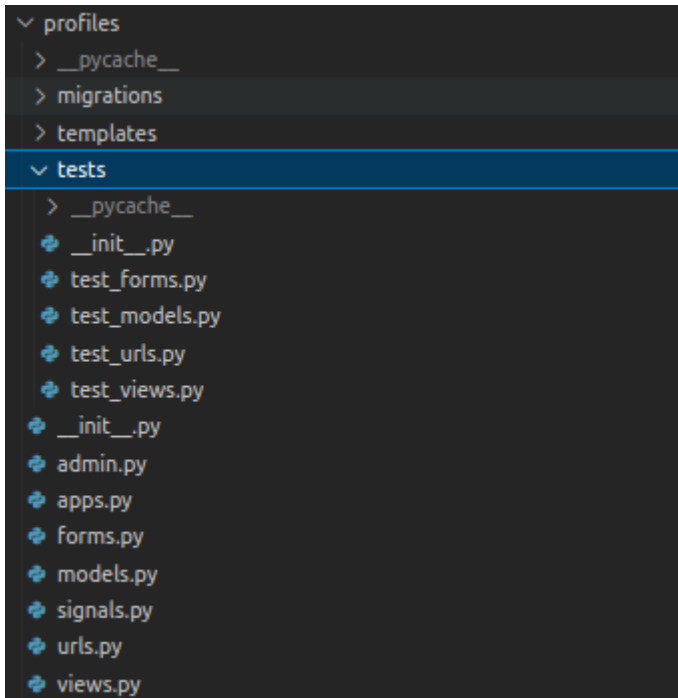


profiles



test_forms.py

All right

Save ▾

Share

Your code

```
1  """Tests for the forms of the profile app."""
2  from django.test import TestCase
3  from django.contrib.auth.models import User
4  from profiles.forms import ProfileForm, AddressForm
5  from profiles.models import Role, Profile, Address
6
7
8  class TestForms(TestCase):
9      """Tests for the forms of the profile app."""
10     def setUp(self):
11         """Set up the test."""
12         self.role1 = Role.objects.create(
13             name='Customer',
14             description='Customer'
15         )
```

test_models.py

All right

Save ▾ Share

Your code

```
1  """Tests for the models of the profiles app."""
2  from django.test import TestCase
3  from django.contrib.auth.models import User
4  from profiles.models import Role, Profile, Address
5  import cloudinary
6  import cloudinary.uploader
7  from datetime import date
8
9
10 class TestModels(TestCase):
11     """Tests for the Roles model."""
12
13     def setUp(self):
14         """Set up the test."""
15         self.role1 = Role.objects.create(
```

test_urls.py

All right

Save ▾

Share

Your code

```
1  """Test Profiles URLs."""
2  from django.test import SimpleTestCase
3  from django.urls import reverse, resolve
4  from profiles.views import (
5      UserProfileView,
6      EditAvatarAjaxView,
7      ResetAvatarView,
8      EditUserProfileView,
9      DeleteProfileView,
10     AddressesView,
11     AddAddressView,
12     EditAddressView,
13     DeleteAddressView,
14     ChangePrimaryAddressView,
15 )
```

Check again

test_views.py

All right

Save ▾

Share

Your code

```
1  """Tests for the views of the profile app."""
2  from django.test import TestCase, Client
3  from django.urls import reverse
4  from django.contrib.auth.models import User
5  from profiles.models import Role, Profile, Address
6  from django.contrib.auth.hashers import make_password
7
8
9  class TestViews(TestCase):
10     """Tests for the Roles model."""
11
12     def setUp(self):
13         """Set up the test."""
14
15         self.role1 = Role.objects.create(
```

admin.py

All right

Save ▾ Share

Your code

```
1 """Admin for the profile app."""
2 from django.contrib import admin
3 from .models import Role, Profile, Address
4
5
6 @admin.register(Role)
7 class RoleAdmin(admin.ModelAdmin):
8     """Admin for the Role model."""
9     list_display = ('name', 'description')
10    list_filter = ('name',)
11    search_fields = ('name',)
12
13
14 @admin.register(Profile)
15 class ProfileAdmin(admin.ModelAdmin):
```

apps.py

All right

Save ▾ Share

Your code

```
1 """Apps for the profiles app."""
2 from django.apps import AppConfig
3 from django.db.models.signals import post_save
4
5
6 class ProfilesConfig(AppConfig):
7     """Config for the profiles app."""
8     default_auto_field = 'django.db.models.BigAutoField'
9     name = 'profiles'
10
11     def ready(self):
12         from profiles.signals import create_profile
13         from django.contrib.auth.models import User
14         post_save.connect(create_profile, sender=User)
15         from profiles.signals import save_profile
```

forms.py

All right

Save ▾

Share

Your code

```
1  """Forms for the profiles app."""
2  from django import forms
3  from .models import Profile, Address
4
5
6  class ProfileForm(forms.ModelForm):
7      """Form for the Profile model."""
8      class Meta:
9          model = Profile
10         fields = ['first_name', 'last_name', 'birthday', 'subscription']
11         widgets = {
12             'first_name': forms.TextInput(
13                 attrs={
14                     'class': 'form-control',
15                     'title': 'max_length=50',
```


models.py

All right

Save ▾

Share

Your code

```
1  """Models for the profiles app."""
2  from django.db import models
3  from django.contrib.auth.models import User
4  from cloudinary.models import CloudinaryField
5  from datetime import date
6
7
8  class Role(models.Model):
9      """Model for the roles."""
10     name = models.CharField(
11         max_length=50,
12         unique=True,
13         blank=False,
14         null=False,
15         verbose_name='Role name',
```

Check again

signals.py

All right

Save ▾

Share

Your code

```
1  """Signals for the profiles app."""
2  from django.db.models.signals import post_save
3  from django.dispatch import receiver
4  from profiles.models import Profile
5  from django.contrib.auth.models import User
6  from wishlist.models import Wishlist
7
8
9  @receiver(post_save, sender=User)
10 def create_profile(sender, instance, created, **kwargs):
11     if created:
12         Profile.objects.create(user=instance)
13         Wishlist.objects.create(user=instance)
14
15
```

urls.py

All right

Save ▾

Share

Your code

```
1  """Urls for the profile app."""
2  from django.urls import path
3  from .views import (
4      UserProfileView,
5      EditAvatarAjaxView,
6      ResetAvatarView,
7      EditUserProfileView,
8      DeleteProfileView,
9      AddressesView,
10     AddAddressView,
11     EditAddressView,
12     DeleteAddressView,
13     ChangePrimaryAddressView,
14 )
15
```

views.py

All right

Save ▾ Share

Your code

```
1  """Views for the profiles app."""
2  from django.views import View
3  from django.shortcuts import render, get_object_or_404, redirect, reverse
4  from django.http import JsonResponse, HttpResponseRedirect
5  from .models import Role, Profile, Address
6  from .forms import ProfileForm, AddressForm
7  from django.contrib.auth.forms import PasswordChangeForm
8  from django.contrib.auth import update_session_auth_hash
9
10
11 class UserProfileView(View):
12     """View for the profile page."""
13     def get(self, request, *args, **kwargs):
14         """Get the profile page."""
15         if request.user.is_authenticated:
```