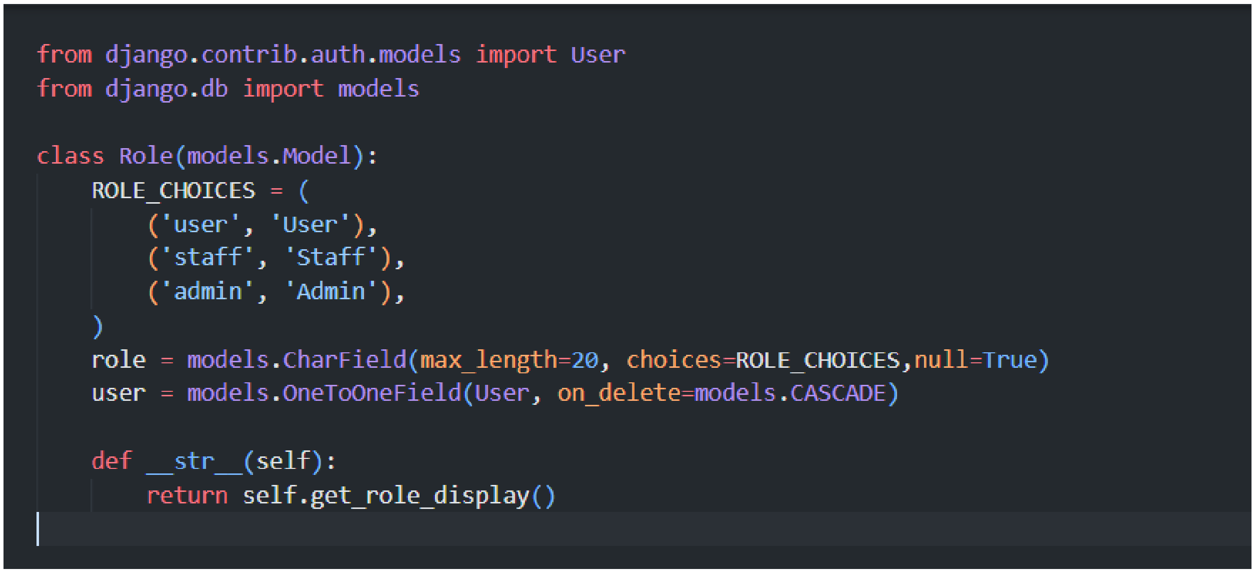
3) Create a model role and link with the user model

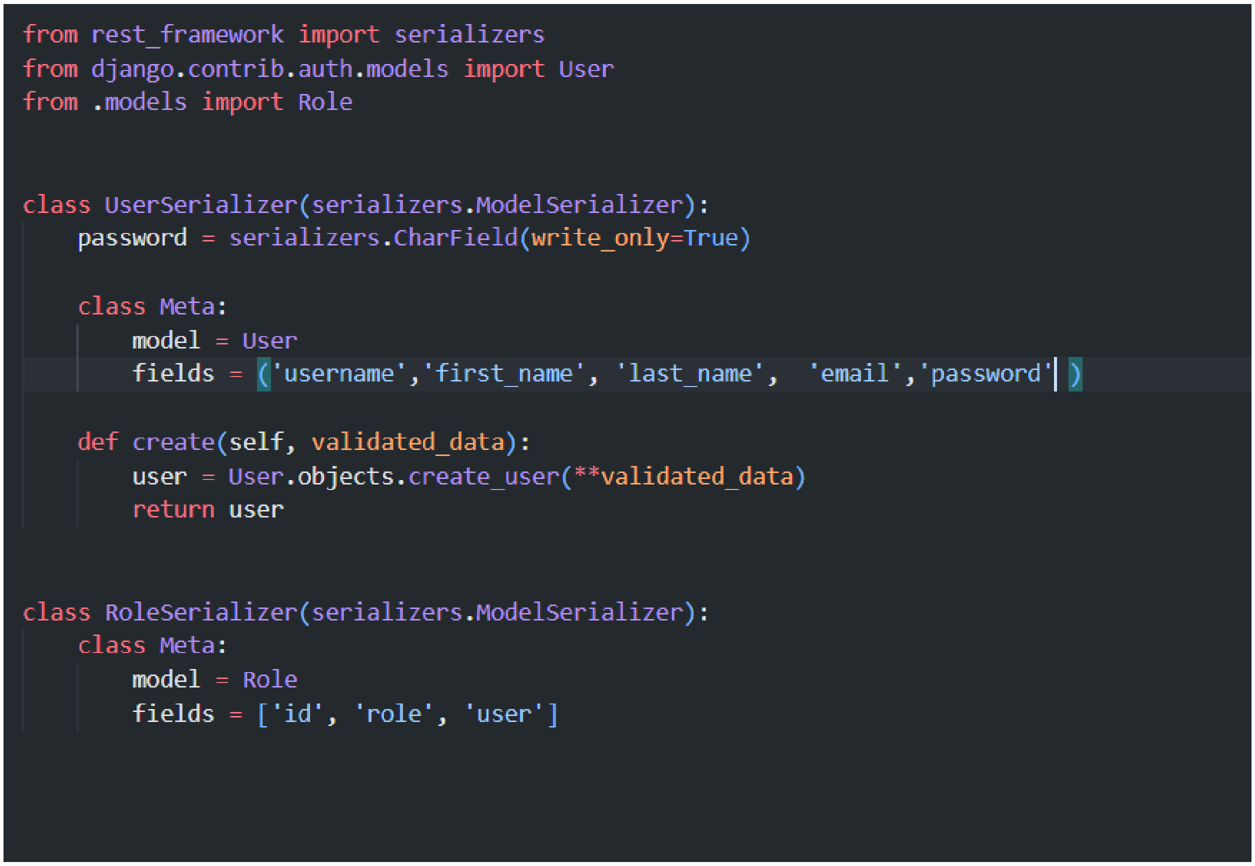
4) Create an Api to register user

models.py



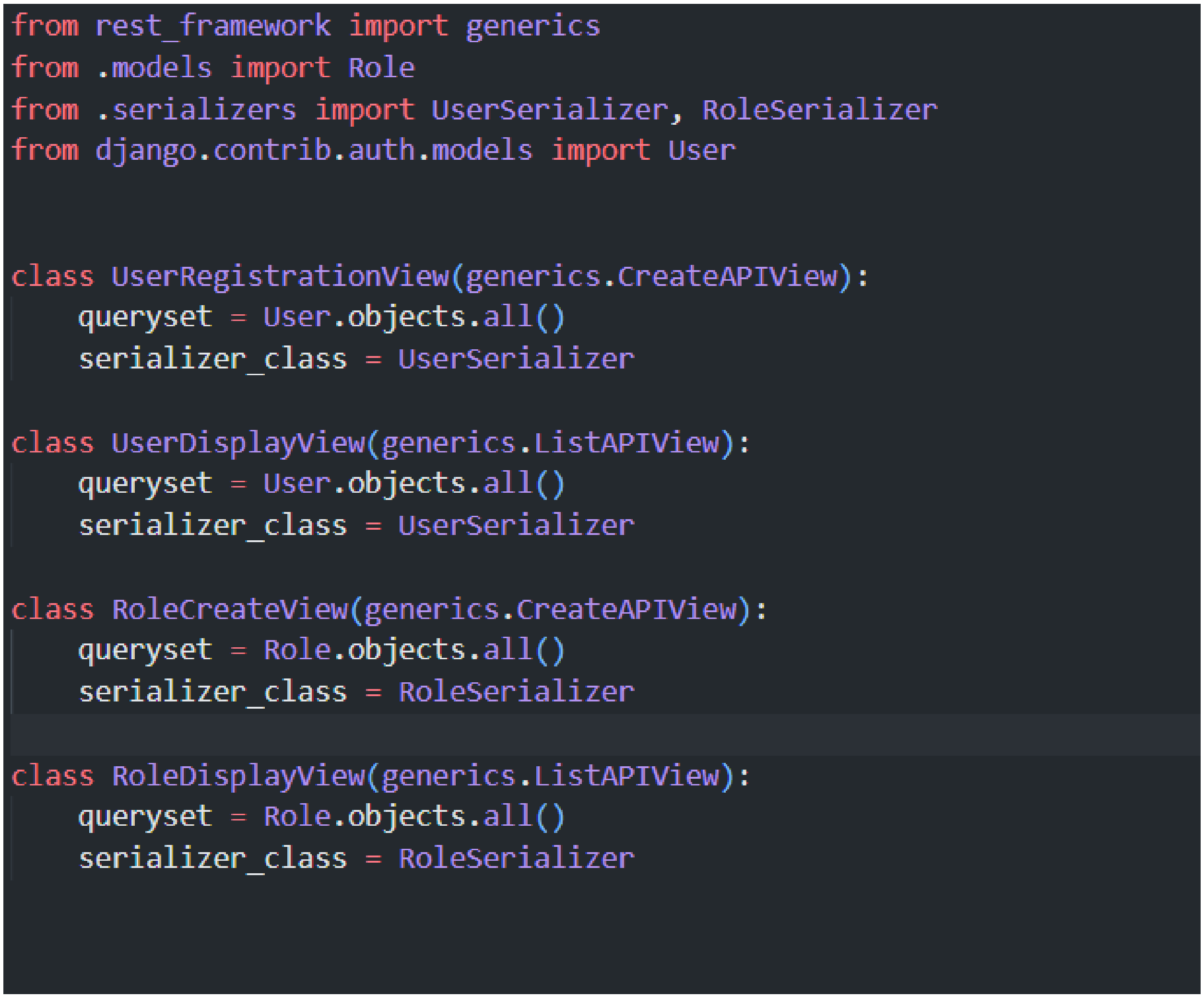
```python
from django.contrib.auth.models import User
from django.db import models

class Role(models.Model):
    ROLE_CHOICES = (
        ('user', 'User'),
        ('staff', 'Staff'),
        ('admin', 'Admin'),
    )
    role = models.CharField(max_length=20, choices=ROLE_CHOICES,null=True)
    user = models.OneToOneField(User, on_delete=models.CASCADE)

    def __str__(self):
        return self.get_role_display()
```

Serializers.py



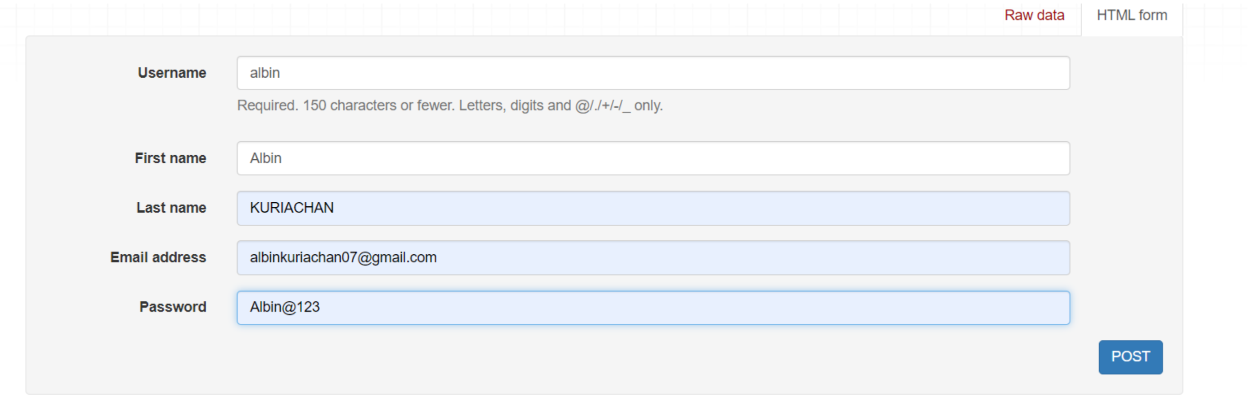
```python
from rest_framework import serializers
from django.contrib.auth.models import User
from .models import Role


class UserSerializer(serializers.ModelSerializer):
    password = serializers.CharField(write_only=True)

    class Meta:
        model = User
        fields = ('username','first_name', 'last_name',  'email','password' )

    def create(self, validated_data):
        user = User.objects.create_user(**validated_data)
        return user


class RoleSerializer(serializers.ModelSerializer):
    class Meta:
        model = Role
        fields = ['id', 'role', 'user']
```

Views.py



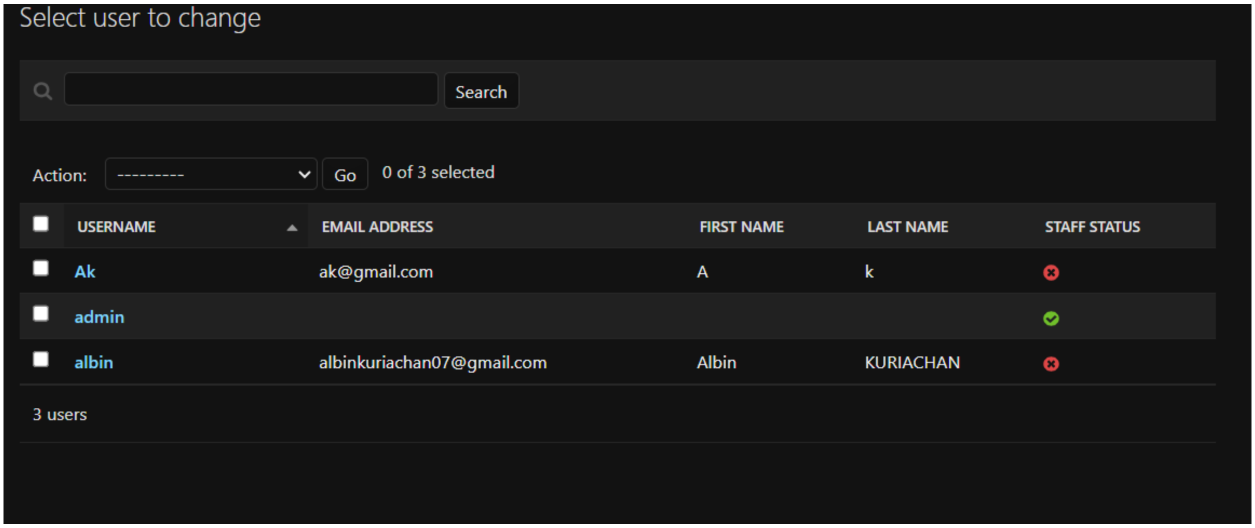
```
from rest_framework import generics
from .models import Role
from .serializers import UserSerializer, RoleSerializer
from django.contrib.auth.models import User


class UserRegistrationView(generics.CreateAPIView):
    queryset = User.objects.all()
    serializer_class = UserSerializer

class UserDisplayView(generics.ListAPIView):
    queryset = User.objects.all()
    serializer_class = UserSerializer

class RoleCreateView(generics.CreateAPIView):
    queryset = Role.objects.all()
    serializer_class = RoleSerializer

class RoleDisplayView(generics.ListAPIView):
    queryset = Role.objects.all()
    serializer_class = RoleSerializer
```

User Creation



Username	albin Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.
First name	Albin
Last name	KURIACHAN
Email address	albinkuriachan07@gmail.com
Password	Albin@123

Admin Pannel –User

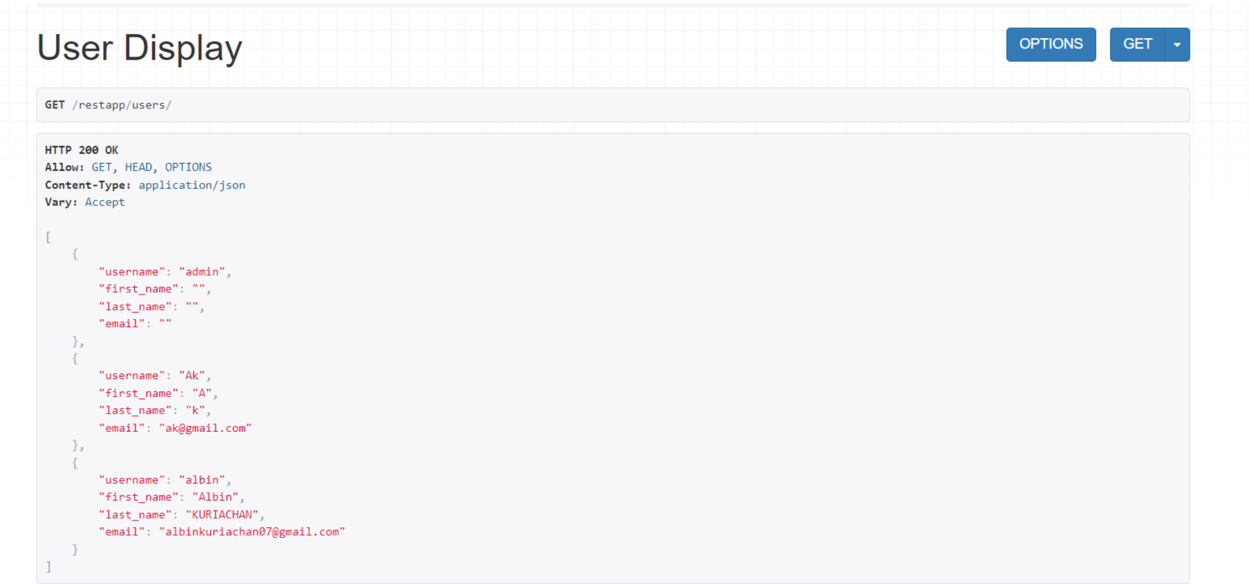


Select user to change

Action: --------- Go 0 of 3 selected

□	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
□	Ak	ak@gmail.com	A	k	✗
□	admin				✓
□	albin	albinkuriachan07@gmail.com	Albin	KURIACHAN	✗

3 users

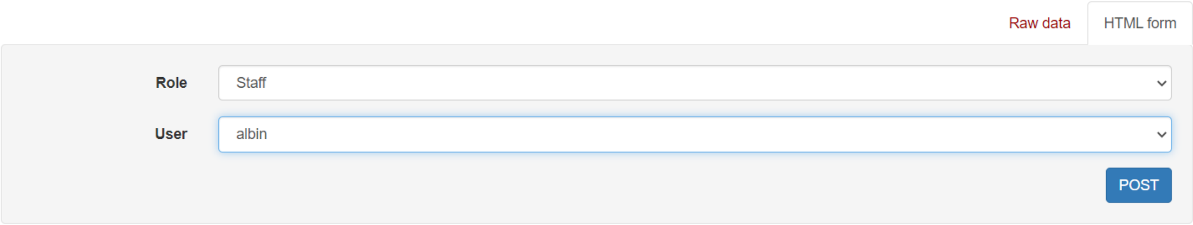


User Display

```
GET /restapp/users/
```

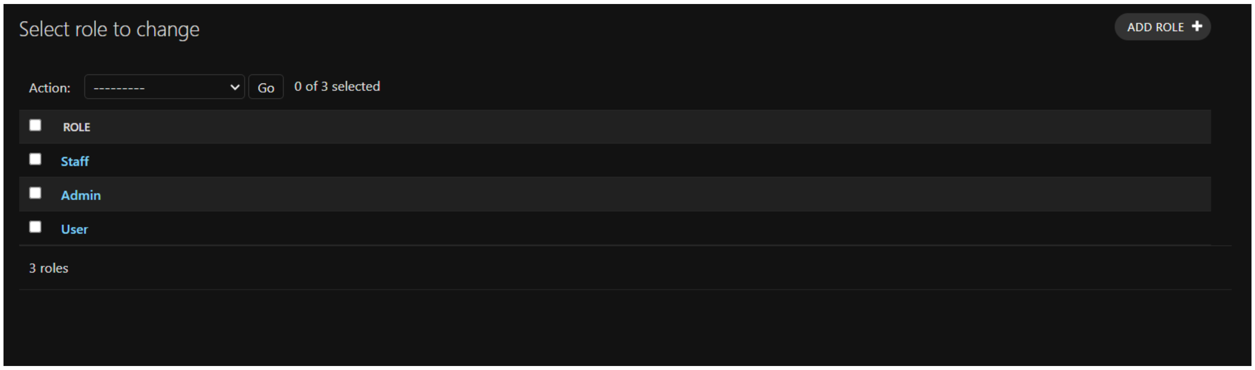
```
HTTP 200 OK
Allow: GET, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[
    {
        "username": "admin",
        "first_name": "",
        "last_name": "",
        "email": ""
    },
    {
        "username": "Ak",
        "first_name": "A",
        "last_name": "k",
        "email": "ak@gmail.com"
    },
    {
        "username": "albin",
        "first_name": "Albin",
        "last_name": "KURIACHAN",
        "email": "albinkuriachan07@gmail.com"
    }
]
```

Post-Role



Role	Staff
User	albin

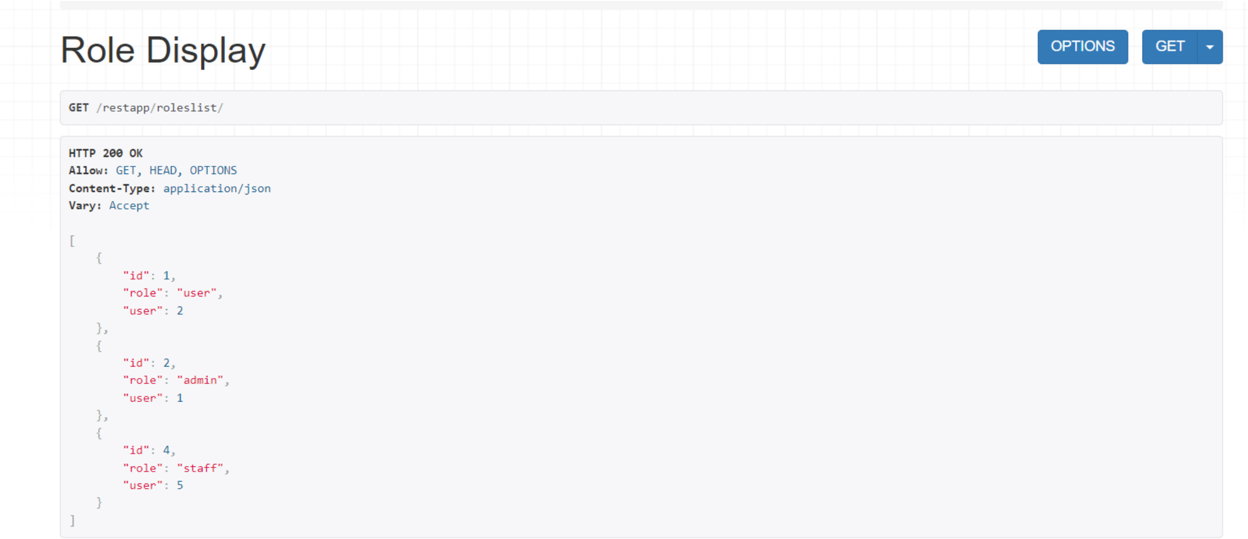
Admin Pannel – Role



0 of 3 selected

□	ROLE
□	Staff
□	Admin
□	User

3 roles



```
GET /restapp/roleslist/
```

```
HTTP 200 OK
Allow: GET, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[
    {
        "id": 1,
        "role": "user",
        "user": 2
    },
    {
        "id": 2,
        "role": "admin",
        "user": 1
    },
    {
        "id": 4,
        "role": "staff",
        "user": 5
    }
]
```