Successfully completed"
}
```

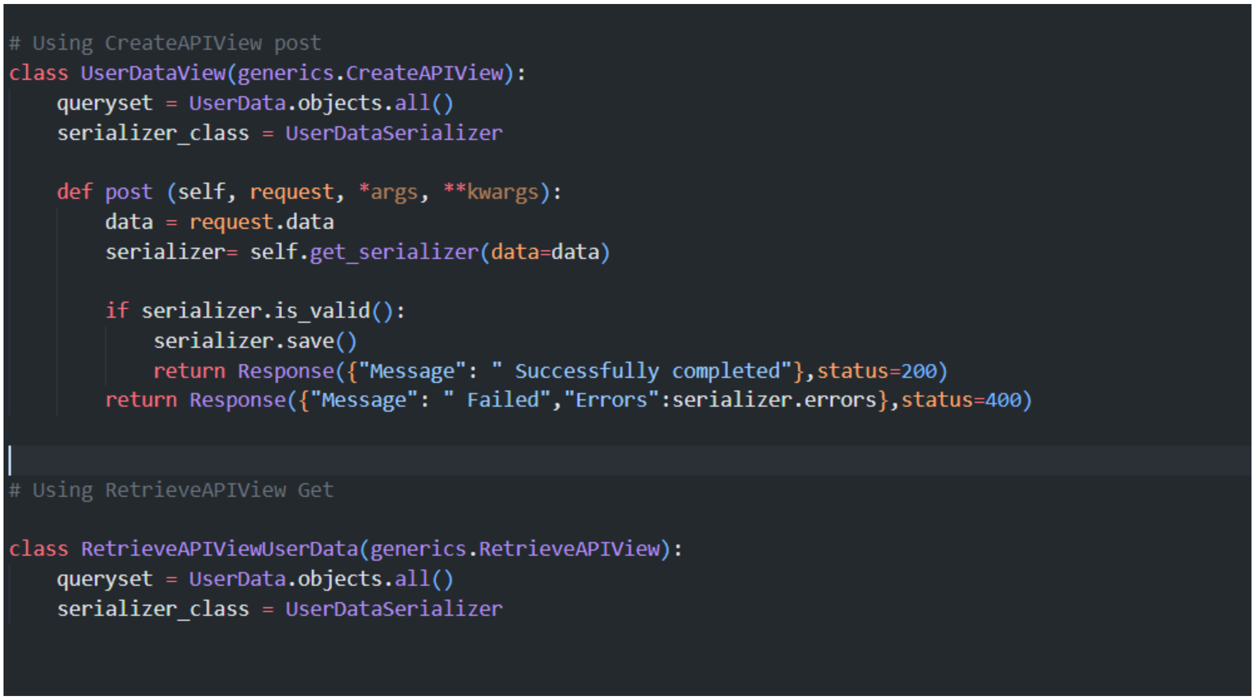
APIView Get



```
GET http://127.0.0.1:8000/retriev_api_view_userdata/7/
```

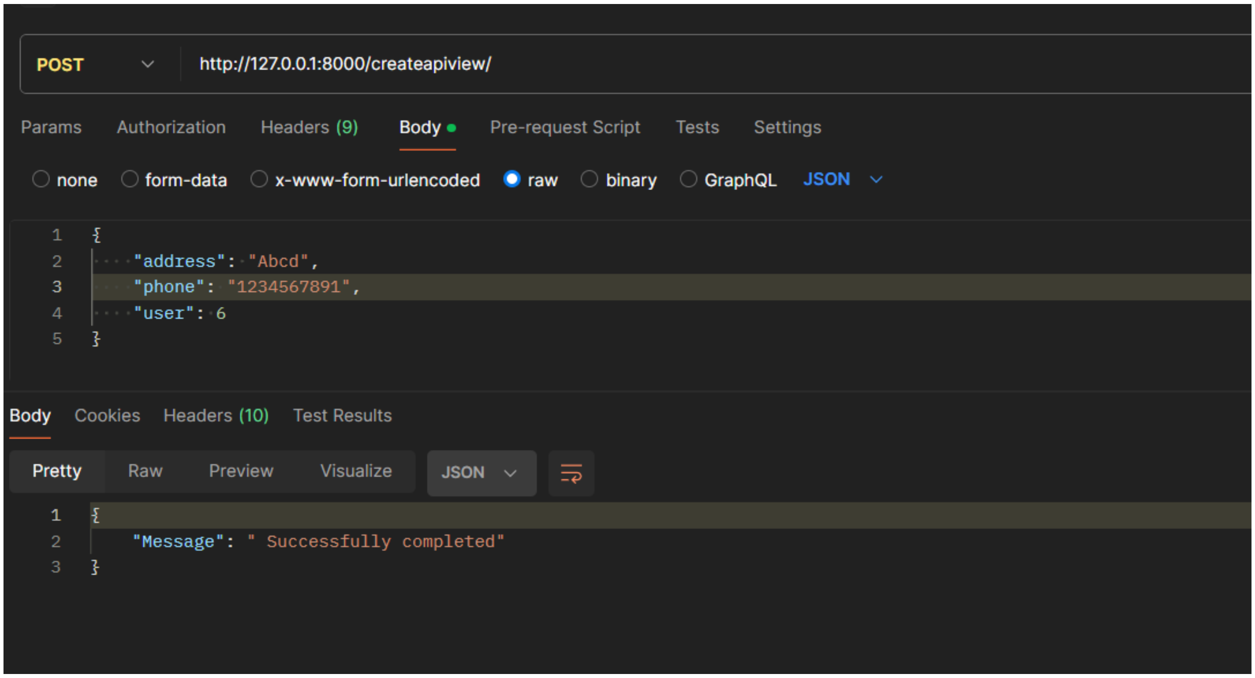
```
{
    "id": 7,
    "address": "Ak",
    "phone": "200",
    "user": 5
}
```

CreateAPIView



```python
# Using CreateAPIView post
class UserDataView(generics.CreateAPIView):
    queryset = UserData.objects.all()
    serializer_class = UserDataSerializer

    def post (self, request, *args, **kwargs):
        data = request.data
        serializer= self.get_serializer(data=data)

        if serializer.is_valid():
            serializer.save()
            return Response({"Message": " Successfully completed"},status=200)
        return Response({"Message": " Failed","Errors":serializer.errors},status=400)

# Using RetrieveAPIView Get

class RetrieveAPIViewUserData(generics.RetrieveAPIView):
    queryset = UserData.objects.all()
    serializer_class = UserDataSerializer
```

CreateAPIView Post

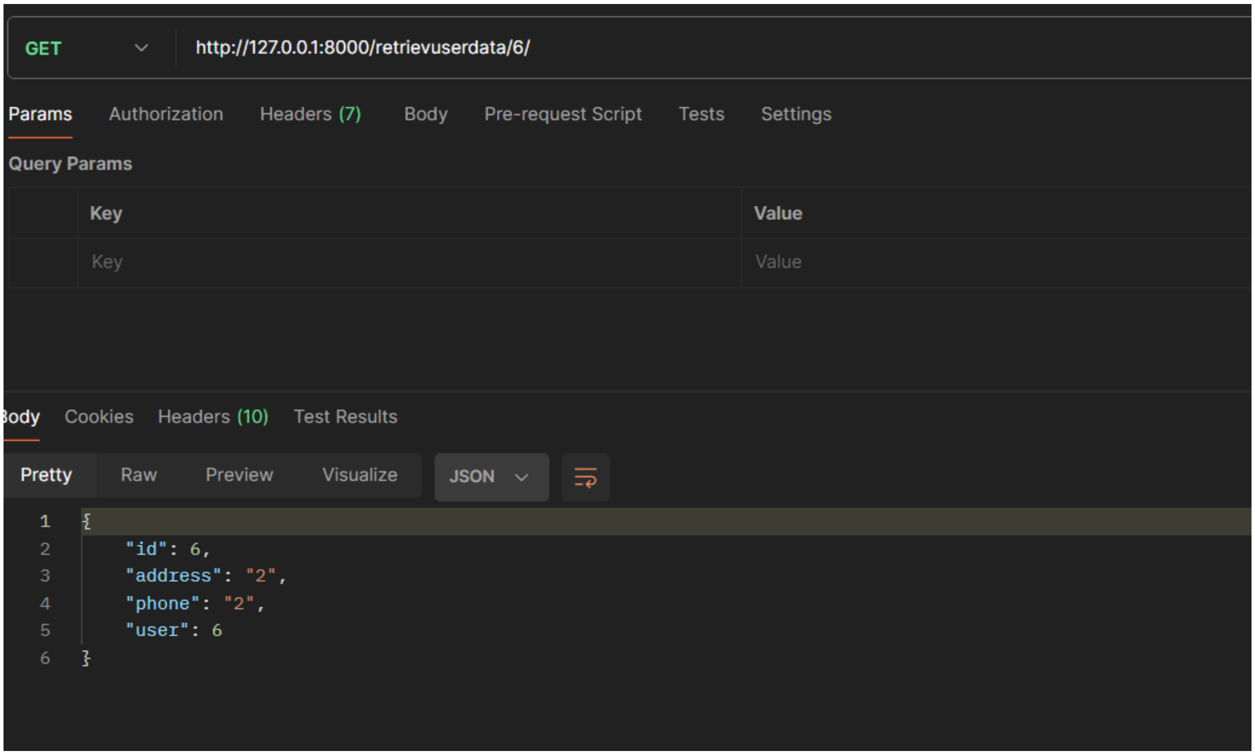


POST http://127.0.0.1:8000/createapiview/

```
{
    "address": "Abcd",
    "phone": "1234567891",
    "user": 6
}
```

```
{
    "Message": " Successfully completed"
}
```

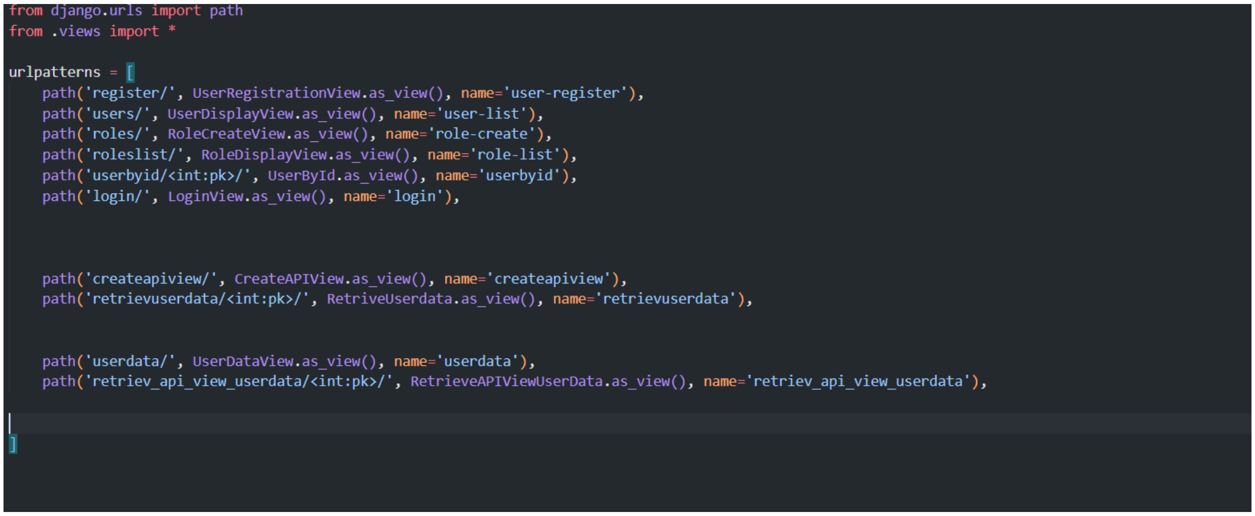
CreateAPIView Get



GET http://127.0.0.1:8000/retrievuserdata/6/

```
{
    "id": 6,
    "address": "2",
    "phone": "2",
    "user": 6
}
```

Url.py



```python
from django.urls import path
from .views import *

urlpatterns = [
    path('register/', UserRegistrationView.as_view(), name='user-register'),
    path('users/', UserDisplayView.as_view(), name='user-list'),
    path('roles/', RoleCreateView.as_view(), name='role-create'),
    path('roleslist/', RoleDisplayView.as_view(), name='role-list'),
    path('userbyid/<int:pk>/', UserById.as_view(), name='userbyid'),
    path('login/', LoginView.as_view(), name='login'),



    path('createapiview/', CreateAPIView.as_view(), name='createapiview'),
    path('retrievuserdata/<int:pk>/', RetriveUserdata.as_view(), name='retrievuserdata'),


    path('userdata/', UserDataView.as_view(), name='userdata'),
    path('retriev_api_view_userdata/<int:pk>/', RetrieveAPIViewUserData.as_view(), name='retriev_api_view_userdata'),

]
```

Project

https://github.com/Albin-Kuriachan/python_beniex/tree/main/17.04.2024