

admin.py

All right

Save ▼

Share

Your code

```
1 from django.contrib import admin
2
3 # Register your models here.
4
```

apps.py

All right

Save ▼

Share

Your code

```
1 from django.apps import AppConfig
2
3
4 class ApiConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'api'
7
```

models.py

All right

Save ▼

Share

Your code

```
1 from django.db import models
2
3 # Create your models here.
4
```

urls.py

All right

[Save ▼](#)[Share](#)

Your code

```
1 from django.urls import path
2 from .views import CountriesService
3 from authentication.views import (
4     LogoutAPIView,
5     RegisterAPIView,
6     LoginAPIView,
7     UserAPIView,
8     RefreshTokenAPIView,
9     ForgotPasswordAPIView,
```

views.py

All right

Save ▼

Share

Your code

```
1 from django.shortcuts import render
2 from rest_framework.views import APIView
3 from rest_framework.response import Response
4 from rest_framework import status
5 from django_countries.data import COUNTRIES
6
7
8 # Create your views here.
9
```

✓ authentication

> migrations

> templates

🔗 __init__.py

🔗 admin.py

🔗 apps.py

🔗 authentication.py

🔗 exceptions.py

🔗 models.py

🔗 serializers.py

🔗 tests.py

🔗 views.py

admin.py

All right

Save ▾

Share

Your code

```
1 from django.contrib import admin
2 from .models import ForgotPasswordToken
3
4 # Register your models here.
5
6
7 @admin.register(ForgotPasswordToken)
8 class ForgotPasswordTokenAdmin(admin.ModelAdmin):
9     list_display = ("email", "token")
10     list_filter = ("email",)
11     search_fields = ("email",)
12
```


apps.py

All right

Save ▼

Share

Your code

```
1 from django.apps import AppConfig
2
3
4 class AuthenticationConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'authentication'
7
```

authentication.py

All right

Save ▾

Share

Your code

```
1 import jwt
2 import datetime
3 from rest_framework import status, exceptions
4 from rest_framework.response import Response
5 from rest_framework.authentication import BaseAuthentication
6 from profiles.models import User
7
8
9 class JWTAuthentication(BaseAuthentication):
10     """Custom JWT authentication class for DRF"""
```

exceptions.py

All right

Save ▾ Share

Your code

```
1 from rest_framework.views import exception_handler
2
3
4 def status_code_handler(exc, context):
5     response = exception_handler(exc, context)
6     # print(response)
7
8     if (
9         response is not None and
10         response.status_code == 403 and
```

models.py

All right

Save ▼

Share

Your code

```
1 from django.db import models
2
3
4 class ForgotPasswordToken(models.Model):
5     email = models.EmailField()
6     token = models.CharField(max_length=255)
7
```

serializers.py

All right

Save ▼

Share

Your code

```
1 from rest_framework.serializers import ModelSerializer
2 from profiles.models import User
3
4
5 class UserSerializer(ModelSerializer):
6     """User Serializer"""
7
8     class Meta:
9         model = User
10        fields = [
11            "id",
12            "first_name",
```

views.py

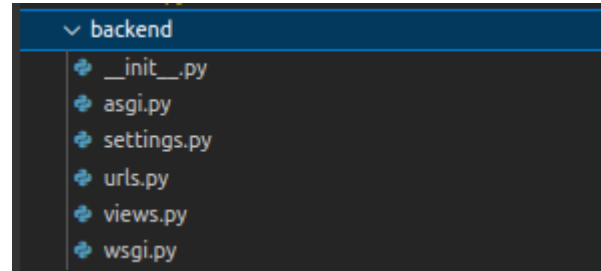
All right

Save ▾

Share

Your code

```
1 import datetime
2 import random
3 import string
4 from rest_framework.views import APIView
5 from rest_framework.response import Response
6 from rest_framework import status
7 from .serializers import UserSerializer
8 from profiles.models import User
9 from .authentication import (
10     JWTAuthentication,
```



asgi.py

All right

Save ▼

Share

Your code

```
1 """
2 ASGI config for backend project.
3
4 It exposes the ASGI callable as a module-level variable named ``application``.
5
6 For more information on this file, see
7 https://docs.djangoproject.com/en/4.0/howto/deployment/asgi/
8 """
9
10 import os
```


urls.py

All right

Save ▾ Share

Your code

```
1 """backend URL Configuration
2
3 ▾ The `urlpatterns` list routes URLs to views. For more information please see:
4     https://docs.djangoproject.com/en/4.0/topics/http/urls/
5 ▾ Examples:
6     Function views
7         1. Add an import:  from my_app import views
8         2. Add a URL to urlpatterns:  path('', views.home, name='home')
9     Class-based views
10        1. Add an import:  from other_app.views import Home
```

views.py

All right

Save ▼

Share

Your code

```
1 from django.http import HttpResponseRedirect
2 from django.conf import settings
3
4 base_url = settings.BASE_URL
5
6
7 def handler404(request, exception):
8     return HttpResponseRedirect(base_url + "/404")
9
10
11 def handler500(request):
12     return HttpResponseRedirect(base_url + "/500")
13
```

wsgi.py

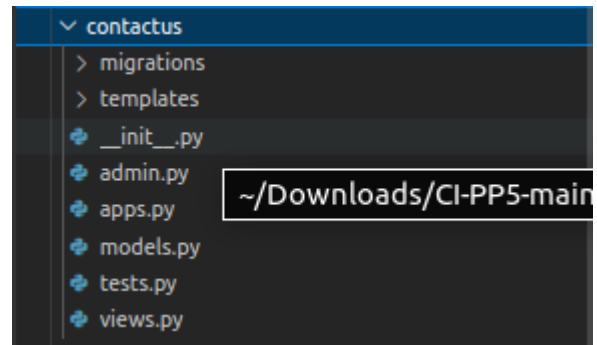
All right

Save ▼

Share

Your code

```
1 """
2 WSGI config for backend project.
3
4 It exposes the WSGI callable as a module-level variable named ``application``.
5
6 For more information on this file, see
7 https://docs.djangoproject.com/en/4.0/howto/deployment/wsgi/
8 """
9
10 import os
11
```



admin.py

All right

Save ▼

Share

Your code

```
1 from django.contrib import admin
2 from .models import ContactUs, NewsletterSubscription
3
4 # Register your models here.
5
6
7 @admin.register(ContactUs)
8 class ContactUsAdmin(admin.ModelAdmin):
9     list_display = ("name", "email", "subject", "message")
10    list_filter = ("name", "email", "subject", "message")
11    search_fields = ("name", "email", "subject", "message")
12
```

apps.py

All right

Save ▾

Share

Your code

```
1 from django.apps import AppConfig
2
3
4 class ContactusConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'contactus'
7
```

models.py

All right

Save ▼

Share

Your code

```
1 from django.db import models
2 from django.core.mail import send_mail
3 from django.template.loader import render_to_string
4 from django.utils.html import strip_tags
5 from django.conf import settings
6
7
8 class ContactUs(models.Model):
9     name = models.CharField(max_length=100)
10    email = models.EmailField()
11    subject = models.CharField(max_length=100)
12    message = models.TextField()
```

views.py

All right

Save ▼

Share

Your code

```
1 from rest_framework.views import APIView
2 from rest_framework.response import Response
3 from rest_framework import status
4 from .models import ContactUs, NewsletterSubscription
5
6
7 class ContactUsAPIView(APIView):
8     def post(self, request):
9         name = request.data.get("name")
10        email = request.data.get("email")
11        subject = request.data.get("subject")
12        message = request.data.get("message")
```


▼ frontend

- ✚ __init__.py
- ✚ admin.py
- ✚ apps.py
- ✚ models.py
- ✚ tests.py
- ✚ urls.py
- ✚ views.py

admin.py

All right

Save ▼

Share

Your code

```
1 from django.contrib import admin
2
3 # Register your models here.
4
```

apps.py

All right

[Save ▼](#)[Share](#)

Your code

```
1 from django.apps import AppConfig
2
3
4 class FrontendConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'frontend'
7
```

urls.py

Check results

[Save](#)[Share](#)

Code	Line	Column	Text
E501	7	80	line too long (88 > 79 characters)

Your code

```
1 from django.urls import path, re_path
2 from .views import index
3
4 urlpatterns = [
5     path("", index, name="index"),
6     re_path(
```

This error is unavoidable as it is coming from the regular expression which is quite long

views.py

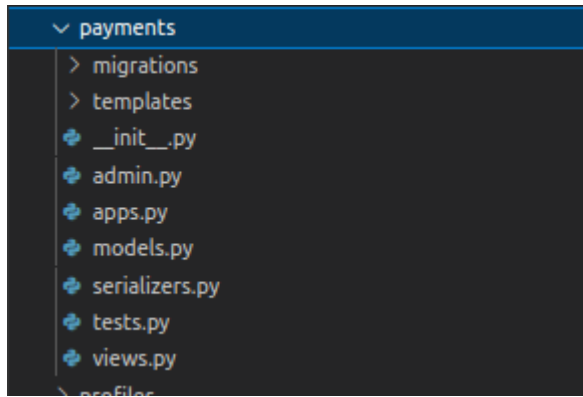
All right

Save ▾

Share

Your code

```
1 from django.shortcuts import render
2
3
4 # Create your views here.
5 def index(request):
6     return render(request, "build/index.html")
7
```



admin.py

All right

Save ▼

Share

Your code

```
1 from django.contrib import admin
2 from .models import Payment, PaymentHistory
3
4
5 @admin.register(Payment)
6 class PaymentAdmin(admin.ModelAdmin):
7     list_display = ("membership", "amount", "created_at")
8
9
10 @admin.register(PaymentHistory)
11 class PaymentHistoryAdmin(admin.ModelAdmin):
12     list_display = ("user", "Amount_of_payments")
13
```

apps.py

All right

Save ▼

Share

Your code

```
1 from django.apps import AppConfig
2
3
4 class PaymentsConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'payments'
7
```


models.py

All right

Save ▾

Share

Your code

```
1 from django.db import models
2 from subscriptions.models import SubscriptionPlan, Membership
3 from profiles.models import User
4
5
6 class Payment(models.Model):
7     membership = models.ForeignKey(
8         Membership, on_delete=models.CASCADE, related_name="payments"
9     )
10    created_at = models.DateTimeField(auto_now_add=True)
```

serializers.py

All right

Save ▾

Share

Your code

```
1 from rest_framework.serializers import ModelSerializer
2 from .models import Payment, PaymentHistory
3
4
5 class PaymentSerializer(ModelSerializer):
6     """Payment Serializer"""
7
8     class Meta:
9         model = Payment
10        fields = [
11            "created_at",
12            "amount",
13            1
```

views.py

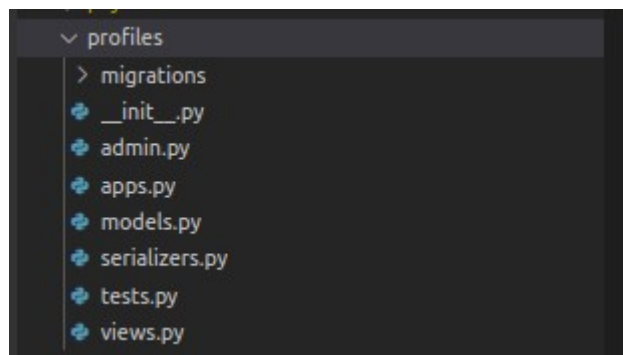
All right

Save ▾

Share

Your code

```
1 from datetime import datetime, timedelta
2 from django.utils.timezone import localtime
3 import json
4 from django.shortcuts import render
5 from rest_framework.views import APIView
6 from rest_framework.response import Response
7 from rest_framework import status
8 import stripe
9 from django.conf import settings
10 from authentication.authentication import JWTAuthentication
```



admin.py

All right

Save ▼

Share

Your code

```
1 from django.contrib import admin
2 from .models import User
3
4
5 @admin.register(User)
6 class AdminUser(admin.ModelAdmin):
7     list_display = (
8         "id",
9         "email",
```

apps.py

All right

Save ▼

Share

Your code

```
1 from django.apps import AppConfig
2
3
4 class ProfilesConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'profiles'
7
```

models.py

All right

Save ▾

Share

Your code

```
1 import datetime
2 from datetime import date
3 from django.db import models
4 from django.contrib.auth.models import AbstractUser
5 from django.contrib.auth.base_user import BaseUserManager
6 from cloudinary.models import CloudinaryField
7
8
9 class UserManager(BaseUserManager):
10     def create_user(
```

serializers.py

All right

Save ▾

Share

Your code

```
1 from rest_framework.serializers import ModelSerializer
2 from .models import User
3
4
5 class EditProfileSerializer(ModelSerializer):
6     class Meta:
7         model = User
8         fields = [
9             "country",
```


views.py

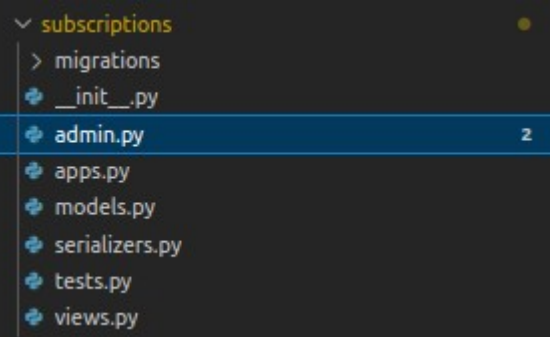
All right

Save ▾

Share

Your code

```
1 from rest_framework import status
2 from rest_framework.views import APIView
3 from rest_framework.response import Response
4 from authentication.authentication import JWTAuthentication
5 from .serializers import EditProfileSerializer
6
7
8 class EditProfileAPIView(APIView):
9     authentication_classes = [JWTAuthentication]
```



admin.py

All right

Save ▾

Share

Your code

```
1 from django.contrib import admin
2 from .models import SubscriptionPlan, Membership
3
4
5 @admin.register(SubscriptionPlan)
6 class SubscriptionPlanAdmin(admin.ModelAdmin):
7     list_display = (
8         "name",
9         "can_create_rooms",
10        "price",
11        "guest_limit",
```

apps.py

All right

Save ▾

Share

Your code

```
1 from django.apps import AppConfig
2
3
4 class SubscriptionsConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'subscriptions'
7
```

models.py

All right

Save ▾

Share

Your code

```
1 from django.db import models
2 from profiles.models import User
3 from django.conf import settings
4 import stripe
5
6 stripe.api_key = settings.STRIPE_SECRET_KEY
7
8
9 class SubscriptionPlan(models.Model):
10     name = models.CharField(max_length=255)
11     price = models.DecimalField(max_digits=10, decimal_places=2)
```

urls.py

All right

Save ▾

Share

Your code

```
1 from rest_framework.serializers import ModelSerializer
2 from .models import Membership, SubscriptionPlan
3
4
5 class SubscriptionPlanSerializer(ModelSerializer):
6     class Meta:
7         model = SubscriptionPlan
8         fields = [
9             "name",
10            "guest_limit",
11            "can_create_rooms",
```

views.py

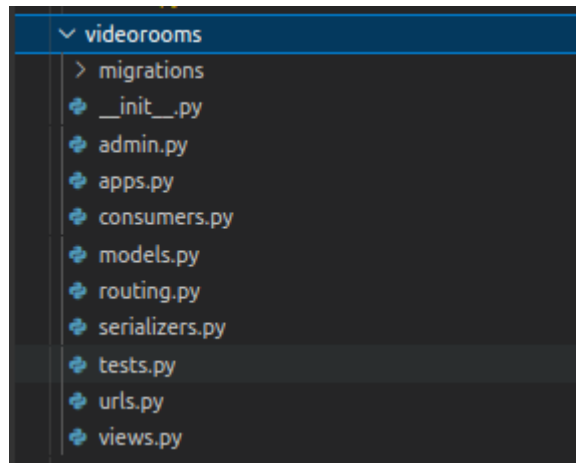
All right

Save ▾

Share

Your code

```
1 from django.shortcuts import render
2 from .models import SubscriptionPlan, Membership
3 from .serializers import SubscriptionPlanSerializer
4 from rest_framework import status
5 from rest_framework.views import APIView
6 from rest_framework.response import Response
7 from authentication.authentication import JWTAuthentication
8 import stripe
9 from django.conf import settings
10
11 stripe.api_key = settings.STRIPE_SECRET_KEY
```



admin.py

All right

Save ▾

Share

Your code

```
1 from django.contrib import admin
2 from .models import VideoRoom
3
4 admin.site.register(VideoRoom)
5
```

apps.py

All right

Save ▼

Share

Your code

```
1 from django.apps import AppConfig
2
3
4 class VideoroomsConfig(AppConfig):
5     default_auto_field = 'django.db.models.BigAutoField'
6     name = 'videorooms'
7
```

consumers.py

All right

Save ▾ Share

Your code

```
1 import json
2 from channels.generic.websocket import AsyncWebsocketConsumer
3 from channels.db import database_sync_to_async
4 from .models import VideoRoom
5 from .serializers import VideoRoomSerializer
6 from authentication.authentication import JWTAuthentication
7 from profiles.models import User
8 from authentication.serializers import UserSerializer
9 from channels.layers import get_channel_layer
10
```

models.py

All right

Save ▼

Share

Your code

```
1 from django.urls import re_path, path
2 from .consumers import VideoRoomConsumer
3
4 websocket_urlpatterns = [
5     re_path(
6         r"ws/videorooms/(?P<room_token>[^/]+)/$", VideoRoomConsumer.as_asgi()
7     ),
8     re_path(
```

routing.py

All right

Save ▼

Share

Your code

```
1 from django.urls import re_path, path
2 from .consumers import VideoRoomConsumer
3
4 websocket_urlpatterns = [
5     re_path(
6         r"ws/videorooms/(?P<room_token>[^/]+)/$", VideoRoomConsumer.as_asgi()
7     ),
8     re_path(
```

serializers.py

All right

Save ▼

Share

Your code

```
1 from rest_framework.serializers import ModelSerializer
2 from .models import VideoRoom
3
4
5 class VideoRoomSerializer(ModelSerializer):
6     class Meta:
7         model = VideoRoom
8         fields = [
```

views.py

All right

Save ▾

Share

Your code

```
1 from rest_framework.views import APIView
2 from rest_framework.response import Response
3 from rest_framework import status
4 from .serializers import VideoRoomSerializer
5 from authentication.authentication import JWTAuthentication
6 from .models import VideoRoom
7
8
9 ▾ class CreateRoomAPIView(APIView):
10     authentication_classes = [JWTAuthentication]
```