BACKEND WEB DEVELOPMENT

DJANGO – A PYTHON FRAMEWORK

SOURABH PAL

LIST OF CLASSWORK

- Day 1: Display on client side (HttpResponse)
- Day 2: Display a html page from templates
- Day 3: Admin panel (adding members)
- Day 4: Pull data from database (admin panel added members information display)
- Day 5: Creating a form (Get and post methods) – applying CSS
- Day 6: function evaluation – output display
- Day 7: local library models

LIST OF CLASSWORK

- Day 8:
- TASK 1: Decorators
- TASK 2: JSON File
- TASK 3: API using Django rest Framework – POSTMAN TOOL
- Day 9: restful API – GET and POST method
- DAY 10: Project Web Scrapper

DOWNLOAD VISUAL STUDIO CODE AND PYTHON

DOWNLOAD LINKS:-

- https://code.visualstudio.com/download
- https://www.python.org/downloads/

DAY 1

Creating first Django project

DISPLAY CONTENT ON THE CLIENT SIDE (BROWSER)

ppt_by_sourabh_pal

Creating project (use vs code terminal OR cmd)

EXECUTE THE FOLLOWING CODES SEQUENTIALLY

- mkdir <FolderName>
- cd <FolderName>
- python –m venv env
- cd env
- Scripts\activate.bat
- Python –m pip install Django
- Django-admin startproject <ProjectName>
- Cd <ProjectName>
- Python manage.py runserver
- Crtl + C
- Code .

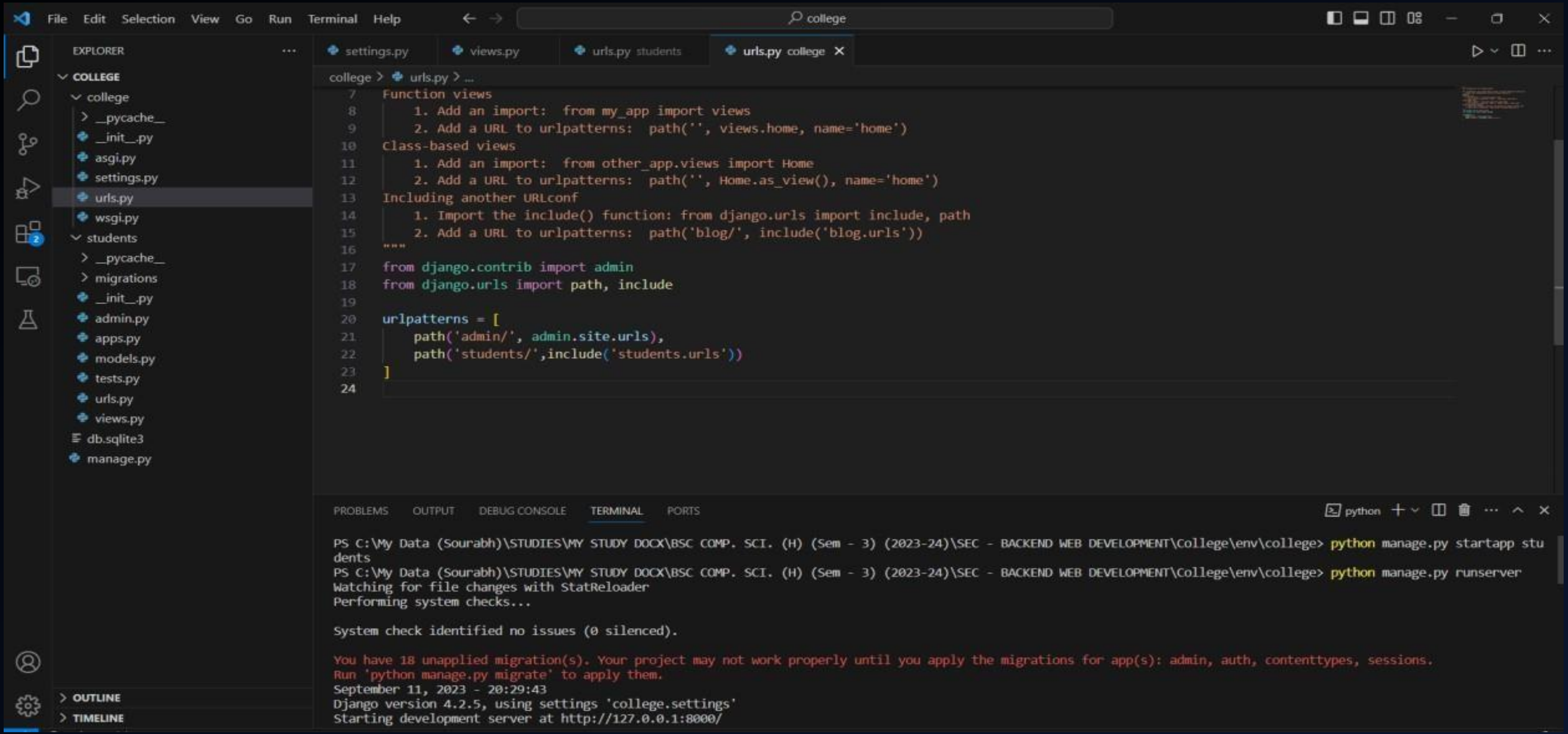
NOTE THAT

- Runserver command will generate a url which shows if the project creation is successful or not.
- To break the connection from server, press ctrl + C.
- Code . Command will open the present directories (project) in vs code (ide)

FOLLOW THESE STEPS

- Create app using the following command
- Python manage.py startapp <appName>
- Create urls.py in app directory
- Add the appName in quotes in the installed apps section of settings.py of the project
- Then make the following changes in views.py, urls.py (project and app).
- In the given example
- Directory name = College
- Project name = college
- App name = students

PROJECT – URLS.PY

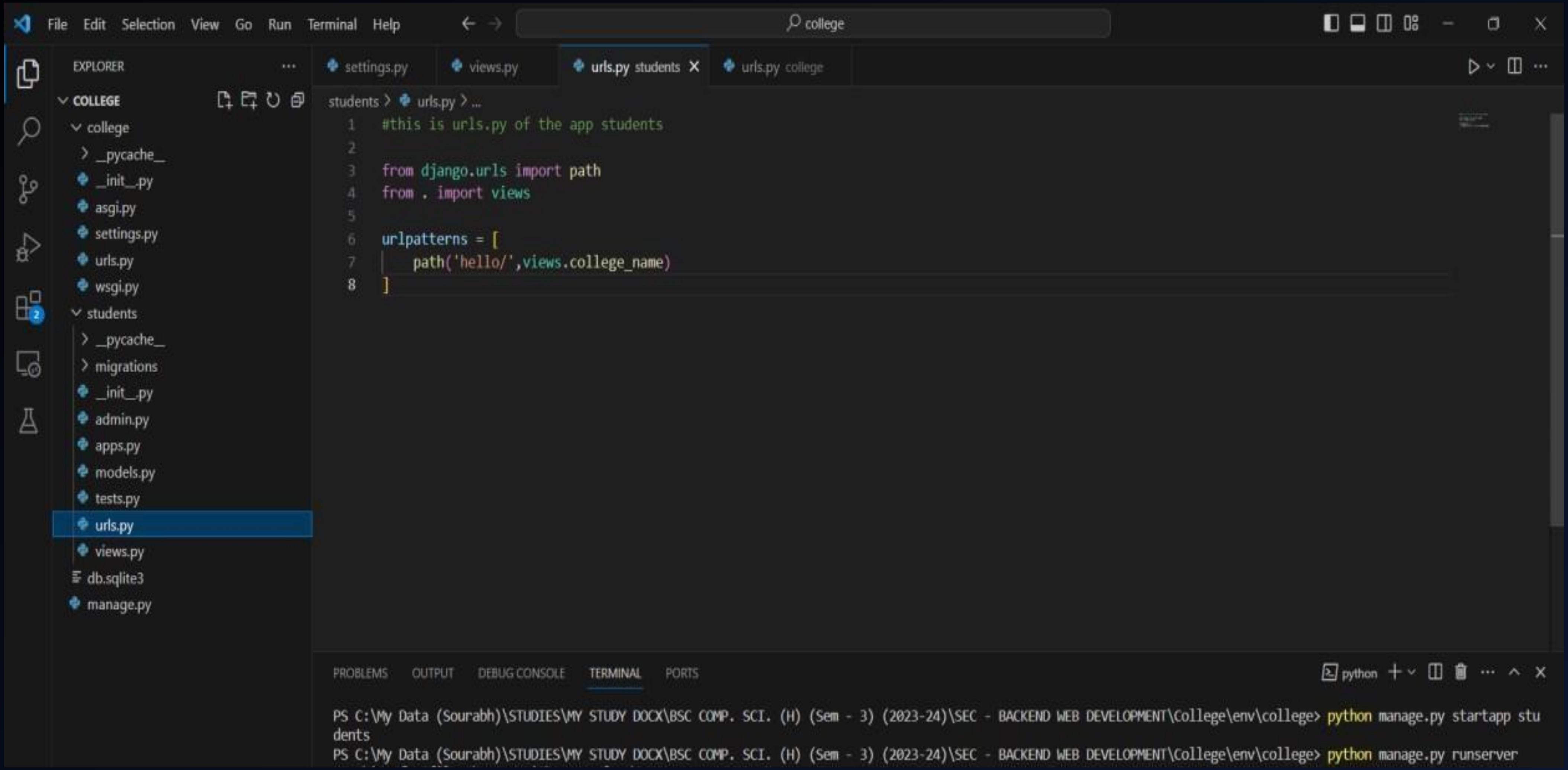


```python
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('students/',include('students.urls'))
]

```

```
PS C:\My Data (Sourabh)\STUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\College\env\college> python manage.py startapp stu
dents
PS C:\My Data (Sourabh)\STUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\College\env\college> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
September 11, 2023 - 20:29:43
Django version 4.2.5, using settings 'college.settings'
Starting development server at http://127.0.0.1:8000/
```

ppt_by_sourabh_pal

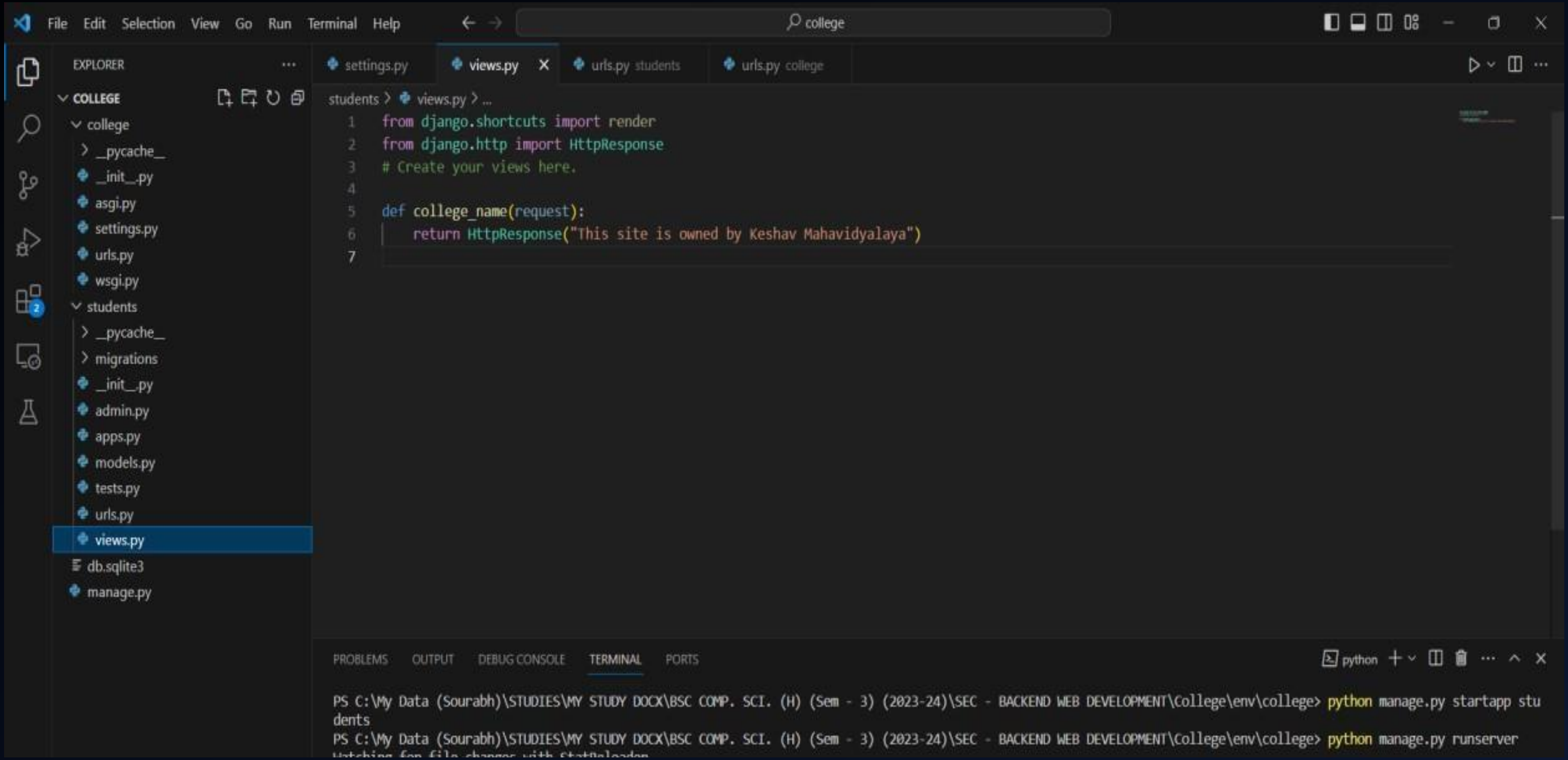
APP – URLS.PY



```python
#this is urls.py of the app students

from django.urls import path
from . import views

urlpatterns = [
    path('hello/',views.college_name)
]
```

```
PS C:\My Data (Sourabh)\STUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\College\env\college> python manage.py startapp stu
dents
PS C:\My Data (Sourabh)\STUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\College\env\college> python manage.py runserver
```

VIEWS.PY



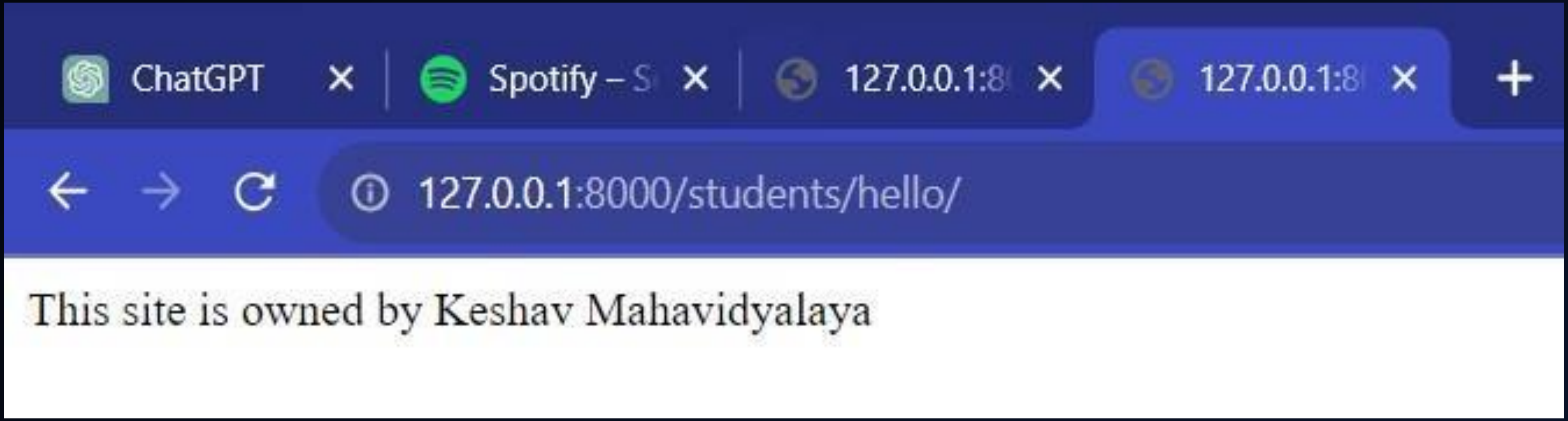
```python
from django.shortcuts import render
from django.http import HttpResponse
# Create your views here.

def college_name(request):
    return HttpResponse("This site is owned by Keshav Mahavidyalaya")
```

```
PS C:\My Data (Sourabh)\STUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\College\env\college> python manage.py startapp stu
dents
PS C:\My Data (Sourabh)\STUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\College\env\college> python manage.py runserver
Watching for file changes with StatReloader
```

BROWSER

- Run the following command
- Python manage.py runserver
- Then go to the url generated
- Then append the url with /students/hello

RESULT



This site is owned by Keshav Mahavidyalaya

DAY 2

Creating templates

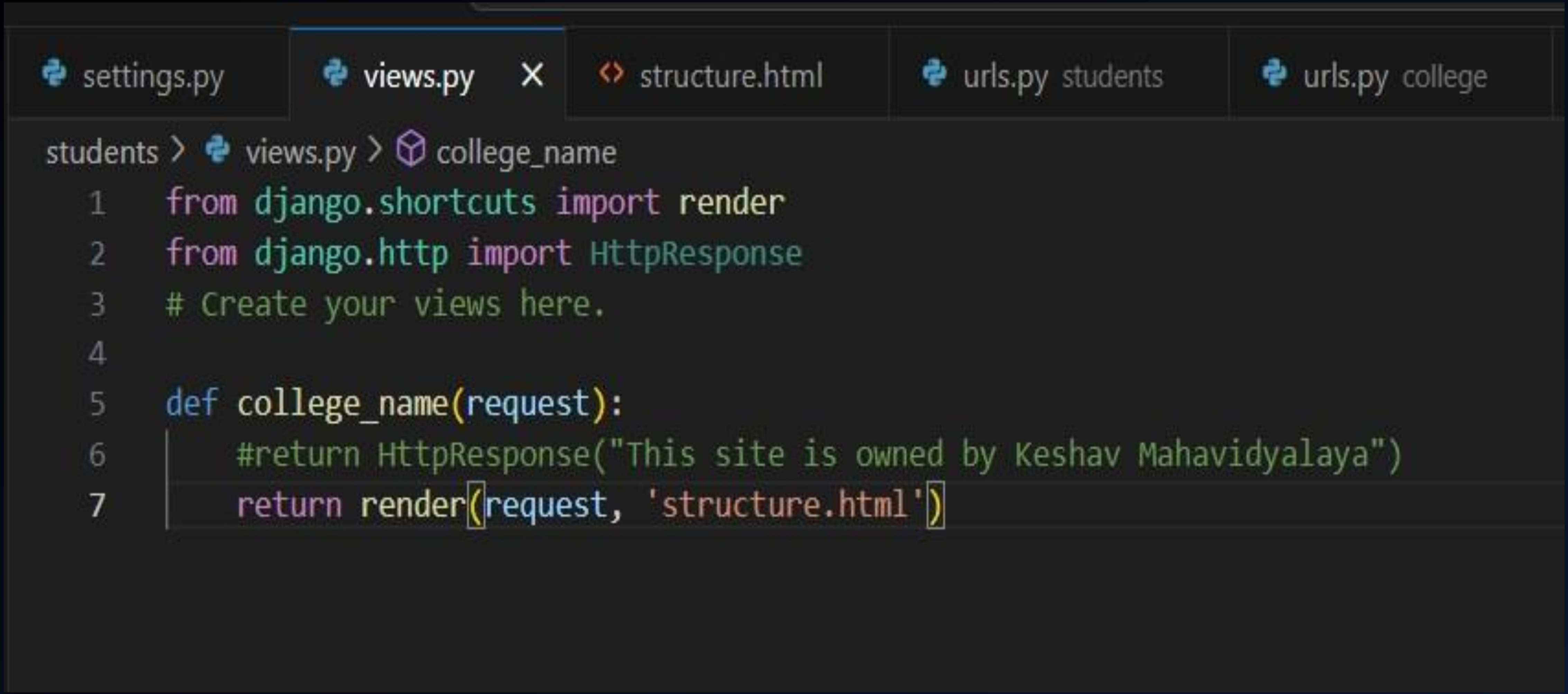
DISPLAY HTML PAGES ON THE CLIENT SIDE (BROWSER)

ppt_by_sourabh_pal

TEMPLATE – HTML PAGE

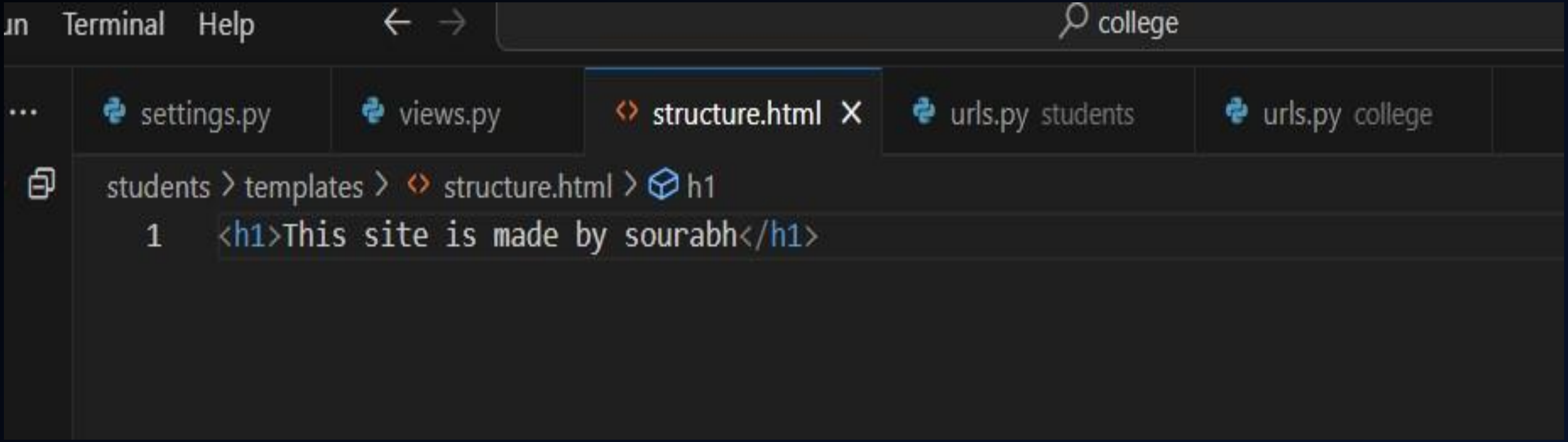
- To the same project, make the following changes
- Create templates folder inside the app directory and create html page inside it.
- Then Change views.py

VIEWS.PY



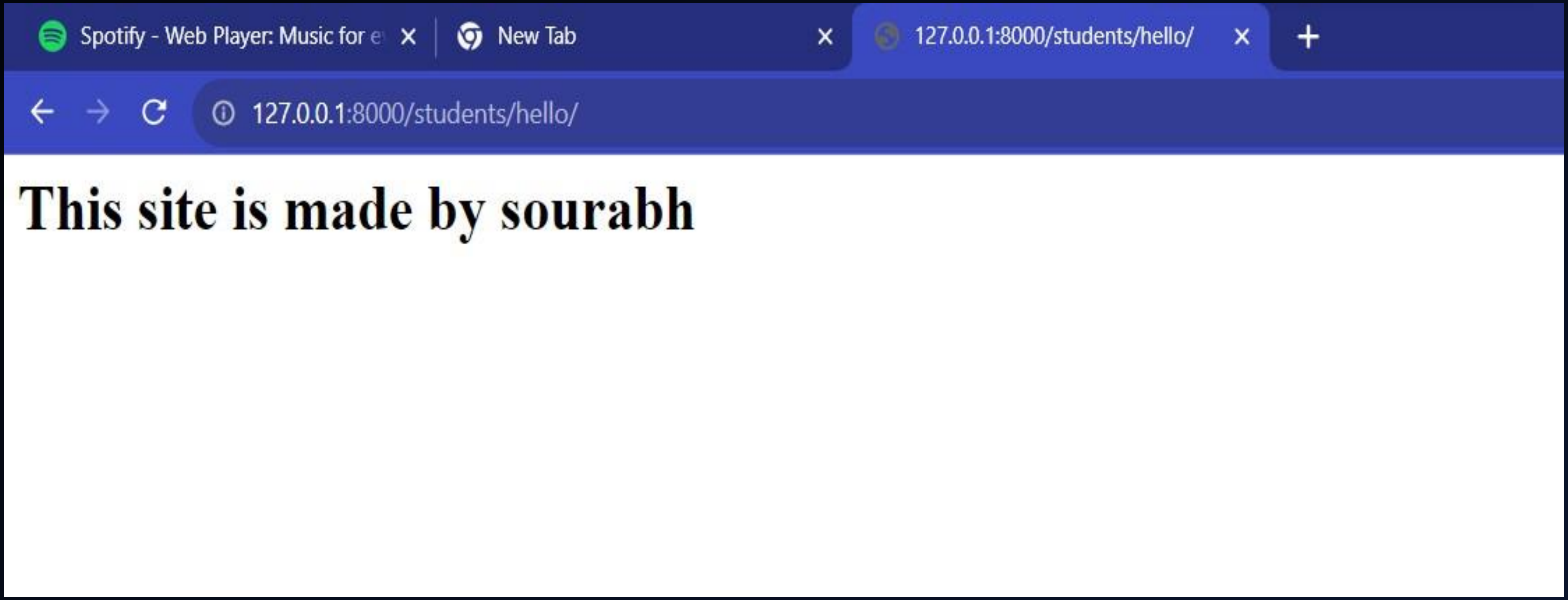
```python
from django.shortcuts import render
from django.http import HttpResponse
# Create your views here.

def college_name(request):
    #return HttpResponse("This site is owned by Keshav Mahavidyalaya")
    return render(request, 'structure.html')
```

HTML PAGE INSIDE TEMPLATES



```html
<h1>This site is made by sourabh</h1>
```

BROWSER


Spotify - Web Player: Music for e
New Tab
127.0.0.1:8000/students/hello/

127.0.0.1:8000/students/hello/

This site is made by sourabh

DAY 3

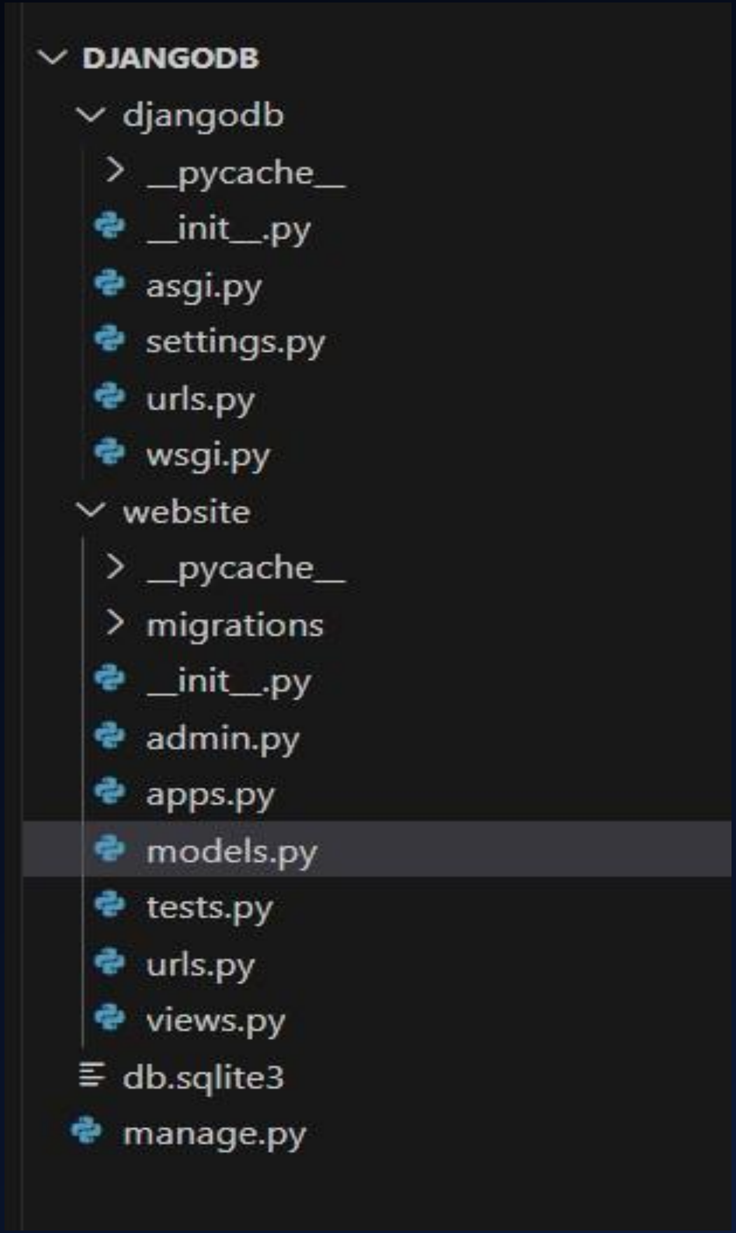
The Admin panel

USER ADDING DATA TO THE DATABASE

ppt_by_sourabh_pal

INFORMATION

- Create another project
- Exampe given:
- Directory name = DjangoDb
- Project name = djangodb
- Appname = website



```
DJANGODB
  djangodb
    __pycache__
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
  website
    __pycache__
    migrations
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  db.sqlite3
  manage.py
```

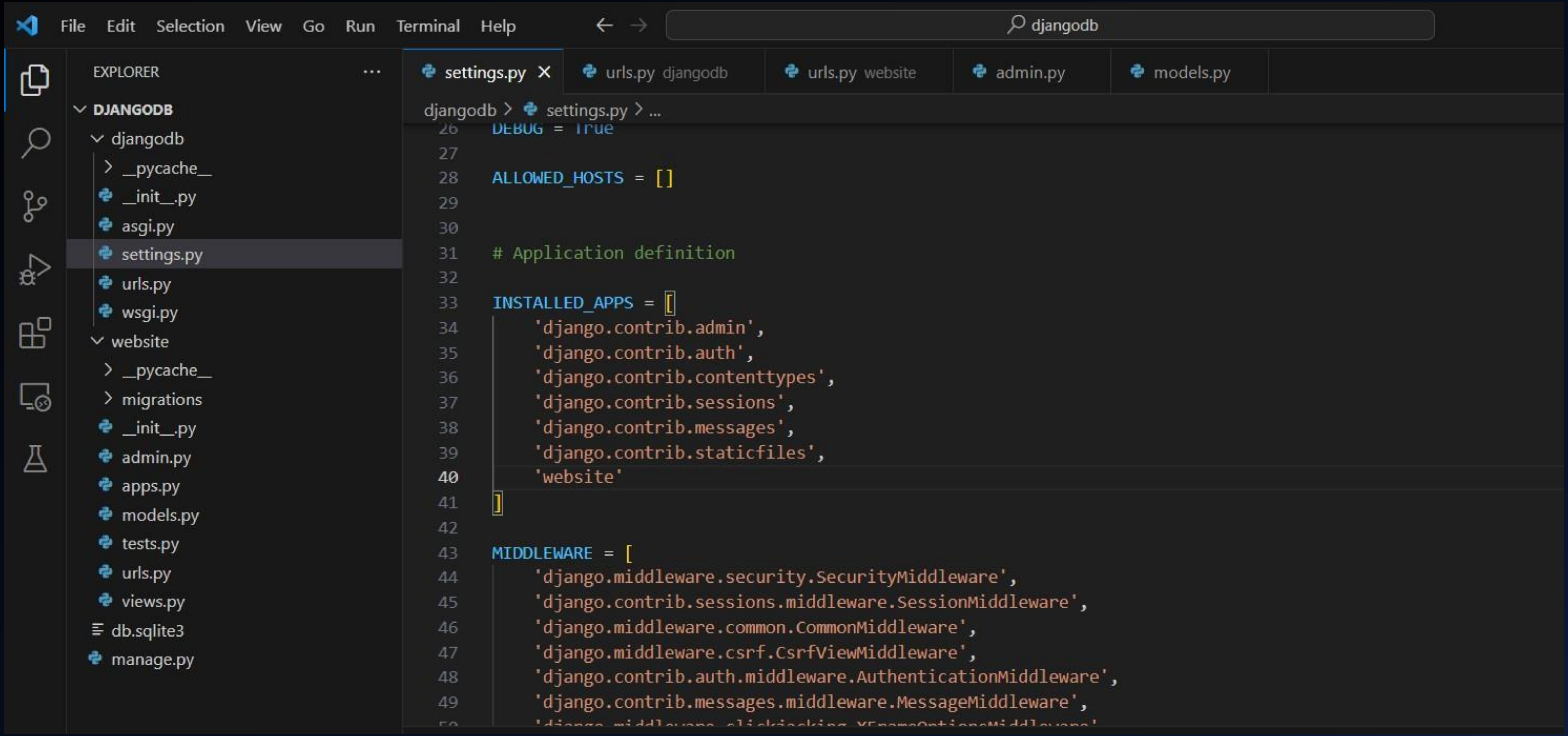
INSTRUCTIONS

- Run -> python manage.py migrate
- Then make changes in the models.py and other files as shown in next slides
- Then create an admin user by running the command -> python manage.py createsuperuser
- Then create a user – remember the username and password
- Then run the command -> python manage.py makemigrations
- Then again -> python manage.py migrate
- Python manage.py runserver

CHANGES BEFORE RUNNING THE INSTRUCTED COMMANDS

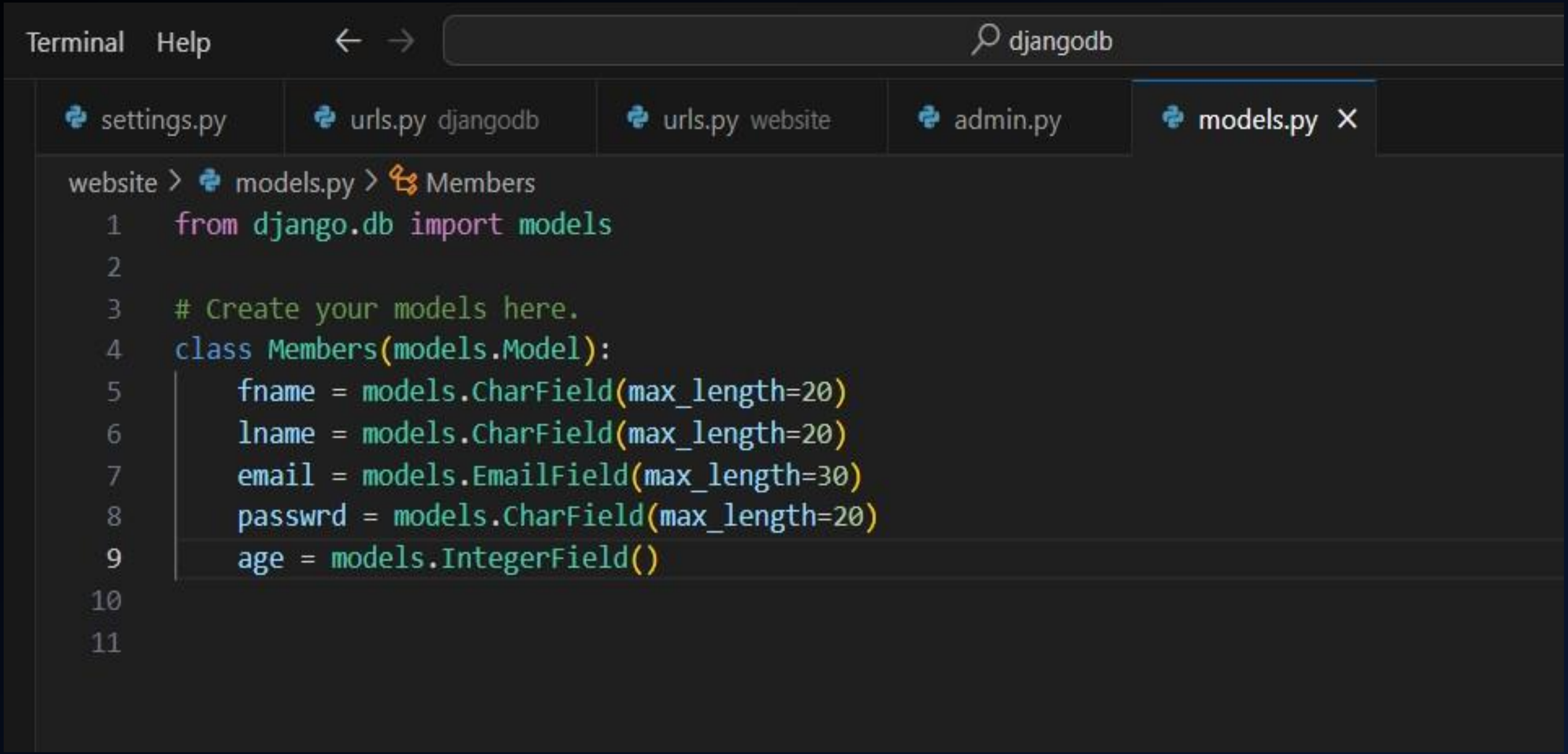
ppt_by_sourabh_pal

SETTINGS.PY



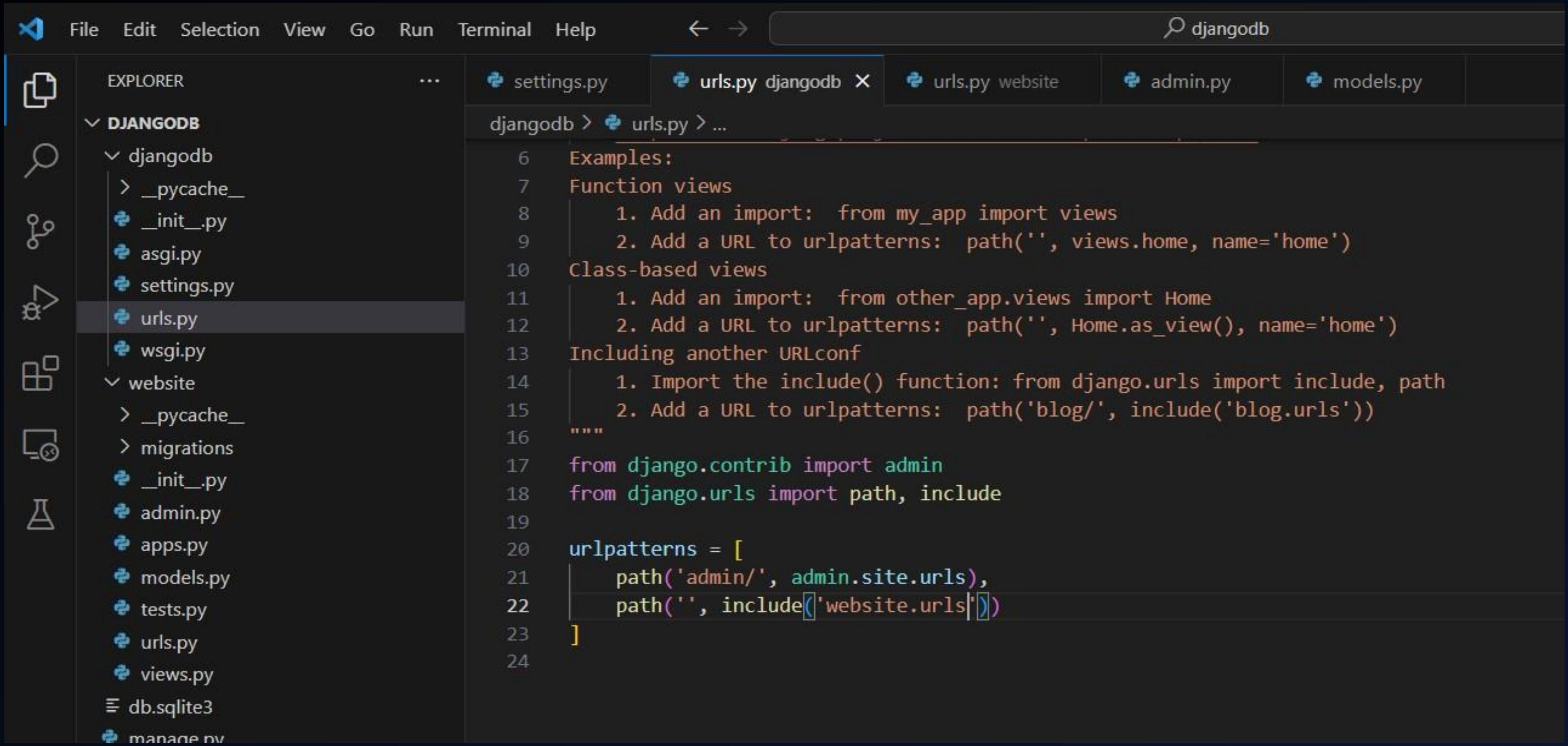
```python
DEBUG = True

ALLOWED_HOSTS = []


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'website'
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
```

MODELS.PY



```python
from django.db import models

# Create your models here.
class Members(models.Model):
    fname = models.CharField(max_length=20)
    lname = models.CharField(max_length=20)
    email = models.EmailField(max_length=30)
    passwrd = models.CharField(max_length=20)
    age = models.IntegerField()
```

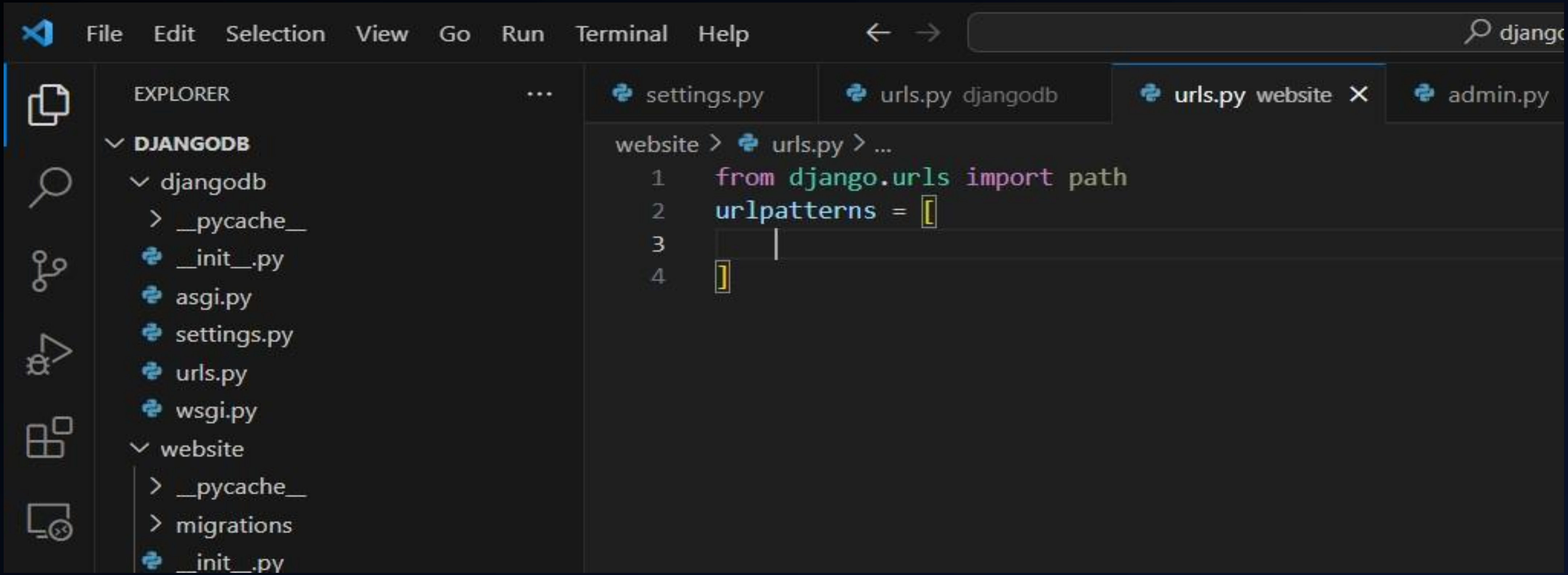
PROJECT – URLS.PY



```
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('website.urls'))
]
```

ppt_by_sourabh_pal

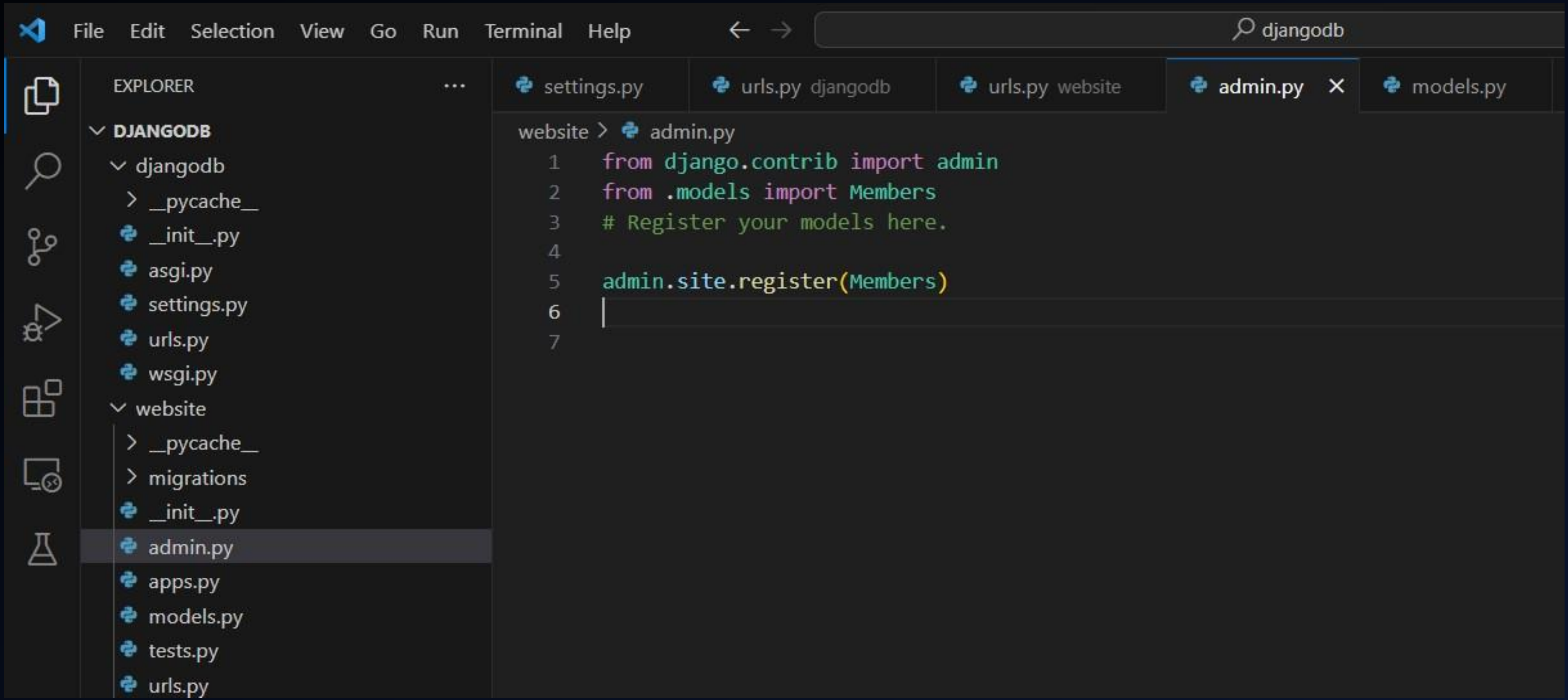
APP – URLS.PY



```python
from django.urls import path
urlpatterns = [
    
]
```

ppt_by_sourabh_pal

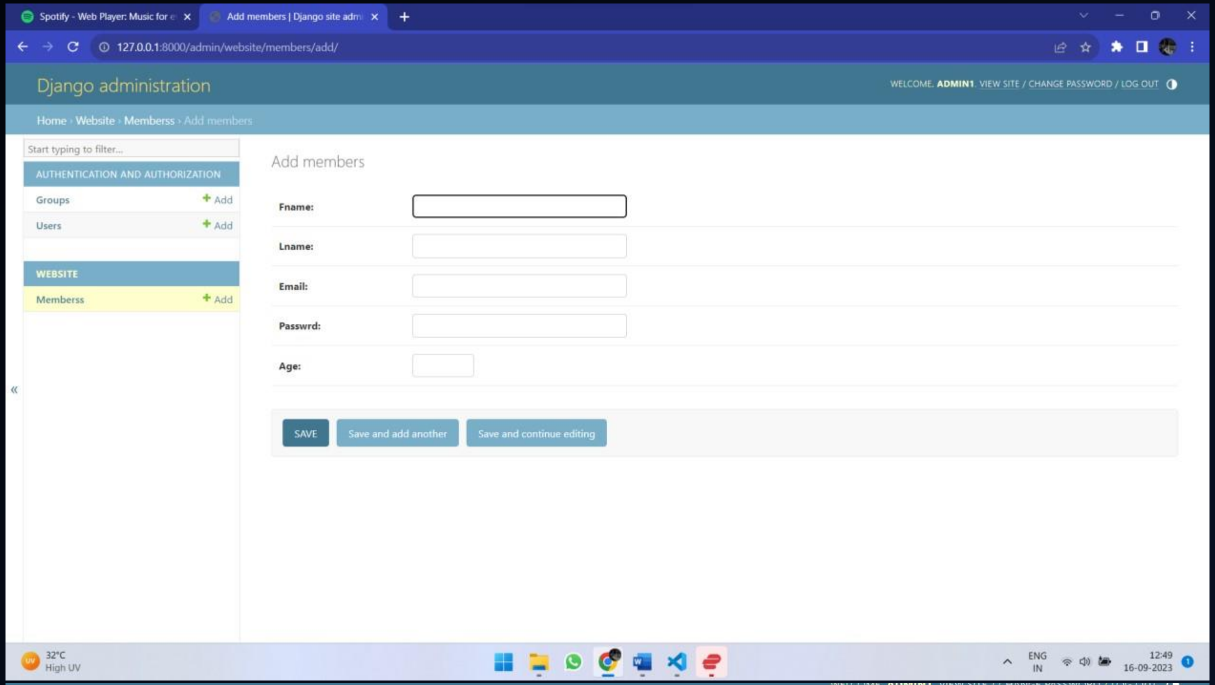
ADMIN.PY



```python
from django.contrib import admin
from .models import Members
# Register your models here.

admin.site.register(Members)

```

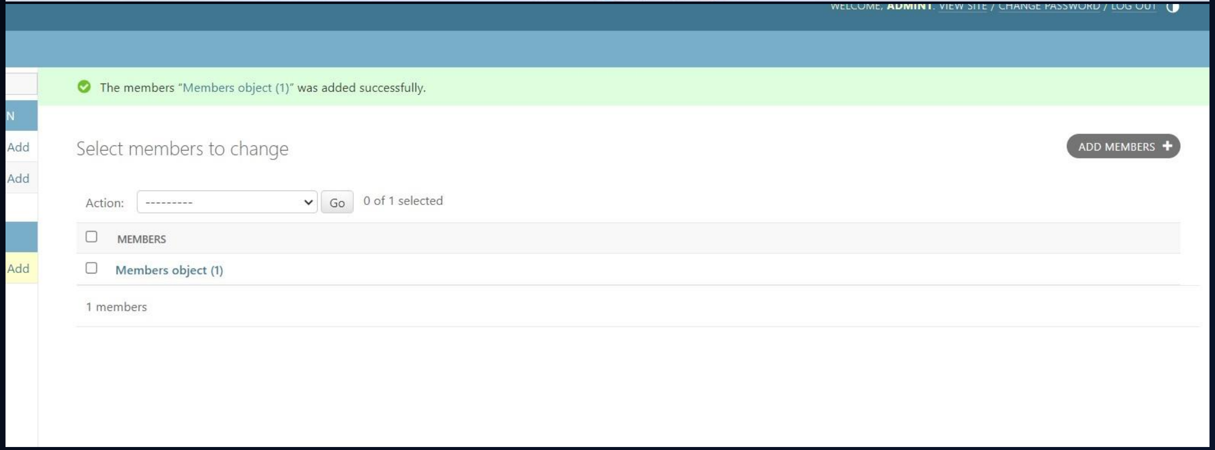
ppt_by_sourabh_pal

ADDING MEMBERS ON THE CLIENT SIDE


Spotify - Web Player: Music for e
Add members | Django site adm
127.0.0.1:8000/admin/website/members/add/
Django administration
WELCOME, ADMIN1. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Website › Memberss › Add members
Start typing to filter...
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
WEBSITE
Memberss + Add
SAVE
Save and add another
Save and continue editing
32°C High UV
ENG IN
12:49 16-09-2023

Add members

Fname:	
Lname:	
Email:	
Passwrd:	
Age:	



The members "Members object (1)" was added successfully.

Select members to change

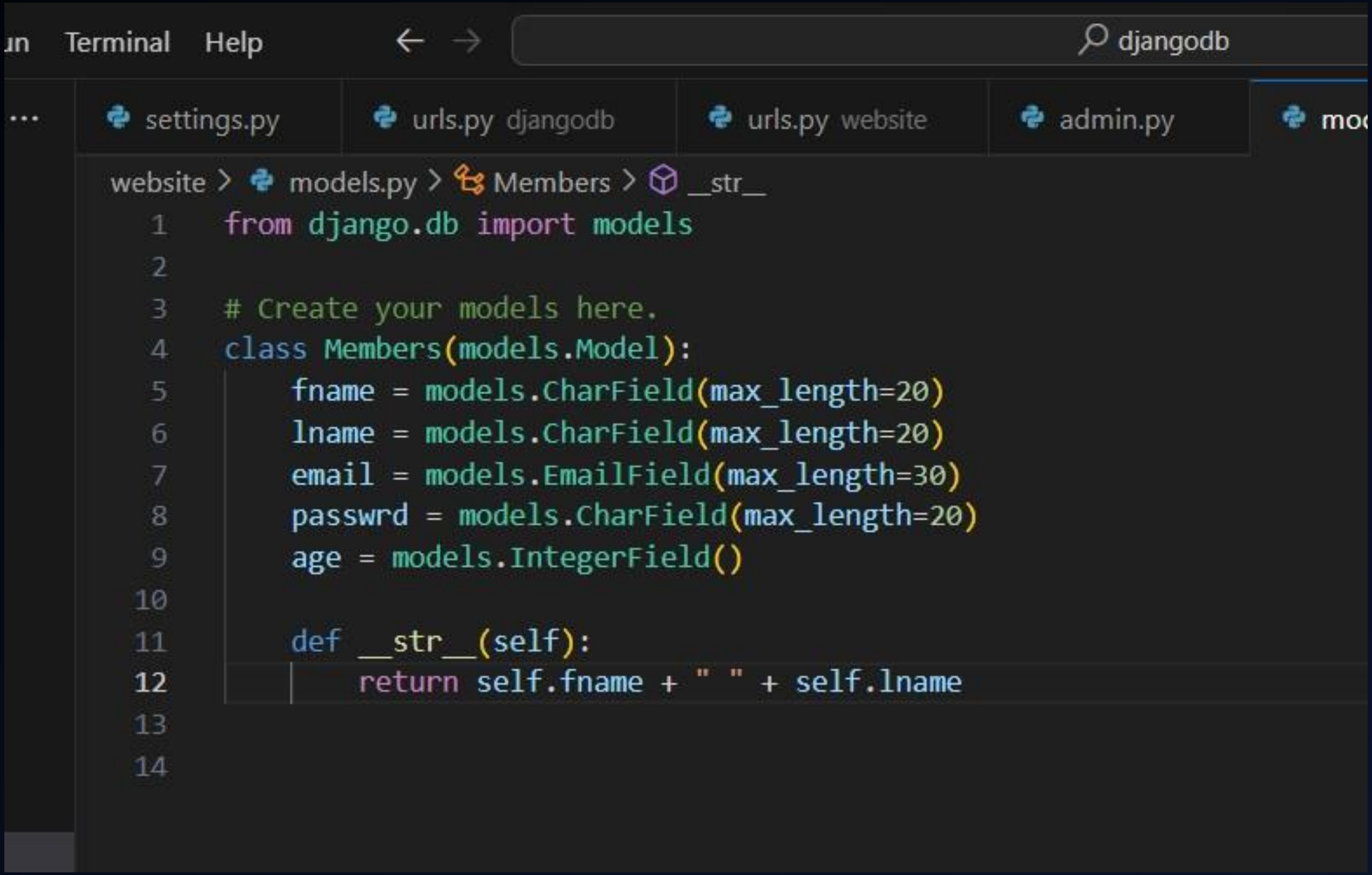
Action: --------- 0 of 1 selected

☐	MEMBERS
☐	Members object (1)

1 members

ppt_by_sourabh_pal

UPDATED MODELS.PY – TO DISPLAY NAME LIST



```python
from django.db import models

# Create your models here.
class Members(models.Model):
    fname = models.CharField(max_length=20)
    lname = models.CharField(max_length=20)
    email = models.EmailField(max_length=30)
    passwrd = models.CharField(max_length=20)
    age = models.IntegerField()

    def __str__(self):
        return self.fname + " " + self.lname
```

ppt_by_sourabh_pal

DAY 4

Pull data from database

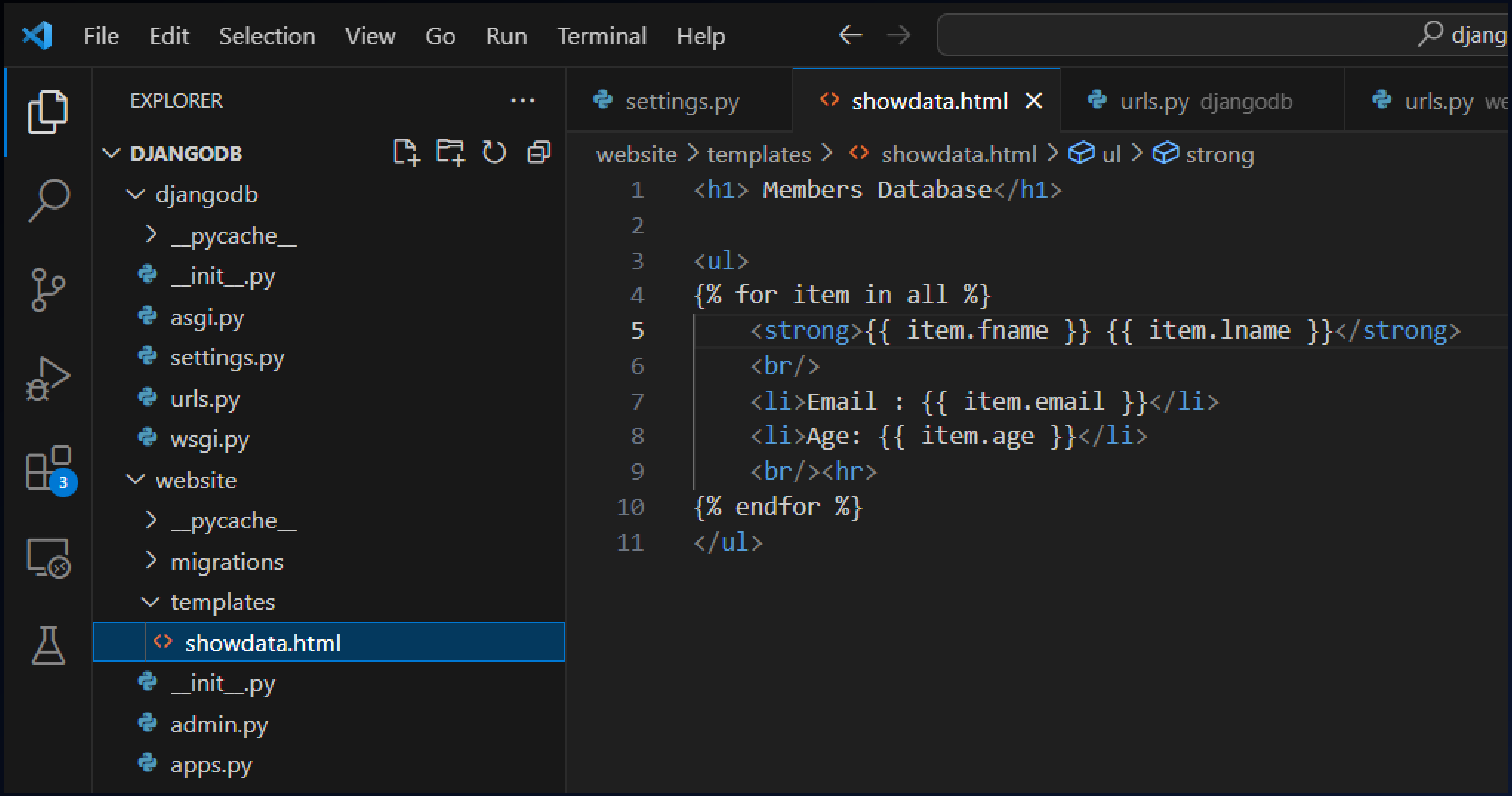
EXTRACT DATA FROM DATABASE AND DISPLAY ON THE CLIENT SIDE

ppt_by_sourabh_pal

INSTRUCTIONS

- This is continuation of the previous project in which we display the content of the database on to the client screen.
- These data are entered by the user in the admin panel earlier.
- We need to create an html file for this
- Let’s name it showdata.html
- Make changes to the html page in the templates

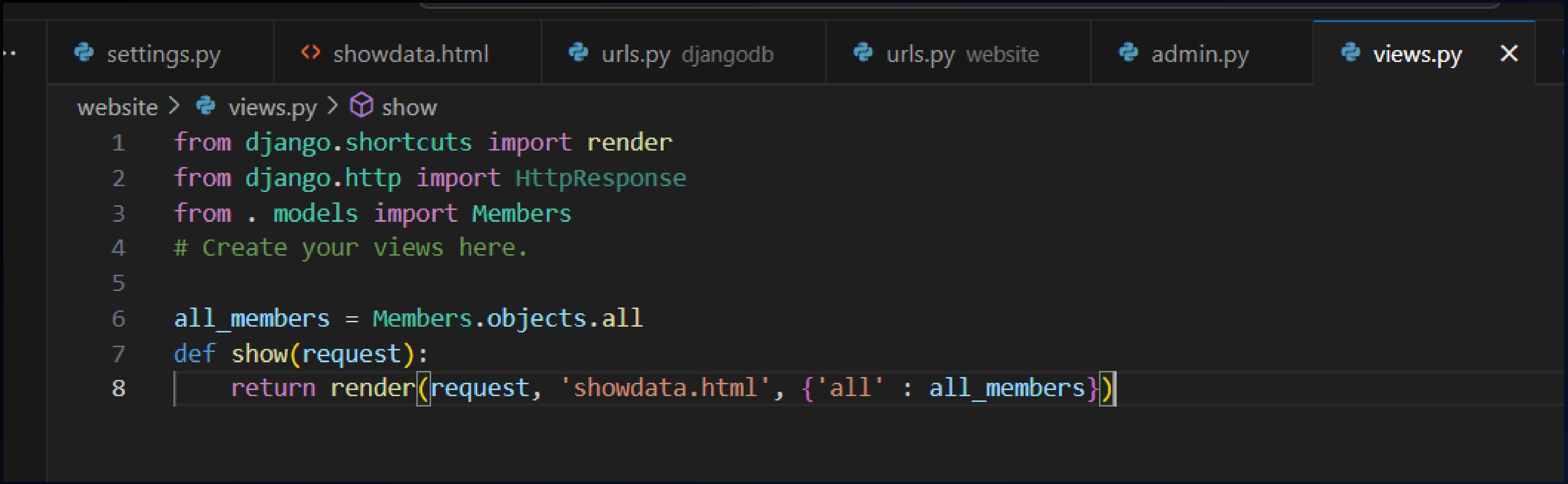
SHOWDATA.HTML



```
<h1> Members Database</h1>

<ul>
{% for item in all %}
    <strong>{{ item.fname }} {{ item.lname }}</strong>
    <br/>
    <li>Email : {{ item.email }}</li>
    <li>Age: {{ item.age }}</li>
    <br/><hr>
{% endfor %}
</ul>
```

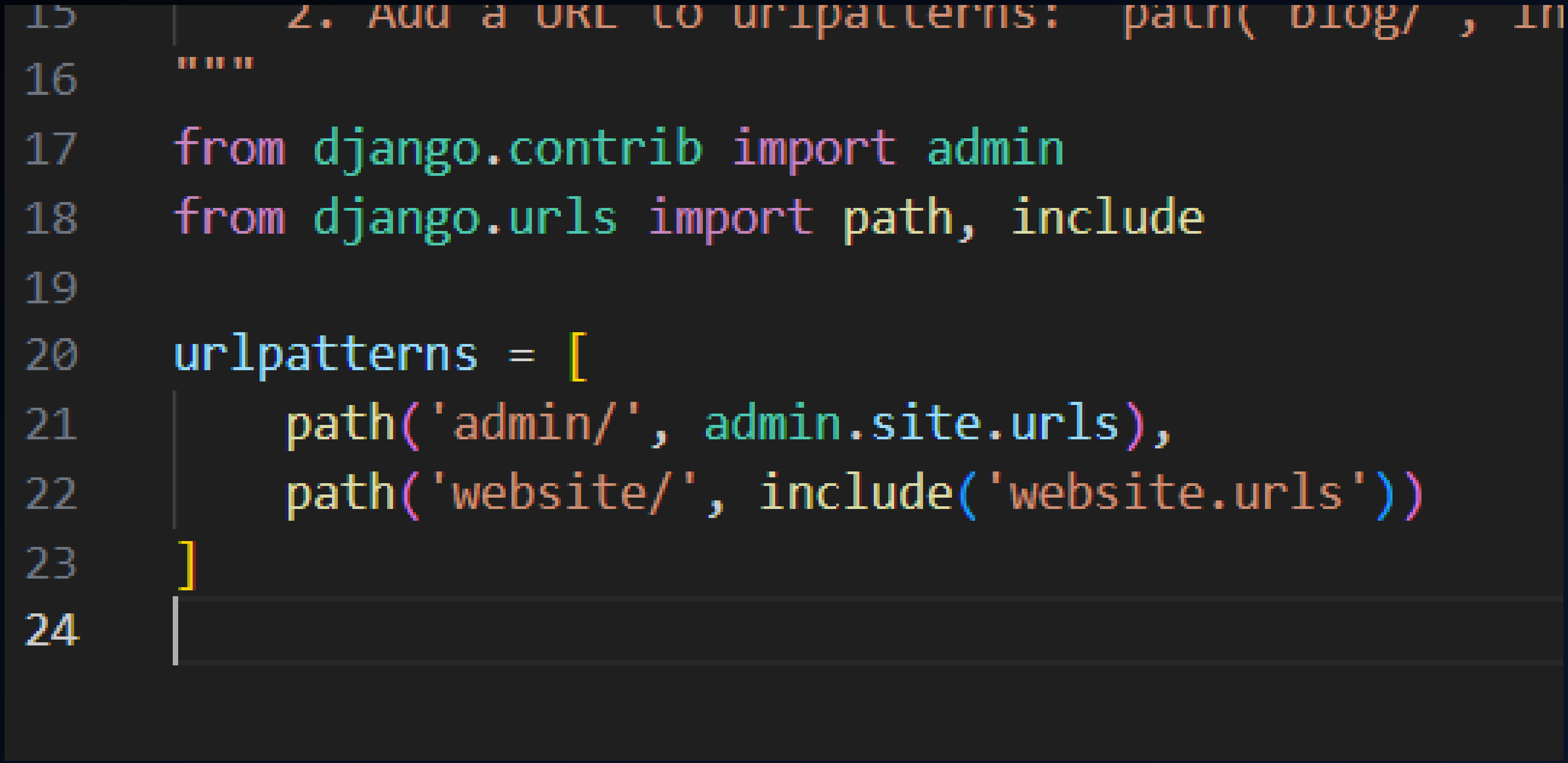
ppt_by_sourabh_pal

VIEWS.PY



```python
from django.shortcuts import render
from django.http import HttpResponse
from . models import Members
# Create your views here.

all_members = Members.objects.all
def show(request):
    return render(request, 'showdata.html', {'all' : all_members})
```

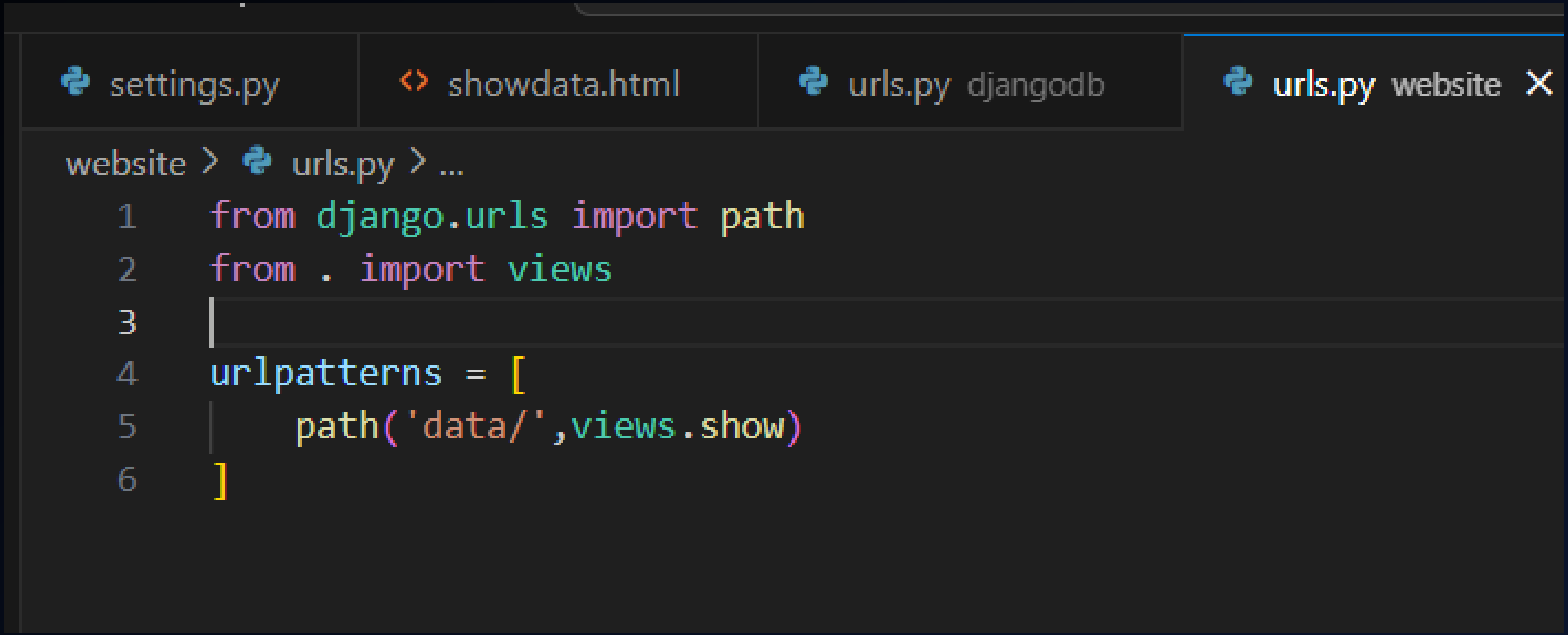
PROJECT – URLS.PY



```
    2. Add a URL to urlpatterns:  path('blog/', in
"""
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('website/', include('website.urls'))
]

```

ppt_by_sourabh_pal

APP – URLS.PY

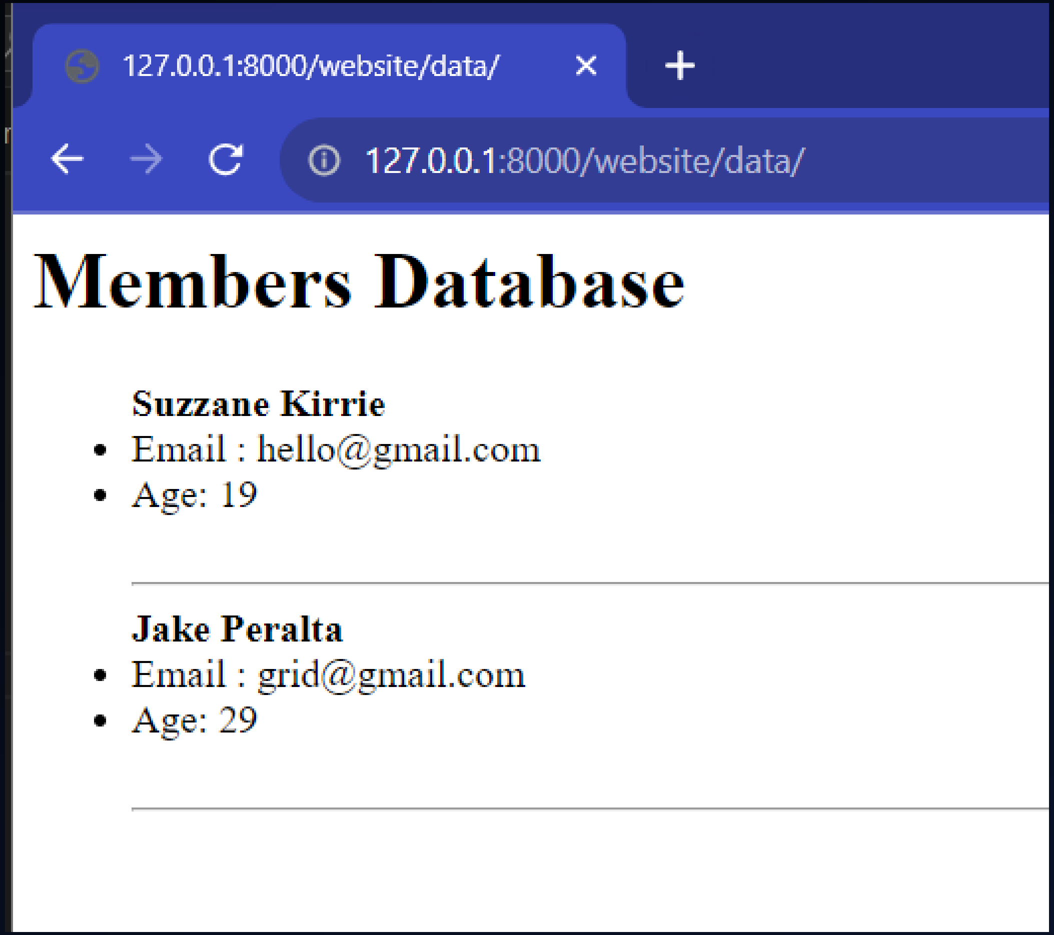


```python
from django.urls import path
from . import views

urlpatterns = [
    path('data/',views.show)
]
```

No change in admin.py and models.py

ppt_by_sourabh_pal

CLIENT SIDE



Members Database

Suzzane Kirrie

- Email : hello@gmail.com
- Age: 19

Jake Peralta

- Email : grid@gmail.com
- Age: 29

DAY 5

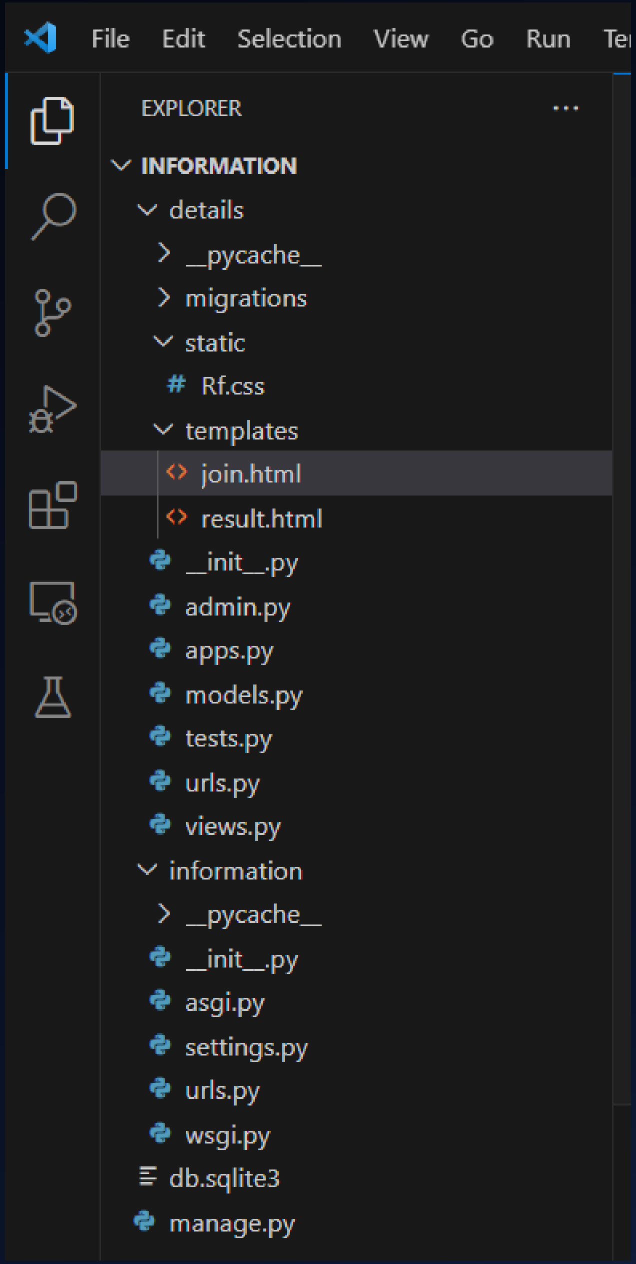
Creating Form and applying CSS

TAKING INPUT FROM USER AND SHOWING DESIRED OUTPUT

ppt_by_sourabh_pal

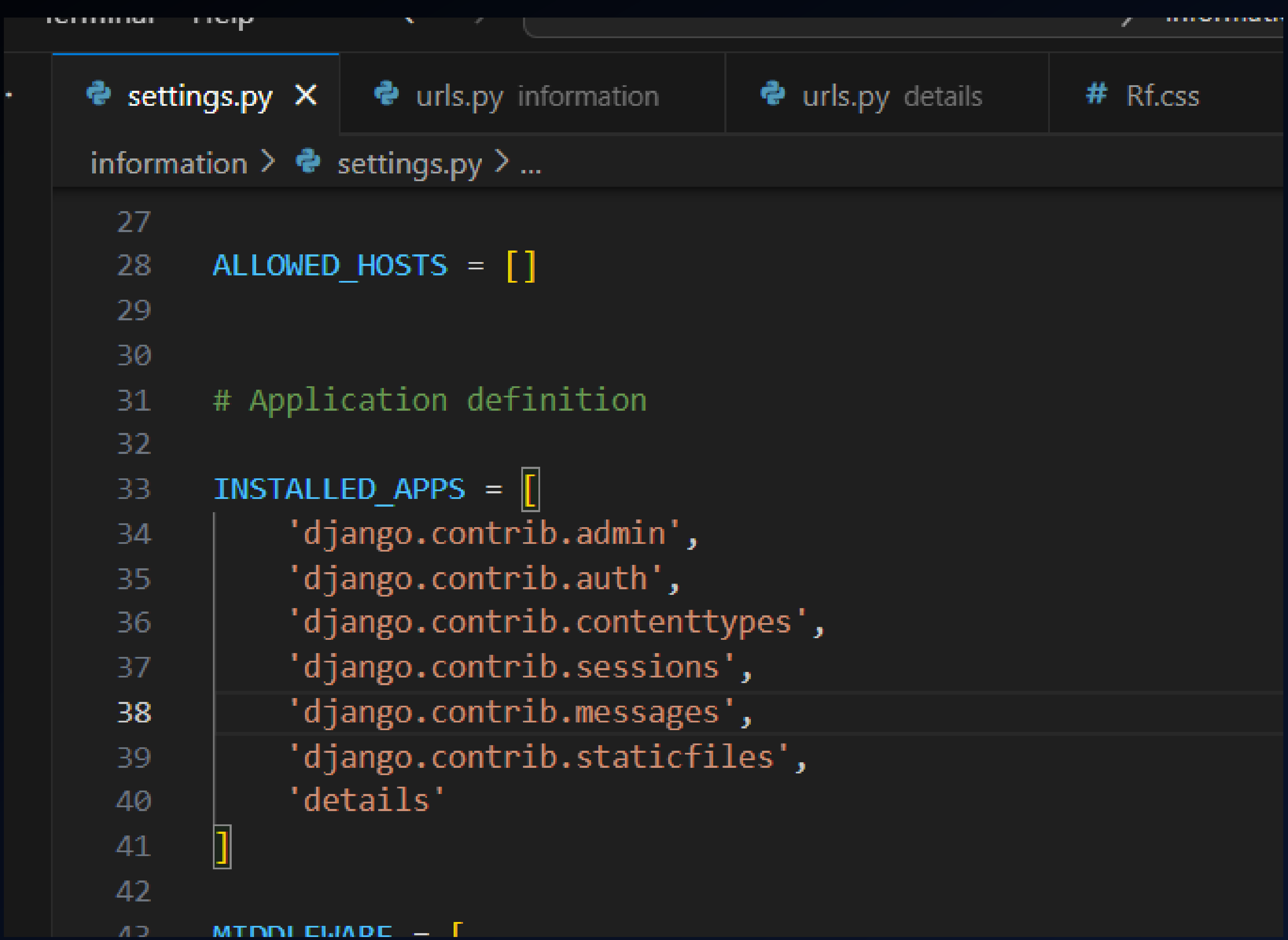
INSTRUCTIONS

- Directory name = INFO
- Project name = information
- App name = details
- Html page 1 = join.html
- Html page 2 = result.html
- Views.py functions = join(), result()
- Make the following changes



```
INFORMATION
  details
    __pycache__
    migrations
    static
      Rf.css
    templates
      join.html
      result.html
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  information
    __pycache__
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
  db.sqlite3
  manage.py
```

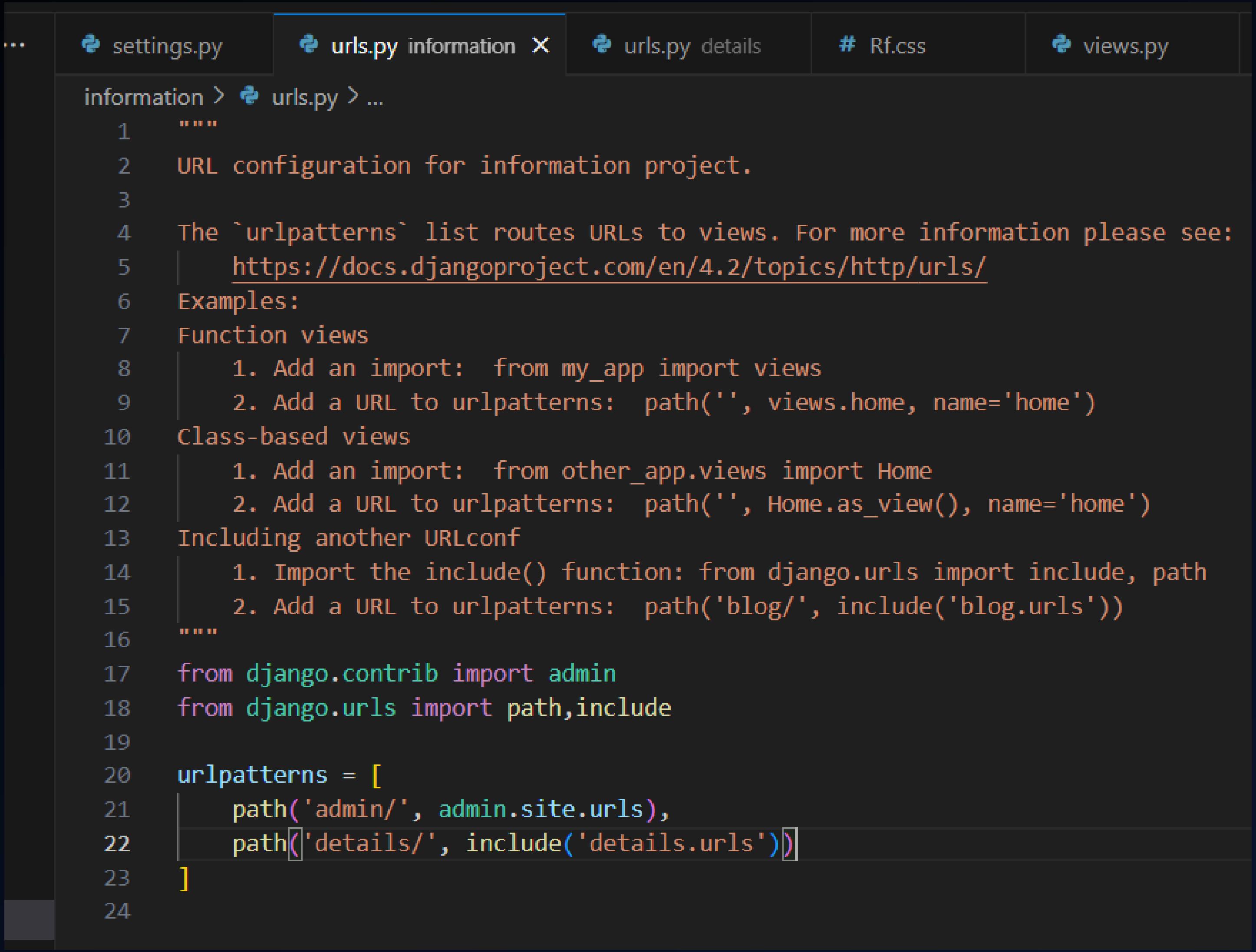
SETTINGS.PY



```python

ALLOWED_HOSTS = []


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'details'
]

```

ppt_by_sourabh_pal

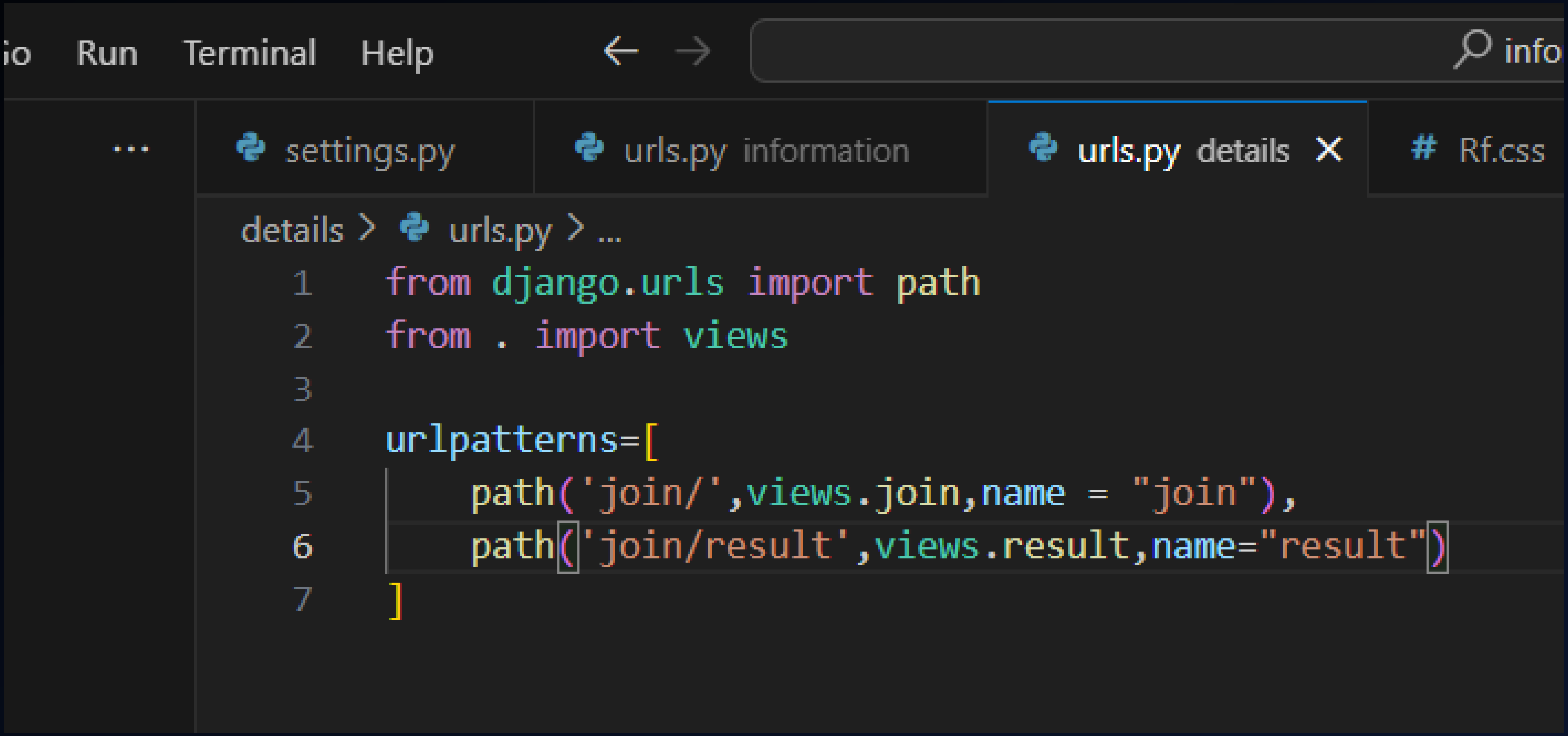
PROJECT – URLS.PY



```python
"""
URL configuration for information project.

The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/4.2/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path,include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('details/', include('details.urls'))
]

```

ppt_by_sourabh_pal

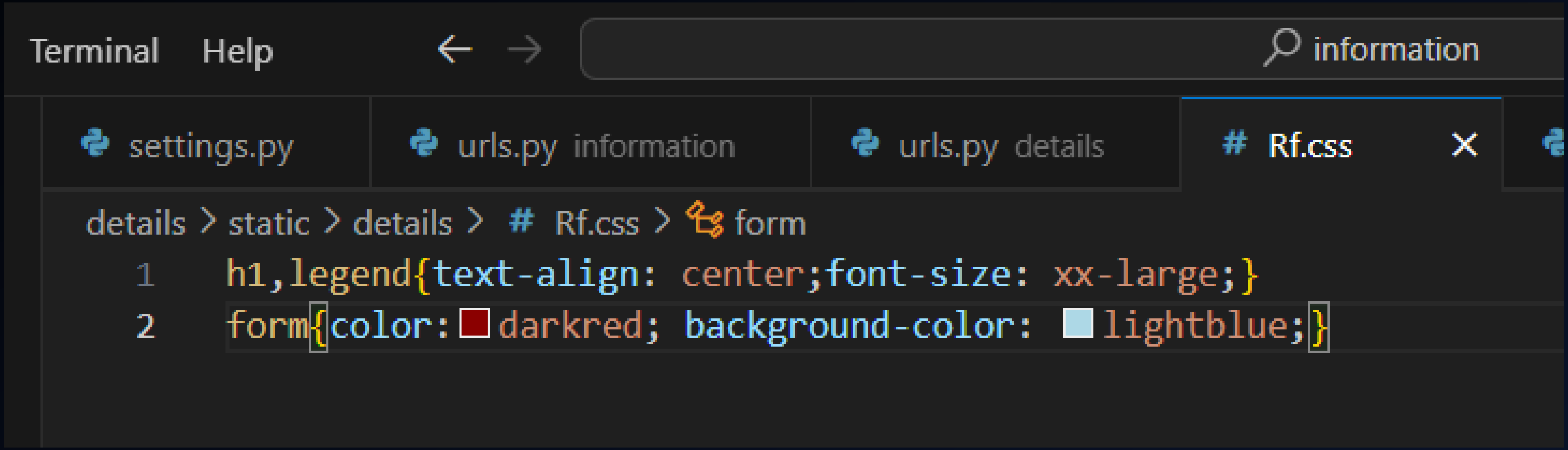
APP – URLS.PY



```python
from django.urls import path
from . import views

urlpatterns=[
    path('join/',views.join,name = "join"),
    path('join/result',views.result,name="result")
]
```

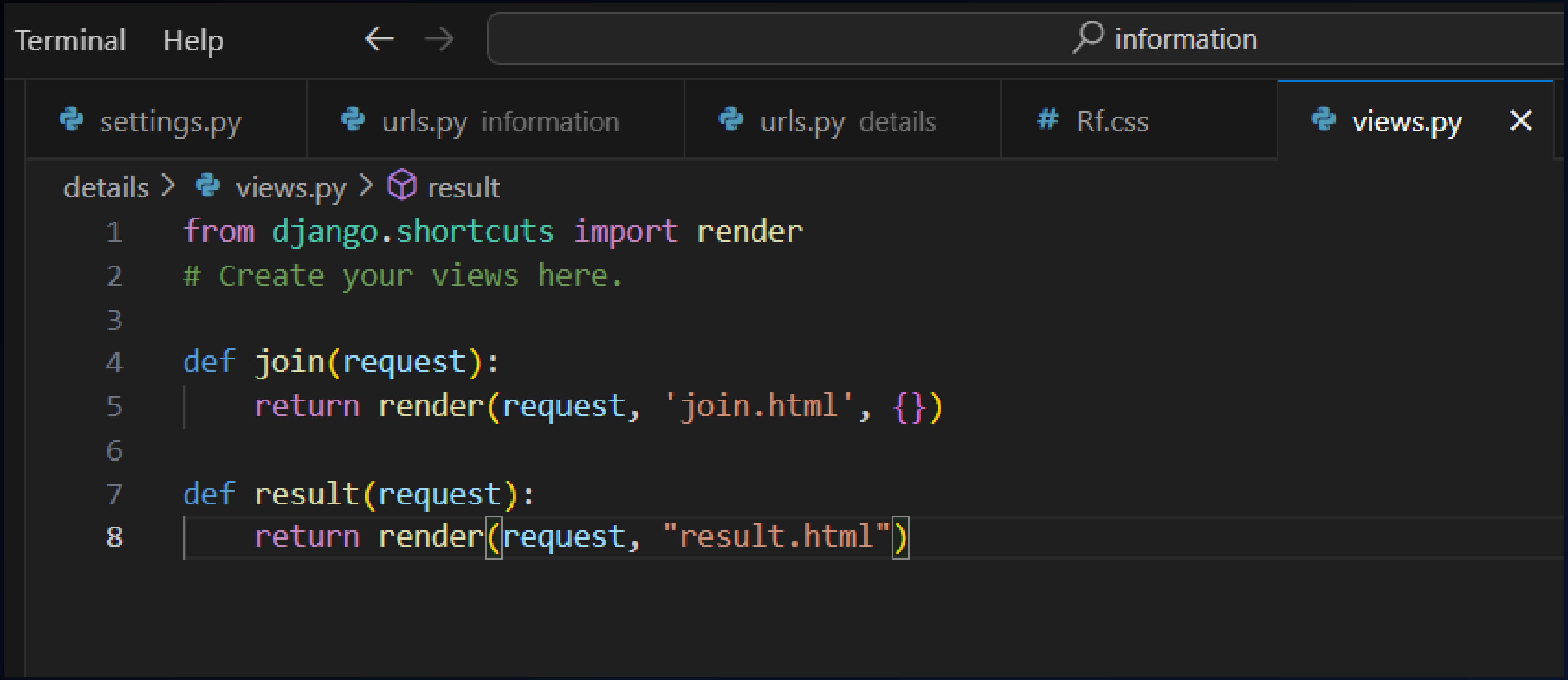
ppt_by_sourabh_pal

STYLESHEET – RF.CSS



```css
h1,legend{text-align: center;font-size: xx-large;}
form{color: darkred; background-color: lightblue;}
```

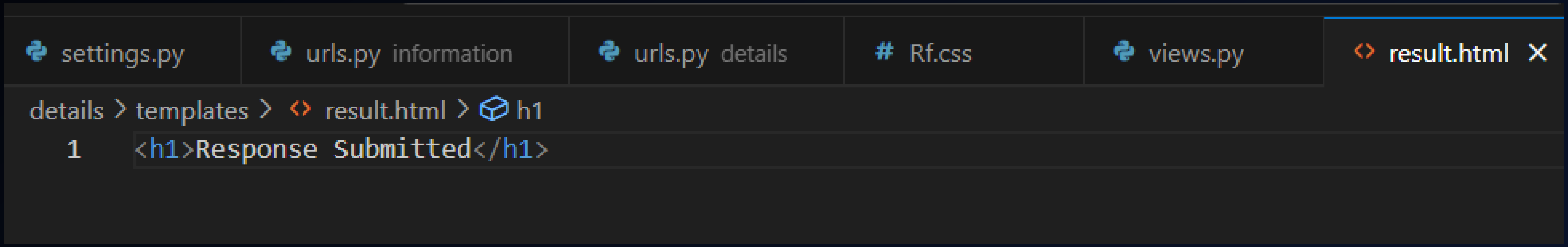
VIEWS.PY



```python
from django.shortcuts import render
# Create your views here.

def join(request):
    return render(request, 'join.html', {})

def result(request):
    return render(request, "result.html")
```

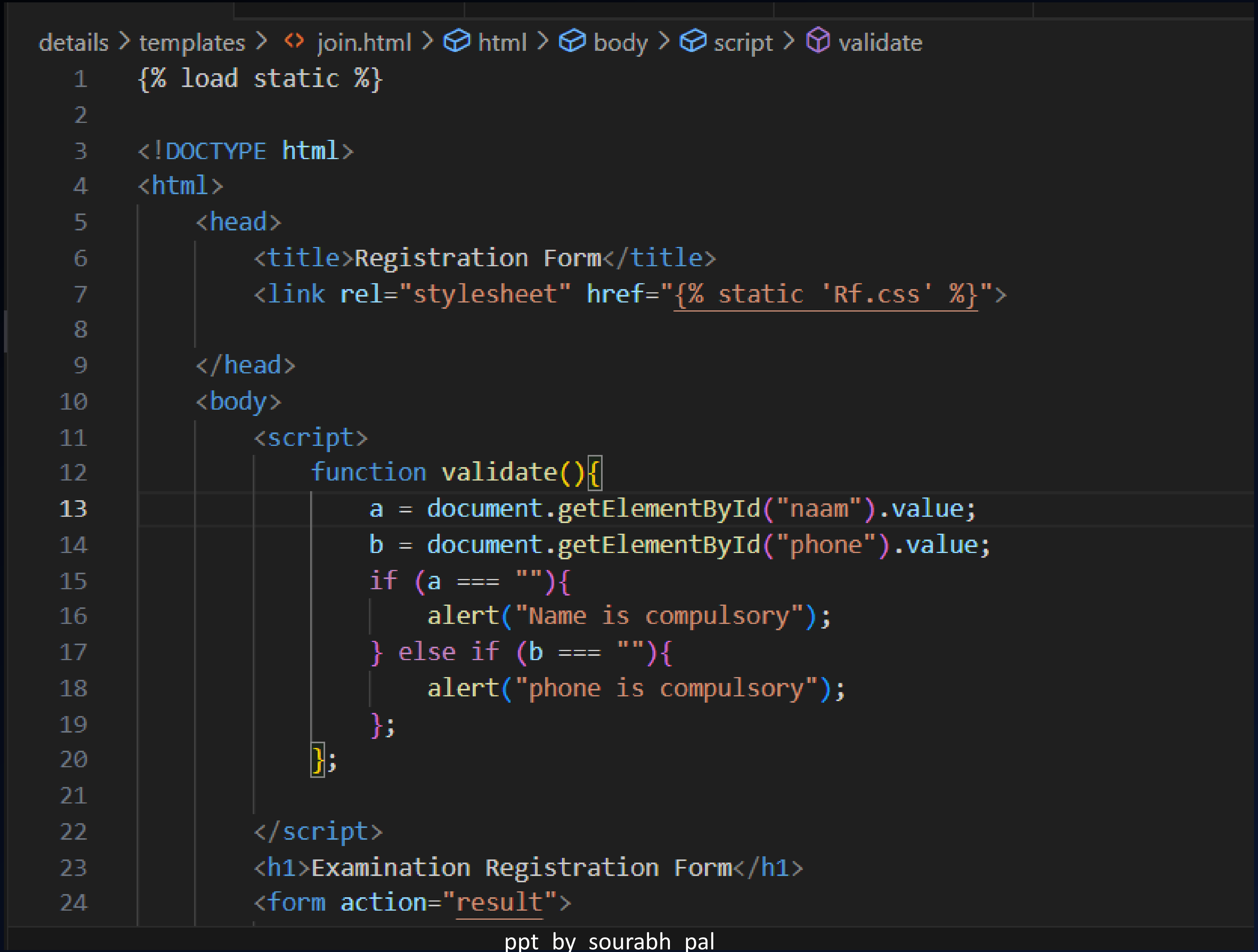
ppt_by_sourabh_pal

RESULT.HTML



```html
<h1>Response Submitted</h1>
```

JOIN.HTML - 1

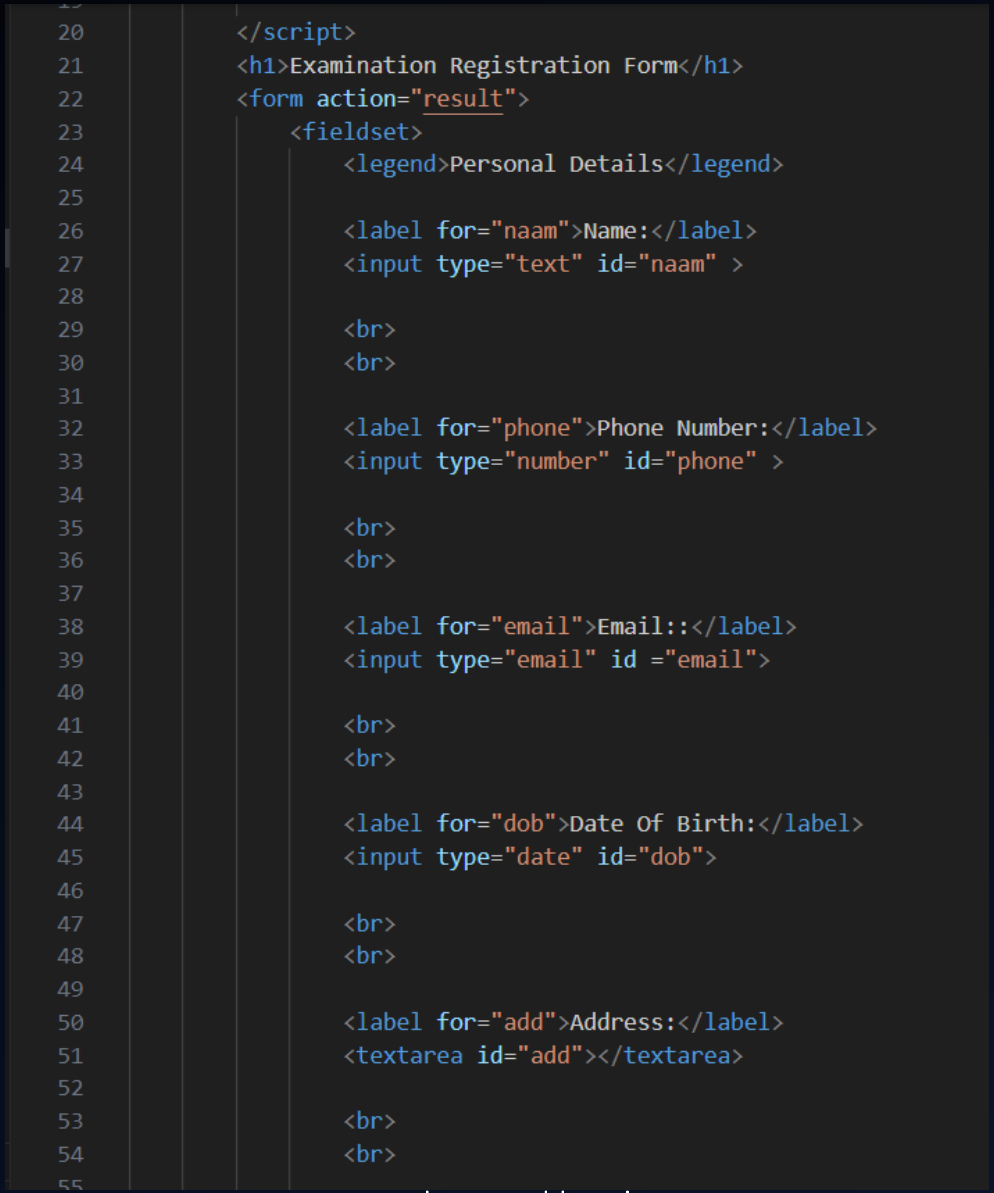


details > templates > join.html > html > body > script > validate

```
{% load static %}

<!DOCTYPE html>
<html>
    <head>
        <title>Registration Form</title>
        <link rel="stylesheet" href="{% static 'Rf.css' %}">

    </head>
    <body>
        <script>
            function validate(){
                a = document.getElementById("naam").value;
                b = document.getElementById("phone").value;
                if (a === ""){
                    alert("Name is compulsory");
                } else if (b === ""){
                    alert("phone is compulsory");
                };
            };

        </script>
        <h1>Examination Registration Form</h1>
        <form action="result">
```

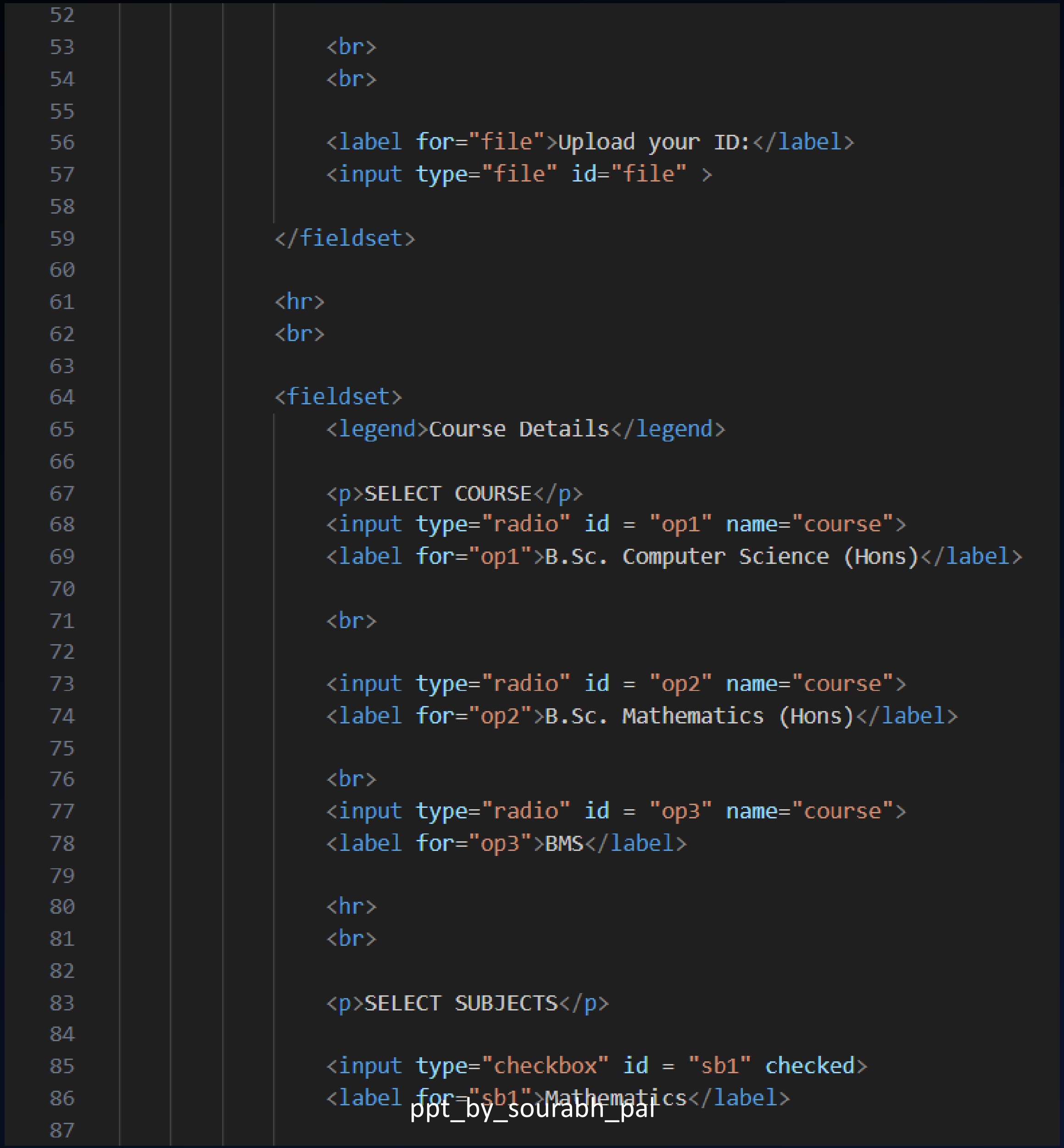
ppt_by_sourabh_pal

JOIN.HTML - 2



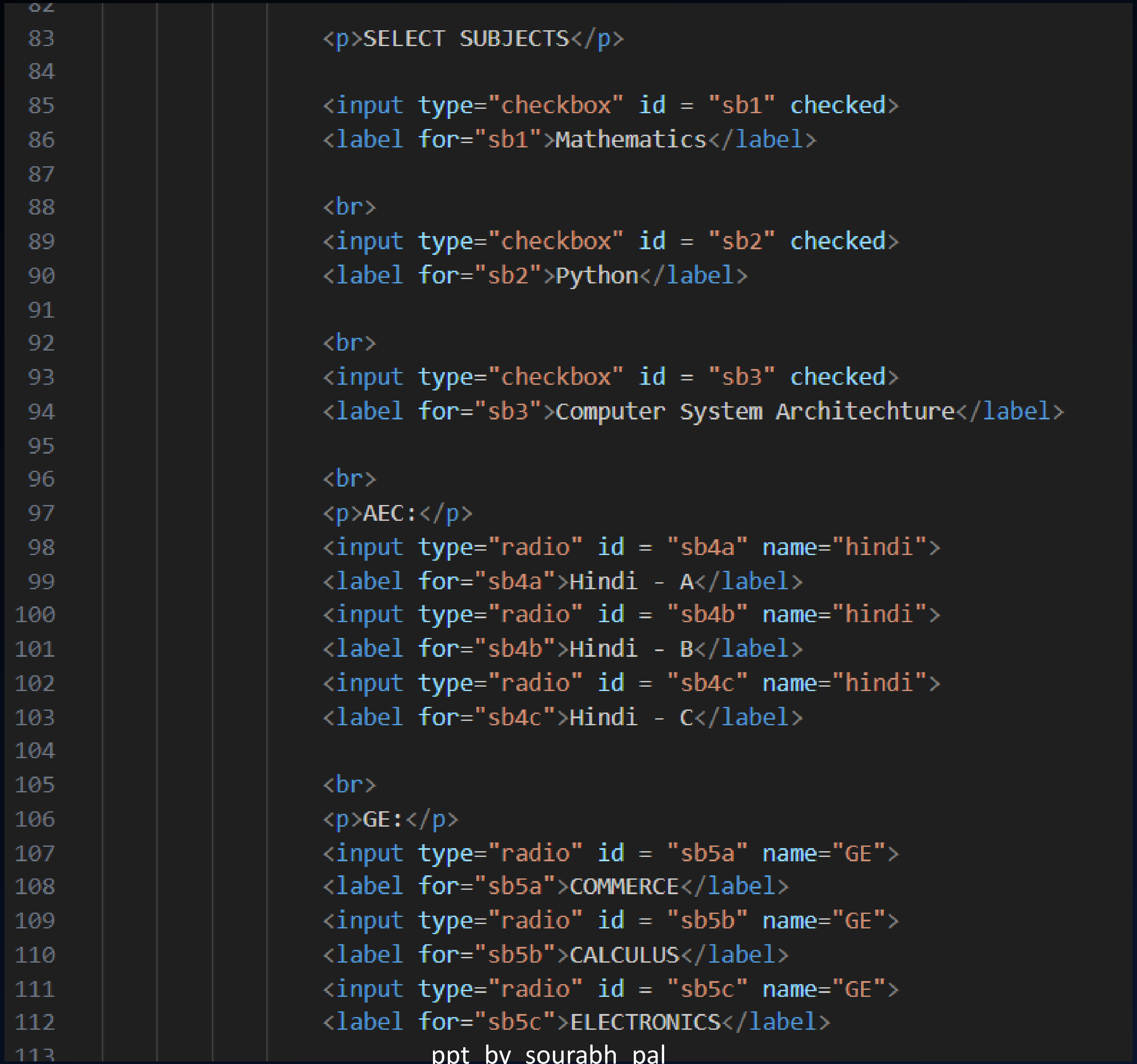
```
    </script>
    <h1>Examination Registration Form</h1>
    <form action="result">
        <fieldset>
            <legend>Personal Details</legend>

            <label for="naam">Name:</label>
            <input type="text" id="naam" >

            <br>
            <br>

            <label for="phone">Phone Number:</label>
            <input type="number" id="phone" >

            <br>
            <br>

            <label for="email">Email::</label>
            <input type="email" id ="email">

            <br>
            <br>

            <label for="dob">Date Of Birth:</label>
            <input type="date" id="dob">

            <br>
            <br>

            <label for="add">Address:</label>
            <textarea id="add"></textarea>

            <br>
            <br>
```

JOIN.HTML - 3



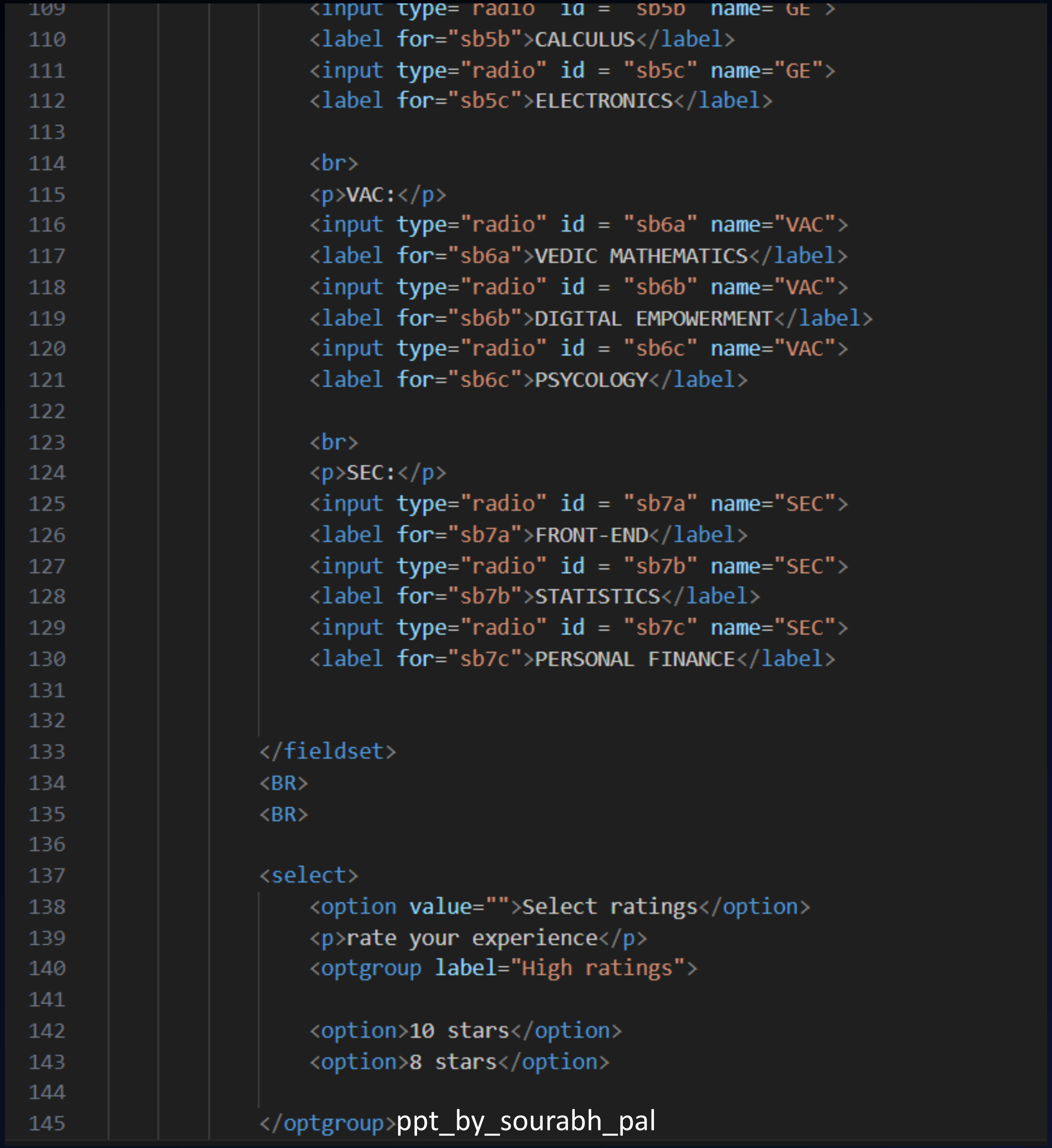
```html

                <br>
                <br>

                <label for="file">Upload your ID:</label>
                <input type="file" id="file" >

            </fieldset>

            <hr>
            <br>

            <fieldset>
                <legend>Course Details</legend>

                <p>SELECT COURSE</p>
                <input type="radio" id = "op1" name="course">
                <label for="op1">B.Sc. Computer Science (Hons)</label>

                <br>

                <input type="radio" id = "op2" name="course">
                <label for="op2">B.Sc. Mathematics (Hons)</label>

                <br>
                <input type="radio" id = "op3" name="course">
                <label for="op3">BMS</label>

                <hr>
                <br>

                <p>SELECT SUBJECTS</p>

                <input type="checkbox" id = "sb1" checked>
                <label for="sb1">Mathematics</label>

```

JOIN.HTML - 4



```
<p>SELECT SUBJECTS</p>

<input type="checkbox" id = "sb1" checked>
<label for="sb1">Mathematics</label>

<br>
<input type="checkbox" id = "sb2" checked>
<label for="sb2">Python</label>

<br>
<input type="checkbox" id = "sb3" checked>
<label for="sb3">Computer System Architechture</label>

<br>
<p>AEC:</p>
<input type="radio" id = "sb4a" name="hindi">
<label for="sb4a">Hindi - A</label>
<input type="radio" id = "sb4b" name="hindi">
<label for="sb4b">Hindi - B</label>
<input type="radio" id = "sb4c" name="hindi">
<label for="sb4c">Hindi - C</label>

<br>
<p>GE:</p>
<input type="radio" id = "sb5a" name="GE">
<label for="sb5a">COMMERCE</label>
<input type="radio" id = "sb5b" name="GE">
<label for="sb5b">CALCULUS</label>
<input type="radio" id = "sb5c" name="GE">
<label for="sb5c">ELECTRONICS</label>
```

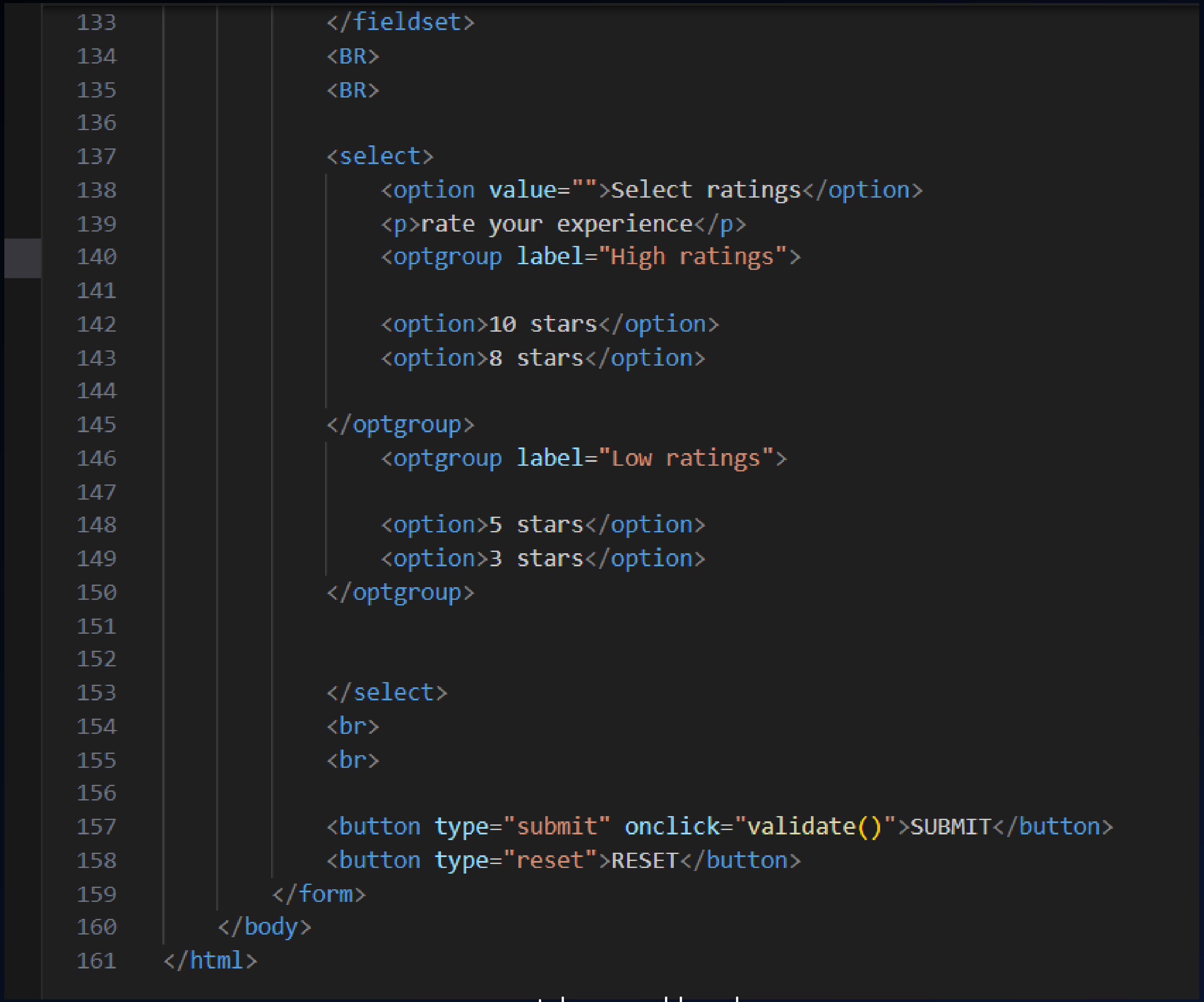
JOIN.HTML - 5



```
                <input type="radio" id = "sb5b" name="GE">
                <label for="sb5b">CALCULUS</label>
                <input type="radio" id = "sb5c" name="GE">
                <label for="sb5c">ELECTRONICS</label>

                <br>
                <p>VAC:</p>
                <input type="radio" id = "sb6a" name="VAC">
                <label for="sb6a">VEDIC MATHEMATICS</label>
                <input type="radio" id = "sb6b" name="VAC">
                <label for="sb6b">DIGITAL EMPOWERMENT</label>
                <input type="radio" id = "sb6c" name="VAC">
                <label for="sb6c">PSYCOLOGY</label>

                <br>
                <p>SEC:</p>
                <input type="radio" id = "sb7a" name="SEC">
                <label for="sb7a">FRONT-END</label>
                <input type="radio" id = "sb7b" name="SEC">
                <label for="sb7b">STATISTICS</label>
                <input type="radio" id = "sb7c" name="SEC">
                <label for="sb7c">PERSONAL FINANCE</label>


            </fieldset>
            <BR>
            <BR>

            <select>
                <option value="">Select ratings</option>
                <p>rate your experience</p>
                <optgroup label="High ratings">

                <option>10 stars</option>
                <option>8 stars</option>

            </optgroup>
```

ppt_by_sourabh_pal

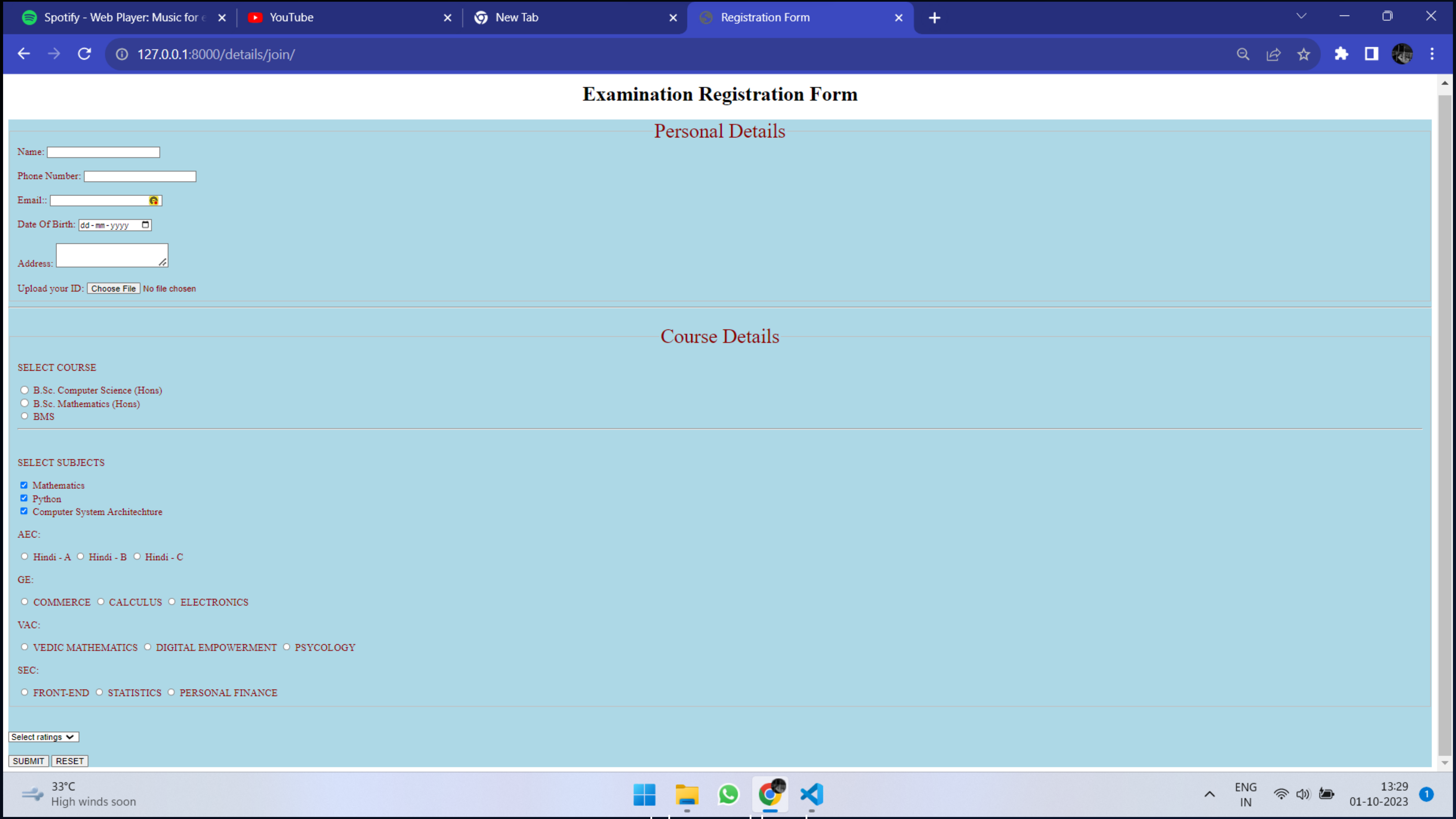
JOIN.HTML - 6



```
            </fieldset>
            <BR>
            <BR>

            <select>
                <option value="">Select ratings</option>
                <p>rate your experience</p>
                <optgroup label="High ratings">

                <option>10 stars</option>
                <option>8 stars</option>

            </optgroup>
                <optgroup label="Low ratings">

                <option>5 stars</option>
                <option>3 stars</option>
            </optgroup>


            </select>
            <br>
            <br>

            <button type="submit" onclick="validate()">SUBMIT</button>
            <button type="reset">RESET</button>
        </form>
    </body>
</html>
```

ppt_by_sourabh_pal

BROWSER



Examination Registration Form

Personal Details

Name:

Phone Number:

Email::

Date Of Birth: dd-mm-yyyy

Address:

Upload your ID: Choose File No file chosen

Course Details

SELECT COURSE

- B.Sc. Computer Science (Hons)
- B.Sc. Mathematics (Hons)
- BMS

SELECT SUBJECTS

- [✓] Mathematics
- [✓] Python
- [✓] Computer System Architechture

AEC:

Hindi - A   Hindi - B   Hindi - C

GE:

COMMERCE   CALCULUS   ELECTRONICS

VAC:

VEDIC MATHEMATICS   DIGITAL EMPOWERMENT   PSYCOLOGY

SEC:

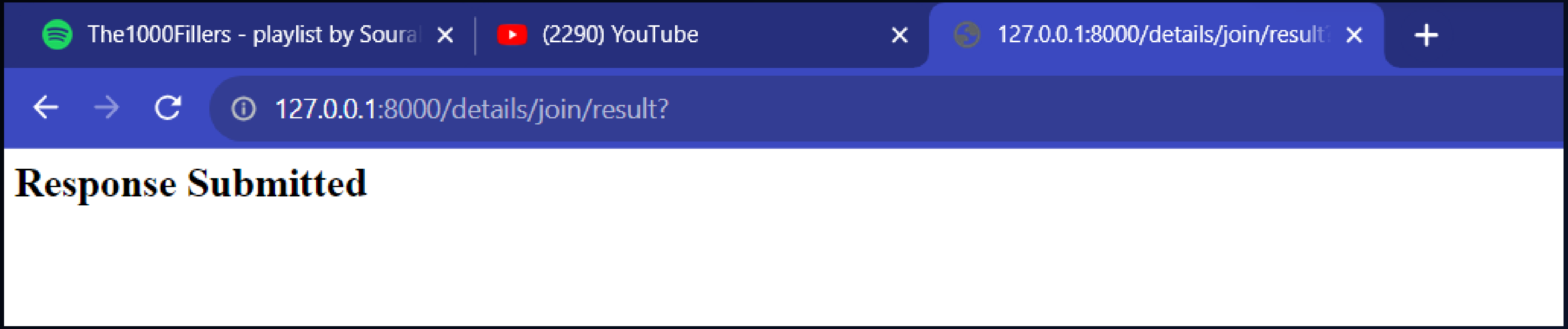
FRONT-END   STATISTICS   PERSONAL FINANCE

Select ratings

SUBMIT   RESET

ppt_by_sourabh_pal

BROWSER – AFTER SUBMIT


The1000Fillers - playlist by Soura
(2290) YouTube
127.0.0.1:8000/details/join/result
127.0.0.1:8000/details/join/result?

Response Submitted

DAY 6

EVALUATION FUNCTIONS

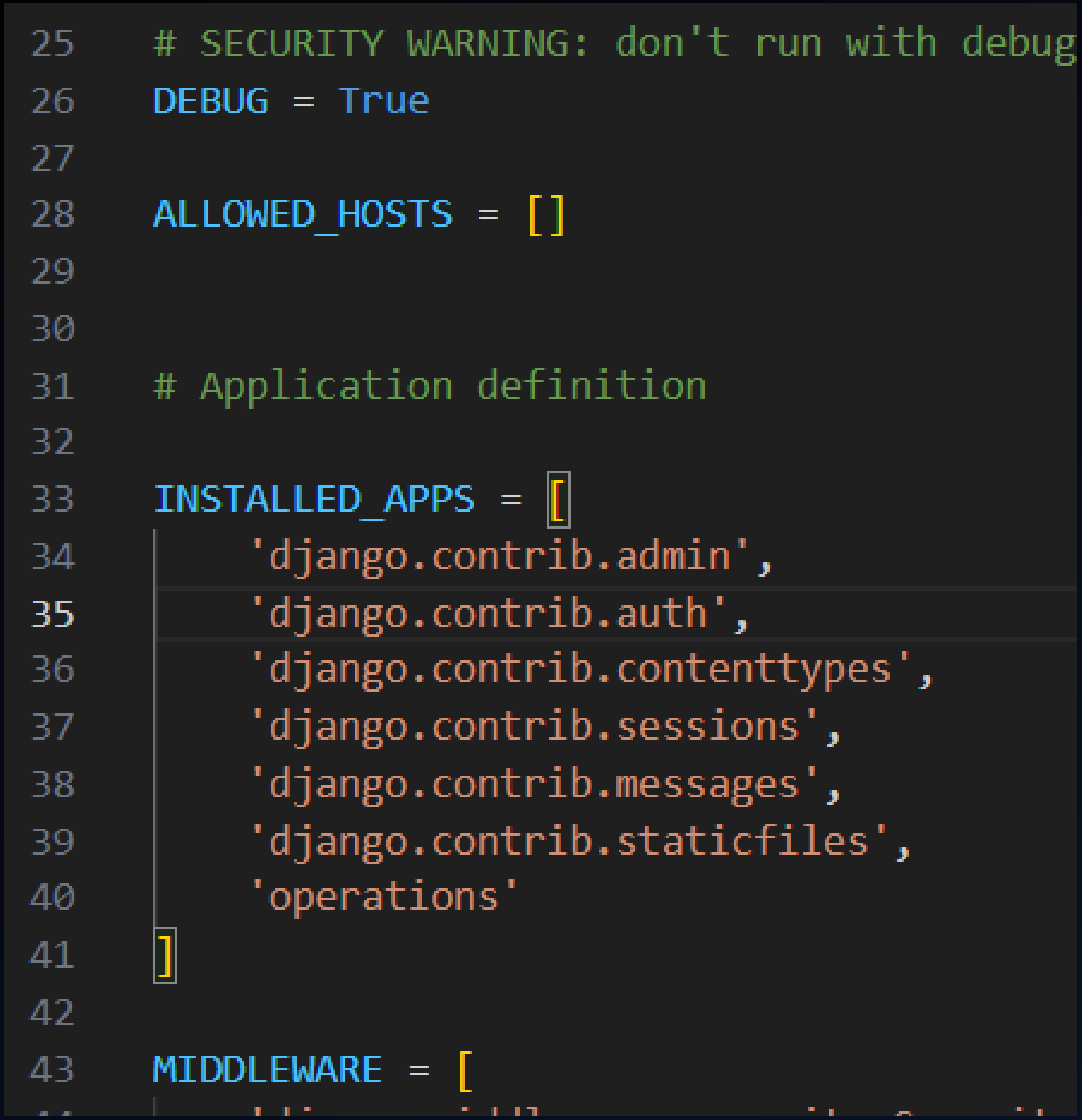
ARITHMETIC OPERATIONS AND RESULT DISPLAY ON THE CLIENT PAGE

ppt_by_sourabh_pal

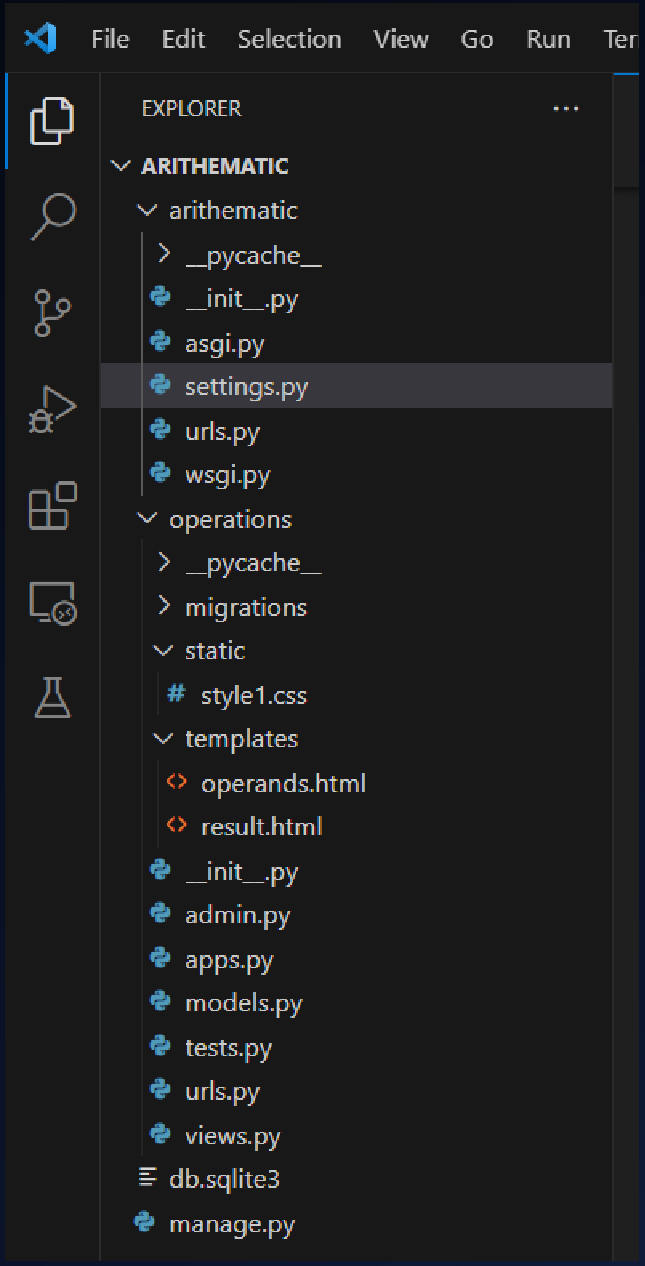
INSTRUCTIONS

- Directory name = Arithematic
- Project name = arithematic
- App name = operations
- Html page 1 =operands.html
- Html page 2 = result.html
- Views.py functions = enter(), add()
- Make the following changes

SETTINGS.PY

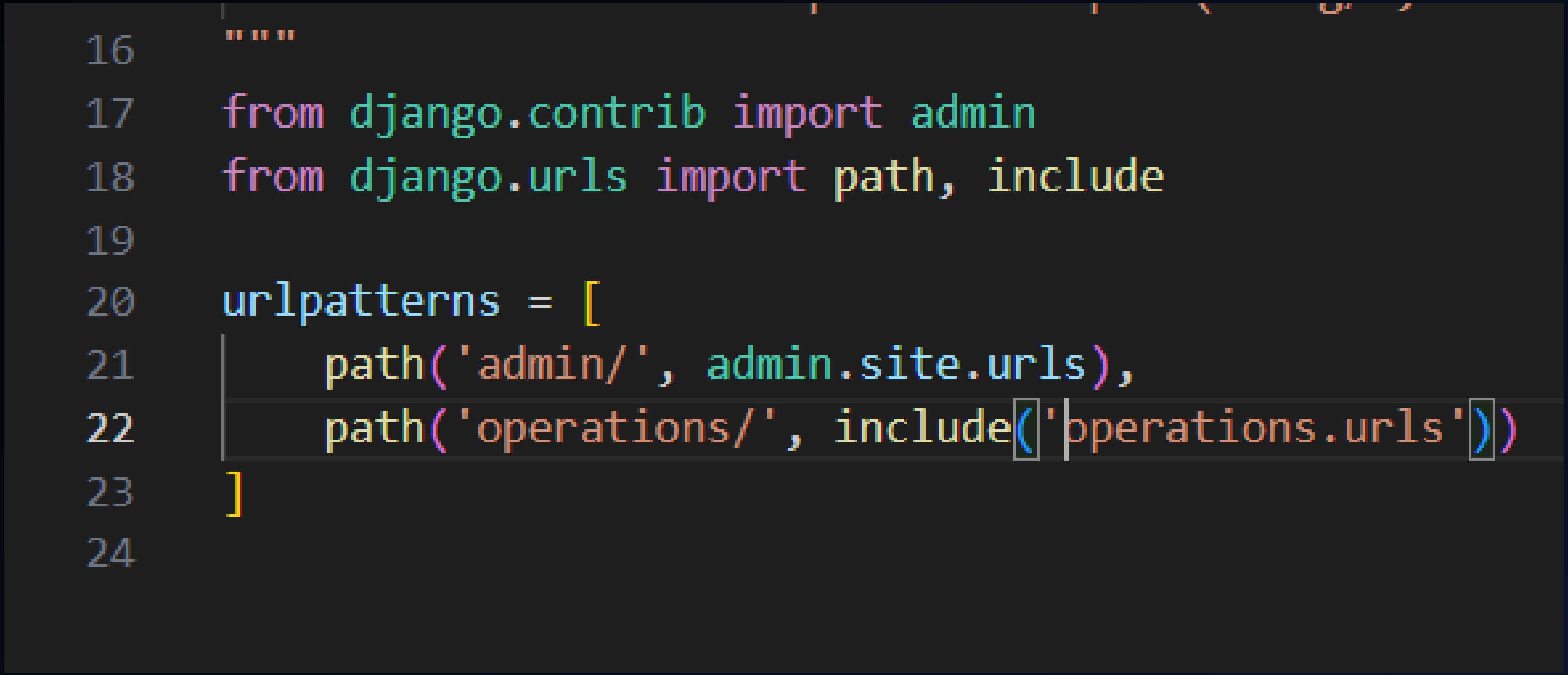


```python
# SECURITY WARNING: don't run with debug
DEBUG = True

ALLOWED_HOSTS = []


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'operations'
]

MIDDLEWARE = [
```



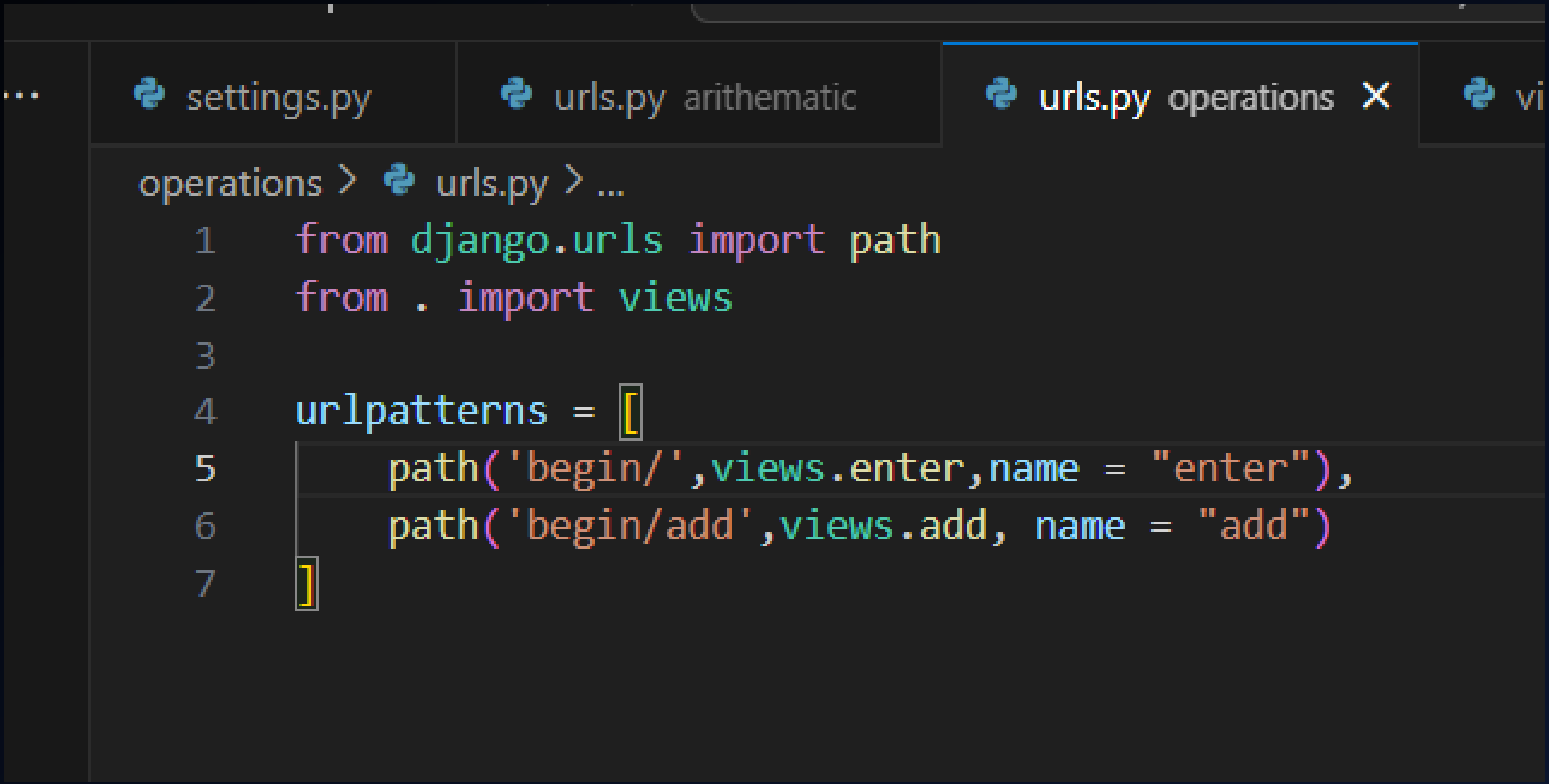
```
ARITHEMATIC
  arithematic
    __pycache__
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
  operations
    __pycache__
    migrations
    static
      style1.css
    templates
      operands.html
      result.html
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  db.sqlite3
  manage.py
```

PROJECT – URLS.PY



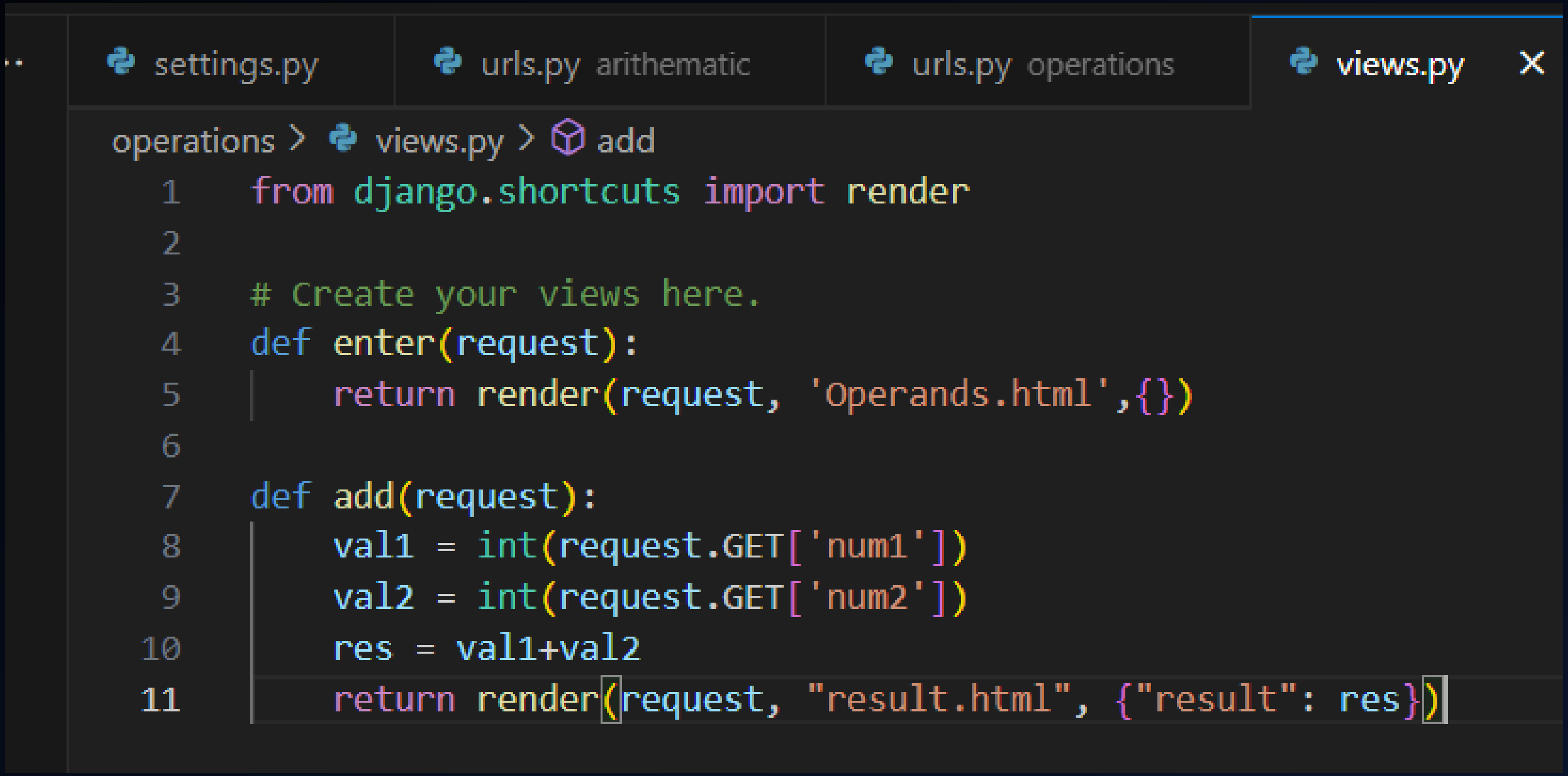
```
"""
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('operations/', include('operations.urls'))
]

```

APP – URLS.PY



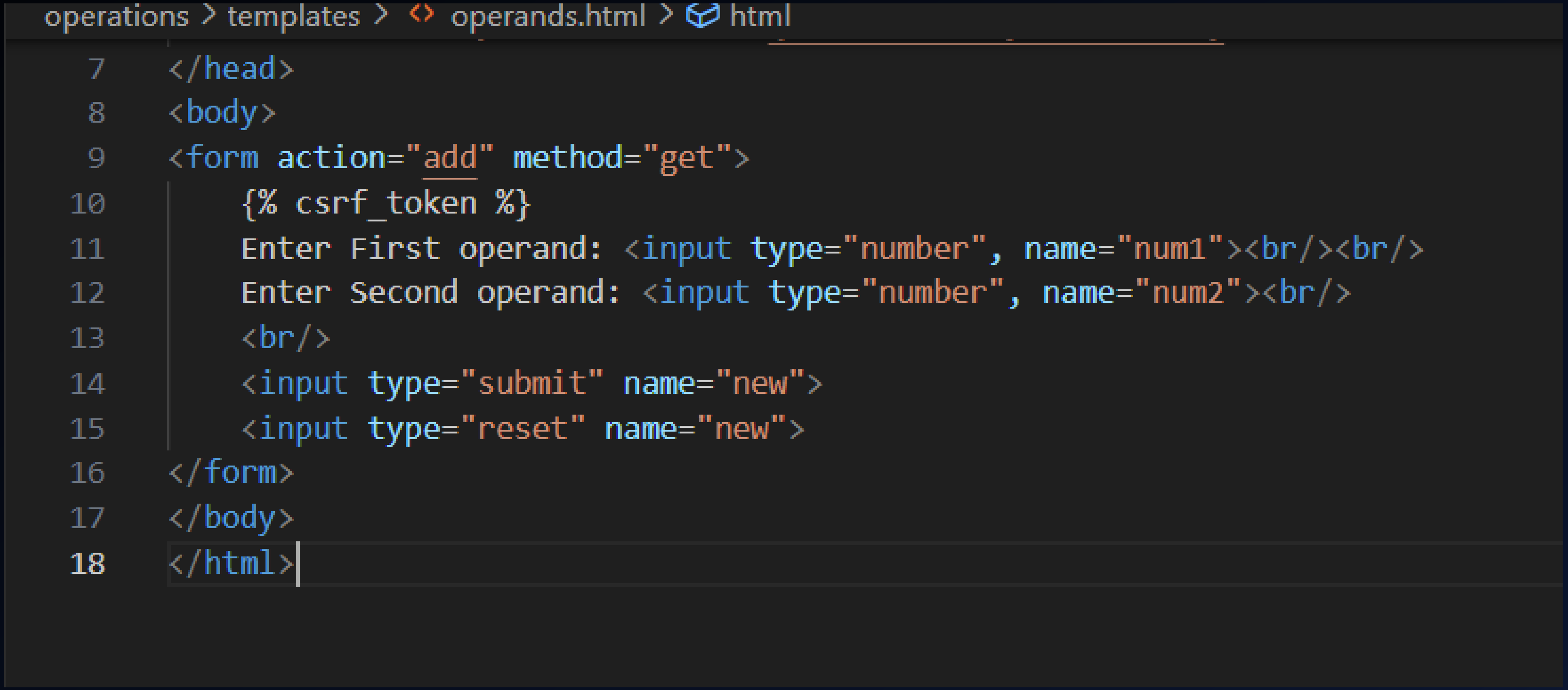
```python
from django.urls import path
from . import views

urlpatterns = [
    path('begin/',views.enter,name = "enter"),
    path('begin/add',views.add, name = "add")
]
```

VIEWS.PY



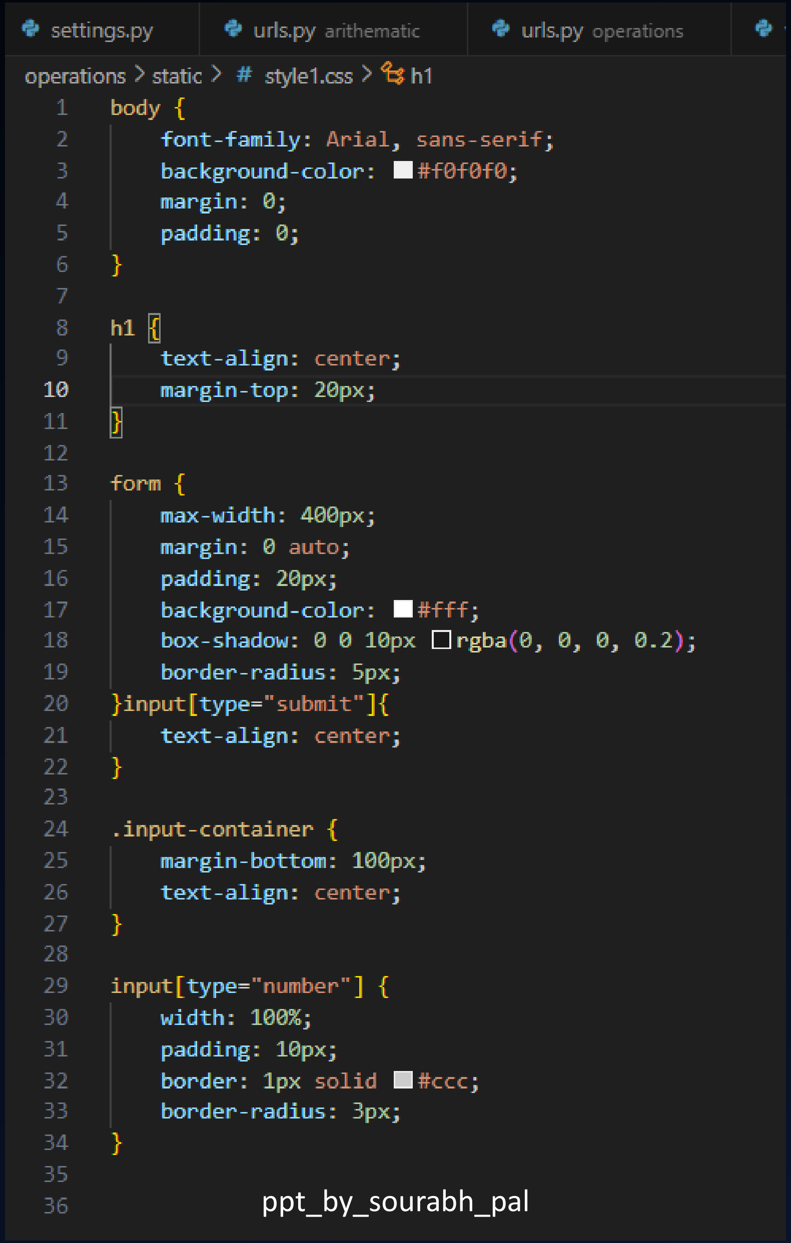
```python
from django.shortcuts import render

# Create your views here.
def enter(request):
    return render(request, 'Operands.html',{})

def add(request):
    val1 = int(request.GET['num1'])
    val2 = int(request.GET['num2'])
    res = val1+val2
    return render(request, "result.html", {"result": res})
```

OPERANDS.HTML



```
</head>
<body>
<form action="add" method="get">
    {% csrf_token %}
    Enter First operand: <input type="number", name="num1"><br/><br/>
    Enter Second operand: <input type="number", name="num2"><br/>
    <br/>
    <input type="submit" name="new">
    <input type="reset" name="new">
</form>
</body>
</html>
```

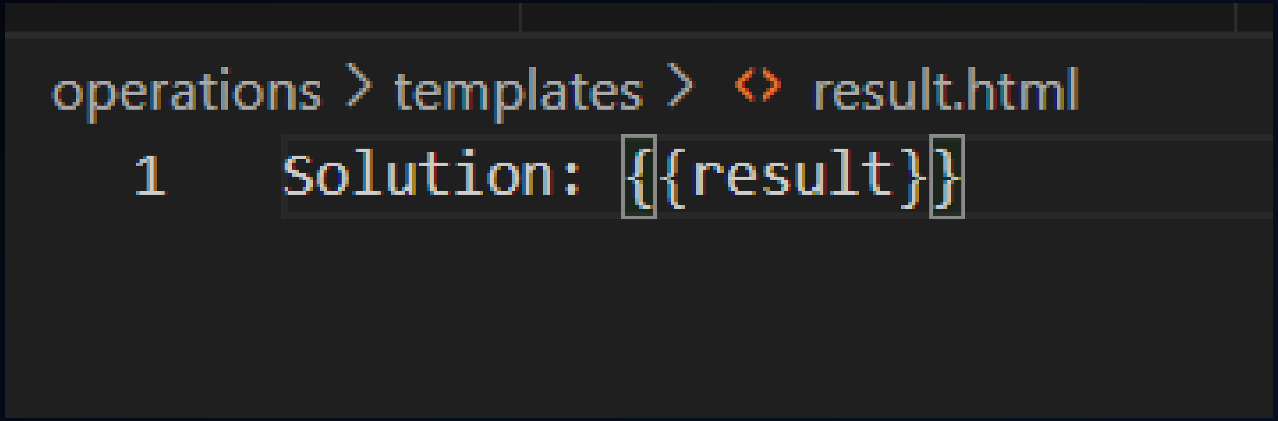
STYLE1.CSS



```css
body {
    font-family: Arial, sans-serif;
    background-color: #f0f0f0;
    margin: 0;
    padding: 0;
}

h1 {
    text-align: center;
    margin-top: 20px;
}

form {
    max-width: 400px;
    margin: 0 auto;
    padding: 20px;
    background-color: #fff;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
    border-radius: 5px;
}input[type="submit"]{
    text-align: center;
}

.input-container {
    margin-bottom: 100px;
    text-align: center;
}

input[type="number"] {
    width: 100%;
    padding: 10px;
    border: 1px solid #ccc;
    border-radius: 3px;
}

```

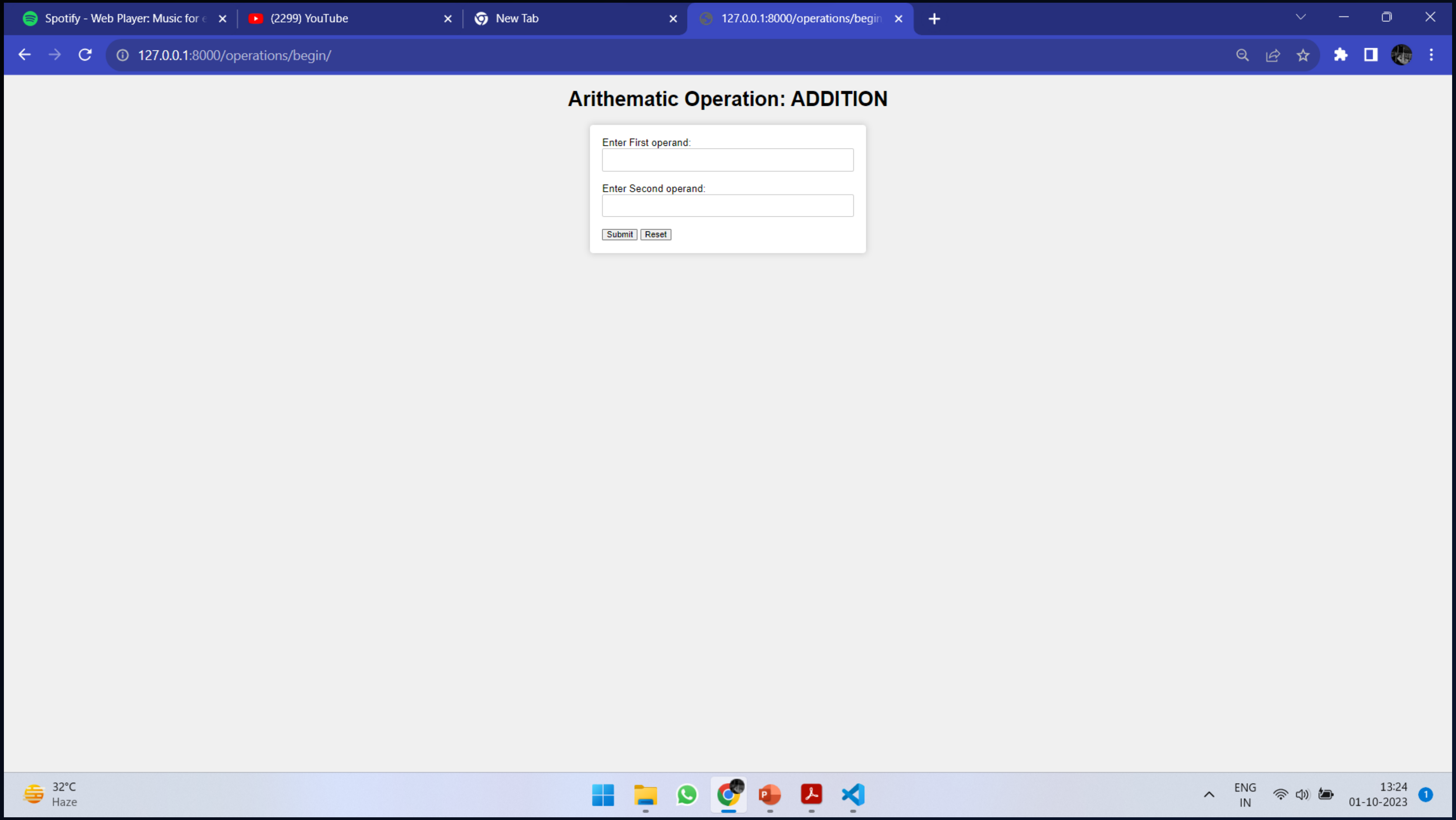
ppt_by_sourabh_pal

RESULT.HTML



```
Solution: {{result}}
```

BROWSER


Spotify - Web Player: Music for
(2299) YouTube
New Tab
127.0.0.1:8000/operations/begin
127.0.0.1:8000/operations/begin/

Arithematic Operation: ADDITION

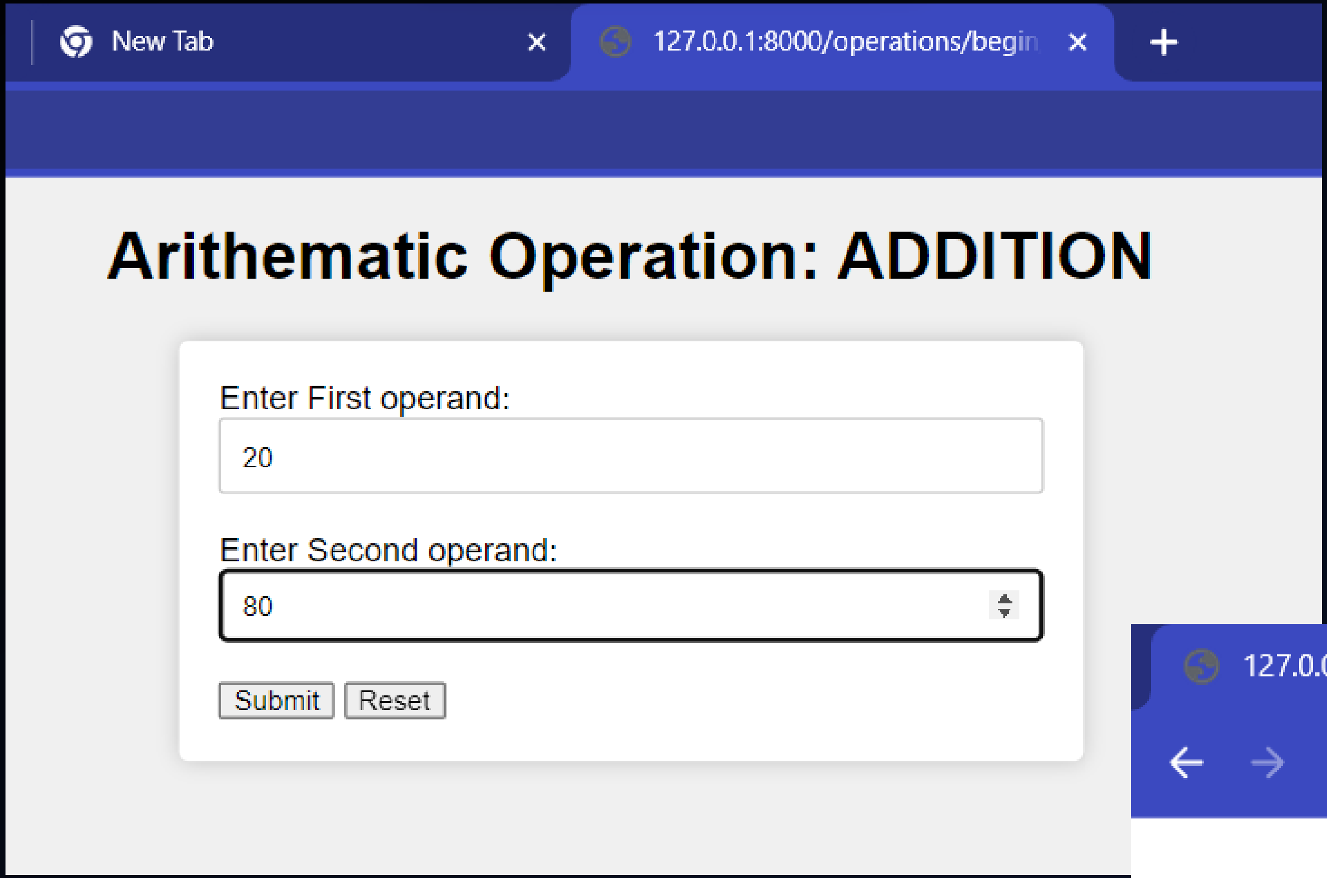
Enter First operand:

Enter Second operand:

Submit
Reset
32°C
Haze
ENG
IN
13:24
01-10-2023

ppt_by_sourabh_pal

BROWSER

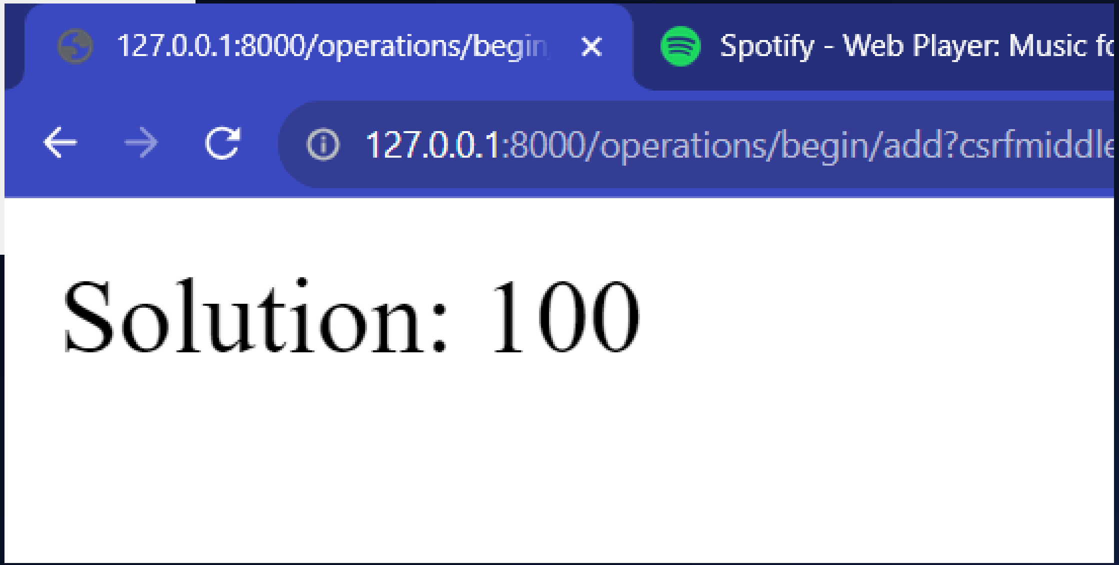

New Tab
127.0.0.1:8000/operations/begin

Arithematic Operation: ADDITION

Enter First operand:
20

Enter Second operand:
80

Submit Reset


127.0.0.1:8000/operations/begin
Spotify - Web Player: Music fo

127.0.0.1:8000/operations/begin/add?csrfmiddle

Solution: 100

ppt_by_sourabh_pal

DAY 7

Creating a Local Library Model

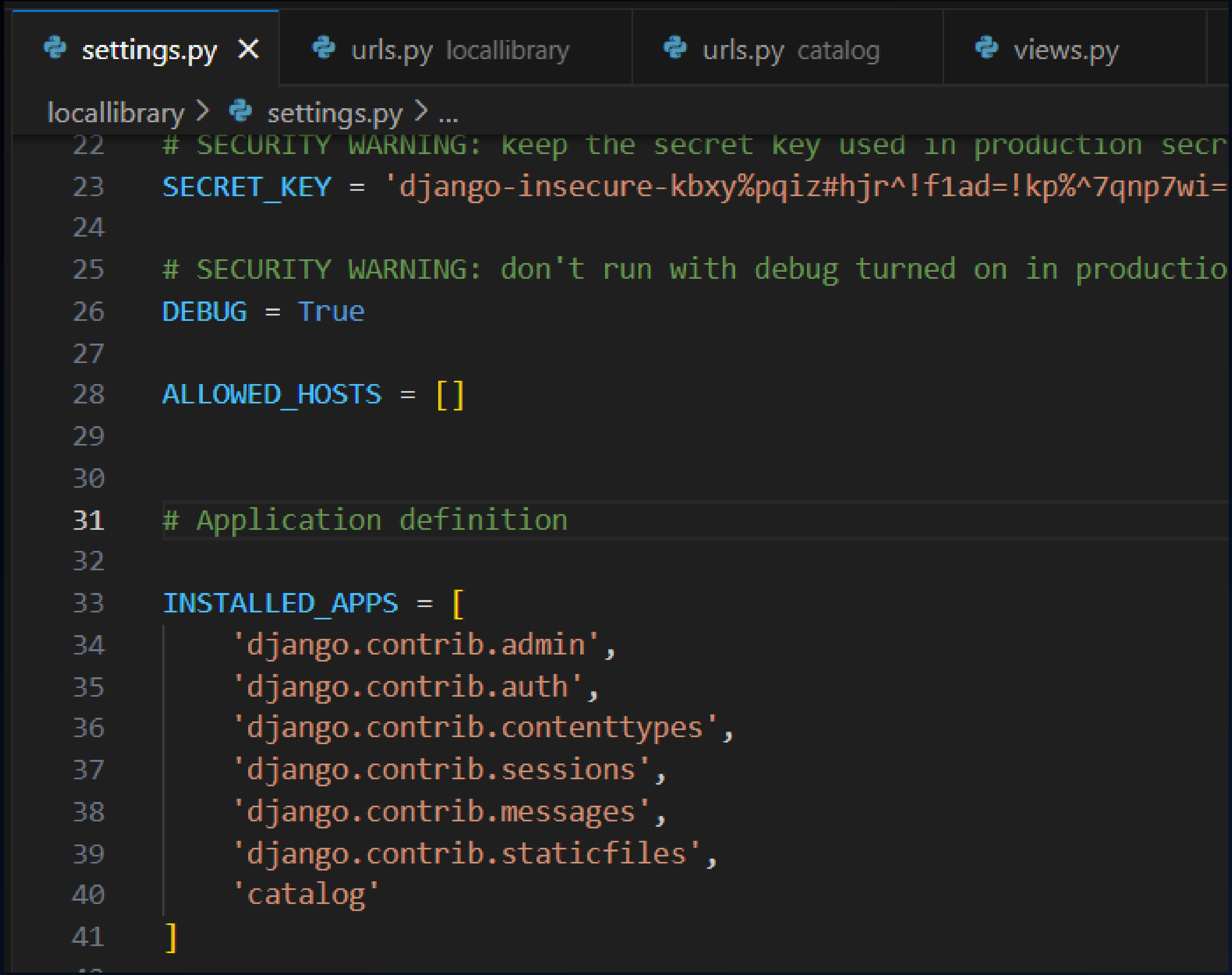
AUTHORS AND THEIR BOOKS

ppt_by_sourabh_pal

INSTRUCTIONS

- DIRECTORY NAME: django_project1
- PROJECT NAME: locallibrary
- APP NAME: catalog
- Create environment, project, app, superuser, directories [templates, static (image and stylesheet), urls.py(apps)], HTML page (index.html)
- After creating the models, migrate the contents using the following commands
- Python manage.py makemigrations
- Python manage.py migrate

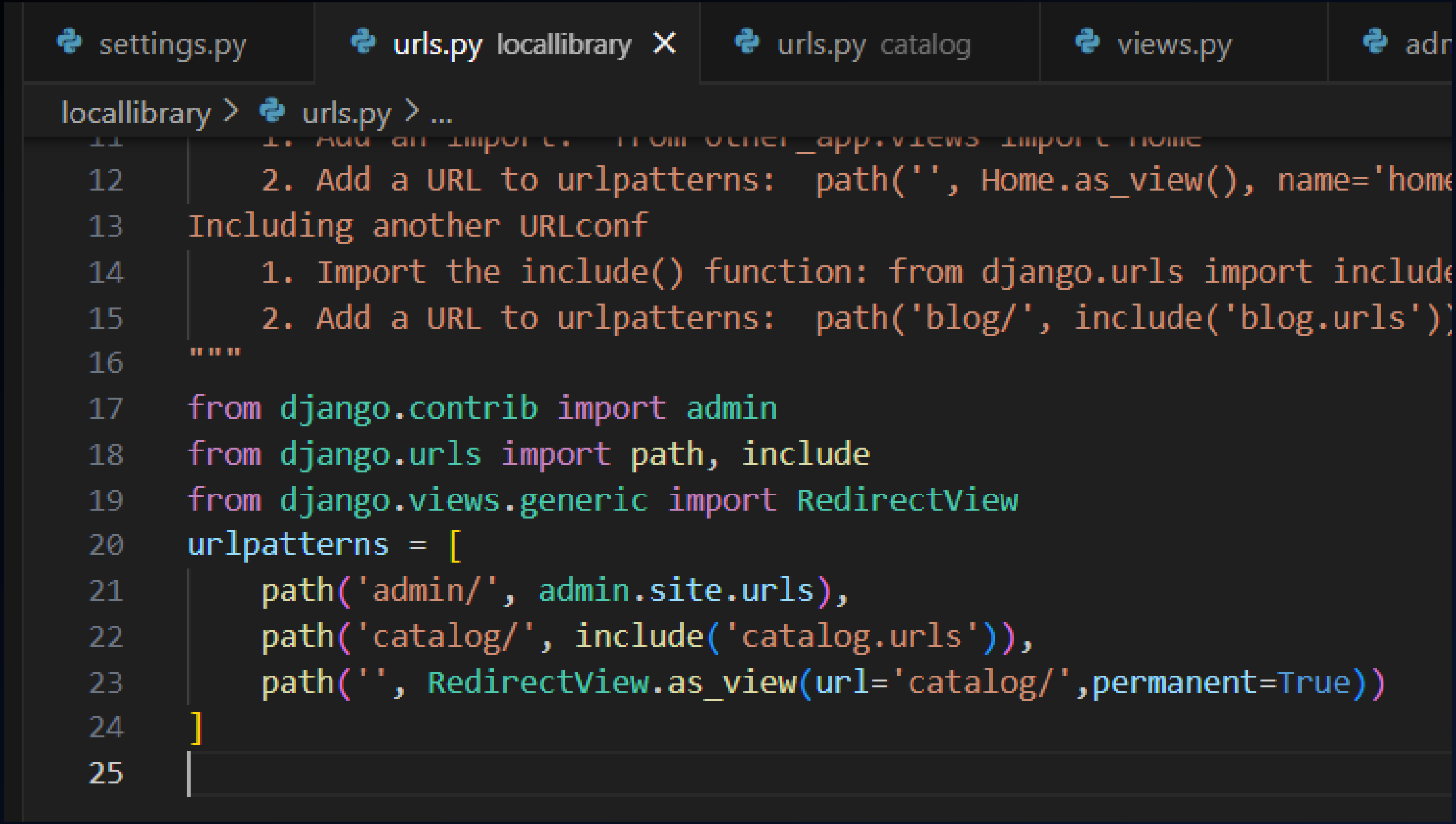
SETTINGS.PY



```
# SECURITY WARNING: keep the secret key used in production secr
SECRET_KEY = 'django-insecure-kbxy%pqiz#hjr^!f1ad=!kp%^7qnp7wi=

# SECURITY WARNING: don't run with debug turned on in productio
DEBUG = True

ALLOWED_HOSTS = []


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'catalog'
]
```

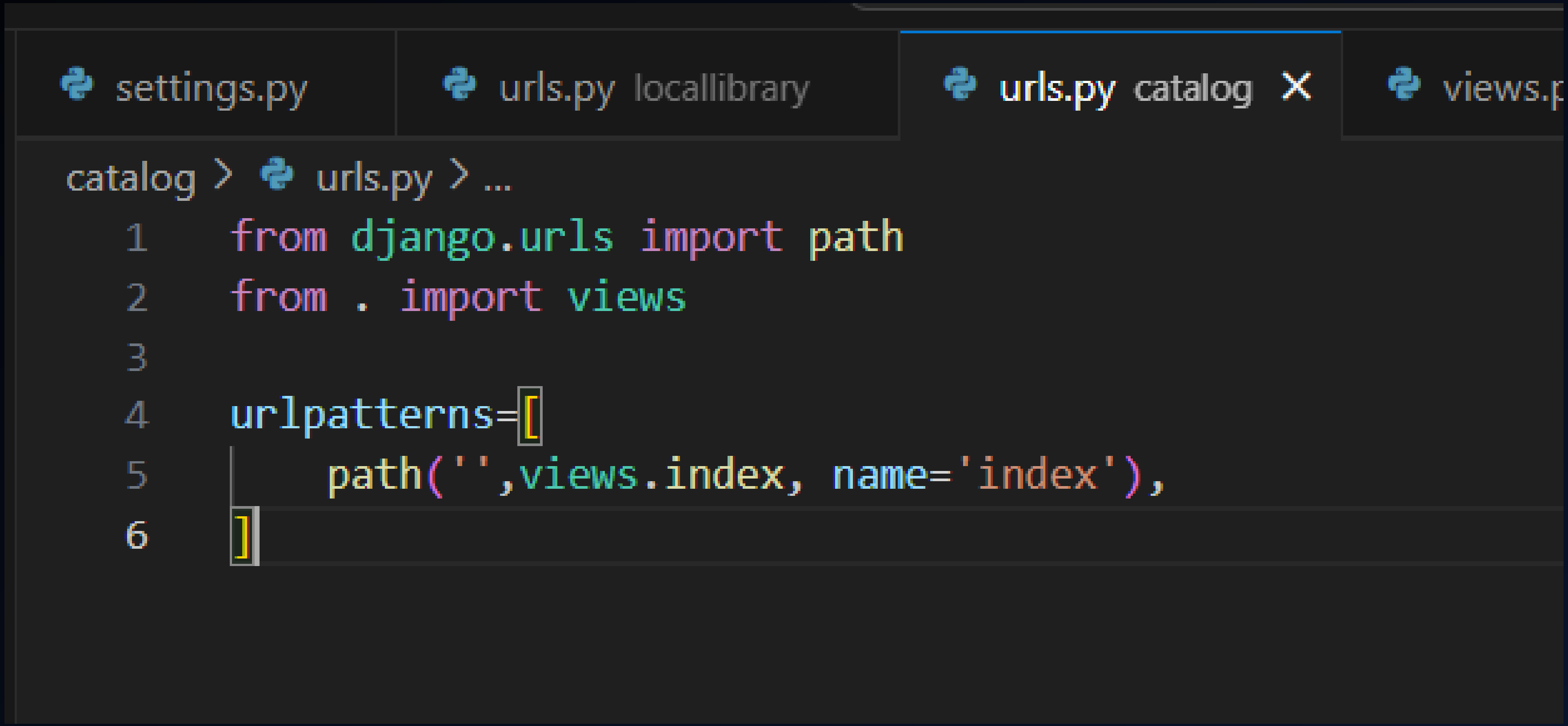
ppt_by_sourabh_pal

URLS.PY - PROJECT



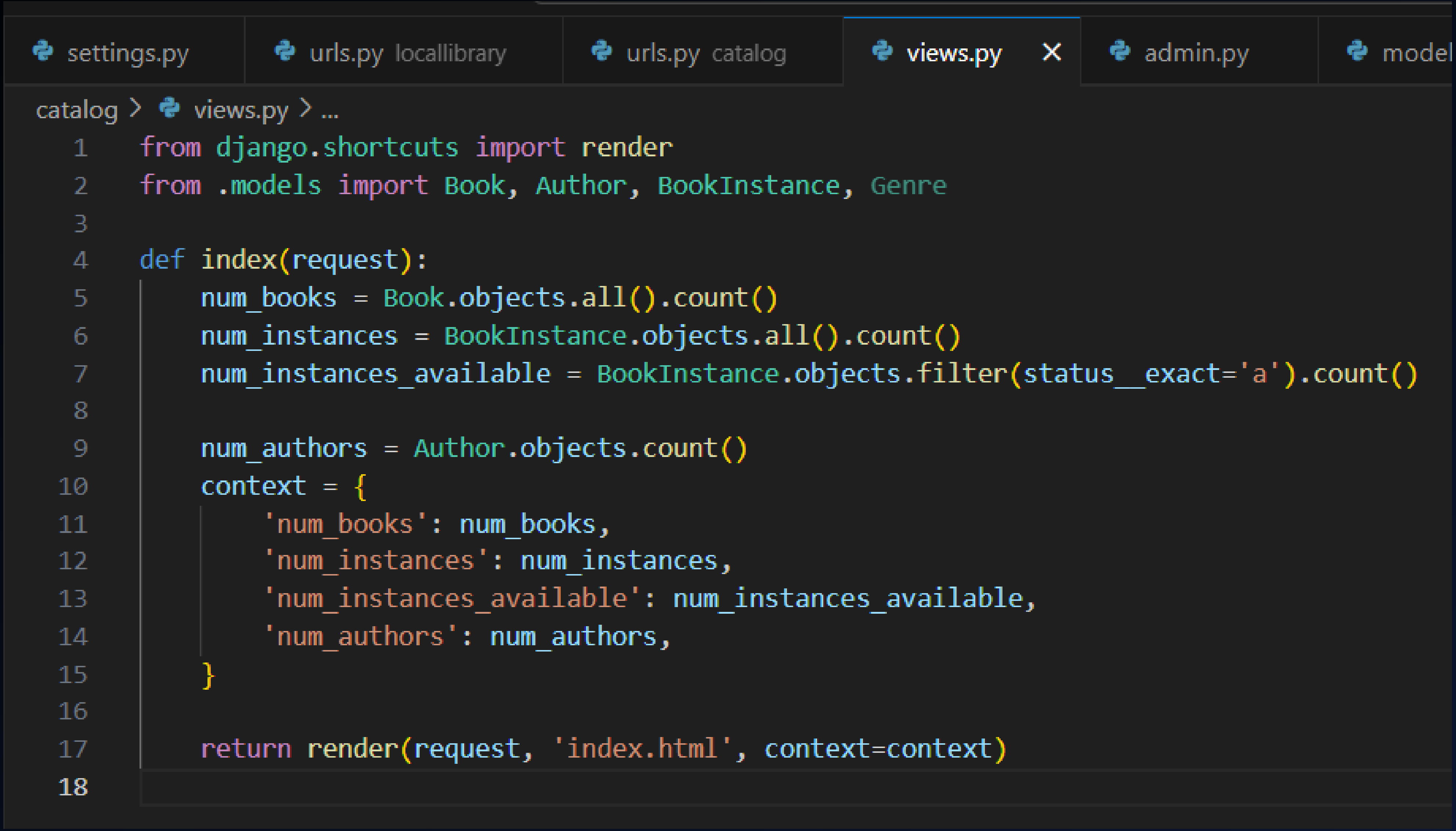
```
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home
Including another URLconf
    1. Import the include() function: from django.urls import include
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path, include
from django.views.generic import RedirectView
urlpatterns = [
    path('admin/', admin.site.urls),
    path('catalog/', include('catalog.urls')),
    path('', RedirectView.as_view(url='catalog/',permanent=True))
]

```

URLS.PY - APP



```python
from django.urls import path
from . import views

urlpatterns=[
    path('',views.index, name='index'),
]
```

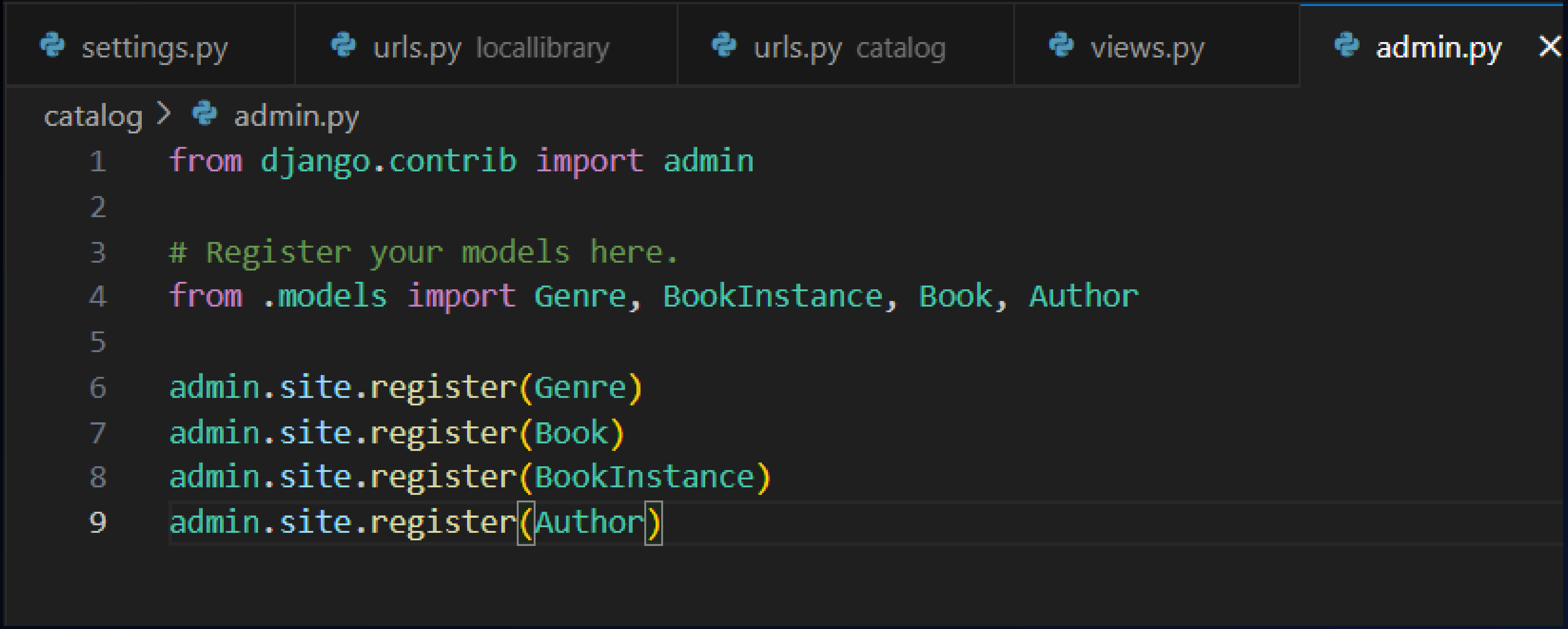
VIEWS.PY



```python
from django.shortcuts import render
from .models import Book, Author, BookInstance, Genre

def index(request):
    num_books = Book.objects.all().count()
    num_instances = BookInstance.objects.all().count()
    num_instances_available = BookInstance.objects.filter(status__exact='a').count()

    num_authors = Author.objects.count()
    context = {
        'num_books': num_books,
        'num_instances': num_instances,
        'num_instances_available': num_instances_available,
        'num_authors': num_authors,
    }

    return render(request, 'index.html', context=context)

```

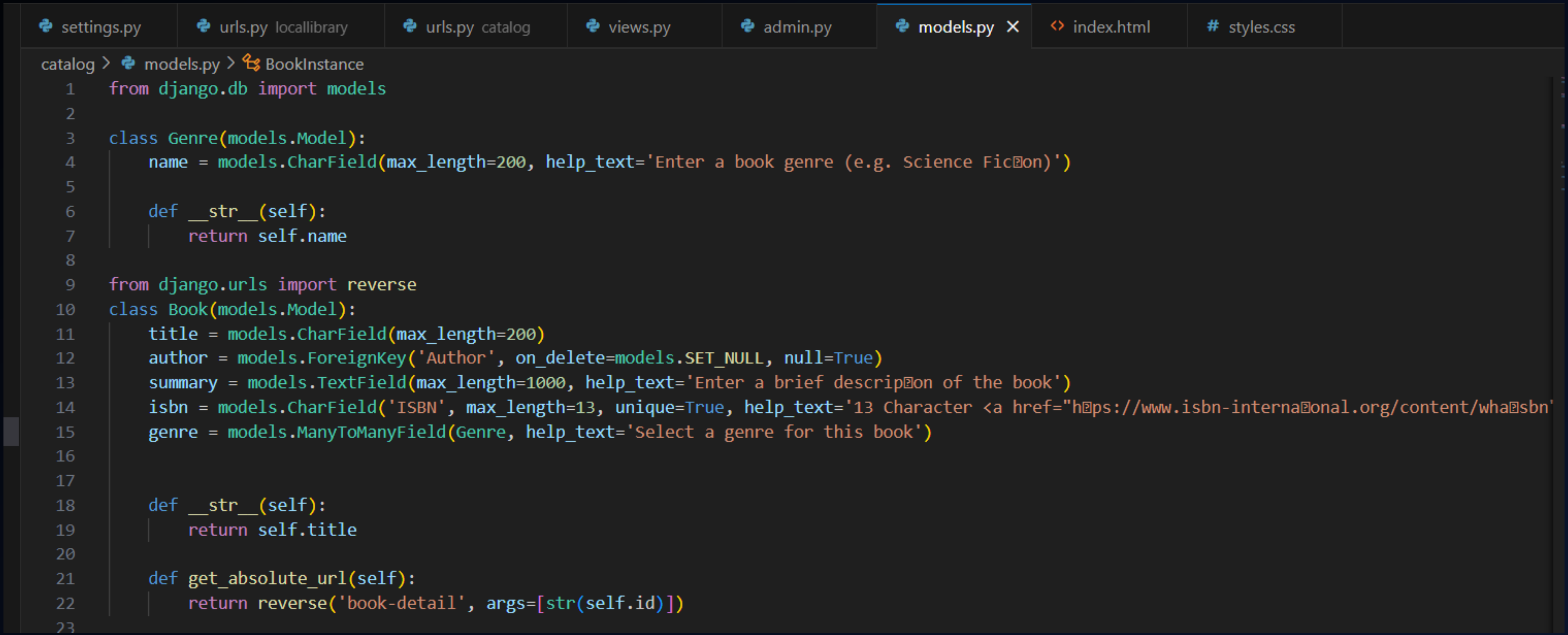
ppt_by_sourabh_pal

ADMIN.PY



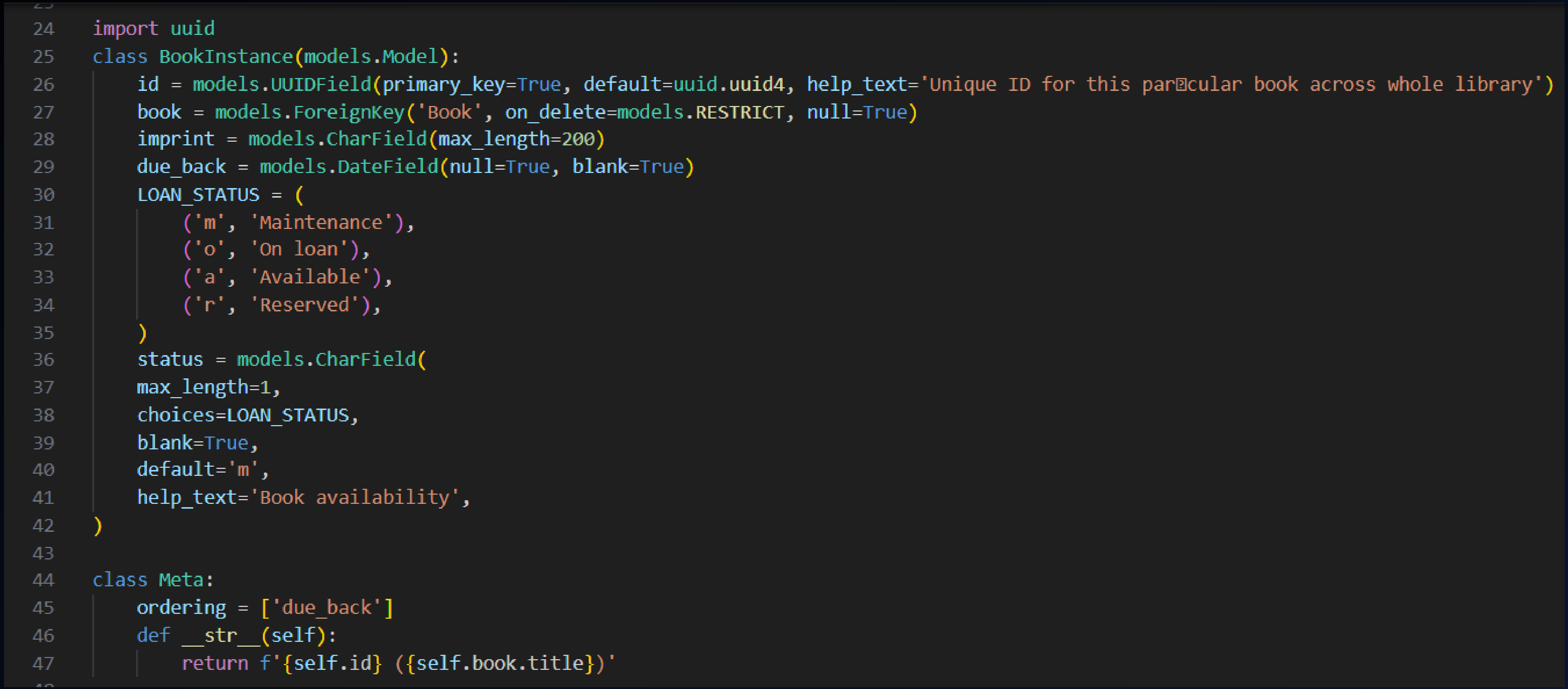
```python
from django.contrib import admin

# Register your models here.
from .models import Genre, BookInstance, Book, Author

admin.site.register(Genre)
admin.site.register(Book)
admin.site.register(BookInstance)
admin.site.register(Author)
```

MODELS.PY



```python
from django.db import models

class Genre(models.Model):
    name = models.CharField(max_length=200, help_text='Enter a book genre (e.g. Science Fic on)')

    def __str__(self):
        return self.name

from django.urls import reverse
class Book(models.Model):
    title = models.CharField(max_length=200)
    author = models.ForeignKey('Author', on_delete=models.SET_NULL, null=True)
    summary = models.TextField(max_length=1000, help_text='Enter a brief descrip on of the book')
    isbn = models.CharField('ISBN', max_length=13, unique=True, help_text='13 Character <a href="h ps://www.isbn-interna onal.org/content/wha sbn'
    genre = models.ManyToManyField(Genre, help_text='Select a genre for this book')


    def __str__(self):
        return self.title

    def get_absolute_url(self):
        return reverse('book-detail', args=[str(self.id)])
```

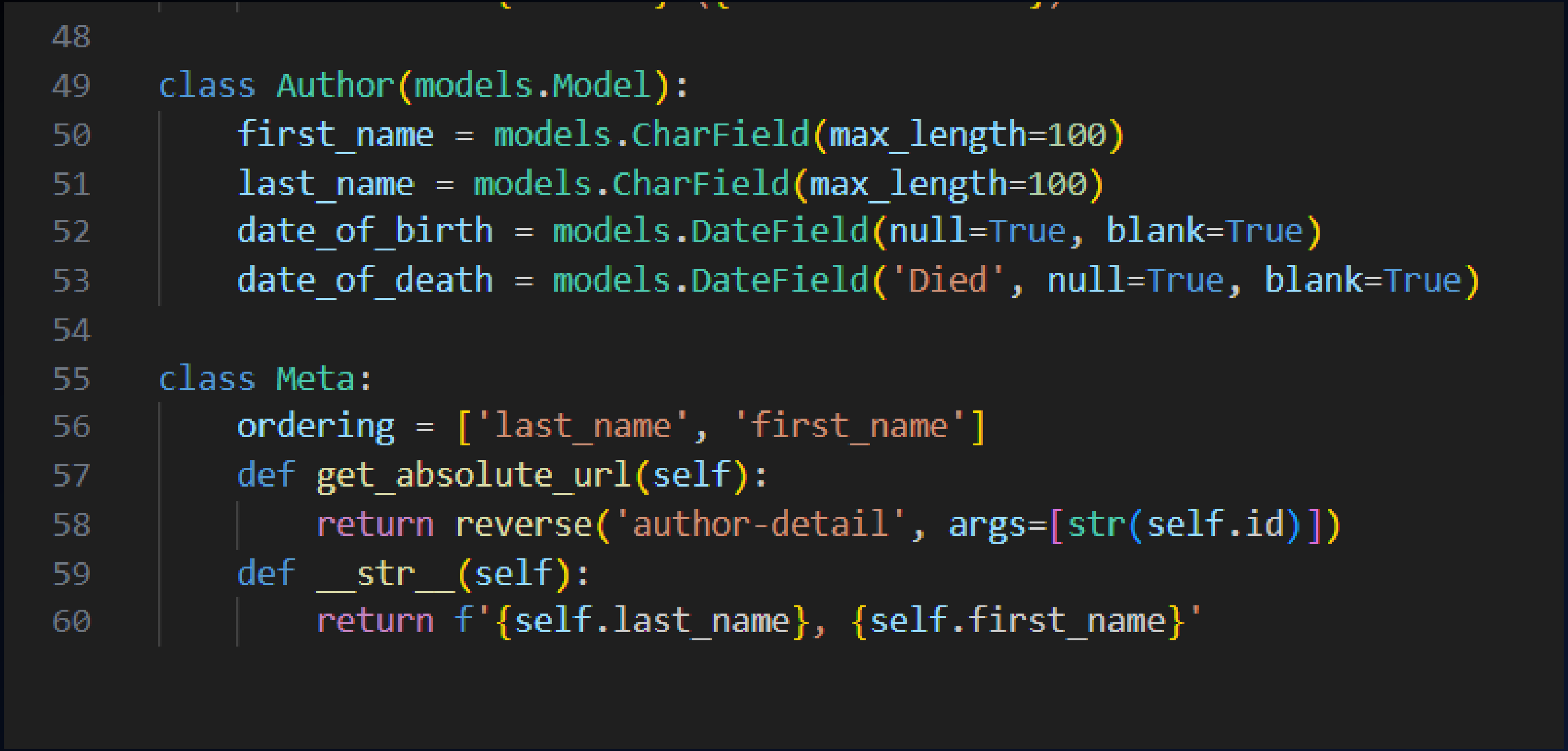
MODELS.PY



```python
import uuid
class BookInstance(models.Model):
    id = models.UUIDField(primary_key=True, default=uuid.uuid4, help_text='Unique ID for this par▯cular book across whole library')
    book = models.ForeignKey('Book', on_delete=models.RESTRICT, null=True)
    imprint = models.CharField(max_length=200)
    due_back = models.DateField(null=True, blank=True)
    LOAN_STATUS = (
        ('m', 'Maintenance'),
        ('o', 'On loan'),
        ('a', 'Available'),
        ('r', 'Reserved'),
    )
    status = models.CharField(
    max_length=1,
    choices=LOAN_STATUS,
    blank=True,
    default='m',
    help_text='Book availability',
)

class Meta:
    ordering = ['due_back']
    def __str__(self):
        return f'{self.id} ({self.book.title})'
```

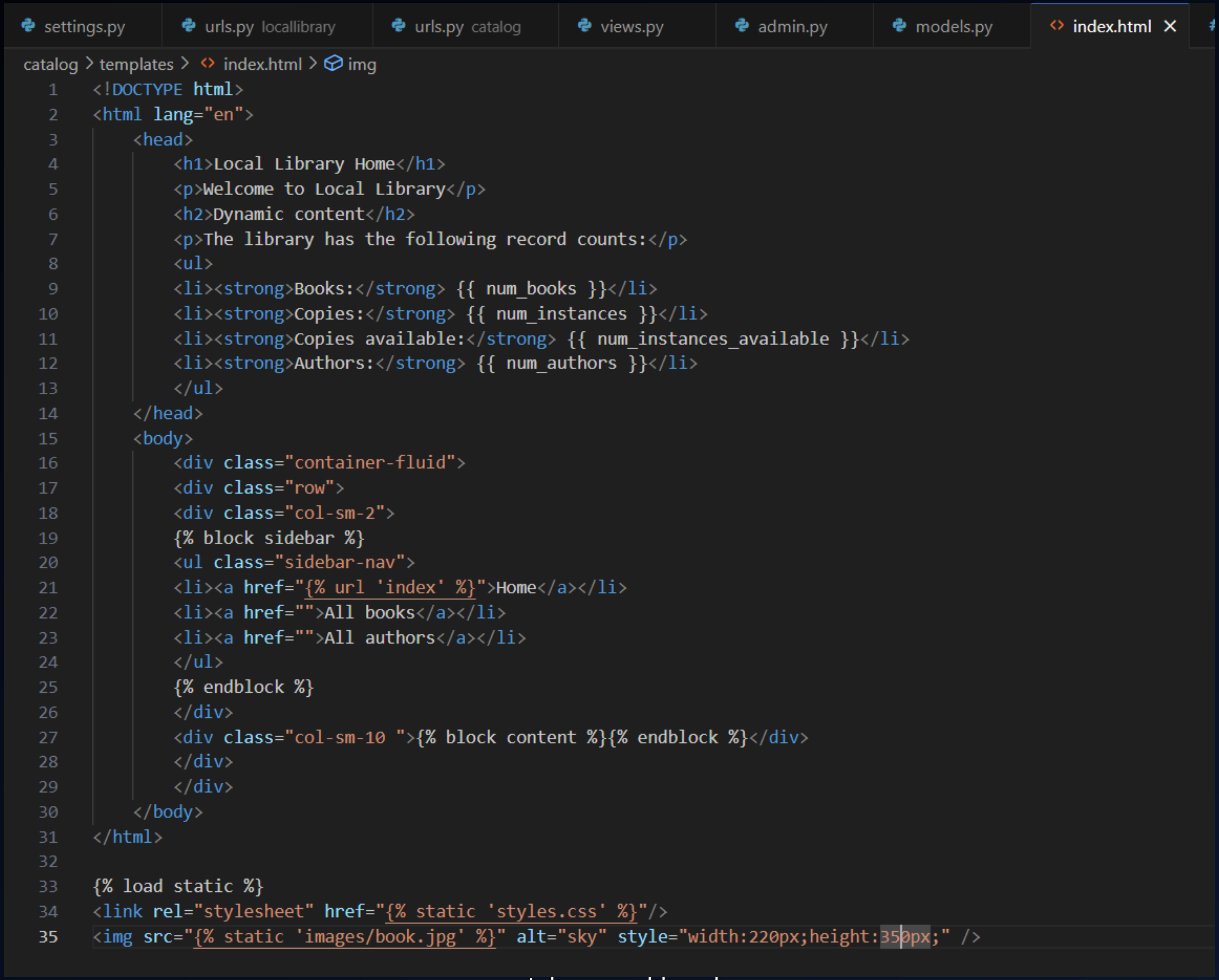
ppt_by_sourabh_pal

MODELS.PY



```python

class Author(models.Model):
    first_name = models.CharField(max_length=100)
    last_name = models.CharField(max_length=100)
    date_of_birth = models.DateField(null=True, blank=True)
    date_of_death = models.DateField('Died', null=True, blank=True)

class Meta:
    ordering = ['last_name', 'first_name']
    def get_absolute_url(self):
        return reverse('author-detail', args=[str(self.id)])
    def __str__(self):
        return f'{self.last_name}, {self.first_name}'
```

INDEX.HTML



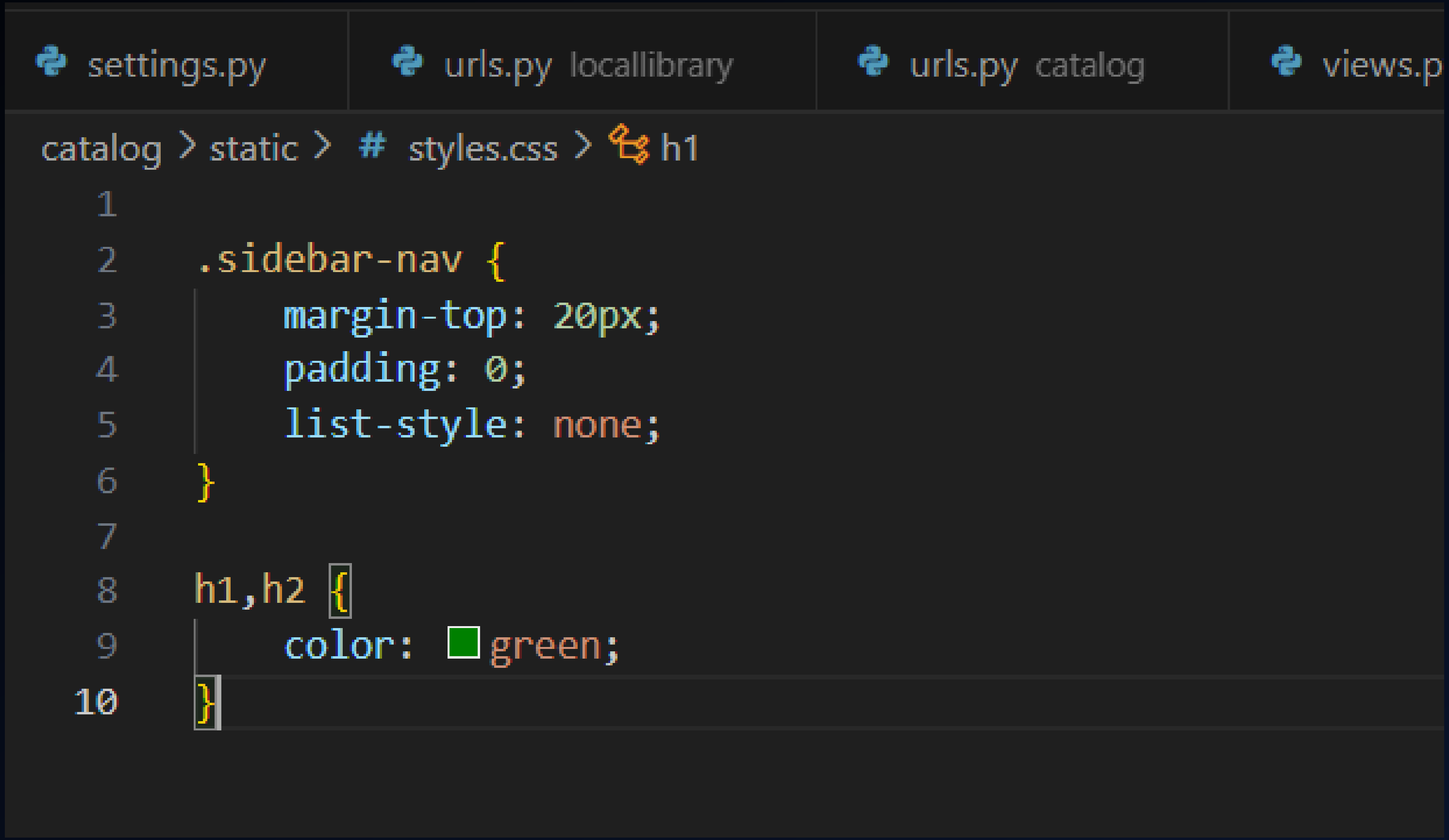
settings.py | urls.py locallibrary | urls.py catalog | views.py | admin.py | models.py | index.html

catalog > templates > index.html > img

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <h1>Local Library Home</h1>
        <p>Welcome to Local Library</p>
        <h2>Dynamic content</h2>
        <p>The library has the following record counts:</p>
        <ul>
        <li><strong>Books:</strong> {{ num_books }}</li>
        <li><strong>Copies:</strong> {{ num_instances }}</li>
        <li><strong>Copies available:</strong> {{ num_instances_available }}</li>
        <li><strong>Authors:</strong> {{ num_authors }}</li>
        </ul>
    </head>
    <body>
        <div class="container-fluid">
        <div class="row">
        <div class="col-sm-2">
        {% block sidebar %}
        <ul class="sidebar-nav">
        <li><a href="{% url 'index' %}">Home</a></li>
        <li><a href="">All books</a></li>
        <li><a href="">All authors</a></li>
        </ul>
        {% endblock %}
        </div>
        <div class="col-sm-10 ">{% block content %}{% endblock %}</div>
        </div>
        </div>
    </body>
</html>

{% load static %}
<link rel="stylesheet" href="{% static 'styles.css' %}"/>
<img src="{% static 'images/book.jpg' %}" alt="sky" style="width:220px;height:350px;" />
```

ppt_by_sourabh_pal

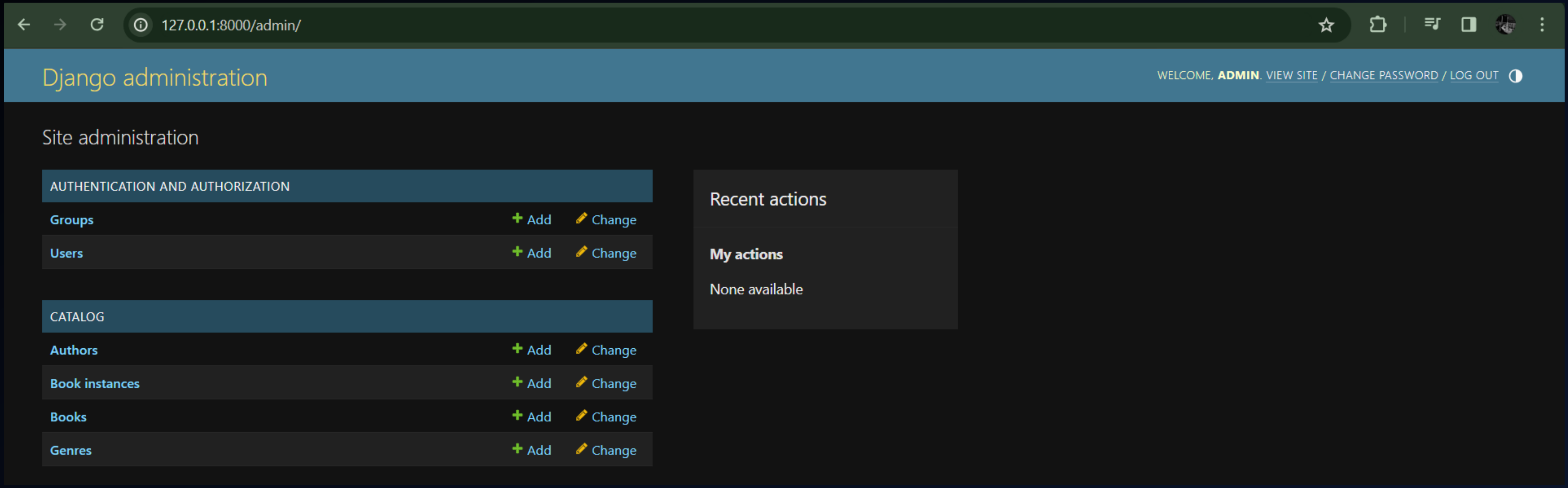
STYLES.CSS



```css

.sidebar-nav {
    margin-top: 20px;
    padding: 0;
    list-style: none;
}

h1,h2 {
    color: green;
}
```

ppt_by_sourabh_pal

ADMIN PAGE (CLIENT)



127.0.0.1:8000/admin/

Django administration

WELCOME, **ADMIN**. VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

AUTHENTICATION AND AUTHORIZATION		
Groups	Add	Change
Users	Add	Change

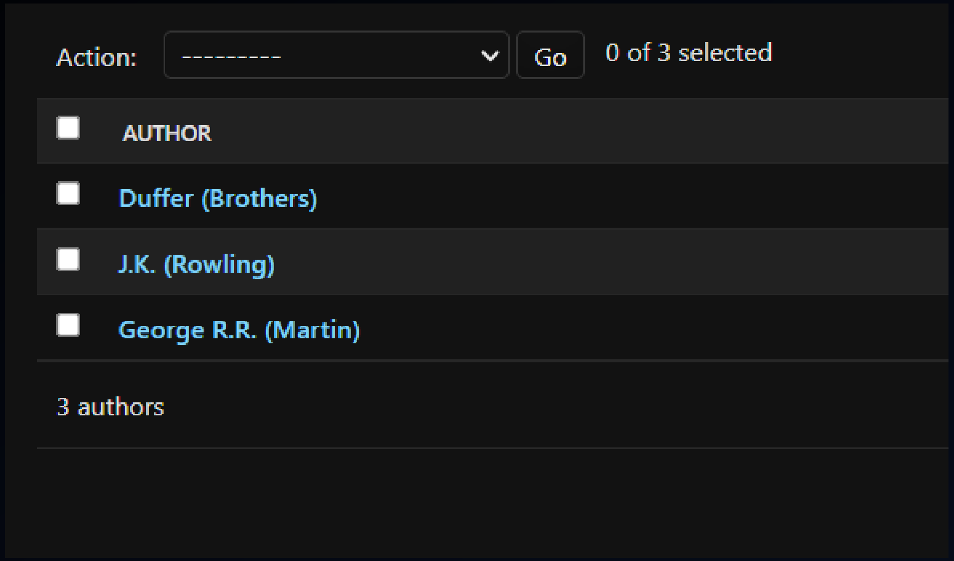
CATALOG		
Authors	Add	Change
Book instances	Add	Change
Books	Add	Change
Genres	Add	Change

Recent actions

My actions

None available

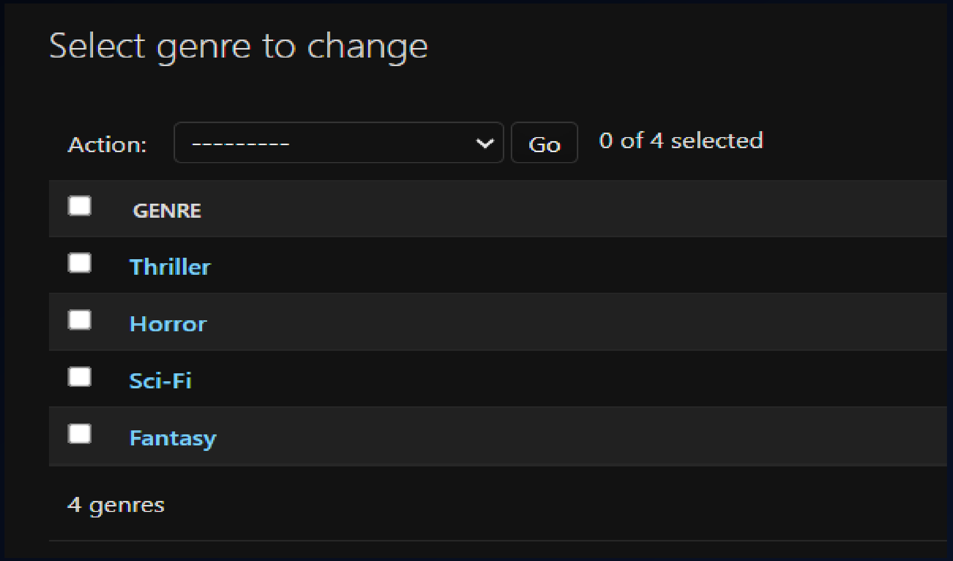
ADMIN PAGE (CLIENT)



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

	AUTHOR
☐	Duffer (Brothers)
☐	J.K. (Rowling)
☐	George R.R. (Martin)

3 authors



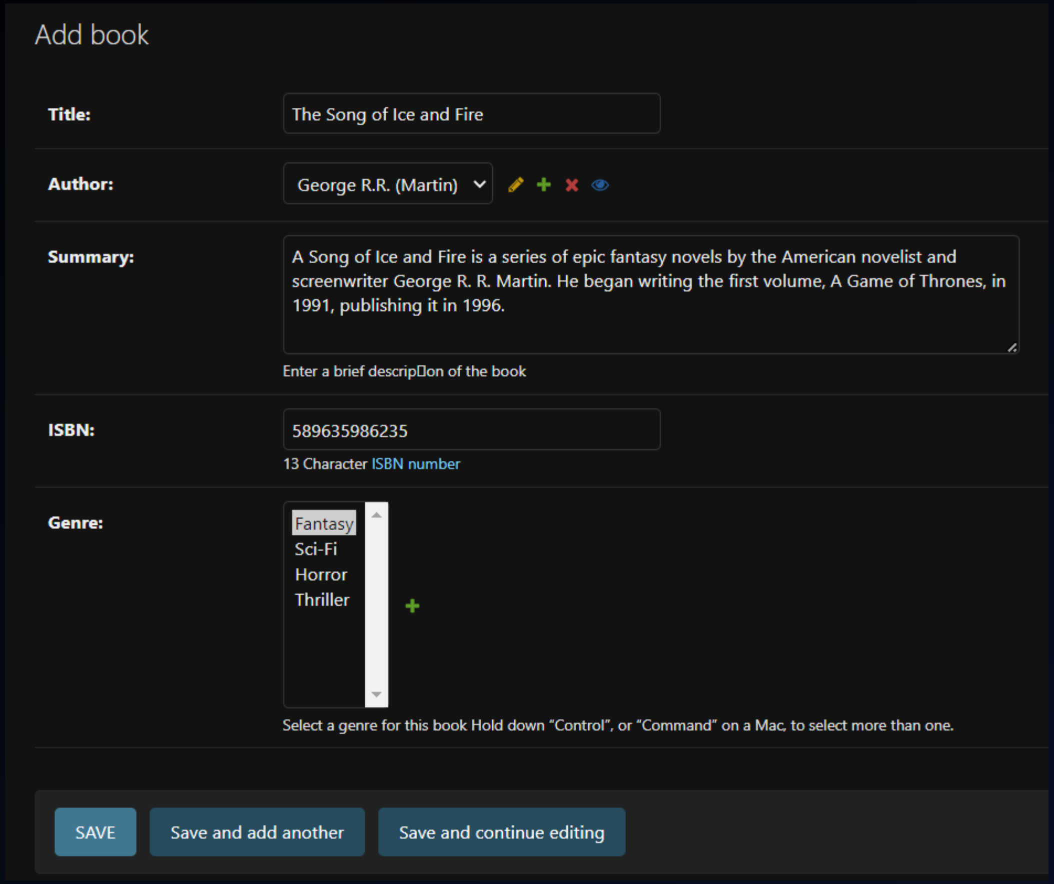
Select genre to change

Action: --------- Go 0 of 4 selected

	GENRE
☐	Thriller
☐	Horror
☐	Sci-Fi
☐	Fantasy

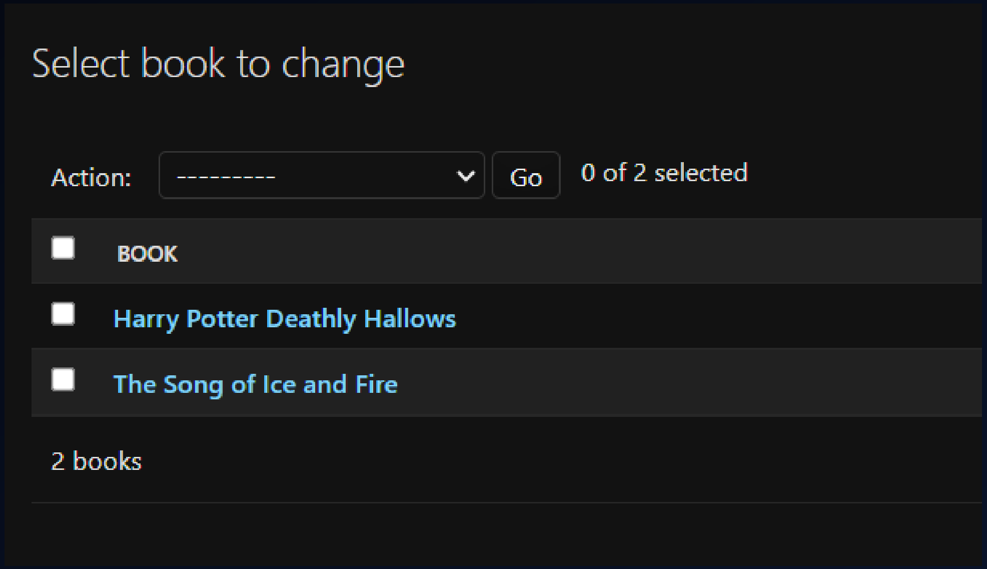
4 genres

ADMIN PAGE (CLIENT)



Add book

Title:	The Song of Ice and Fire
Author:	George R.R. (Martin)
Summary:	A Song of Ice and Fire is a series of epic fantasy novels by the American novelist and screenwriter George R. R. Martin. He began writing the first volume, A Game of Thrones, in 1991, publishing it in 1996. Enter a brief descrip􀀀on of the book
ISBN:	589635986235 13 Character ISBN number
Genre:	Fantasy Sci-Fi Horror Thriller Select a genre for this book Hold down "Control", or "Command" on a Mac, to select more than one.



Select book to change

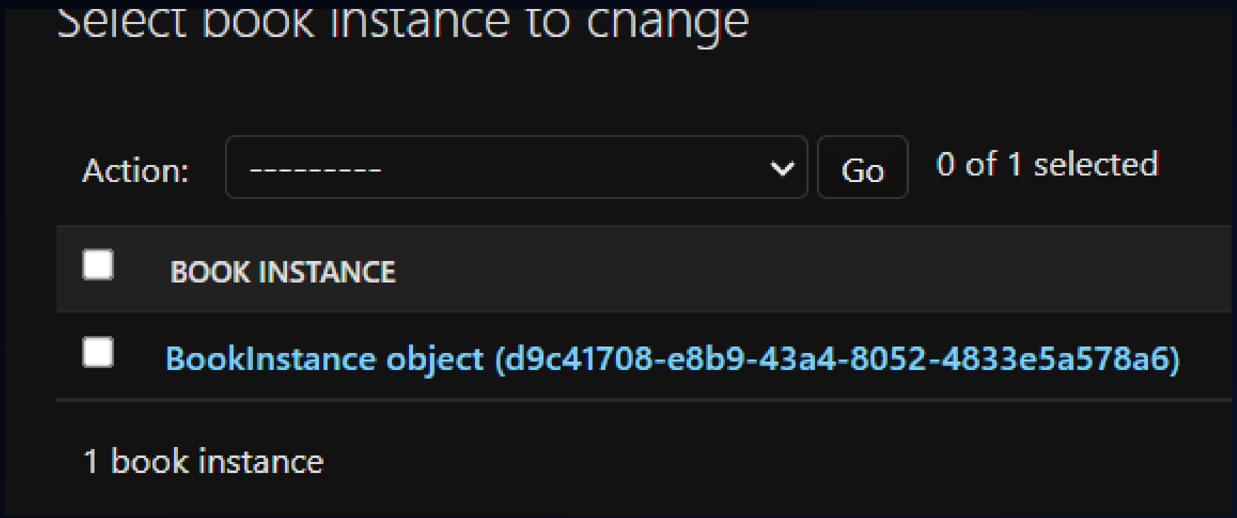
Action: --------- Go 0 of 2 selected

BOOK
Harry Potter Deathly Hallows
The Song of Ice and Fire

2 books

ppt_by_sourabh_pal

ADMIN PAGE (CLIENT)

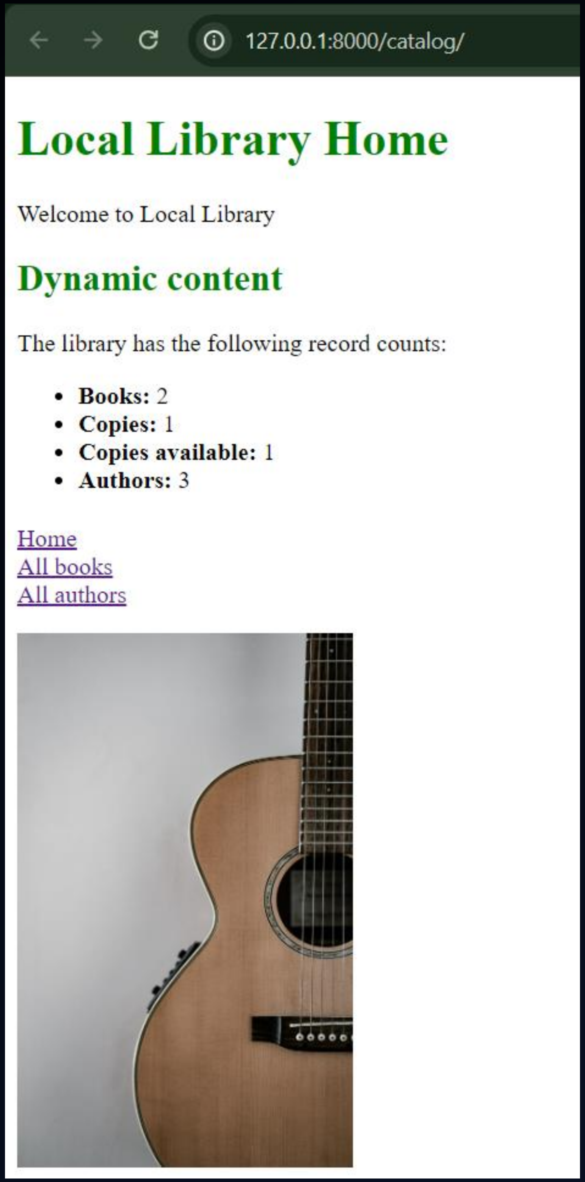

Select book instance to change

Action: --------- Go 0 of 1 selected

	BOOK INSTANCE
☐	BookInstance object (d9c41708-e8b9-43a4-8052-4833e5a578a6)

1 book instance

/CATALOG/



127.0.0.1:8000/catalog/

Local Library Home

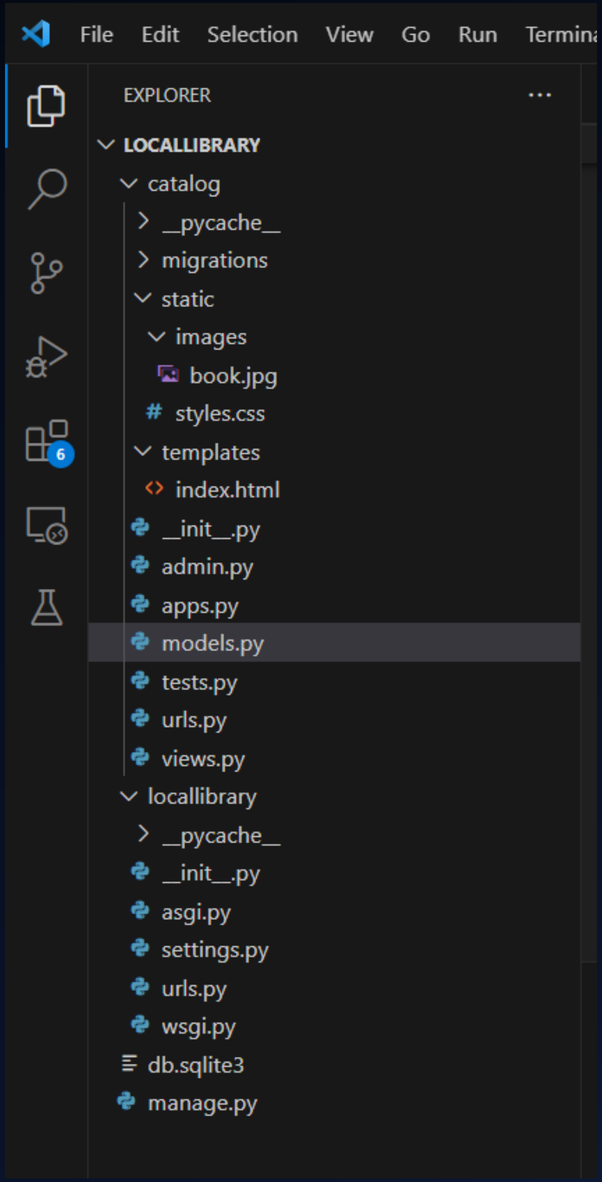
Welcome to Local Library

Dynamic content

The library has the following record counts:

- **Books:** 2
- **Copies:** 1
- **Copies available:** 1
- **Authors:** 3

Home
All books
All authors



```
LOCALLIBRARY
  catalog
    __pycache__
    migrations
    static
      images
        book.jpg
      styles.css
    templates
      index.html
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  locallibrary
    __pycache__
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
  db.sqlite3
  manage.py
```

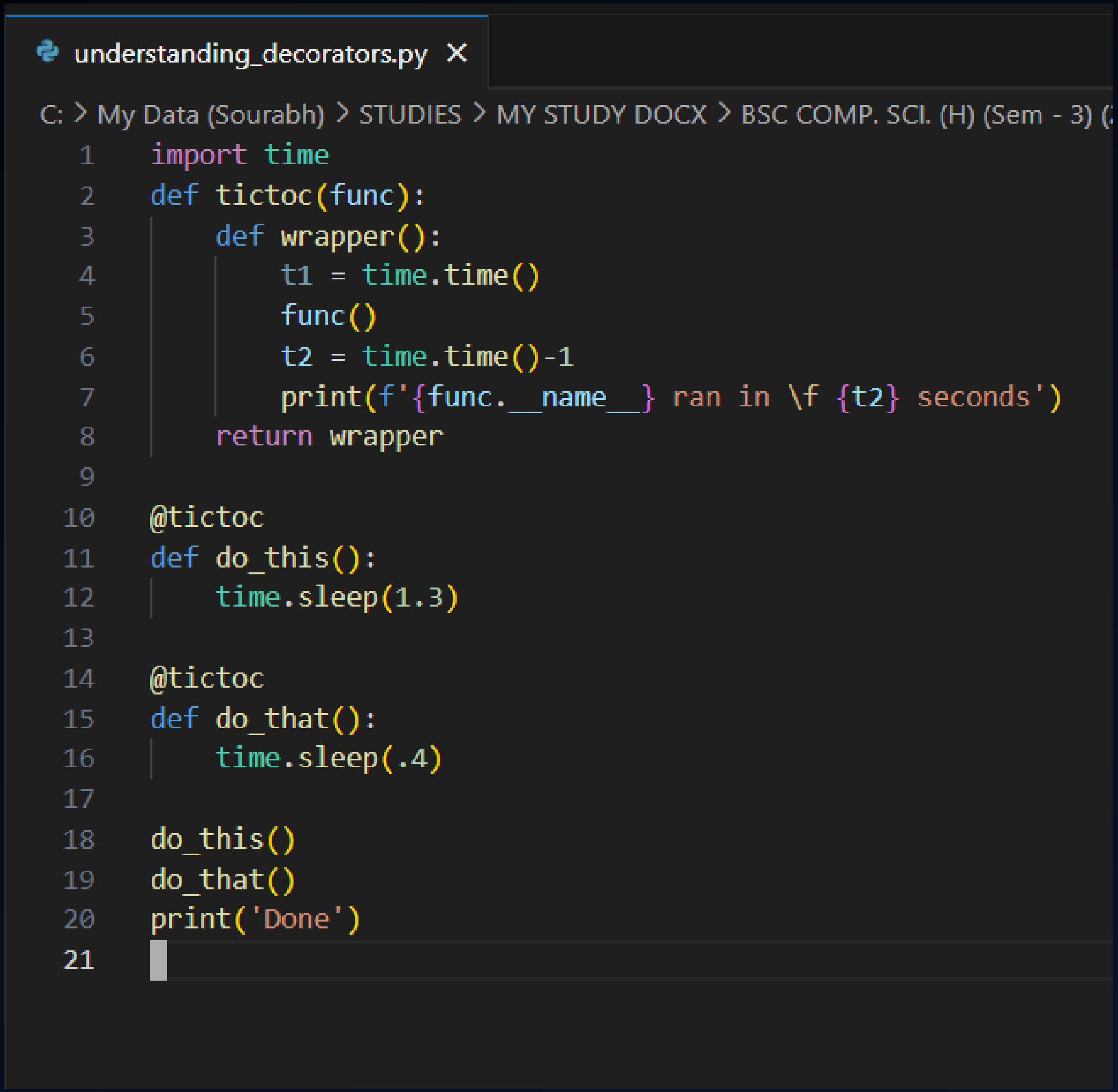
DAY 8 – TASK 1

DECORATORS

USING FUNCTIONS - TIME

ppt_by_sourabh_pal

DECORATORS



understanding_decorators.py

C: > My Data (Sourabh) > STUDIES > MY STUDY DOCX > BSC COMP. SCI. (H) (Sem - 3) (

```python
import time
def tictoc(func):
    def wrapper():
        t1 = time.time()
        func()
        t2 = time.time()-1
        print(f'{func.__name__} ran in \f {t2} seconds')
    return wrapper

@tictoc
def do_this():
    time.sleep(1.3)

@tictoc
def do_that():
    time.sleep(.4)

do_this()
do_that()
print('Done')

```

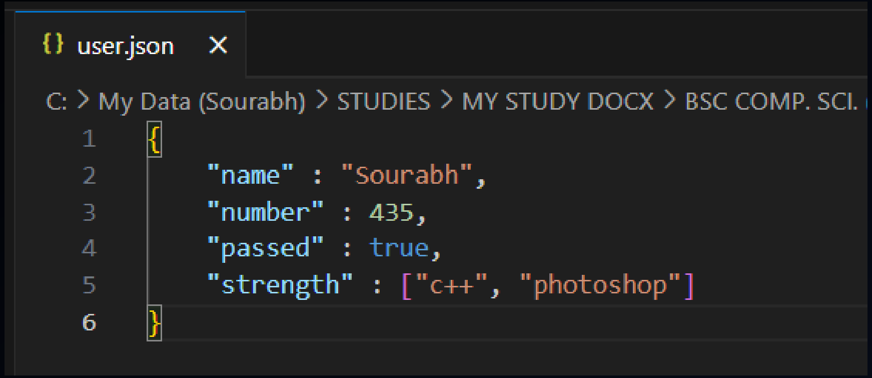
DAY 8 – TASK 2

INTRODUCING TO JSON

JAVA SCRIPT OBJECT NOTATION

ppt_by_sourabh_pal

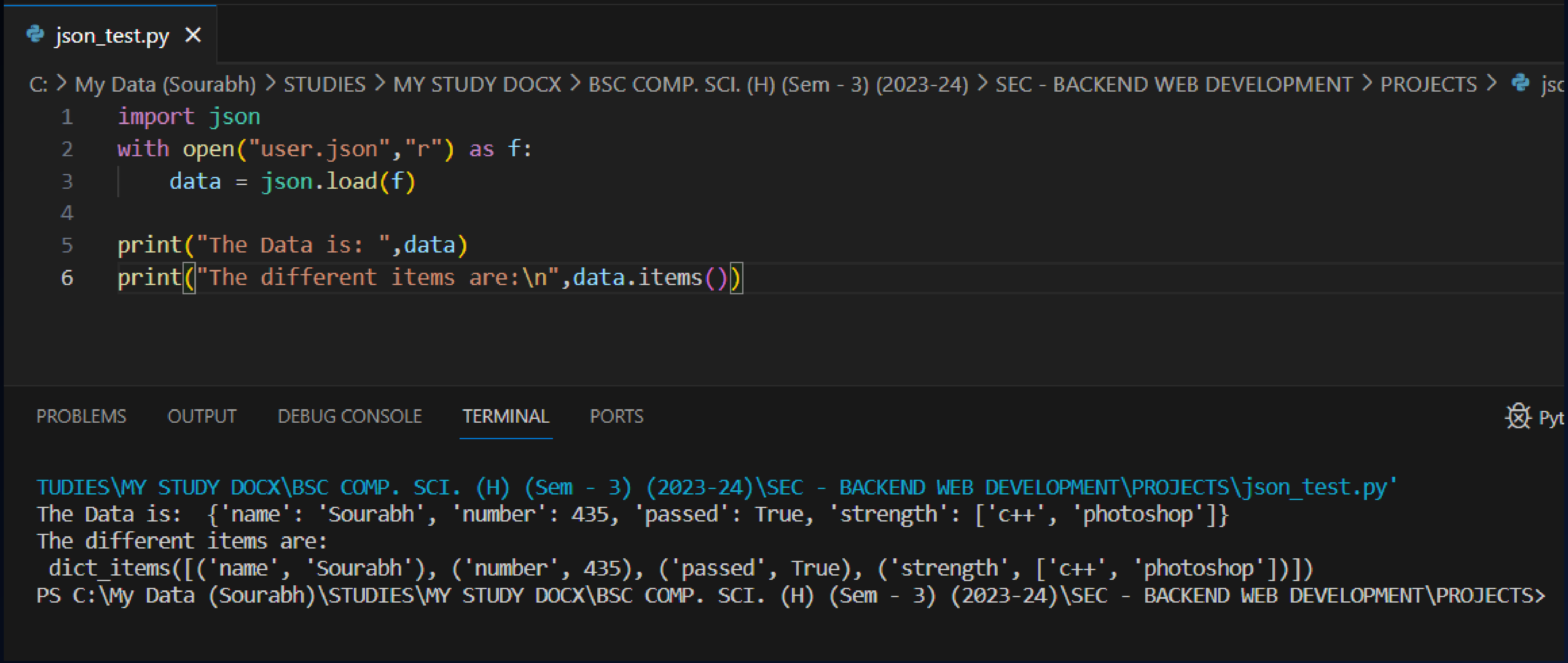
DECORATORS



{} user.json ✕

C: > My Data (Sourabh) > STUDIES > MY STUDY DOCX > BSC COMP. SCI.

```json
{
    "name" : "Sourabh",
    "number" : 435,
    "passed" : true,
    "strength" : ["c++", "photoshop"]
}
```



json_test.py ✕

C: > My Data (Sourabh) > STUDIES > MY STUDY DOCX > BSC COMP. SCI. (H) (Sem - 3) (2023-24) > SEC - BACKEND WEB DEVELOPMENT > PROJECTS > jsc

```python
import json
with open("user.json","r") as f:
    data = json.load(f)

print("The Data is: ",data)
print("The different items are:\n",data.items())
```

```
TUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\PROJECTS\json_test.py'
The Data is:  {'name': 'Sourabh', 'number': 435, 'passed': True, 'strength': ['c++', 'photoshop']}
The different items are:
 dict_items([('name', 'Sourabh'), ('number', 435), ('passed', True), ('strength', ['c++', 'photoshop'])])
PS C:\My Data (Sourabh)\STUDIES\MY STUDY DOCX\BSC COMP. SCI. (H) (Sem - 3) (2023-24)\SEC - BACKEND WEB DEVELOPMENT\PROJECTS>
```

DAY 8 – TASK 3

API using Django rest Framework

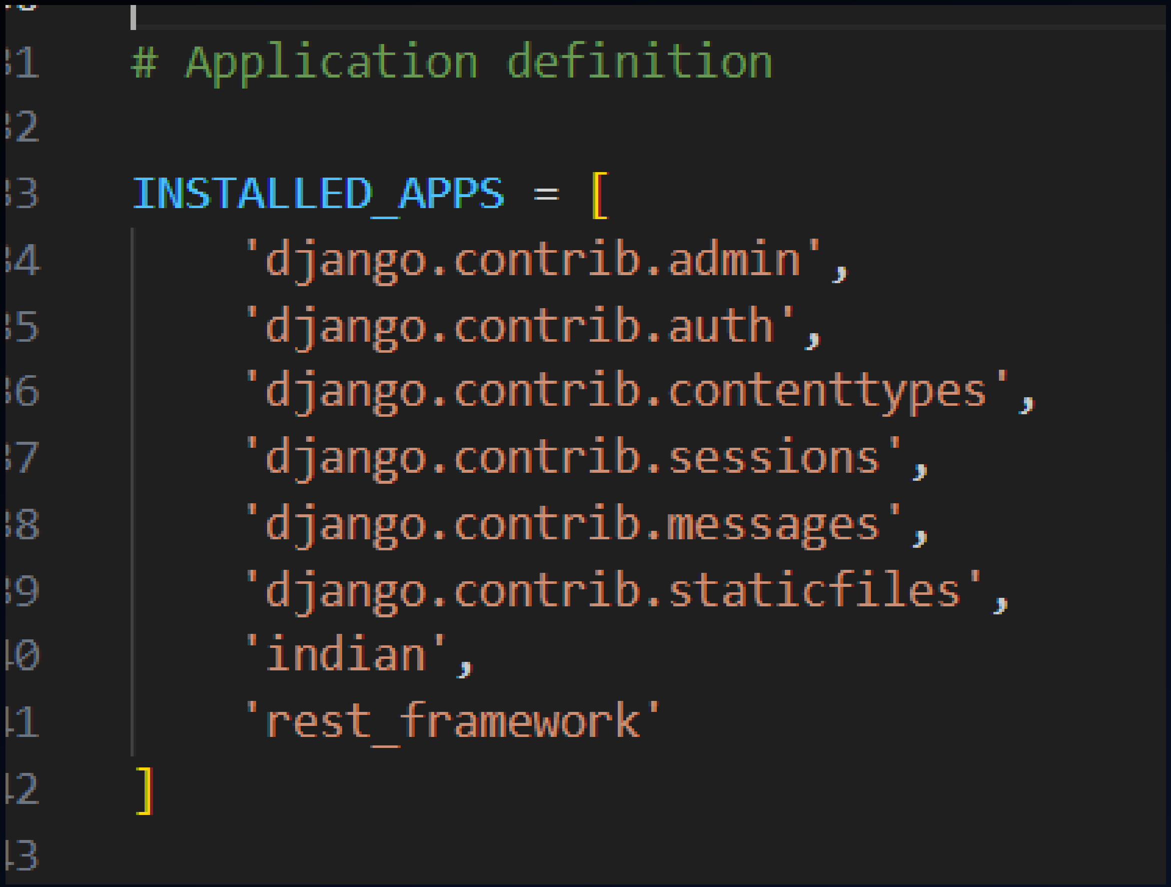
POSTMAN TOOL

ppt_by_sourabh_pal

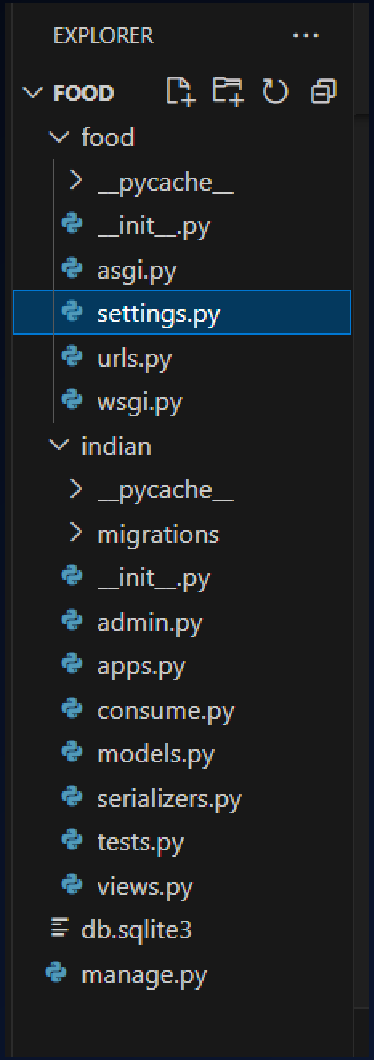
INSTRUCTIONS

- DIRECTORY NAME: foods
- PROJECT NAME: food
- APP NAME: indian

SETTINGS.PY

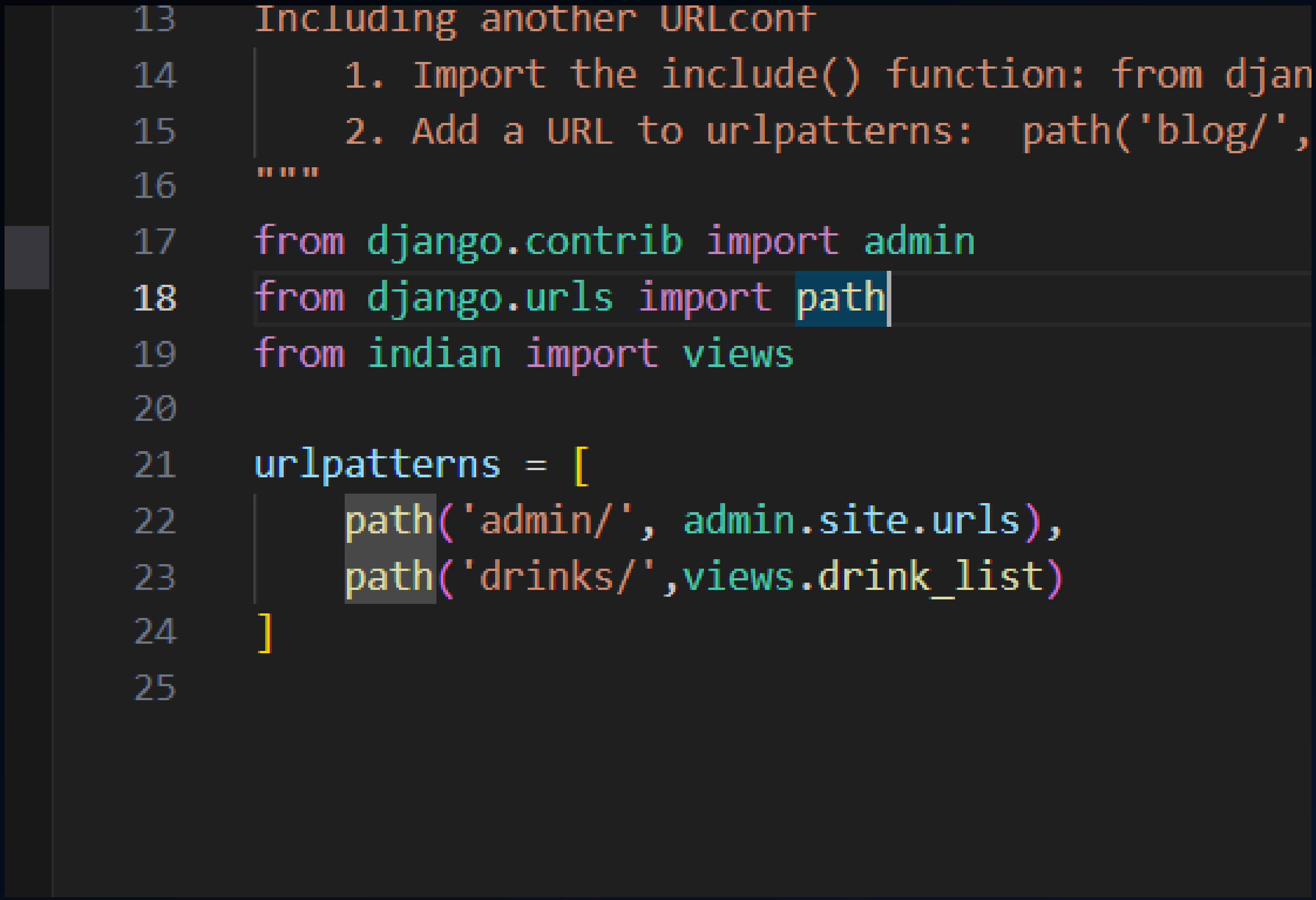


```python
# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'indian',
    'rest_framework'
]
```



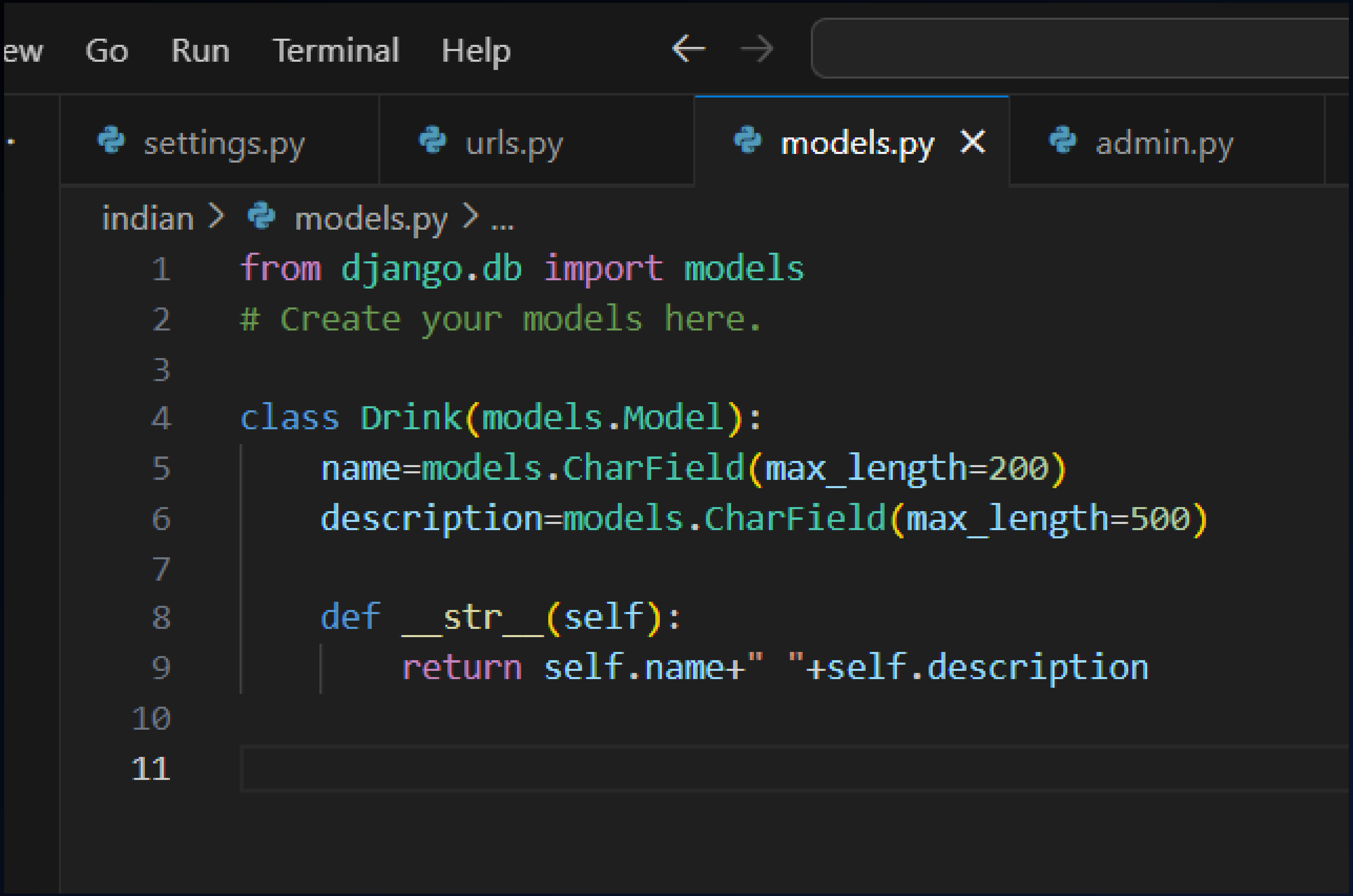
```
EXPLORER
FOOD
  food
    __pycache__
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
  indian
    __pycache__
    migrations
    __init__.py
    admin.py
    apps.py
    consume.py
    models.py
    serializers.py
    tests.py
    views.py
  db.sqlite3
  manage.py
```

URLS.PY - PROJECT



```
Including another URLconf
    1. Import the include() function: from djan
    2. Add a URL to urlpatterns:  path('blog/',
"""
from django.contrib import admin
from django.urls import path
from indian import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('drinks/',views.drink_list)
]

```

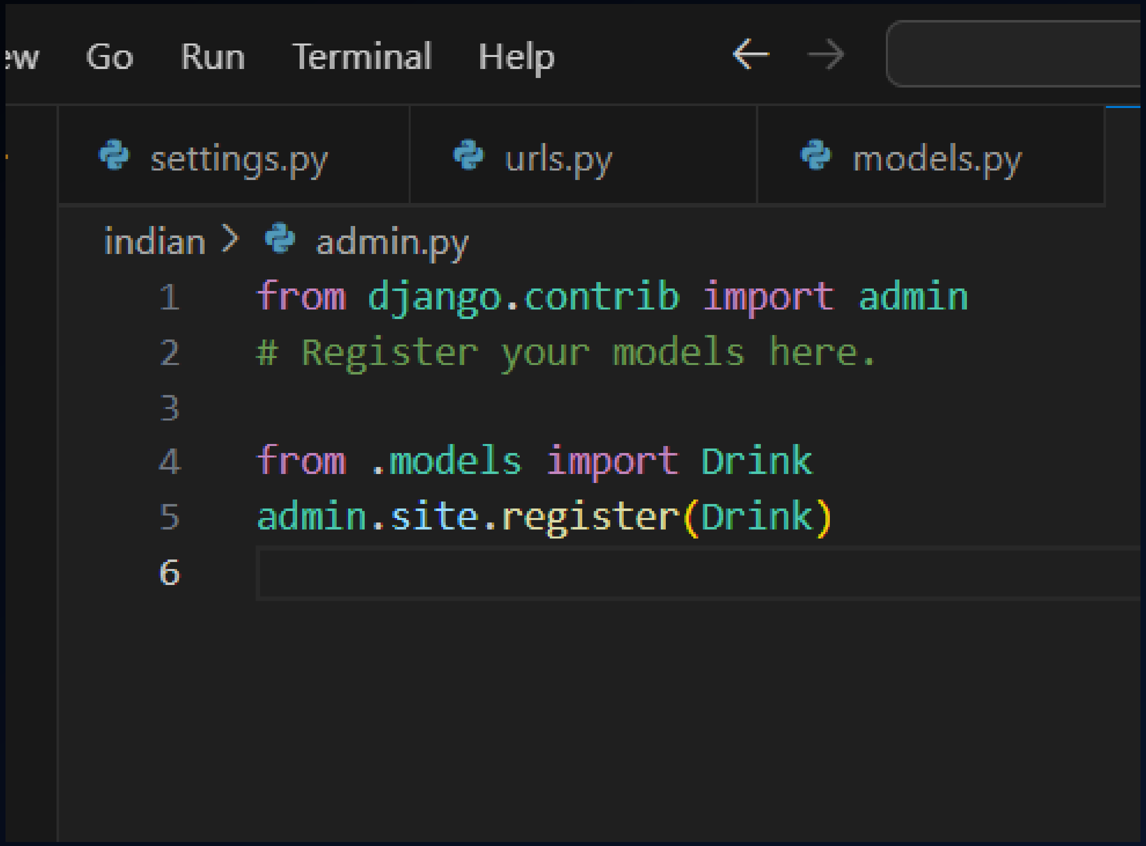
MODELS.PY



```python
from django.db import models
# Create your models here.

class Drink(models.Model):
    name=models.CharField(max_length=200)
    description=models.CharField(max_length=500)

    def __str__(self):
        return self.name+" "+self.description

```

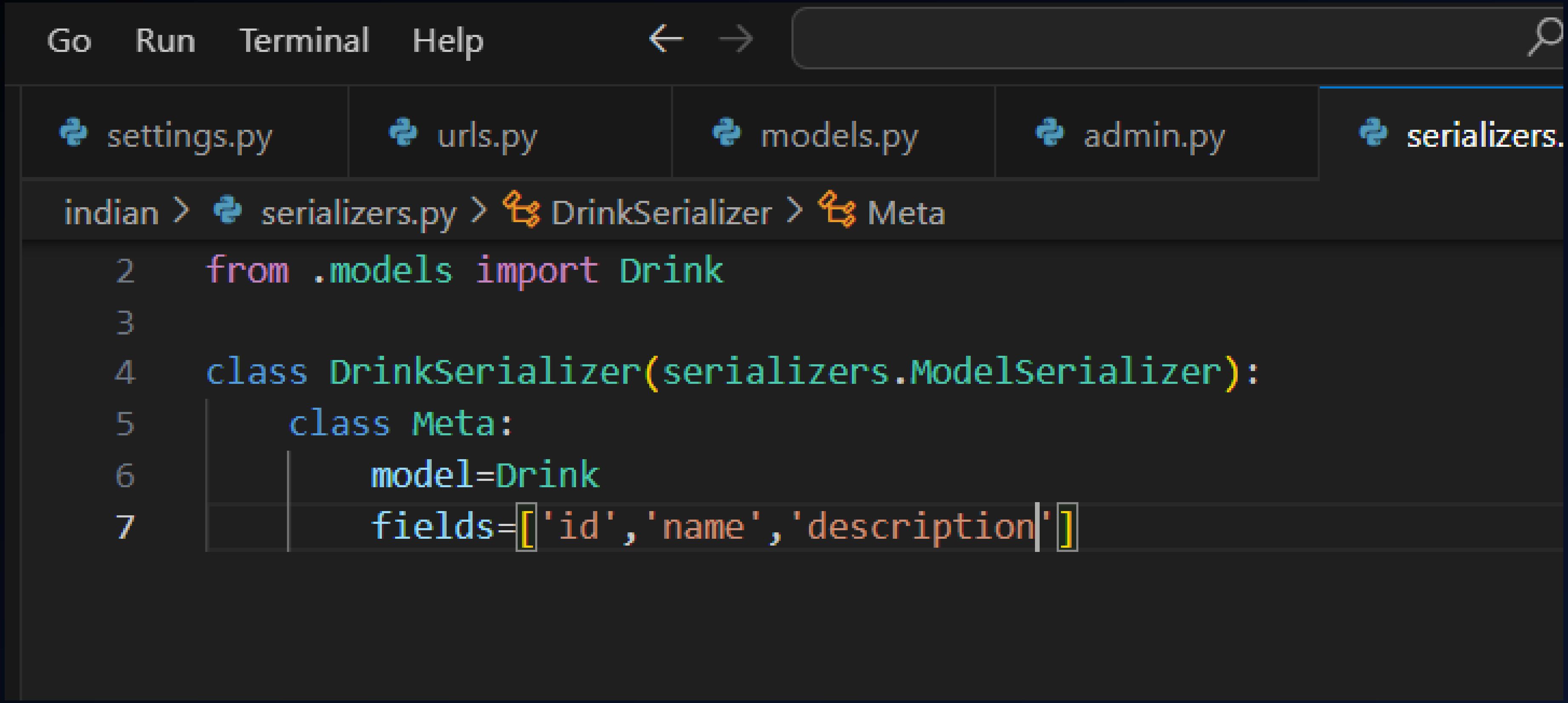
ppt_by_sourabh_pal

ADMIN.PY



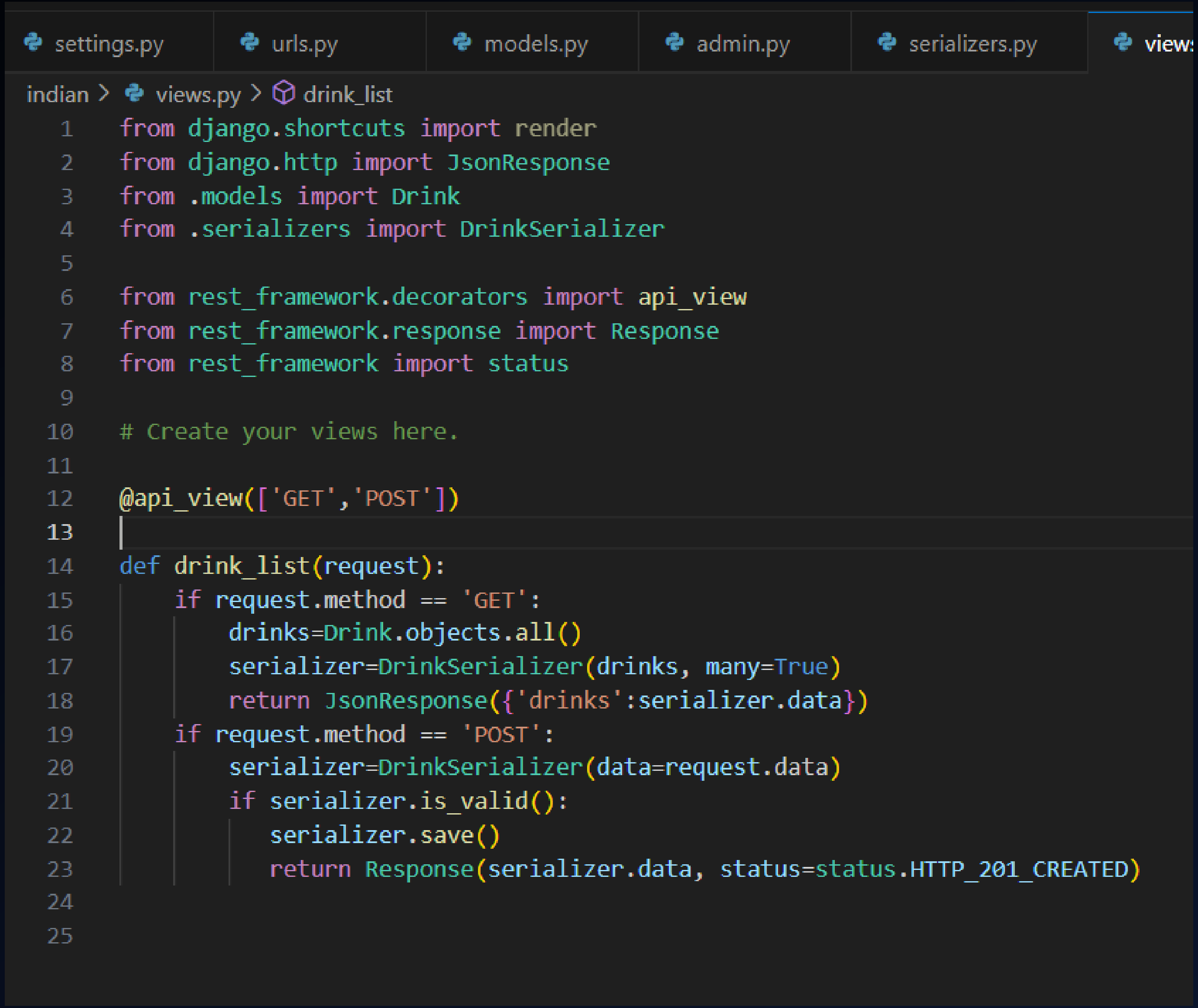
```python
from django.contrib import admin
# Register your models here.

from .models import Drink
admin.site.register(Drink)

```

SERIALIZERS.PY



```
from .models import Drink

class DrinkSerializer(serializers.ModelSerializer):
    class Meta:
        model=Drink
        fields=['id','name','description']
```

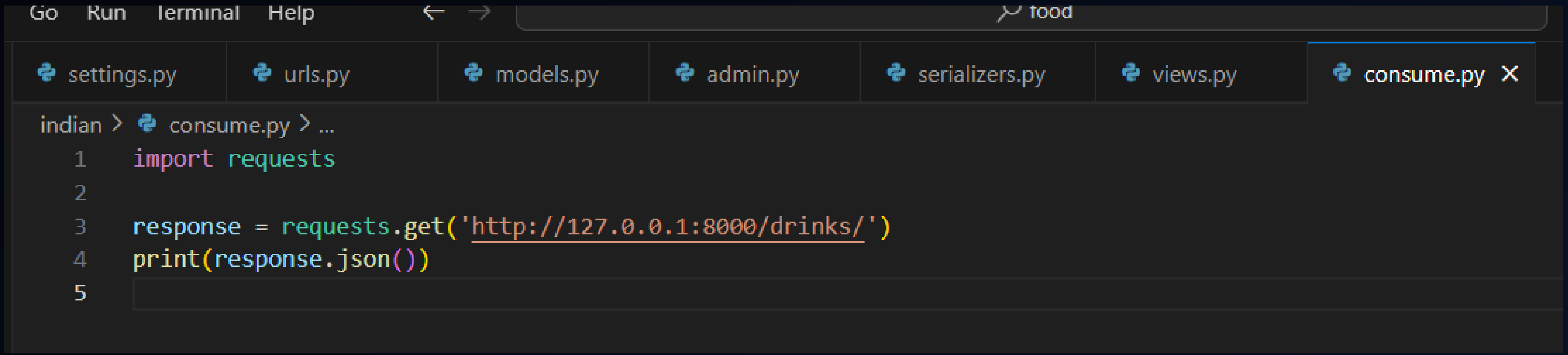
VIEWS.PY



```python
from django.shortcuts import render
from django.http import JsonResponse
from .models import Drink
from .serializers import DrinkSerializer

from rest_framework.decorators import api_view
from rest_framework.response import Response
from rest_framework import status

# Create your views here.

@api_view(['GET','POST'])

def drink_list(request):
    if request.method == 'GET':
        drinks=Drink.objects.all()
        serializer=DrinkSerializer(drinks, many=True)
        return JsonResponse({'drinks':serializer.data})
    if request.method == 'POST':
        serializer=DrinkSerializer(data=request.data)
        if serializer.is_valid():
            serializer.save()
            return Response(serializer.data, status=status.HTTP_201_CREATED)


```

ppt_by_sourabh_pal

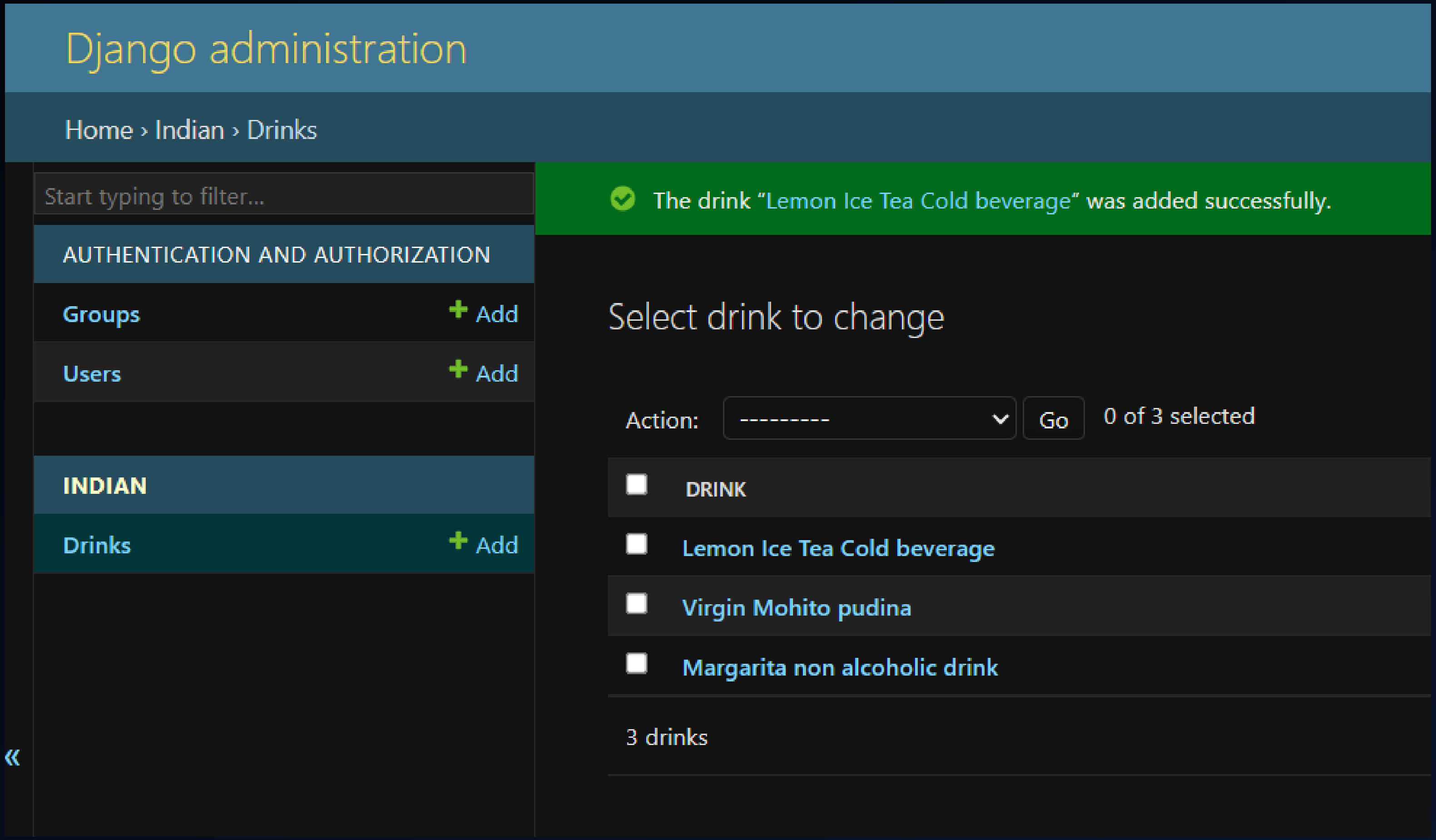
CONSUME.PY



```python
import requests

response = requests.get('http://127.0.0.1:8000/drinks/')
print(response.json())
```

ppt_by_sourabh_pal

ADMIN PANEL



Django administration

Home › Indian › Drinks

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

INDIAN

Drinks + Add

«

The drink "Lemon Ice Tea Cold beverage" was added successfully.

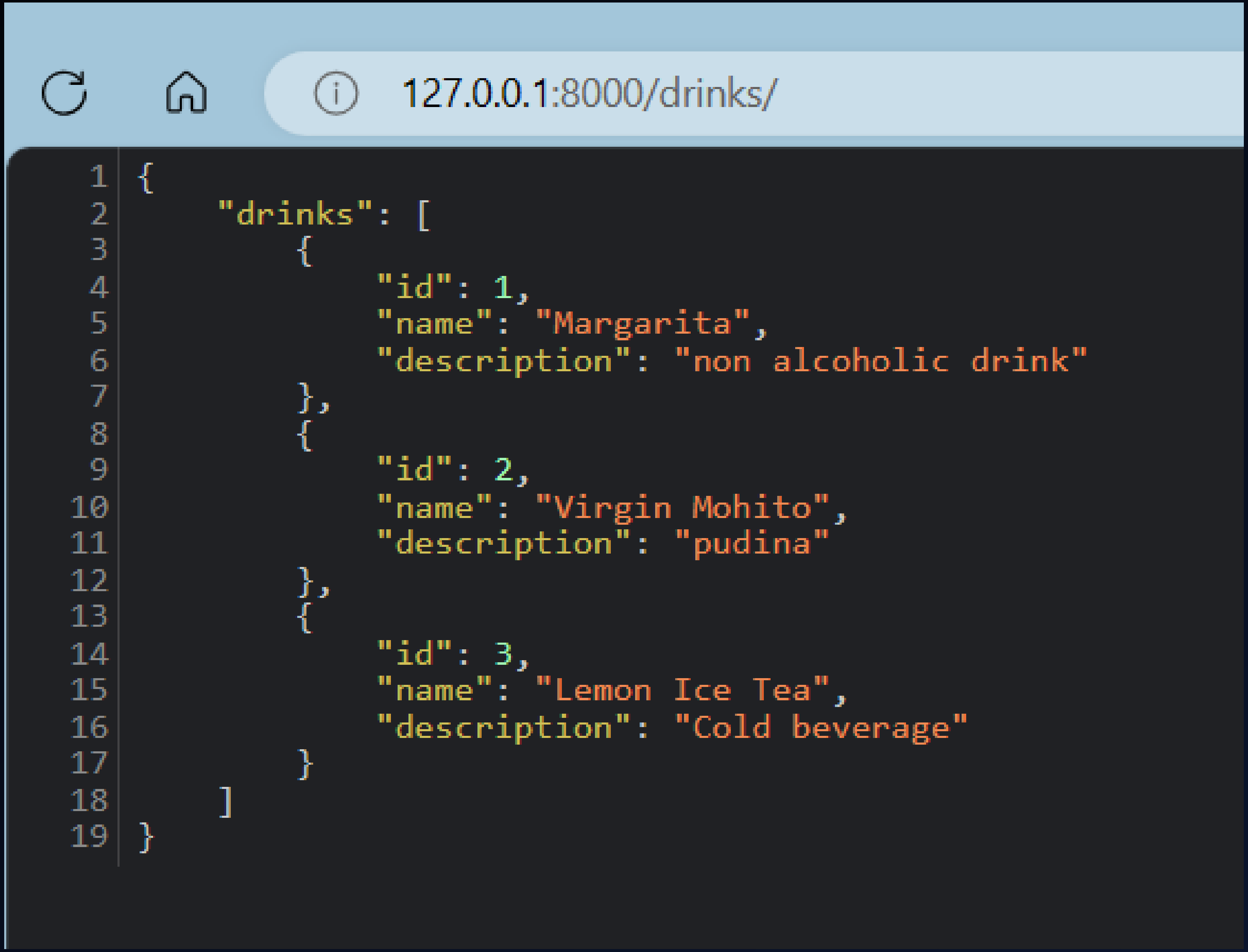
Select drink to change

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

DRINK
Lemon Ice Tea Cold beverage
Virgin Mohito pudina
Margarita non alcoholic drink

3 drinks

CLIENT PAGE

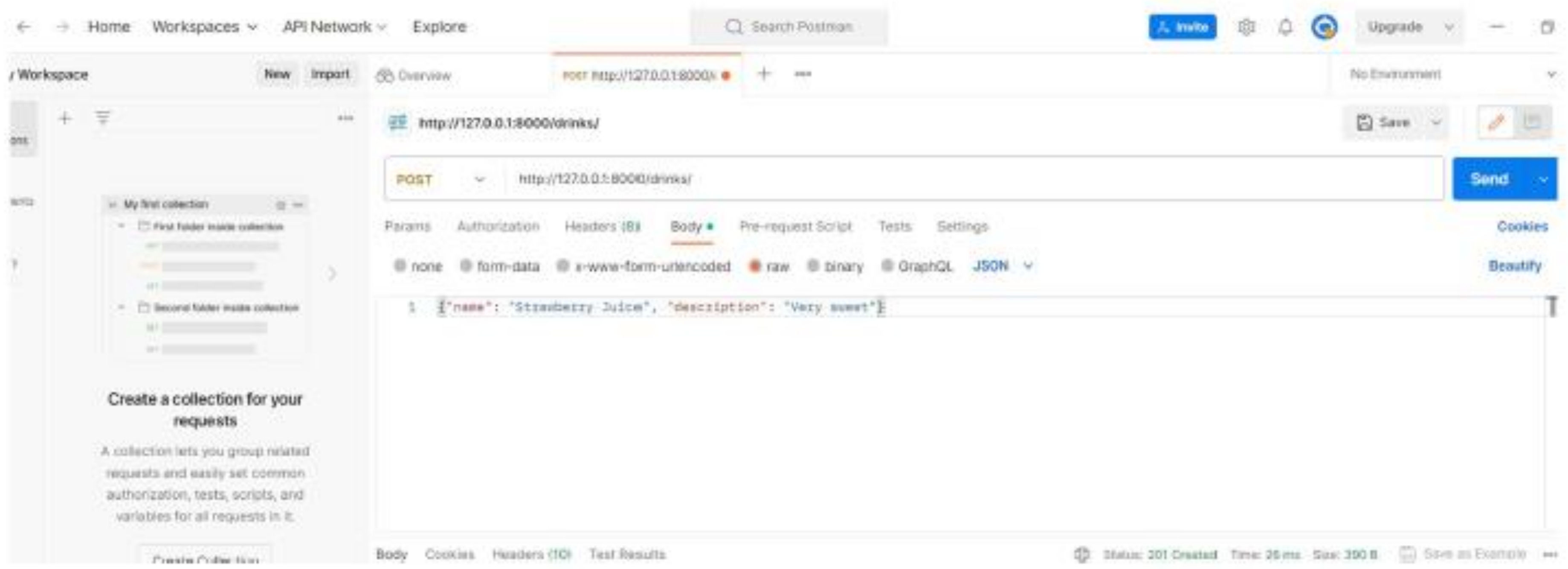


127.0.0.1:8000/drinks/

```
{
    "drinks": [
        {
            "id": 1,
            "name": "Margarita",
            "description": "non alcoholic drink"
        },
        {
            "id": 2,
            "name": "Virgin Mohito",
            "description": "pudina"
        },
        {
            "id": 3,
            "name": "Lemon Ice Tea",
            "description": "Cold beverage"
        }
    ]
}
```

POSTMAN

Adding data using postman tool



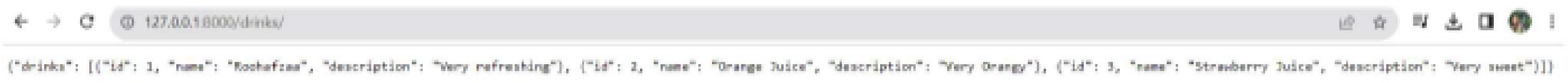
POST http://127.0.0.1:8000/drinks/

```
{"name": "Strawberry Juice", "description": "Very sweet"}
```

Status: 201 Created Time: 26 ms Size: 390 B

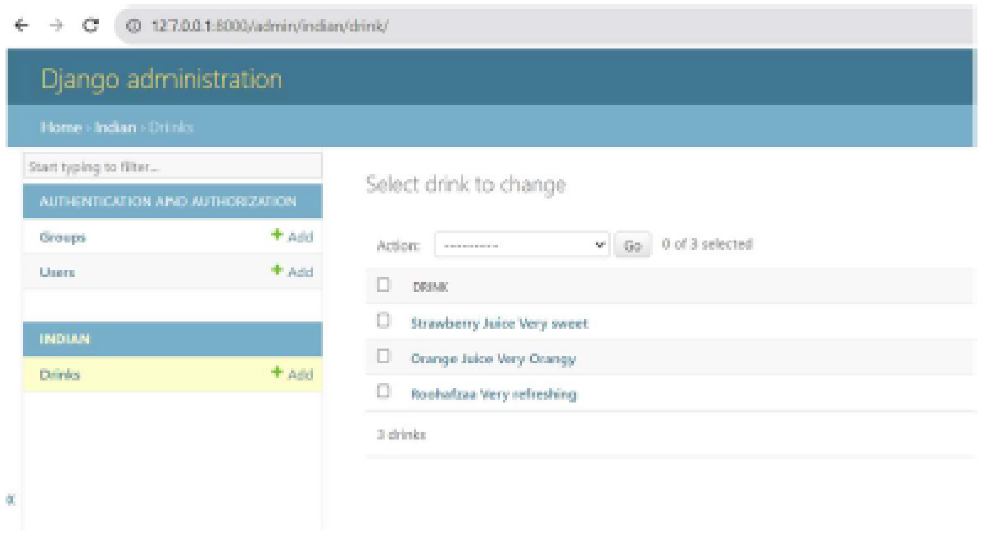
POSTMAN (DATA ADDED TO THE DB)

refresh



127.0.0.1:8000/drinks/

```
{"drinks": [{"id": 1, "name": "Roohafzaa", "description": "Very refreshing"}, {"id": 2, "name": "Orange Juice", "description": "Very Orangy"}, {"id": 3, "name": "Strawberry Juice", "description": "Very sweet"}]}
```



127.0.0.1:8000/admin/indian/drink/

Django administration

Home › Indian › Drinks

Select drink to change

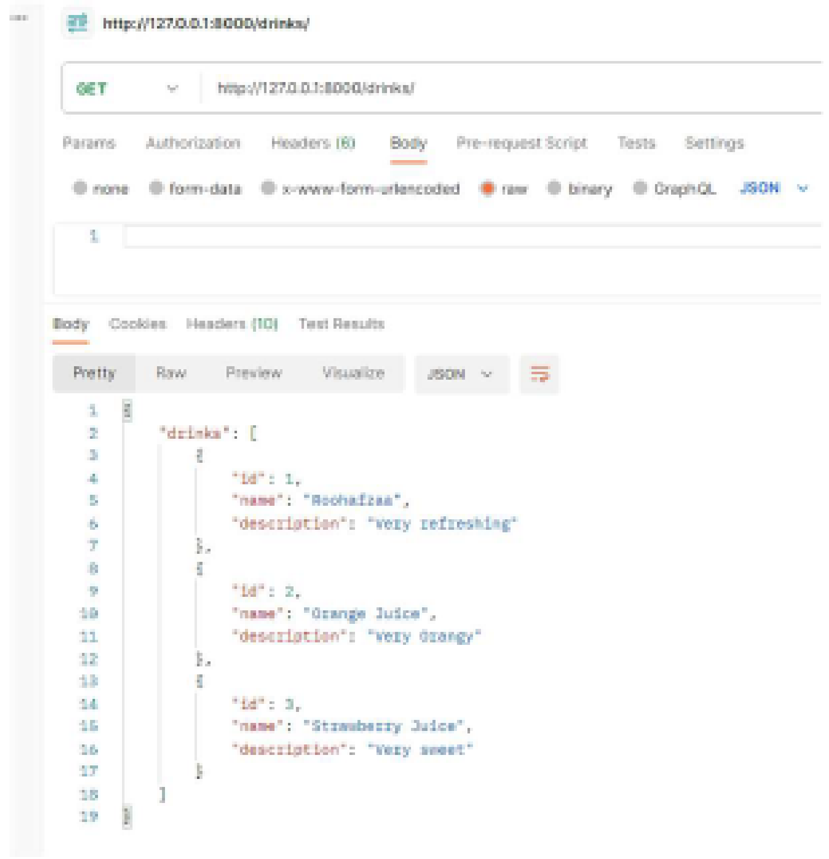
Action: ---------- Go 0 of 3 selected

DRINK
Strawberry Juice Very sweet
Orange Juice Very Orangy
Roohafzaa Very refreshing

3 drinks

POSTMAN

To check what changes has been done can also use GET in postman



GET http://127.0.0.1:8000/drinks/

```
{
    "drinks": [
        {
            "id": 1,
            "name": "Roohafzaa",
            "description": "Very refreshing"
        },
        {
            "id": 2,
            "name": "Orange Juice",
            "description": "Very Orangy"
        },
        {
            "id": 3,
            "name": "Strawberry Juice",
            "description": "Very sweet"
        }
    ]
}
```

DAY 9

Restful API

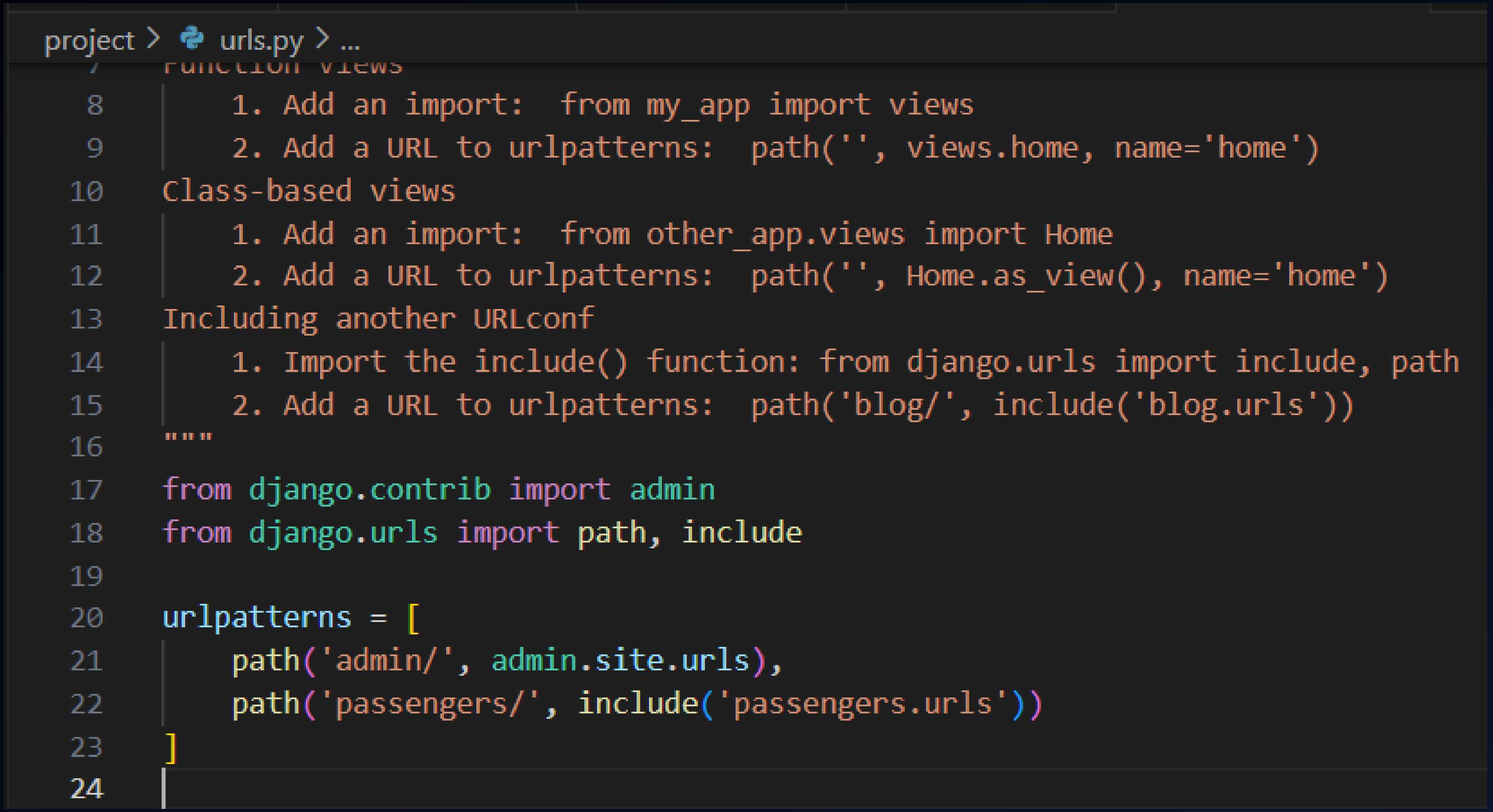
GET AND POST METHODS (POSTMAN)

ppt_by_sourabh_pal

INSTRUCTIONS

- DIRECTORY NAME: project
- PROJECT NAME: project
- APP NAME: passengers
- Pip install requests
- Pip install djangorestframework
- Python manage.py migrate, makemigrations, createsuperuser
- Create files – urls.py (app), serializers.py

URLS.PY - PROJECT

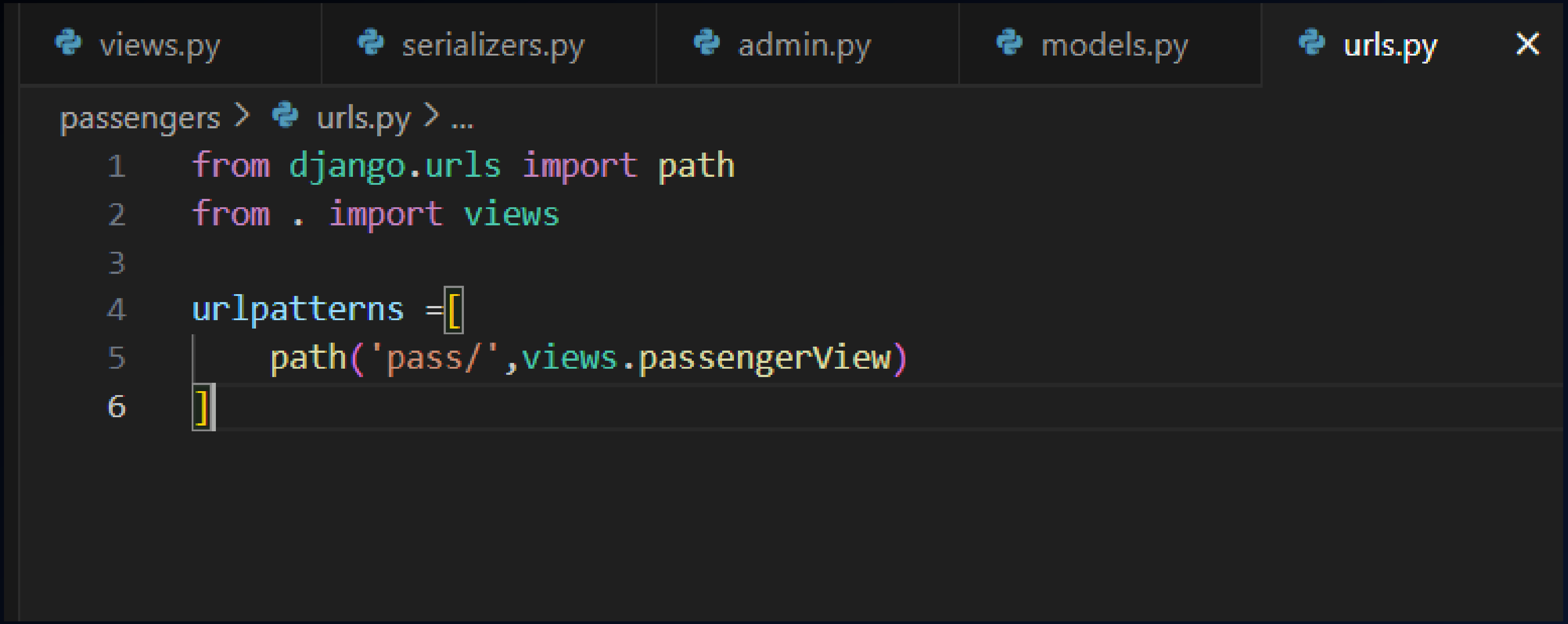


project > urls.py > ...

```
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('passengers/', include('passengers.urls'))
]

```

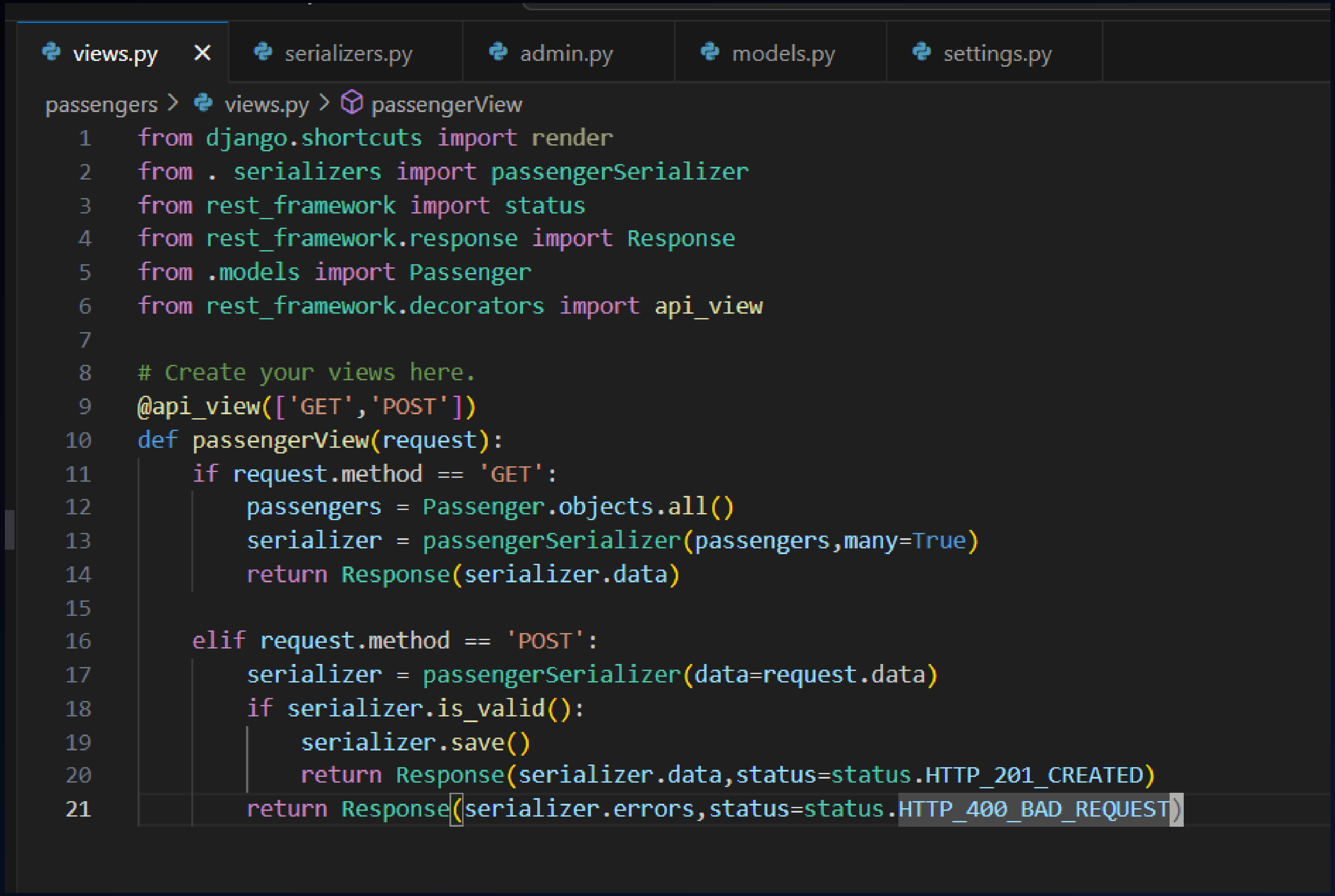
ppt_by_sourabh_pal

URLS.PY - APP



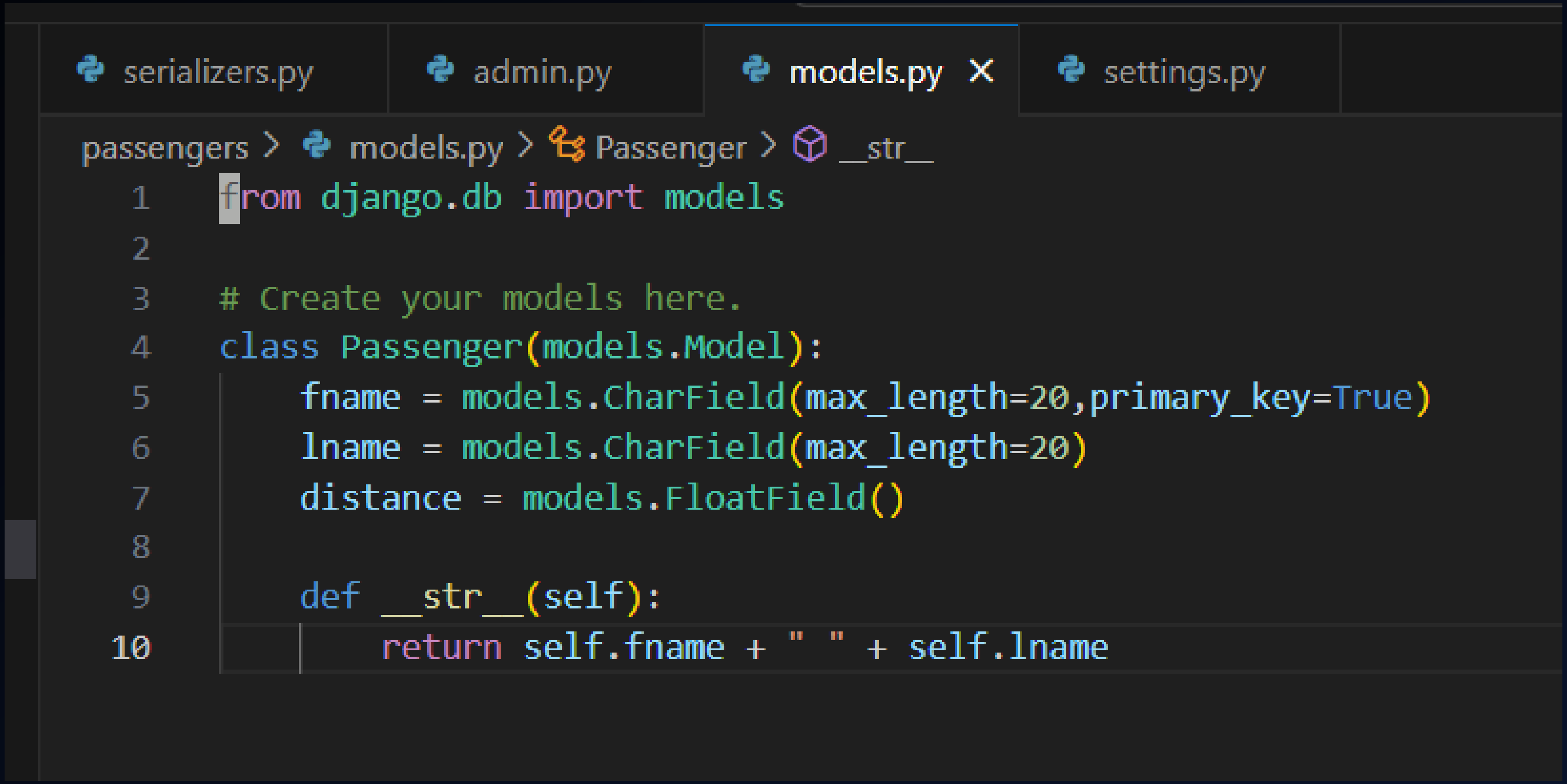
```python
from django.urls import path
from . import views

urlpatterns =[
    path('pass/',views.passengerView)
]
```

VIEWS.PY



```python
from django.shortcuts import render
from . serializers import passengerSerializer
from rest_framework import status
from rest_framework.response import Response
from .models import Passenger
from rest_framework.decorators import api_view

# Create your views here.
@api_view(['GET','POST'])
def passengerView(request):
    if request.method == 'GET':
        passengers = Passenger.objects.all()
        serializer = passengerSerializer(passengers,many=True)
        return Response(serializer.data)

    elif request.method == 'POST':
        serializer = passengerSerializer(data=request.data)
        if serializer.is_valid():
            serializer.save()
            return Response(serializer.data,status=status.HTTP_201_CREATED)
        return Response(serializer.errors,status=status.HTTP_400_BAD_REQUEST)
```

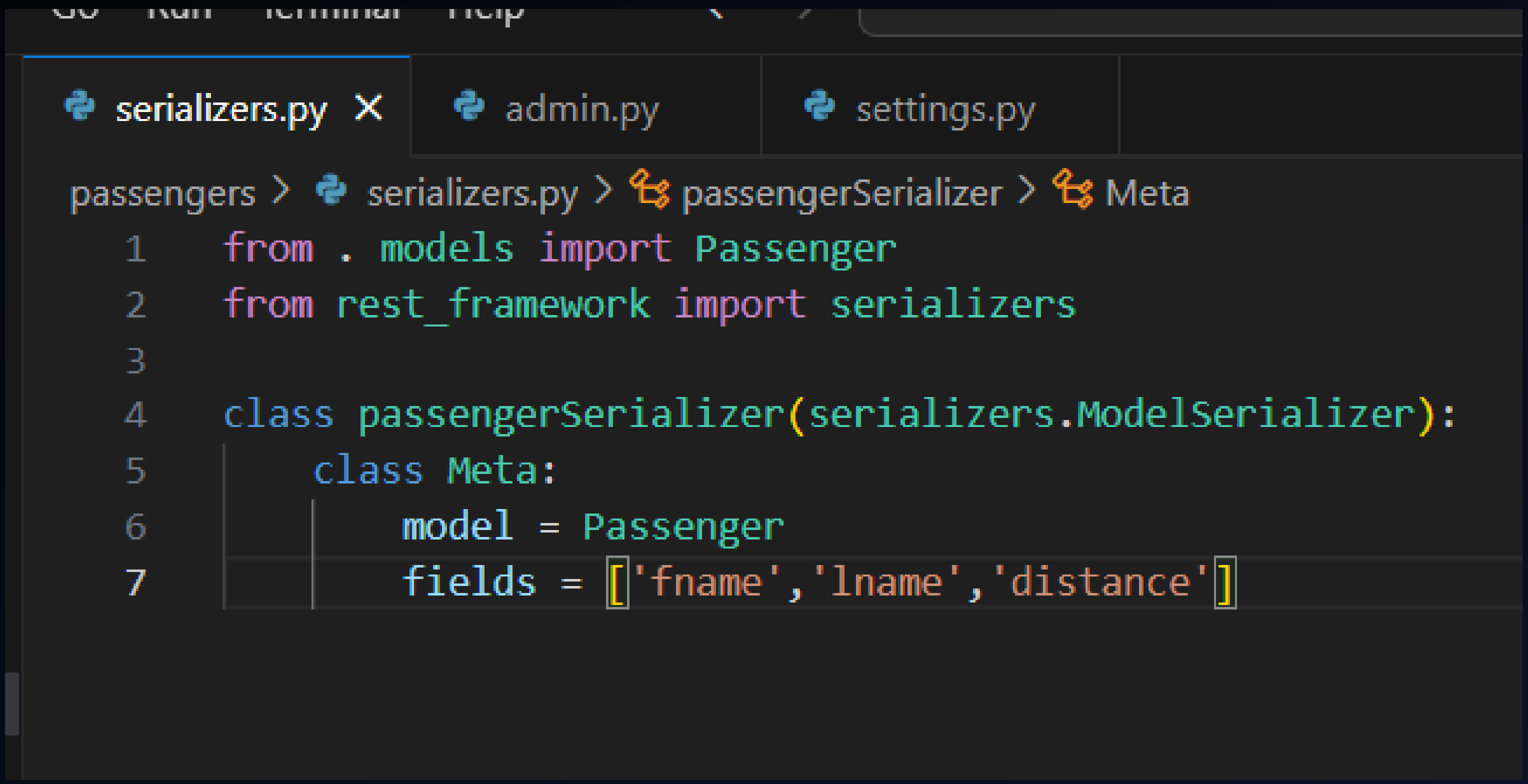
MODELS.PY



```python
from django.db import models

# Create your models here.
class Passenger(models.Model):
    fname = models.CharField(max_length=20,primary_key=True)
    lname = models.CharField(max_length=20)
    distance = models.FloatField()

    def __str__(self):
        return self.fname + " " + self.lname
```

ppt_by_sourabh_pal

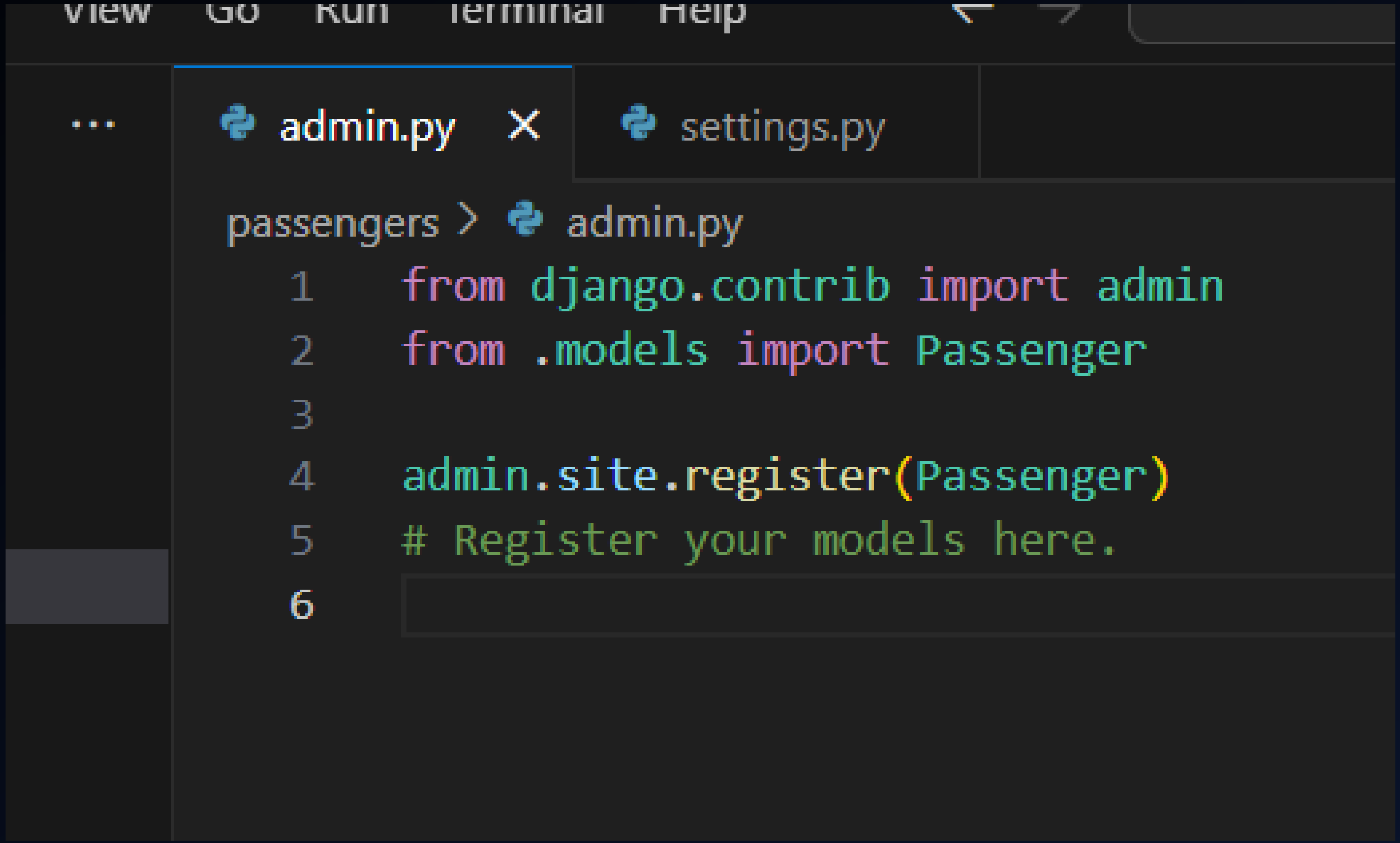
SERIALIZERS.PY



```python
from . models import Passenger
from rest_framework import serializers

class passengerSerializer(serializers.ModelSerializer):
    class Meta:
        model = Passenger
        fields = ['fname','lname','distance']
```

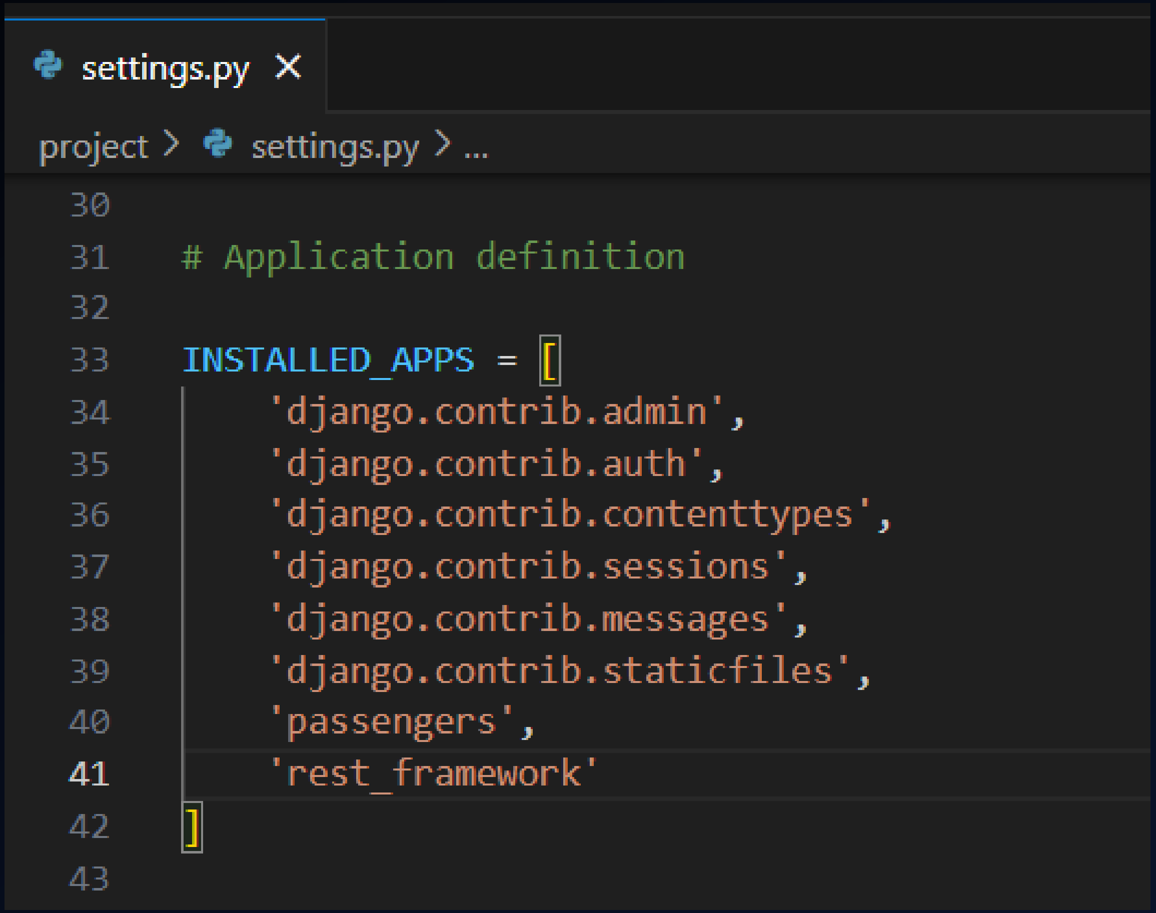
ppt_by_sourabh_pal

ADMIN.PY



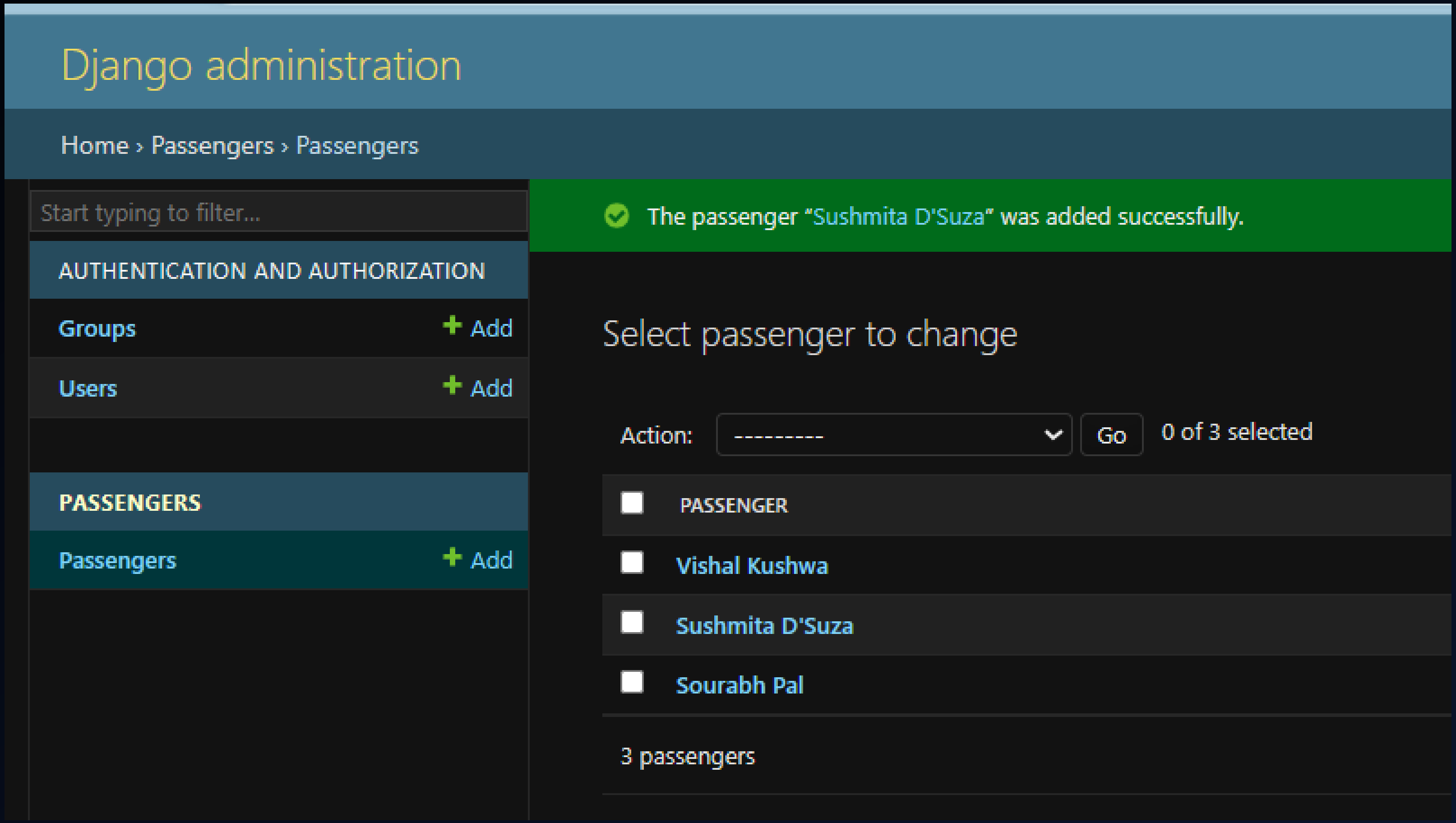
```python
from django.contrib import admin
from .models import Passenger

admin.site.register(Passenger)
# Register your models here.

```

SETTINGS.PY



```

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'passengers',
    'rest_framework'
]

```

ADMIN PAGE



Django administration

Home › Passengers › Passengers

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

PASSENGERS

Passengers + Add

The passenger "Sushmita D'Suza" was added successfully.

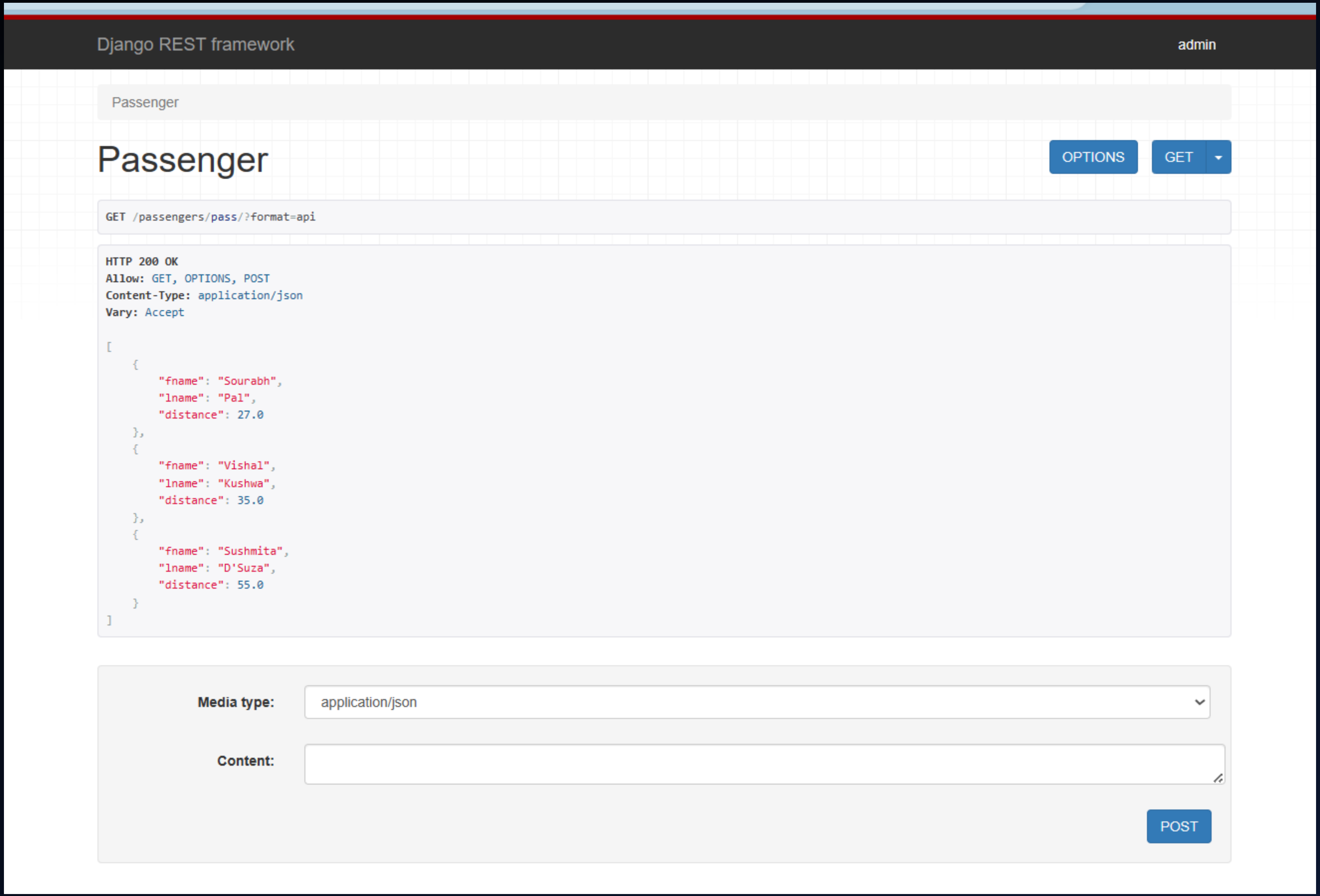
Select passenger to change

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

	PASSENGER
☐	Vishal Kushwa
☐	Sushmita D'Suza
☐	Sourabh Pal

3 passengers

CLIENT PAGE (API) - GET



Passenger

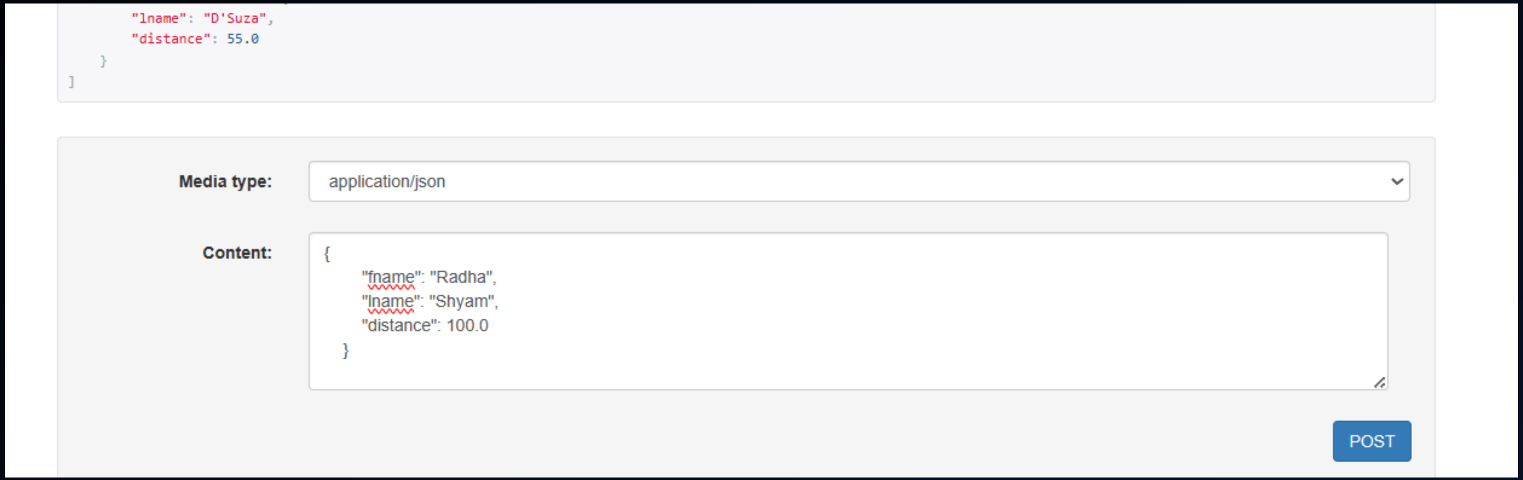
```
GET /passengers/pass/?format=api
```

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

[
    {
        "fname": "Sourabh",
        "lname": "Pal",
        "distance": 27.0
    },
    {
        "fname": "Vishal",
        "lname": "Kushwa",
        "distance": 35.0
    },
    {
        "fname": "Sushmita",
        "lname": "D'Suza",
        "distance": 55.0
    }
]
```

Media type: application/json

Content:

CLIENT PAGE (API) - POST



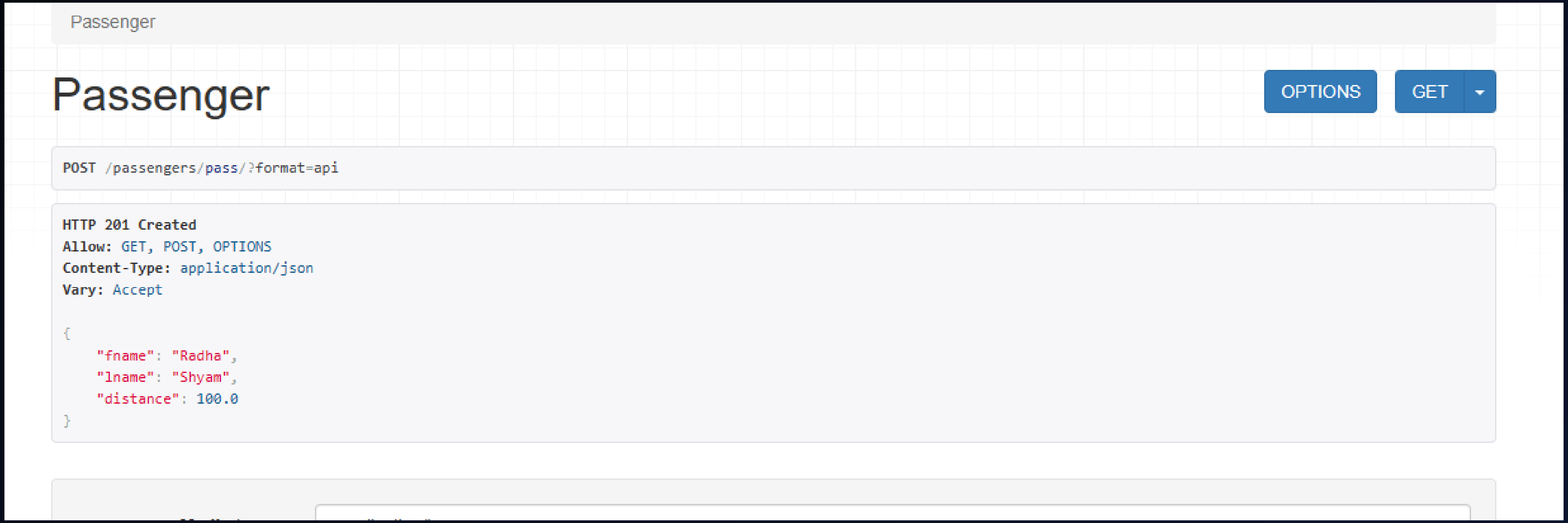
```
        "lname": "D'Suza",
        "distance": 55.0
    }
]
```

Media type: application/json

Content:

```
{
        "fname": "Radha",
        "lname": "Shyam",
        "distance": 100.0
    }
```

POST



Passenger

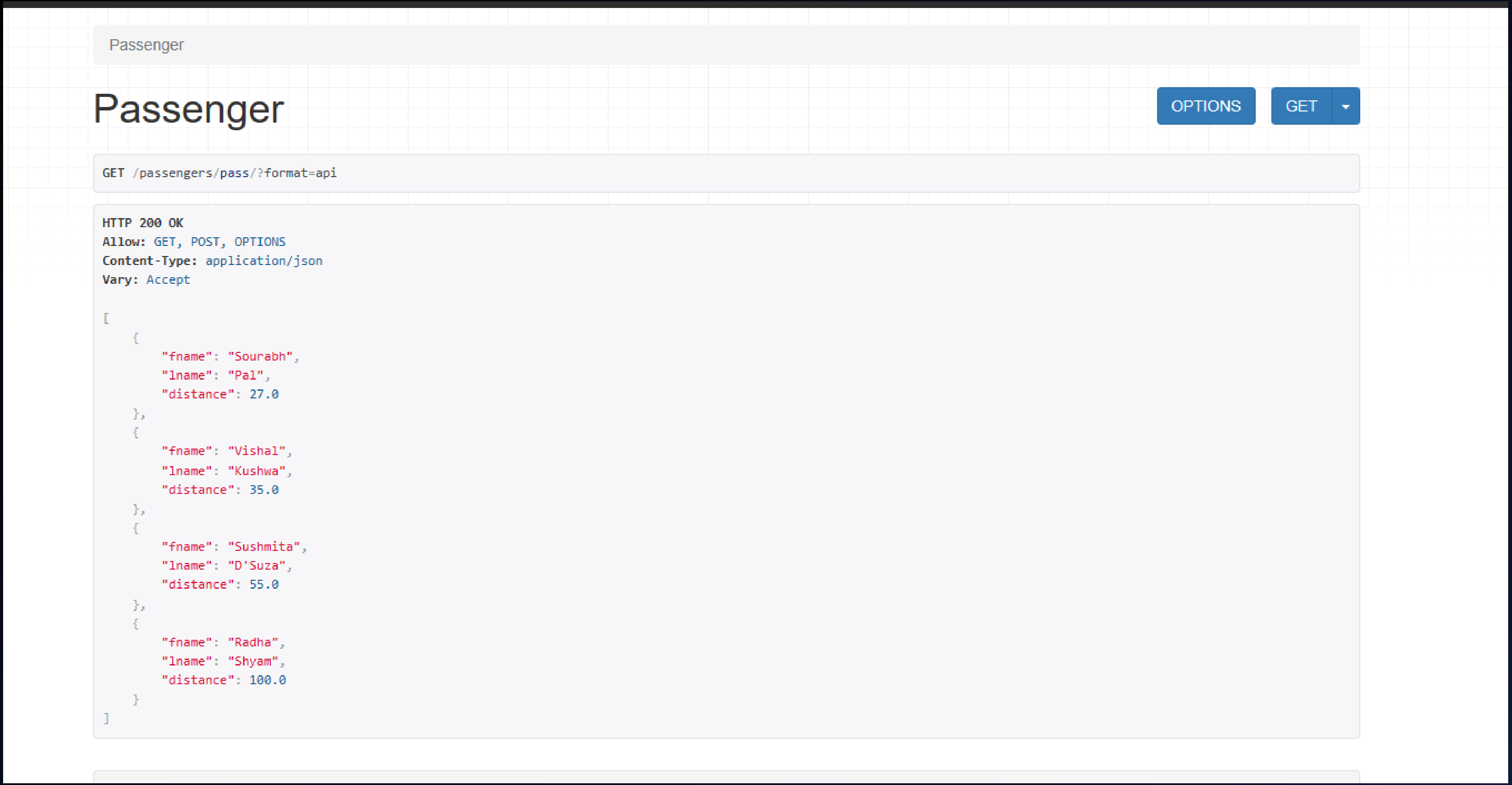
Passenger

OPTIONS GET

```
POST /passengers/pass/?format=api
```

```
HTTP 201 Created
Allow: GET, POST, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "fname": "Radha",
    "lname": "Shyam",
    "distance": 100.0
}
```

CLIENT PAGE (API) - POST



Passenger

```
GET /passengers/pass/?format=api
```

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

[
    {
        "fname": "Sourabh",
        "lname": "Pal",
        "distance": 27.0
    },
    {
        "fname": "Vishal",
        "lname": "Kushwa",
        "distance": 35.0
    },
    {
        "fname": "Sushmita",
        "lname": "D'Suza",
        "distance": 55.0
    },
    {
        "fname": "Radha",
        "lname": "Shyam",
        "distance": 100.0
    }
]
```

DAY 10

Project Web Scrapper

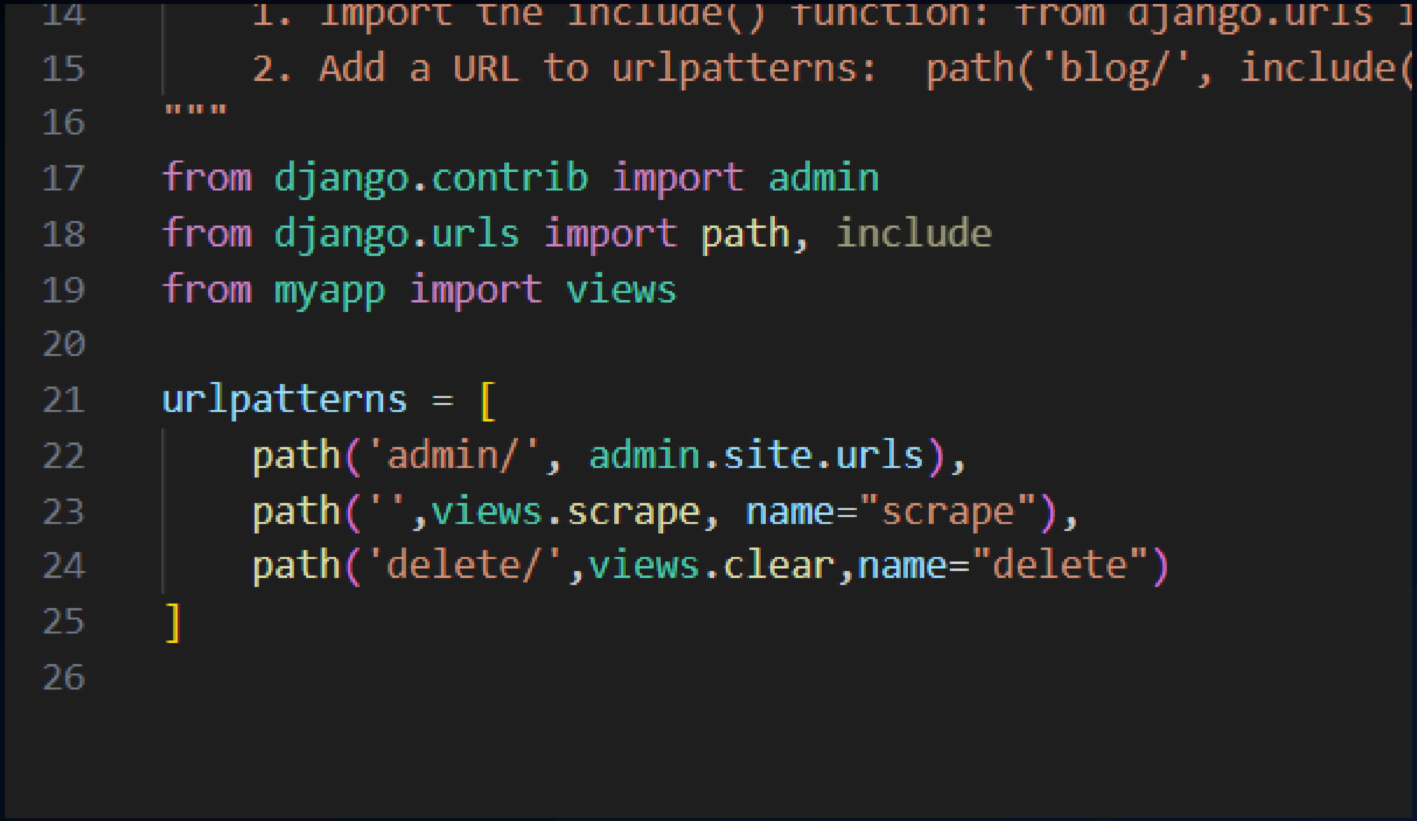
BEAUTIFULSOUP

ppt_by_sourabh_pal

INSTRUCTIONS

- DIRECTORY NAME: scraper
- PROJECT NAME: mysite
- APP NAME: myapp
- Pip install requests
- Pip install djangorestframework
- Pip install beautifulsoup4
- Python manage.py migrate, makemigrations
- Create files – urls.py (app), results.html (in templates)

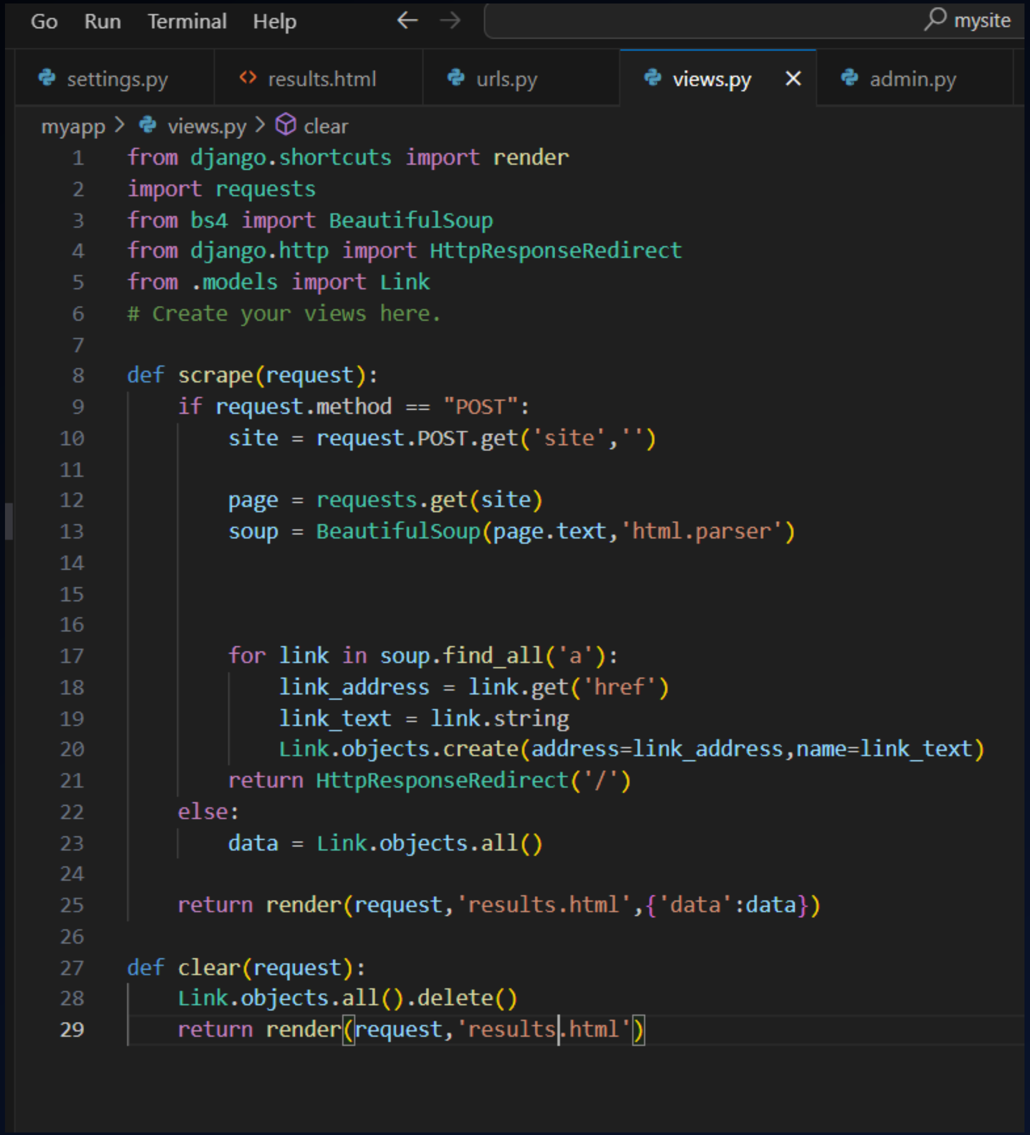
URLS.PY - PROJECT



```
    1. Import the include() function: from django.urls i
    2. Add a URL to urlpatterns:  path('blog/', include(
"""
from django.contrib import admin
from django.urls import path, include
from myapp import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('',views.scrape, name="scrape"),
    path('delete/',views.clear,name="delete")
]

```

ppt_by_sourabh_pal

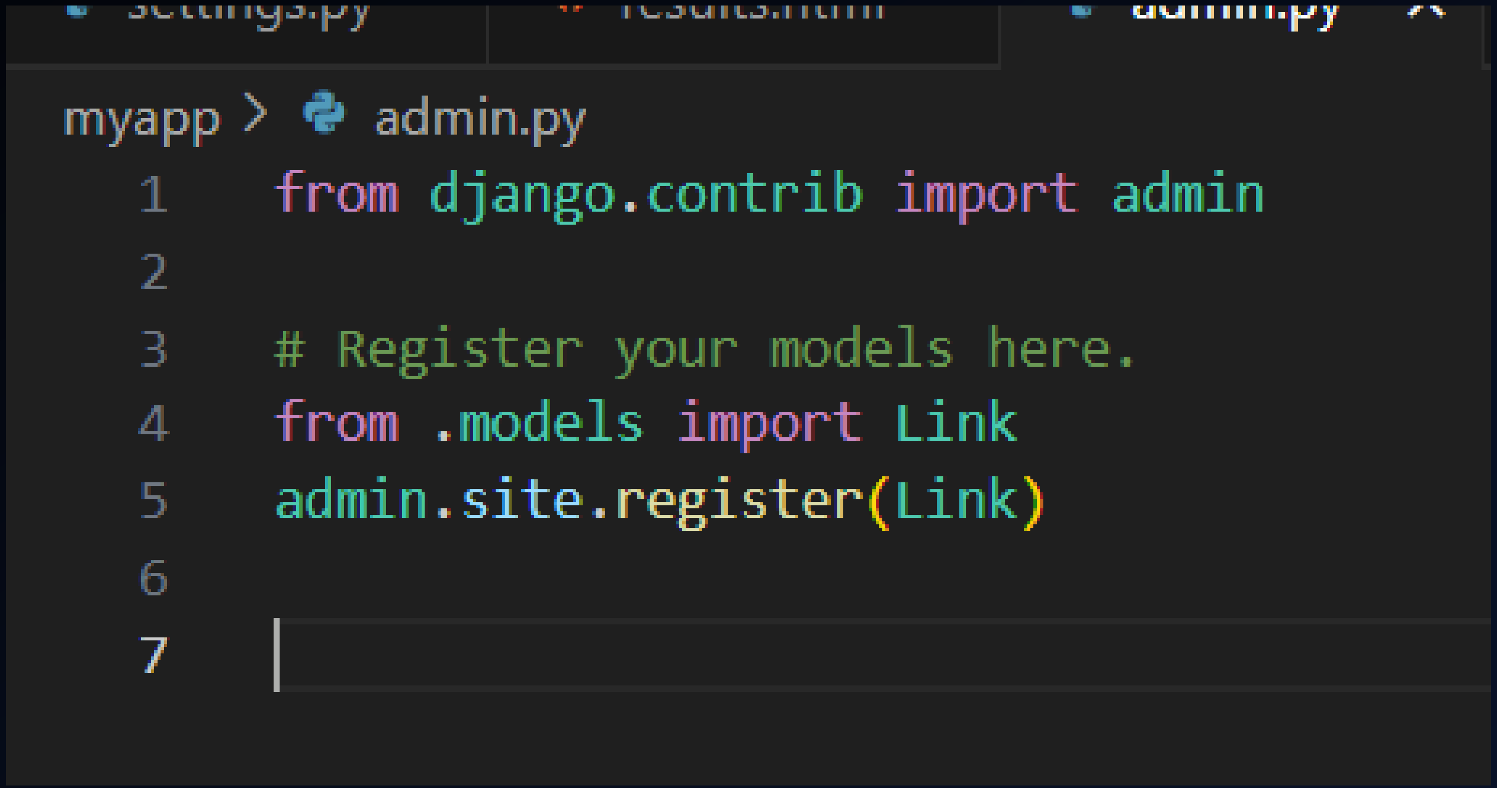
VIEWS.PY



```python
from django.shortcuts import render
import requests
from bs4 import BeautifulSoup
from django.http import HttpResponseRedirect
from .models import Link
# Create your views here.

def scrape(request):
    if request.method == "POST":
        site = request.POST.get('site','')

        page = requests.get(site)
        soup = BeautifulSoup(page.text,'html.parser')



        for link in soup.find_all('a'):
            link_address = link.get('href')
            link_text = link.string
            Link.objects.create(address=link_address,name=link_text)
        return HttpResponseRedirect('/')
    else:
        data = Link.objects.all()

    return render(request,'results.html',{'data':data})

def clear(request):
    Link.objects.all().delete()
    return render(request,'results.html')
```

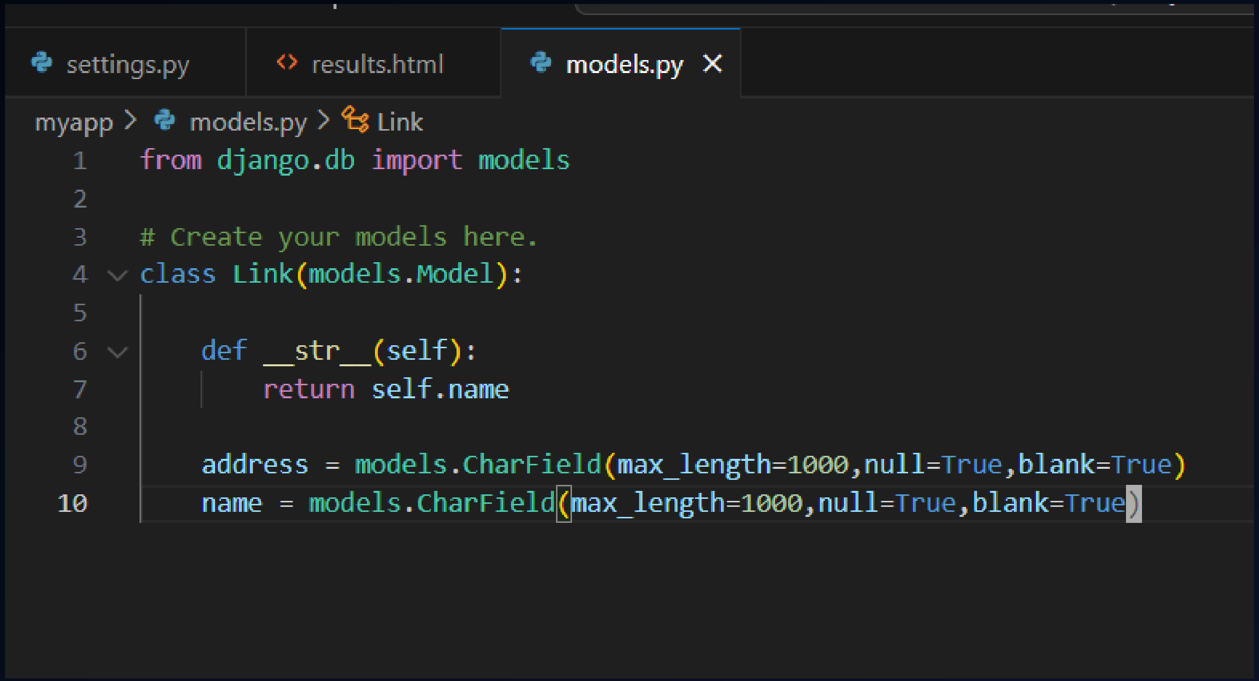
ppt_by_sourabh_pal

ADMIN.PY



```
from django.contrib import admin

# Register your models here.
from .models import Link
admin.site.register(Link)

```

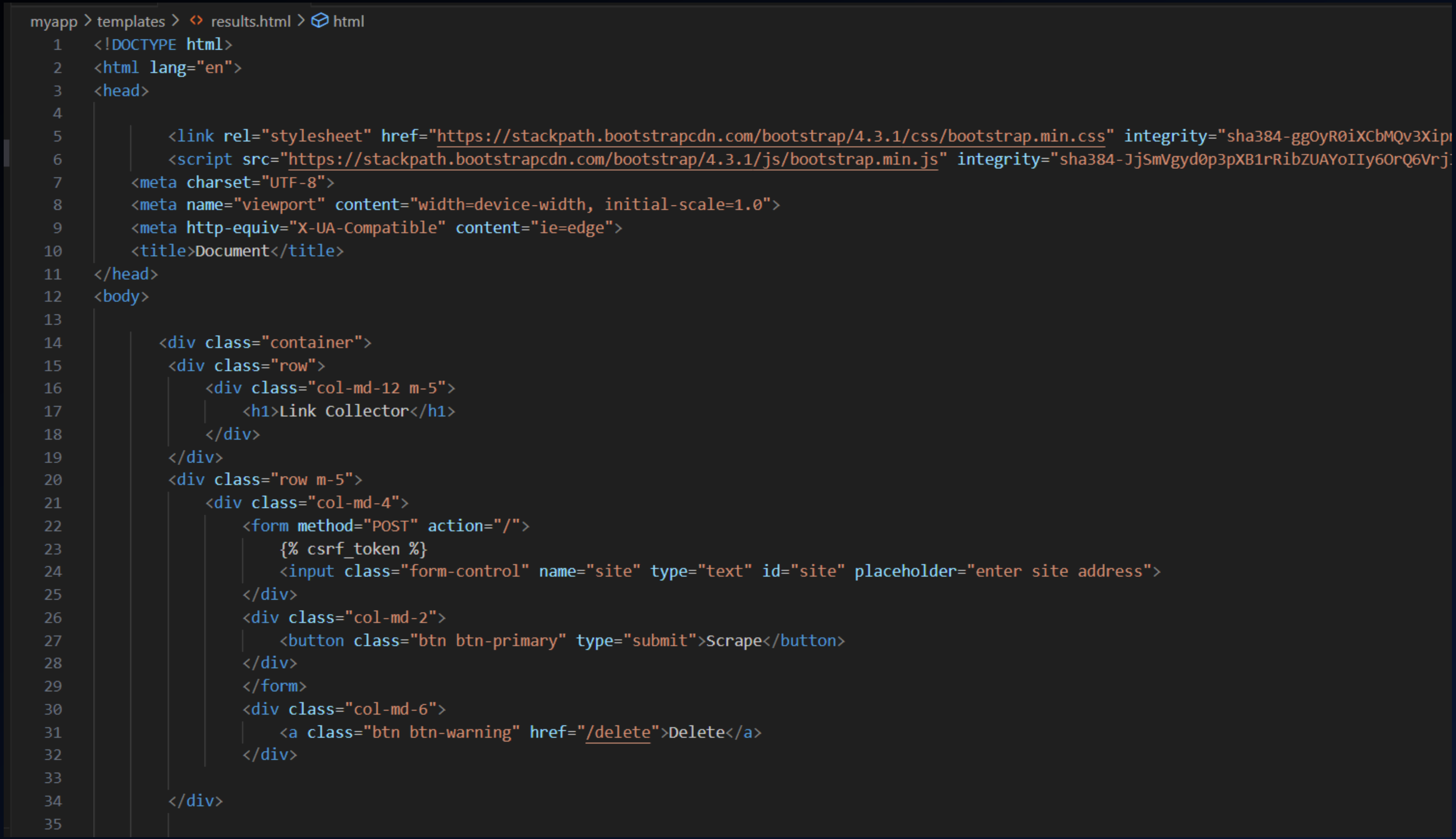
MODELS.PY



```python
from django.db import models

# Create your models here.
class Link(models.Model):

    def __str__(self):
        return self.name

    address = models.CharField(max_length=1000,null=True,blank=True)
    name = models.CharField(max_length=1000,null=True,blank=True)
```

ppt_by_sourabh_pal

RESULTS.HTML

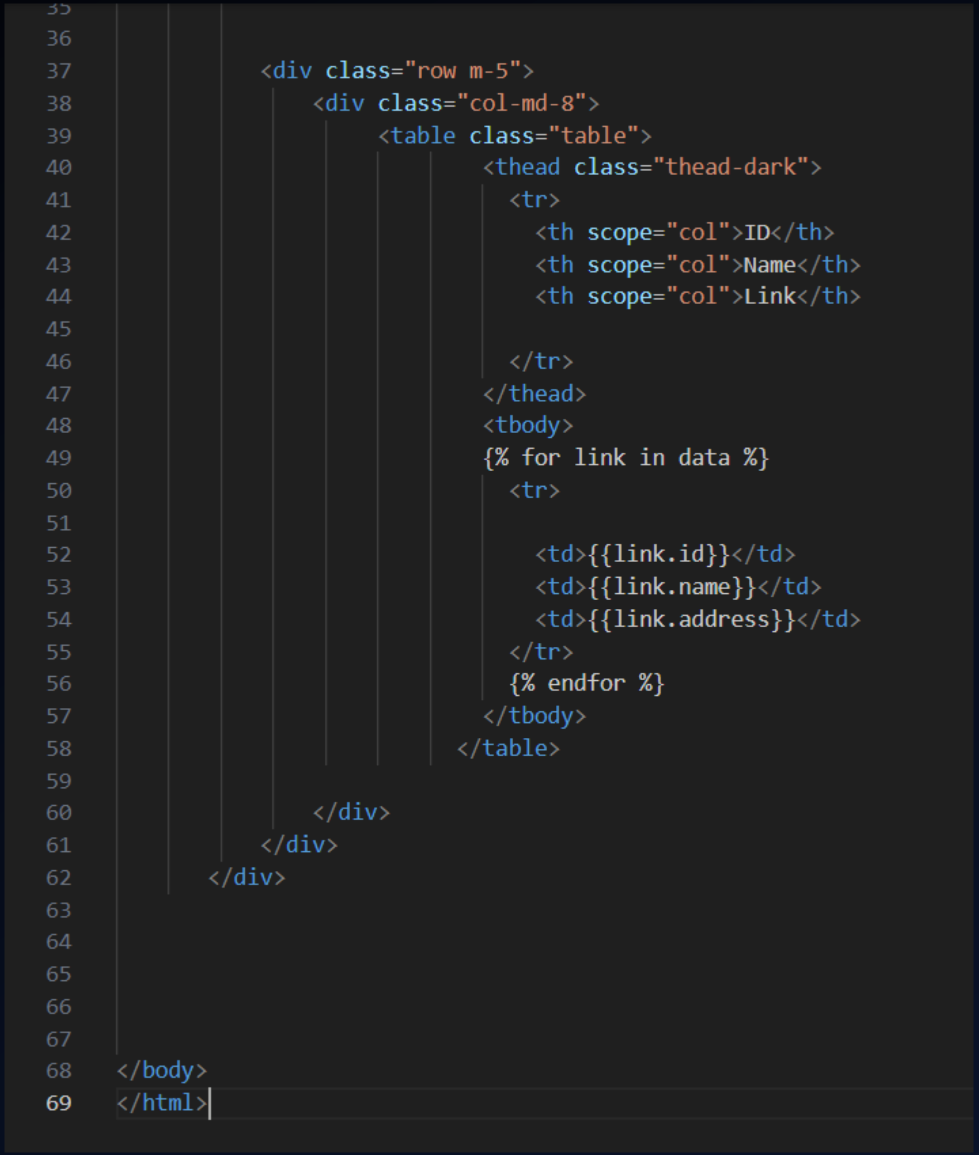


myapp > templates > results.html > html

```
<!DOCTYPE html>
<html lang="en">
<head>

        <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipr
        <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6Vrj
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
</head>
<body>

       <div class="container">
        <div class="row">
            <div class="col-md-12 m-5">
                <h1>Link Collector</h1>
            </div>
        </div>
        <div class="row m-5">
            <div class="col-md-4">
                <form method="POST" action="/">
                    {% csrf_token %}
                    <input class="form-control" name="site" type="text" id="site" placeholder="enter site address">
                </div>
                <div class="col-md-2">
                    <button class="btn btn-primary" type="submit">Scrape</button>
                </div>
                </form>
                <div class="col-md-6">
                    <a class="btn btn-warning" href="/delete">Delete</a>
                </div>

        </div>

```

ppt_by_sourabh_pal

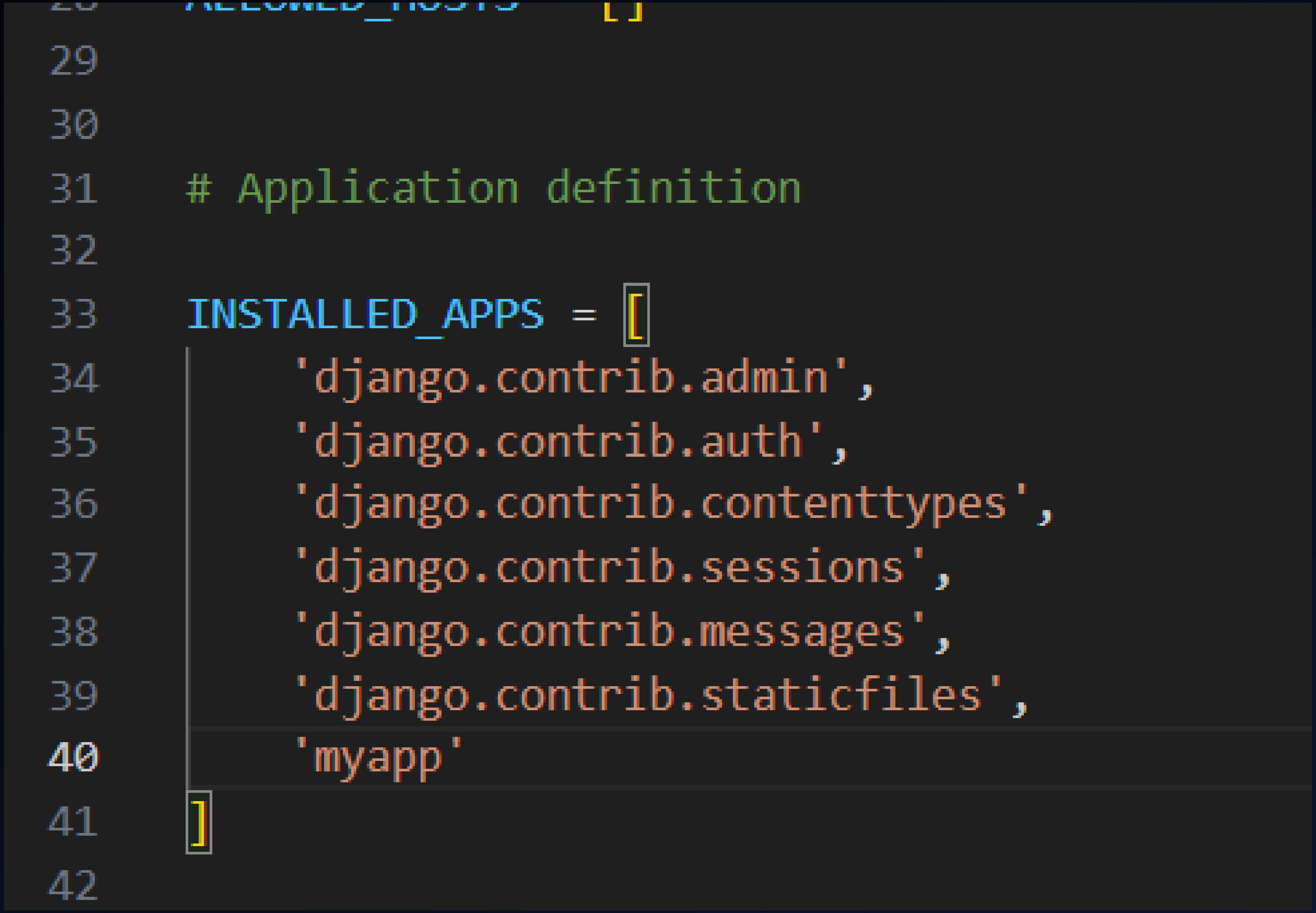
RESULTS.HTML



```


        <div class="row m-5">
            <div class="col-md-8">
                <table class="table">
                        <thead class="thead-dark">
                          <tr>
                            <th scope="col">ID</th>
                            <th scope="col">Name</th>
                            <th scope="col">Link</th>

                          </tr>
                        </thead>
                        <tbody>
                        {% for link in data %}
                          <tr>

                            <td>{{link.id}}</td>
                            <td>{{link.name}}</td>
                            <td>{{link.address}}</td>
                          </tr>
                          {% endfor %}
                        </tbody>
                      </table>

            </div>
        </div>
    </div>





</body>
</html>
```

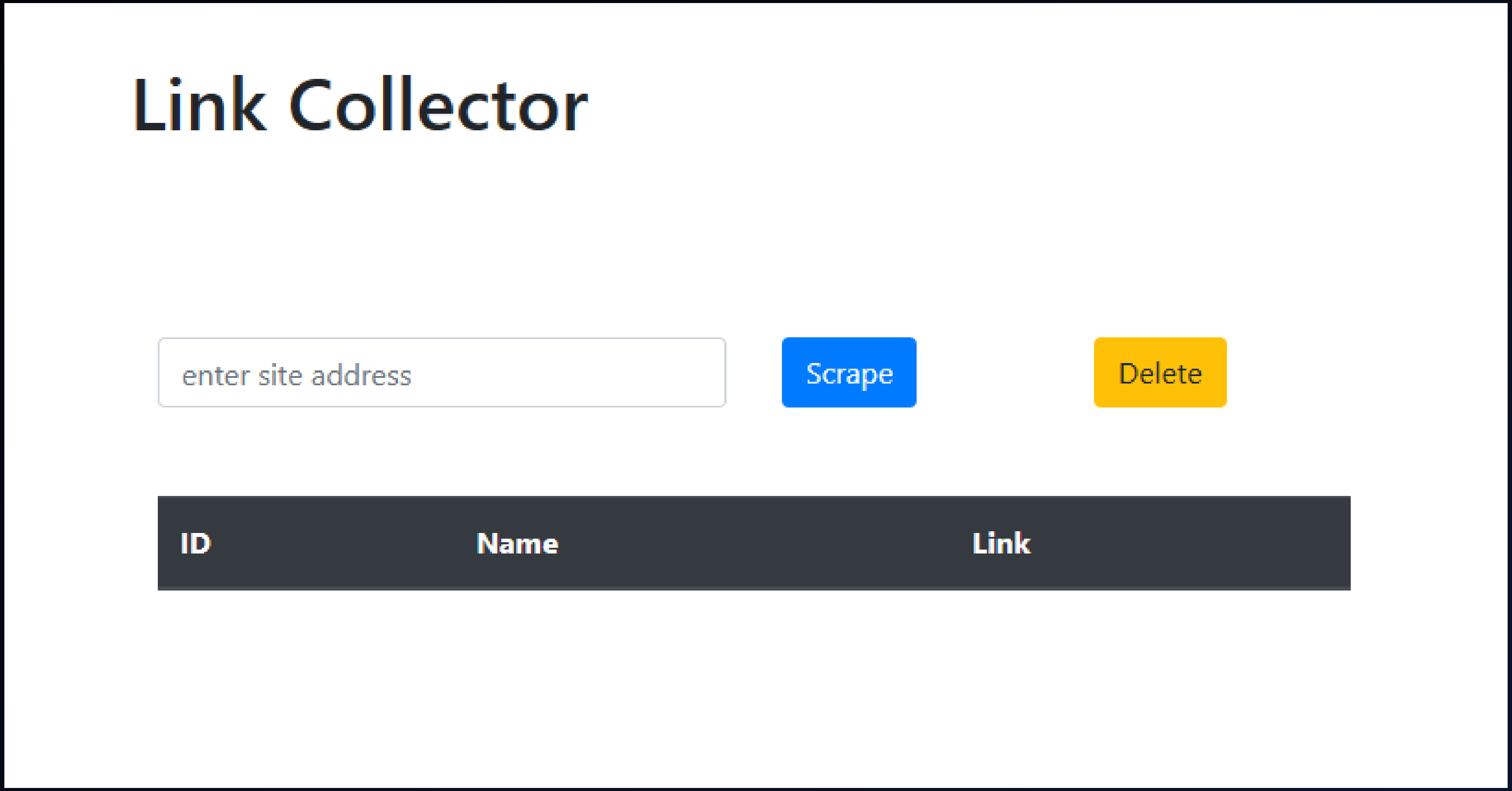
ppt_by_sourabh_pal

SETTINGS.PY



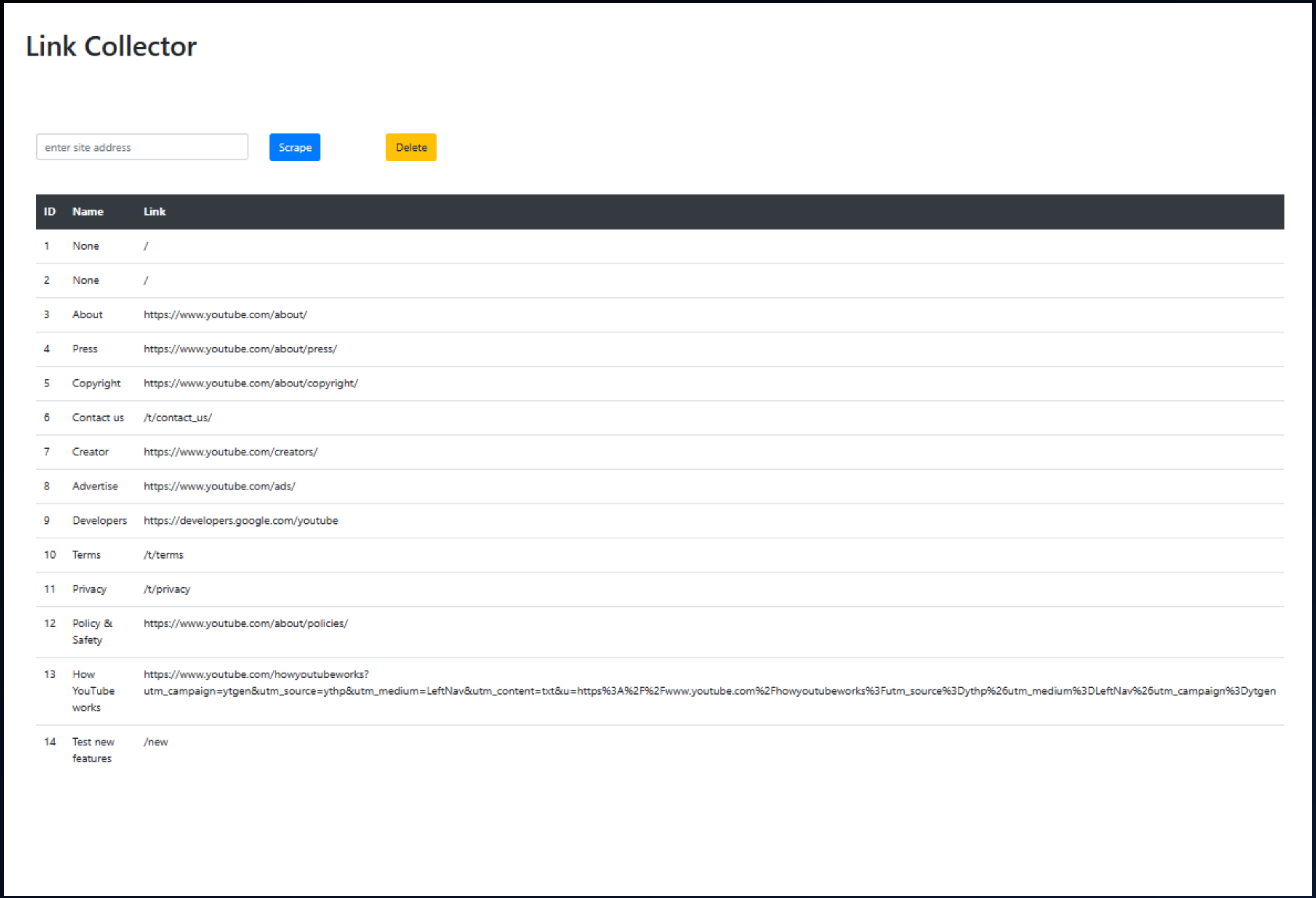
```
ALLOWED_HOSTS = []


# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'myapp'
]

```

ADMIN PAGE


Link Collector

ID	Name	Link

ADMIN PAGE - SCRAPPING

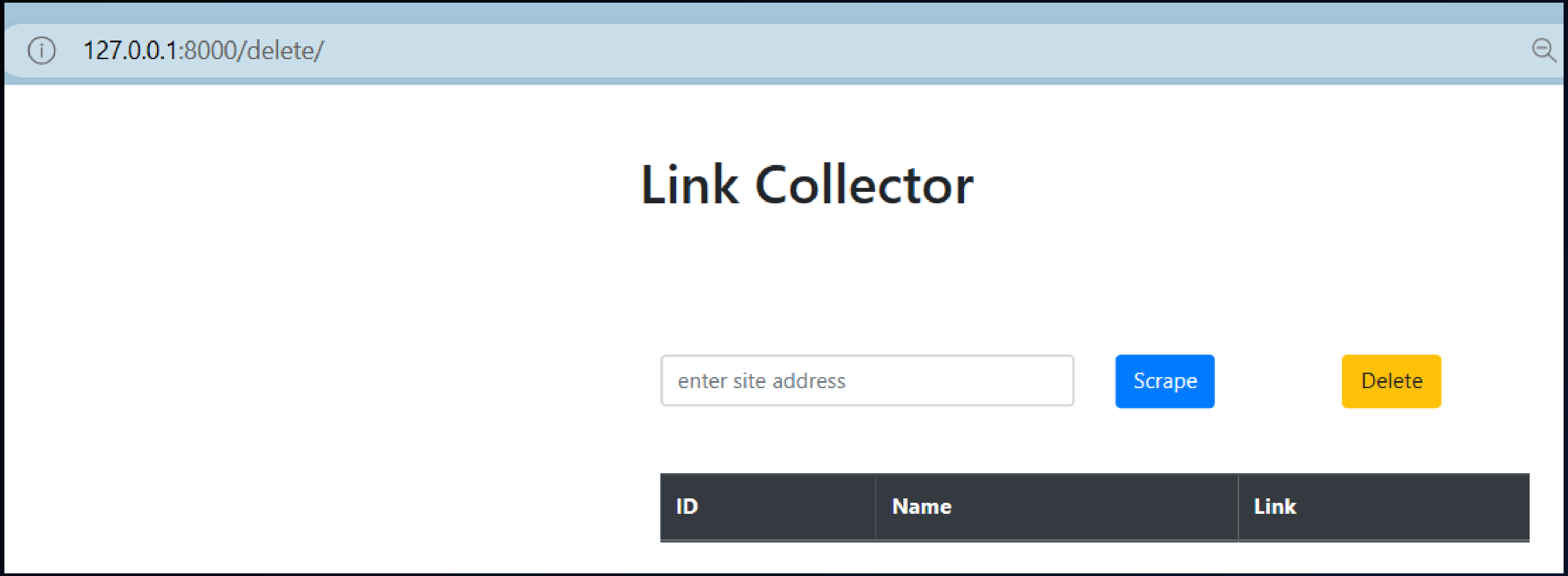


Link Collector

ID	Name	Link
1	None	/
2	None	/
3	About	https://www.youtube.com/about/
4	Press	https://www.youtube.com/about/press/
5	Copyright	https://www.youtube.com/about/copyright/
6	Contact us	/t/contact_us/
7	Creator	https://www.youtube.com/creators/
8	Advertise	https://www.youtube.com/ads/
9	Developers	https://developers.google.com/youtube
10	Terms	/t/terms
11	Privacy	/t/privacy
12	Policy & Safety	https://www.youtube.com/about/policies/
13	How YouTube works	https://www.youtube.com/howyoutubeworks?utm_campaign=ytgen&utm_source=ythp&utm_medium=LeftNav&utm_content=txt&u=https%3A%2F%2Fwww.youtube.com%2Fhowyoutubeworks%3Futm_source%3Dythp%26utm_medium%3DLeftNav%26utm_campaign%3Dytgen
14	Test new features	/new

ppt_by_sourabh_pal

ADMIN PAGE - DELETING


127.0.0.1:8000/delete/

Link Collector

enter site address

Scrape

Delete

ID	Name	Link

ppt_by_sourabh_pal

LAST UPDATED ON: 31ST OCTOBER 2023 (LAST CLASS)