

# Voici mon code entier

---

il s'agit de te présenter le contenu des fichiers models.py, serializer.py, views.py

## models.py

```
from django.db import models
from django.contrib.auth.models import User
# Create your models here.
class UserProfile(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    phone = models.CharField(max_length= 9)
    image = models.ImageField(blank = True, null = True)
```

## views.py

```
from django.shortcuts import render
from rest_framework import viewsets
from django.contrib.auth.models import User
from .serializer import UserSerializer
from rest_framework.response import Response
import json

class UserPostSet(viewsets.ModelViewSet):
    """
    Cette class est publique
    """
    queryset = User.objects.all()
    serializer_class = UserSerializer
    def list(self, request):
        """
        mais j'ai mis all pour afficher tout le monde
        """
        #user = User.objects.get(username=request.user)
        user = User.objects.all()
        user_data = UserSerializer(user, many = True)
        return Response(user_data.data)
```

ceci est une solution qui a été proposé mais elle ne marche pas

```
class UserPostSet(viewsets.ModelViewSet):
    serializer_class = UserSerializer
    queryset = serializer_class.Meta.model.objects.all()

    def update(self, request, *args, **kwargs):
```

```
request.data.update({'user': json.loads(request.data.pop('user', None))})
return suer().update(request, *args, **kwargs)
```

voici le rendu lorsque je fais un PUT. Les autres méthodes fonctionnent correctement. En clair cela n'a pas résolu le problème.

**AttributeError at /userpost/1/**

This QueryDict instance is immutable

Request Method: PUT  
 Request URL: http://127.0.0.1:8000/userpost/1/  
 Django Version: 3.2.5  
 Exception Type: AttributeError  
 Exception Value: This QueryDict instance is immutable  
 Exception Location: C:\Users\Noutcha\AppData\Local\Programs\Python\Python39\lib\site-packages\django\http\request.py, line 510, in \_assert\_mutable  
 Python Executable: C:\Users\Noutcha\AppData\Local\Programs\Python\Python39\python.exe  
 Python Version: 3.9.6  
 Python Path: ['C:\\Users\\Noutcha\\Documents\\API\\userprofile',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39\\python39.zip',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39\\DLLs',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39\\lib',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39\\lib\\site-packages',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39\\lib\\site-packages\\win32',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39\\lib\\site-packages\\win32\\lib',  
 'C:\\Users\\Noutcha\\AppData\\Local\\Programs\\Python\\Python39\\lib\\site-packages\\Pythonwin']  
 Server time: Wed, 20 Oct 2021 15:50:48 +0000

Je précise que je n'ai pas encore utilisé le lien alternatif pour résoudre le problème que je rajoute ici [Premier lien alternatif](#) [second lien alternatif](#)

## serializer.py

```
from django.contrib.auth.models import User
from rest_framework import serializers
from .models import UserProfile
from drf_writable_nested import WritableNestedModelSerializer

class UserProfileSerializer(serializers.ModelSerializer):

    class Meta:
        model = UserProfile
        fields = ('phone', 'image')

class UserSerializer(WritableNestedModelSerializer):
    """
    avec cette méthode la mise à jour ne passe pas
    """
    profile = UserProfileSerializer(source = 'userprofile')
    #profile = UserProfileSerializer(many=True, required = False, read_only=True)
    class Meta:
        model = User
        fields = ('id', 'username', 'first_name', 'last_name', 'email', 'profile')
```

Note: le code mis en commentaire est juste une autre méthode que j'essayais qui n'a pas marché.