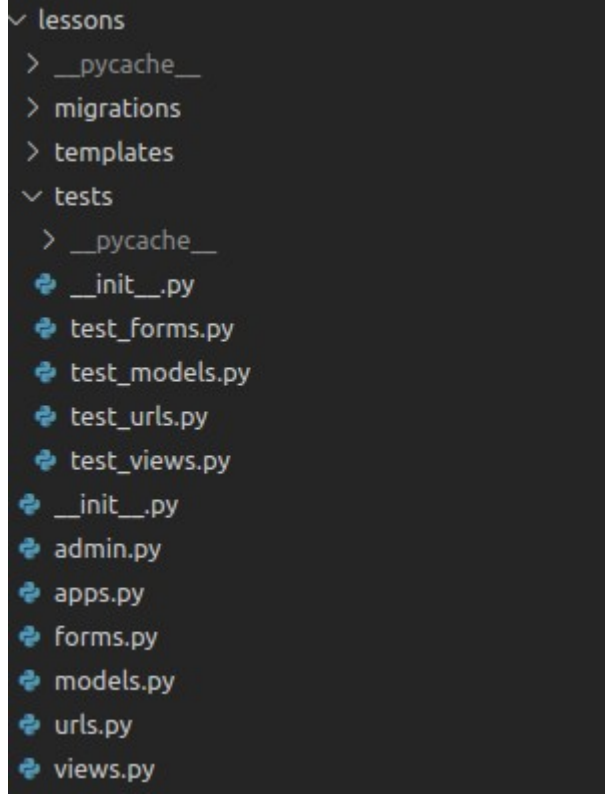


App: lessons



A screenshot of a file explorer window showing the directory structure of a Django application. The 'lessons' directory is expanded, showing subdirectories like '__pycache__', 'migrations', and 'templates'. The 'tests' directory is also expanded, showing files like '__init__.py', 'test_forms.py', 'test_models.py', 'test_urls.py', and 'test_views.py'. Other files in the 'lessons' directory include '__init__.py', 'admin.py', 'apps.py', 'forms.py', 'models.py', 'urls.py', and 'views.py'. Each file is preceded by a blue icon representing a Python file.

- ✓ lessons
 - > __pycache__
 - > migrations
 - > templates
 - ✓ tests
 - > __pycache__
 - __init__.py
 - test_forms.py
 - test_models.py
 - test_urls.py
 - test_views.py
 - __init__.py
 - admin.py
 - apps.py
 - forms.py
 - models.py
 - urls.py
 - views.py

admin.py

All right

Save ▾ Share

Your code

```
1 from django.contrib import admin
2 from .models import Lesson
3
4
5 @admin.register(Lesson)
6 class LessonAdmin(admin.ModelAdmin):
7     """Lesson admin."""
8     list_display = ('id', 'date', 'time', 'subject')
9     list_filter = ('teachers', 'students')
10     search_fields = ('date', 'time', 'subject', 'teachers', 'students')
11
```

apps.py

All right

Save ▼

Share

Your code

```
1 """Apps for the home app."""
2 from django.apps import AppConfig
3
4
5 class LessonsConfig(AppConfig):
6     """lessons config"""
7     default_auto_field = 'django.db.models.BigAutoField'
8     name = 'lessons'
9
```

forms.py

All right

Save ▾ Share

Your code

```
1  """Forms for the lessons app."""
2  from django import forms
3  from .models import Lesson
4
5
6  class LessonForm(forms.ModelForm):
7      """Lesson form."""
8
9      class Meta:
10         """Meta class."""
11         model = Lesson
12         fields = ['date', 'time', 'subject', 'teachers', 'students']
13         widgets = {
14             'date': forms.DateInput(attrs={'class': 'form-control'}),
15             'time': forms.Select(attrs={'class': 'form-control'}),
16             'subject': forms.Select(attrs={'class': 'form-control'}),
```

models.py

All right

Save ▾ Share

Your code

```
1  """Models for the lessons app"""
2  from django.db import models
3  from profiles.models import Teacher
4  from students.models import Student
5
6
7  # Time periods variations
8  TIME_PERIODS = (
9      (0, '9:00-9:45'),
10     (1, '10:00-10:45'),
11     (2, '11:00-11:45'),
12     (3, '14:00-14:45'),
13     (4, '15:00-15:45'),
14     (5, '16:00-16:45'),
15     (6, '17:00-17:45'),
```

urls.py

All right

Save ▾

Share

Your code

```
1  """Urls for the lessons app."""
2  from django.urls import path
3  from .views import (
4      LessonsView,
5      LessonAddView,
6      LessonEditView,
7      LessonDeleteView,
8      LessonDetailView,
9      TeacherScheduleView
10 )
11
12
13 urlpatterns = [
14     path('', LessonsView.as_view(), name='lessons_list'),
15     path('add/', LessonAddView.as_view(), name='lesson_add'),
```

view.py

All right

Save ▼

Share

Your code

```
1  """Views for the lessons app."""
2  from django.shortcuts import render, reverse, get_object_or_404
3  from django.http import HttpResponseRedirect
4  from django.views import View
5  from .models import Lesson
6  from .forms import LessonForm
7  from profiles.models import Teacher
8
9
10 ▼ class LessonsView(View):
11     """Lessons View"""
12 ▼     def get(self, request, *args, **kwargs):
13         """Receive lessons list"""
14 ▼         if request.user.is_authenticated:
15 ▼             if request.user.role == 5:
```

test_forms.py

All right

Save ▾ Share

Your code

```
1  """Tests for the forms of the lessons app."""
2  from django.test import TestCase
3  from lessons.forms import LessonForm
4  from profiles.models import (
5      CustomUser,
6      Teacher,
7      Receptionist,
8      SalesManager,
9      Parent
10 )
11 from students.models import Student
12 from lessons.models import TIME_PERIODS, SUBJECTS
13
14
15 class TestLessonForm(TestCase):
```


test_models.py

All right

Save ▾

Share

Your code

```
1  """Test cases for the lessons app."""
2  from django.test import TestCase
3  from lessons.models import Lesson, TIME_PERIODS, SUBJECTS
4  from profiles.models import (
5      CustomUser,
6      Teacher,
7      Receptionist,
8      SalesManager,
9      Parent
10 )
11 from students.models import Student
12
13
14 class TestModels(TestCase):
15     """Test the models for the lessons app."""
```

test_urls.py

All right

Save ▾ Share

Your code

```
1  """Test for forms in the lessons app"""
2  from django.test import SimpleTestCase
3  from django.urls import reverse, resolve
4  from lessons.views import (
5      LessonsView,
6      LessonAddView,
7      LessonEditView,
8      LessonDeleteView,
9      LessonDetailView,
10     TeacherScheduleView
11 )
12
13
14 class TestUrls(SimpleTestCase):
15     """Test the urls for the lessons app."""
```

test_views.py

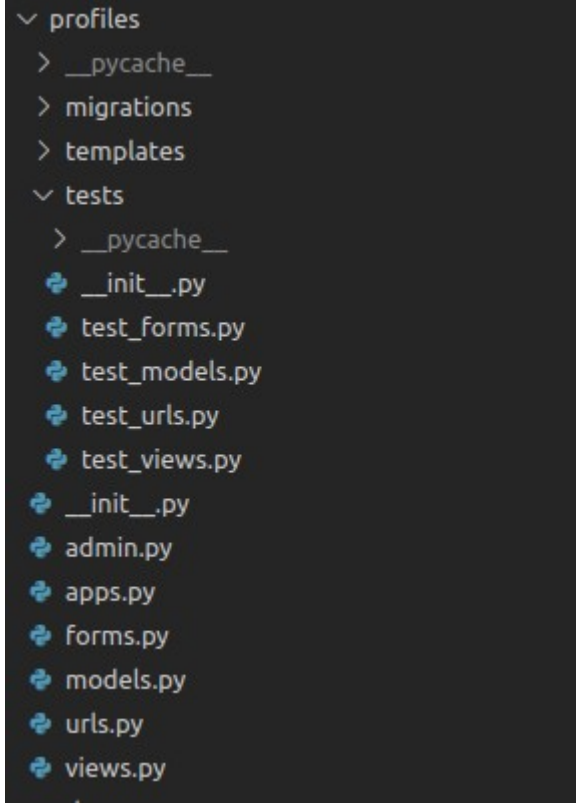
All right

Save ▾ Share

Your code

```
1  """Tests for the views of the lessons app."""
2  from django.test import TestCase, Client
3  from django.urls import reverse
4  from lessons.models import Lesson
5  from profiles.models import (
6      CustomUser,
7      Teacher,
8      SalesManager,
9      Receptionist,
10     Parent
11 )
12 from students.models import Student
13 from lessons.models import TIME_PERIODS, SUBJECTS
14
15
```

App: profiles



```

▼ profiles
  > __pycache__
  > migrations
  > templates
  ▼ tests
    > __pycache__
    + __init__.py
    + test_forms.py
    + test_models.py
    + test_urls.py
    + test_views.py
  + __init__.py
  + admin.py
  + apps.py
  + forms.py
  + models.py
  + urls.py
  + views.py

```

A screenshot of a file explorer interface showing the directory structure of a Django application named 'profiles'. The 'profiles' directory is expanded, revealing subdirectories for '__pycache__', 'migrations', and 'templates', as well as a 'tests' subdirectory. The 'tests' directory is also expanded, showing its own '__pycache__' and several test files: '__init__.py', 'test_forms.py', 'test_models.py', 'test_urls.py', and 'test_views.py'. Below the 'tests' directory, the main application files are listed: '__init__.py', 'admin.py', 'apps.py', 'forms.py', 'models.py', 'urls.py', and 'views.py'. Each file icon is preceded by a small blue icon with a plus sign inside a circle.

admin.py

All right

Save ▼

Share

Your code

```
1  """Admin for the profiles app."""
2  from django.contrib import admin
3  from .models import CustomUser, SalesManager, Teacher, Parent
4
5
6  @admin.register(CustomUser)
7  class UserAdmin(admin.ModelAdmin):
8      """User admin."""
9      list_display = ('id', 'email', 'first_name', 'last_name', 'phone', 'role')
10     search_fields = ('phone', 'first_name', 'last_name')
11
12
13     @admin.register(SalesManager)
14     class SalesManagerAdmin(admin.ModelAdmin):
15         """Sales manager admin."""
```

apps.py

All right

Save ▼

Share

Your code

```
1  """Apps for profiles app"""
2  from django.apps import AppConfig
3
4
5  class ProfilesConfig(AppConfig):
6      """profiles config"""
7      default_auto_field = 'django.db.models.BigAutoField'
8      name = 'profiles'
9
```

forms.py

All right

Save ▼

Share

Your code

```
1  """Forms for profiles app"""
2  from allauth.account.forms import SignupForm
3  from django import forms
4  from .models import CustomUser, Parent
5
6
7  class SimpleSignupForm(SignupForm):
8      """Allauth Signup Form extended"""
9      username = forms.CharField(
10         max_length=50,
11         label='Username',
12         widget=forms.TextInput(attrs={'placeholder': 'Username'}))
13     )
14     first_name = forms.CharField(
15         max_length=30,
```

models.py

All right

Save ▾ Share

Your code

```
1  """Models for profiles app."""
2  from django.db import models
3  from django.contrib.auth.models import AbstractUser
4
5
6  class CustomUser(AbstractUser):
7      """Custom user model."""
8      # Roles to assign to users
9      ROLES = (
10         (0, 'boss'),
11         (1, 'teacher'),
12         (2, 'sales'),
13         (3, 'receptionist'),
14         (4, 'parent'),
15         (5, 'potential user'),
```


urls.py

All right

Save ▾

Share

Your code

```
1  """Urls for the profiles app."""
2  from django.urls import path
3  from .views import (
4      UserProfileView,
5      UserProfileEditView,
6      UserProfileEditPasswordView,
7      NewApplicationsView,
8      NewApplicationsDetailView,
9      NewApplicationsDeleteView,
10     SearchMembersView,
11     DeleteMemberView,
12     AddRelationToParentView,
13 )
14
15
```

views.py

All right

Save ▾

Share

Your code

```
1  """Views for the profiles app."""
2  from django.shortcuts import render, get_object_or_404, reverse
3  from django.views import View
4  from django.views.generic import ListView
5  from django.http import HttpResponseRedirect, JsonResponse
6  from django.contrib.auth.forms import PasswordChangeForm
7  from django.contrib.auth import update_session_auth_hash
8  from lessons.models import Lesson
9  from students.models import Student
10 from sales.models import Sales
11 from .models import CustomUser, Parent, SalesManager, Teacher
12 from .forms import NewApplicationForm, UserProfileEditForm
13
14
15 ▾ class UserProfileView(View):
```

test_forms.py

All right

Save ▼

Share

Your code

```
1  """Test for the forms in profiles app."""
2  from django.test import TestCase
3  from profiles.forms import (
4      SimpleSignupForm,
5      NewApplicationForm,
6      UserProfileEditForm,
7      ParentRelationForm
8  )
9  from profiles.models import CustomUser, Parent
10
11
12  class TestForms(TestCase):
13      """Test the forms in the profiles app."""
14      def test_simple_signup_form(self):
15          """Test the simple signup form."""
```

test_models.py

All right

Save ▾

Share

Your code

```
1  """Testing cases for the profiles' models."""
2  from django.test import TestCase
3  from profiles.models import (
4      CustomUser,
5      Teacher,
6      Receptionist,
7      SalesManager,
8      Parent
9  )
10
11
12  class TestModels(TestCase):
13      """Test the models for the profiles app."""
14      def setUp(self):
15          """Set up the test."""
```

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      UserProfileEditView,
7      UserProfileEditPasswordView,
8      NewApplicationsView,
9      NewApplicationsDetailView,
10     NewApplicationsDeleteView,
11     SearchMembersView,
12     DeleteMemberView,
13     AddRelationToParentView,
14 )
15
```

test views.py

All right

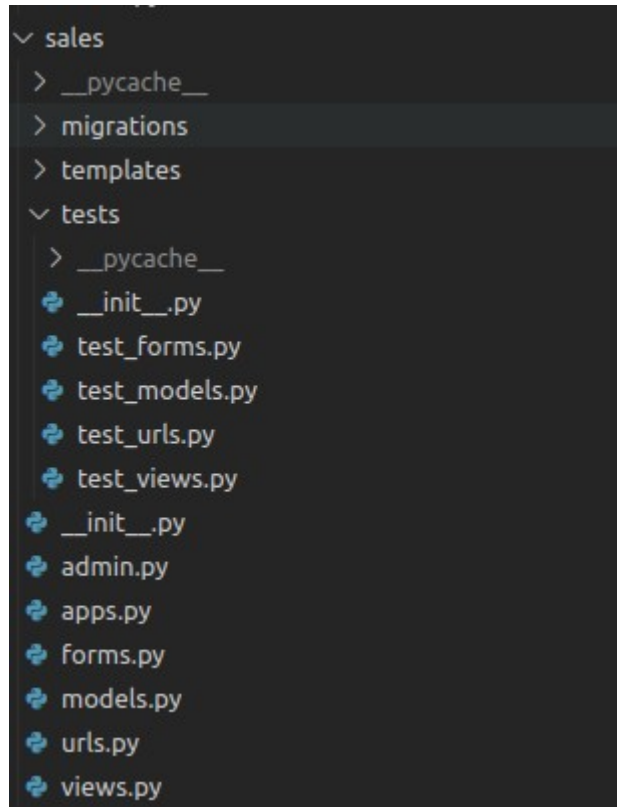
Save ▼

Share

Your code

```
1  """Tests for the views."""
2  from django.test import TestCase, Client
3  from django.urls import reverse
4  from profiles.models import (
5      CustomUser,
6      Teacher,
7      Receptionist,
8      SalesManager,
9      Parent
10 )
11 from students.models import Student
12
13
14 class TestViews(TestCase):
15     """Test the views for the profiles app."""
```

App: sales



admin.py

All right

Save ▼

Share

Your code

```
1  """Admin for the sales app"""
2  from django.contrib import admin
3  from .models import Sales
4
5
6  @admin.register(Sales)
7  class SalesAdmin(admin.ModelAdmin):
8      """Sales admin."""
9      list_display = ('id', 'sold_by', 'sold_to', 'amount', 'date')
10     search_fields = ('sold_by', 'sold_to', 'amount', 'date')
11
```


apps.py

All right

Save ▼

Share

Your code

```
1  """Apps for the sales app."""
2  from django.apps import AppConfig
3
4
5  class SalesConfig(AppConfig):
6      """sales config"""
7      default_auto_field = 'django.db.models.BigAutoField'
8      name = 'sales'
9
```

forms.py

All right

Save ▾ Share

Your code

```
1  """Forms for the sales app."""
2  from django import forms
3  from students.models import Student
4  from .models import Sales
5
6
7  class SalesForm(forms.ModelForm):
8      """Sales form"""
9      student = forms.ModelChoiceField(queryset=Student.objects.all())
10
11  class Meta:
12      """Meta class"""
13      model = Sales
14      fields = ['sold_to', 'amount'] + ['student']
15      widgets = {
```

models.py

All right

Save ▼

Share

Your code

```
1  """Models for the sales app"""
2  from django.db import models
3  from profiles.models import SalesManager, Parent
4
5
6  class Sales(models.Model):
7      """Sales model"""
8      sold_by = models.ForeignKey(
9          SalesManager,
10         on_delete=models.CASCADE,
11         related_name='sold'
12     )
13     sold_to = models.ForeignKey(
14         Parent,
15         on_delete=models.CASCADE,
```

urls.py

All right

Save ▾

Share

Your code

```
1  """Urls for the sales app."""
2  from django.urls import path
3  from .views import SalesView, sales_form, edit_sales, delete_sales
4
5
6  urlpatterns = [
7      path('sales/', SalesView.as_view(), name='sales_list'),
8      path('sales/add/', sales_form, name='sales_form'),
9      path('sales/edit/<int:pk>/', edit_sales, name='edit_sales'),
10     path('sales/delete/<int:pk>/', delete_sales, name='delete_sales'),
11 ]
12
```

views.py

All right

Save ▼

Share

Your code

```
1  """Views for the sales app."""
2  from django.shortcuts import render, reverse
3  from django.http import HttpResponseRedirect
4  from django.views import View
5  from profiles.models import Parent, SalesManager
6  from students.models import Student
7  from .forms import SalesForm
8  from .models import Sales
9
10
11  class SalesView(View):
12      """Sales view"""
13      def get(self, request, *args, **kwargs):
14          """Receive sales"""
15          if request.user.is_authenticated and (
```

test_forms.py

All right

Save ▾

Share

Your code

```
1  """Tests for the forms in the sales app."""
2  from django.test import TestCase
3  from sales.forms import SalesForm
4  from profiles.models import (
5      CustomUser,
6      SalesManager,
7      Parent
8  )
9  from students.models import Student
10
11
12  class SalesFormTest(TestCase):
13      """Test the forms in the sales app."""
14      def setUp(self):
15          """Set up the test."""
```

test_models.py

All right

Save ▼

Share

Your code

```
1  """Testing cases for the sales' models."""
2  from django.test import TestCase
3  from sales.models import Sales
4  from profiles.models import (
5      CustomUser,
6      SalesManager,
7      Parent
8  )
9
10
11 class SalesModelTest(TestCase):
12     """Test the models for the sales app."""
13     def setUp(self):
14         """Set up the test."""
15         # create sales
```

test_urls.py

All right

Save ▼

Share

Your code

```
1  """Tests for the sales urls app."""
2  from django.test import SimpleTestCase
3  from django.urls import reverse, resolve
4  from sales.views import (
5      SalesView, sales_form, edit_sales, delete_sales
6  )
7
8
9  class TestUrls(SimpleTestCase):
10     """Test the urls for the sales app."""
11
12     def test_sales_url_resolves(self):
13         """Test the sales url resolves."""
14         url = reverse('sales_list')
15         self.assertEqual(resolve(url).func.view_class, SalesView)
```


test_views.py

All right

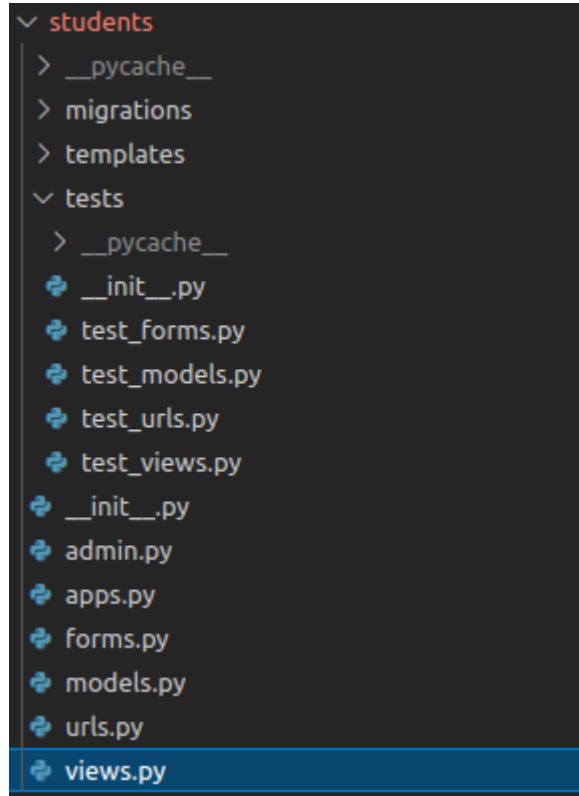
Save ▼

Share

Your code

```
1  """Tests for the views of the sales app."""
2  from django.test import TestCase, Client
3  from django.urls import reverse
4  from sales.models import Sales
5  from profiles.models import (
6      CustomUser,
7      SalesManager,
8      Parent
9  )
10 from students.models import Student
11
12
13 class SalesViewTest(TestCase):
14     """Test the sales view."""
15
```

App: students



admin.py

All right

Save ▾

Share

Your code

```
1  """Admin for the students app."""
2  from django.contrib import admin
3  from .models import Student
4
5
6  @admin.register(Student)
7  class StudentAdmin(admin.ModelAdmin):
8      """Student admin."""
9      list_display = (
10         'id',
11         'first_name',
12         'last_name',
13         'birthday',
14         'enrolled',
15         'classes_left',
```

apps.py

All right

Save ▾

Share

Your code

```
1  """Apps for the students app."""
2  from django.apps import AppConfig
3
4
5  class StudentsConfig(AppConfig):
6      """students config"""
7      default_auto_field = 'django.db.models.BigAutoField'
8      name = 'students'
9
```

forms.py

All right

Save ▾ Share

Your code

```
1  """Forms for students app"""
2  from django import forms
3  from .models import Student
4
5
6  class AddStudentForm(forms.ModelForm):
7      """Form for new student"""
8      class Meta:
9          """Meta class"""
10         model = Student
11         fields = [
12             'first_name',
13             'last_name',
14             'parent',
15             'birthday',
```

models.py

All right

Save ▾

Share

Your code

```
1  """Models for the students app."""
2  from django.db import models
3  from profiles.models import Parent
4  from sales.models import SalesManager
5
6
7  class Student(models.Model):
8      """Student model"""
9      first_name = models.CharField(max_length=50, blank=False, null=False)
10     last_name = models.CharField(max_length=50, blank=False, null=False)
11     parent = models.ManyToManyField(Parent, related_name='child')
12     birthday = models.DateField()
13     address = models.CharField(max_length=100, blank=True, null=True)
14     enrolled = models.DateTimeField(auto_now_add=True)
15     classes_left = models.IntegerField(default=0, blank=True, null=True)
```

urls.py

All right

Save ▾ Share

Your code

```
1  """Urls for the students app."""
2  from django.urls import path
3  from .views import (
4      StudentsView,
5      StudentAddView,
6      StudentView,
7      StudentEditView,
8      StudentDeleteView,
9  )
10
11
12  urlpatterns = [
13      path('students/', StudentsView.as_view(), name='students'),
14      path('students/add/', StudentAddView.as_view(), name='student_add'),
15      path('students/<int:pk>/', StudentView.as_view(), name='student_detail'),
```

views.py

All right

Save ▾

Share

Your code

```
1  """Views for the students app."""
2  from django.shortcuts import render, get_object_or_404, reverse
3  from django.http import HttpResponseRedirect
4  from django.views import View
5  from lessons.models import Lesson
6  from .models import Student
7  from .forms import AddStudentForm
8
9
10 class StudentAddView(View):
11     """Student Add View"""
12     def get(self, request):
13         """Receive student add form"""
14         if request.user.is_authenticated:
15             if request.user.role == 0 or request.user.role == 2:
```


test_forms.py

All right

Save ▾ Share

Your code

```
1  """Tests for the forms in the students app."""
2  from django.test import TestCase
3  from students.forms import AddStudentForm
4  from students.models import Student
5  from profiles.models import (
6      CustomUser,
7      SalesManager,
8      Parent
9  )
10
11
12  class TestAddStudentForm(TestCase):
13      """Test the AddStudentForm."""
14      def setUp(self):
15          """Set up the test."""
```

test_models.py

All right

Save ▾

Share

Your code

```
1  """Testing cases for the students' models."""
2  from django.test import TestCase
3  from students.models import Student
4  from profiles.models import (
5      CustomUser,
6      SalesManager,
7      Parent
8  )
9
10
11 class StudentModelTest(TestCase):
12     """Test the models for the students app."""
13     def setUp(self):
14         """Set up the test."""
15         # create users
```

test_urls.py

All right

Save ▾

Share

Your code

```
1  """Tests for the students app's urls."""
2  from django.test import SimpleTestCase
3  from django.urls import reverse, resolve
4  from students.views import (
5      StudentsView,
6      StudentAddView,
7      StudentView,
8      StudentEditView,
9      StudentDeleteView,
10 )
11
12
13 class TestUrls(SimpleTestCase):
14     """Test the urls for the students app."""
15
```

test views.py

All right

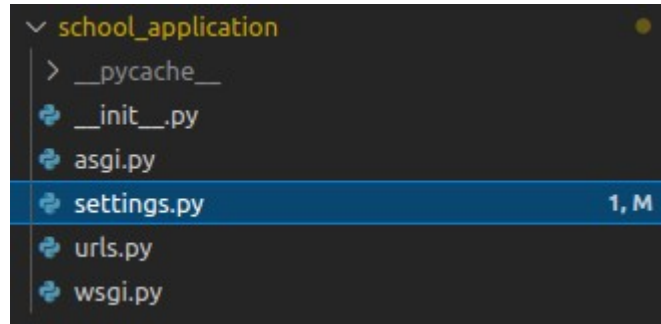
Save ▼

Share

Your code

```
1  """Test for the views in the students app."""
2  from django.test import TestCase, Client
3  from django.urls import reverse
4  from students.models import Student
5  from profiles.models import (
6      CustomUser,
7      SalesManager,
8      Parent
9  )
10
11
12  class TestStudentsViews(TestCase):
13      """Test the views in the students app."""
14
15      def setUp(self):
```

Project: school_application



asgi.py

```
1 import os
2
3 from django.core.asgi import get_asgi_application
4
5 os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'school_application.settings')
6
7 application = get_asgi_application()
8
```

settings.py

Some lines of standard Django code are too long and impossible to reduce.

Check results

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

Code	Line	Column	Text
E501	118	80	line too long (91 > 79 characters)
E501	121	80	line too long (81 > 79 characters)
E501	124	80	line too long (82 > 79 characters)
E501	127	80	line too long (83 > 79 characters)

Your code

```
20 # Application definition
21 INSTALLED_APPS = [
22     'django.contrib.admin',
23     'django.contrib.auth',
24     'django.contrib.contenttypes',
25     'django.contrib.sessions',
26     'django.contrib.messages',
27     'django.contrib.sites',
28     'allauth',
29     'allauth.account',
30     'allauth.socialaccount',
31     'django.contrib.staticfiles',
32     'django_summernote',
33     'crispy_forms',
34     'home',
35     'profile'
```

urls.py

All right

Save ▾ Share

Your code

```
1  """school_application URL Configuration"""
2  from django.contrib import admin
3  from django.urls import path, include
4  from django.conf import settings
5  from django.conf.urls.static import static
6
7  urlpatterns = [
8      path('admin/', admin.site.urls),
9      path('accounts/', include('allauth.urls')),
10     path('', include('home.urls')),
11     path('summernote/', include('django_summernote.urls')),
12     path('profiles/', include('profiles.urls')),
13     path('students/', include('students.urls')),
14     path('sales/', include('sales.urls')),
15     path('lessons/', include('lessons.urls')),
```


wsgi.py

All right

Save ▾ Share

Your code

```
1 """WSGI config for school_application project."""
2
3 import os
4
5 from django.core.wsgi import get_wsgi_application
6
7 os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'school_application.settings')
8
9 application = get_wsgi_application()
10
```

App: home

```
▼ home
  > __pycache__
  ▼ templates/home
    <> index.html
    + __init__.py
    + admin.py
    + apps.py
    + models.py
    + urls.py
    + views.py
```

apps.py

All right

Save ▾ Share

Your code

```
1  """Apps for the home app."""
2  from django.apps import AppConfig
3
4
5  class HomeConfig(AppConfig):
6      """home config"""
7      default_auto_field = 'django.db.models.BigAutoField'
8      name = 'home'
9
```

urls.py

All right

Save ▾

Share

Your code

```
1  """urls for the home app"""
2  from django.urls import path
3  from . import views
4
5  urlpatterns = [
6      path('', views.index, name='home'),
7  ]
8
```

views.py

All right

Save ▾ Share

Your code

```
1  """View for the home page"""
2  from django.shortcuts import render
3
4
5  def index(request):
6      return render(request, 'home/index.html')
7
```