

Lab 5

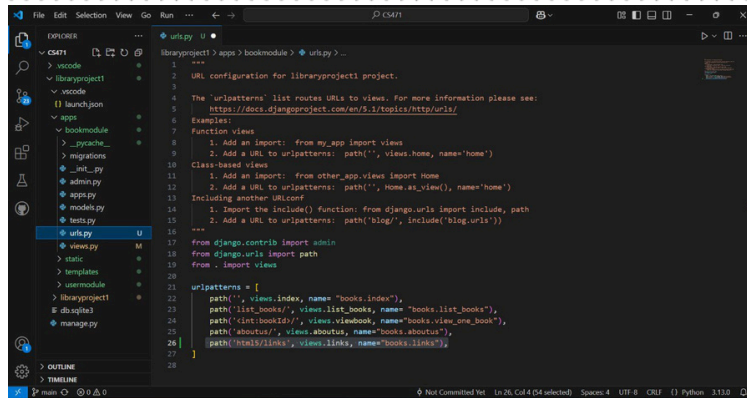


HTML and CSS (Part 2)

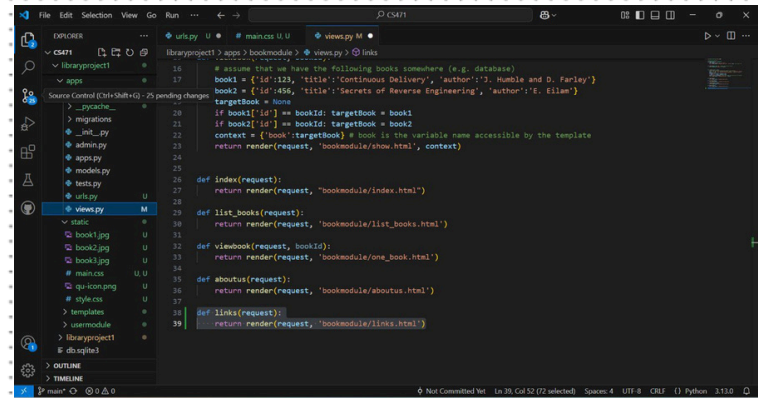
Student Name: Joud Alblaihed
University ID: 422205657
Subject: CS 471
Section number: 3526
Doctor Name: Bshayer Alkhalifa

Task 1

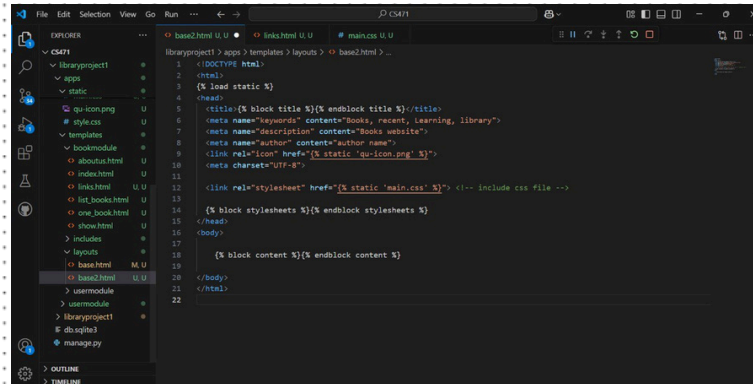
Code



```
1 # -*- coding: utf-8 -*-
2 # URL configuration for libraryproject project.
3
4 The 'urlpatterns' list routes URLs to views. For more information please see:
5 https://docs.djangoproject.com/en/5.1/topics/http/urls/
6 Examples:
7 function views
8 1. Add an import: from my_app import views
9 2. Add a URL to urlpatterns: path('', views.home, name='home')
10 Class-based views
11 1. Add an import: from other_app.views import Home
12 2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
13 Including another URLconf
14 1. Import the include() function: from django.urls import include, path
15 2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
16
17 from django.contrib import admin
18 from django.urls import path
19 from . import views
20
21 urlpatterns = [
22     path('', views.index, name='books.index'),
23     path('list_books/', views.list_books, name='books.list_books'),
24     path('list_bookid/', views.view_book, name='books.view_one_book'),
25     path('aboutus/', views.aboutus, name='books.aboutus'),
26     path('books/links', views.links, name='books.links'),
27 ]
```



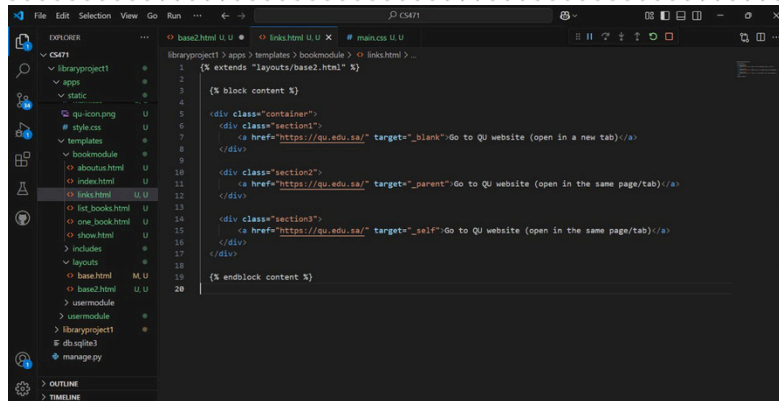
```
1 # -*- coding: utf-8 -*-
2 # Create your views here.
3
4 # assume that we have the following books somewhere (e.g. database)
5 book1 = {'id':123, 'title':'Continuous Delivery', 'author':'J. Humble and D. Farley'}
6 book2 = {'id':1456, 'title':'Secrets of Reverse Engineering', 'author':'E. Eilam'}
7 targetbook = None
8
9 if book1['id'] == bookId: targetbook = book1
10 if book2['id'] == bookId: targetbook = book2
11 context = {'book':targetbook} # book is the variable name accessible by the template
12 return render(request, 'bookmodule/show.html', context)
13
14 def index(request):
15     return render(request, 'bookmodule/index.html')
16
17 def list_books(request):
18     return render(request, 'bookmodule/list_books.html')
19
20 def view_book(request, bookId):
21     return render(request, 'bookmodule/one_book.html')
22
23 # main.py
24
25 def aboutus(request):
26     return render(request, 'bookmodule/aboutus.html')
27
28 # style.css
29
30 # templates
31
32 def links(request):
33     return render(request, 'bookmodule/links.html')
34
35 # libraryproject
36
37 # db.sqlite3
38
39 # manage.py
```



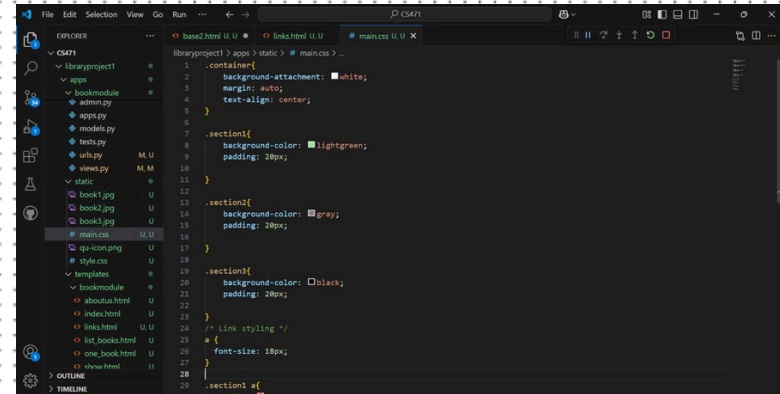
```
1 <!-- DOCTYPE HTML -->
2 <html>
3 <% load static %>
4 <head>
5 <title{% block title %}{% endblock title %}</title>
6 <meta name="keywords" content="Books, recent, learning, library">
7 <meta name="description" content="Books website">
8 <meta name="author" content="author name">
9 <link rel="icon" href="{% static 'qu-icon.png' %}">
10 <meta charset="UTF-8">
11
12 <link rel="stylesheet" href="{% static 'main.css' %}"> <!-- include css file -->
13
14 <% block stylesheets %}{% endblock stylesheets %>
15 </head>
16 <body>
17
18 <% block content %}{% endblock content %>
19
20 </body>
21 </html>
```

Task 1

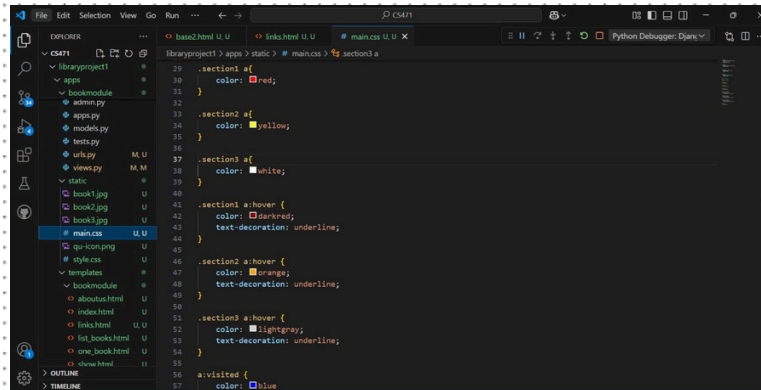
Code



```
1 {% extends "layouts/base2.html" %}
2
3 {% block content %}
4
5 <div class="container">
6   <div class="section1">
7     <a href="https://qu.edu.sa/" target="_blank">Go to QU website (open in a new tab)</a>
8   </div>
9   <div class="section2">
10    <a href="https://qu.edu.sa/" target="_parent">Go to QU website (open in the same page/tab)</a>
11  </div>
12   <div class="section3">
13    <a href="https://qu.edu.sa/" target="_self">Go to QU website (open in the same page/tab)</a>
14  </div>
15 </div>
16
17 {% endblock content %}
```



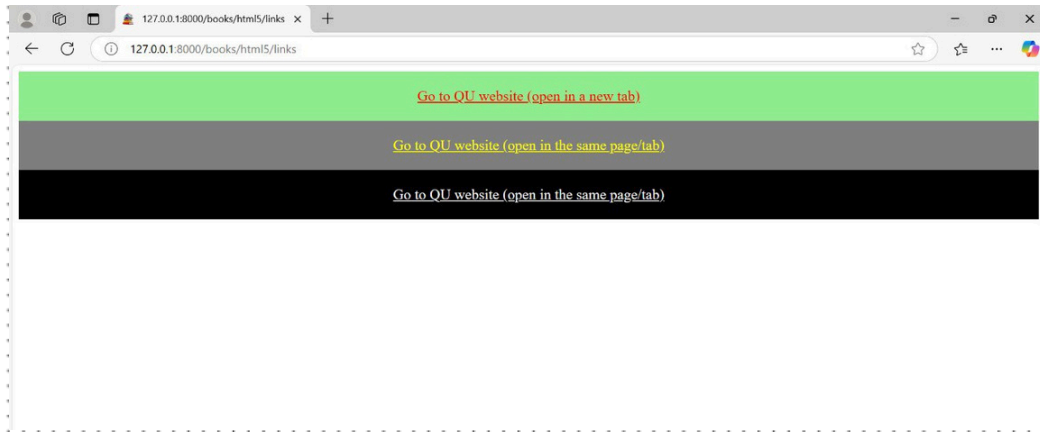
```
1 .container{
2   background-attachment: white;
3   margin: auto;
4   text-align: center;
5 }
6
7 .section1{
8   background-color: lightgreen;
9   padding: 20px;
10 }
11
12 .section2{
13   background-color: gray;
14   padding: 20px;
15 }
16
17 .section3{
18   background-color: black;
19   padding: 20px;
20 }
21
22 /* Link styling */
23 a {
24   font-size: 18px;
25 }
26
27 .section1 a{
28 }
```



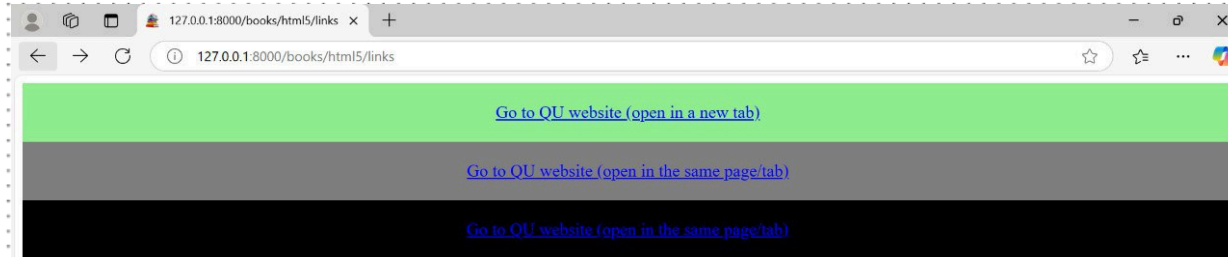
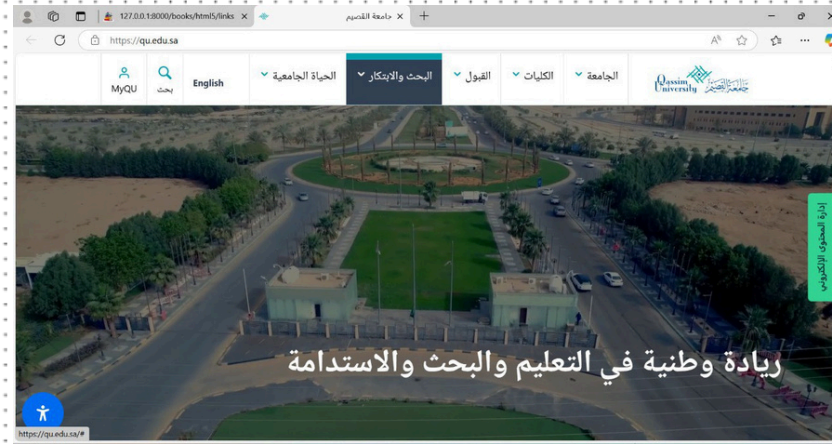
```
29 .section1 a{
30   color: red;
31 }
32
33 .section2 a{
34   color: yellow;
35 }
36
37 .section3 a{
38   color: white;
39 }
40
41 .section1 a:hover {
42   color: darkred;
43   text-decoration: underline;
44 }
45
46 .section2 a:hover {
47   color: orange;
48   text-decoration: underline;
49 }
50
51 .section3 a:hover {
52   color: lightgray;
53   text-decoration: underline;
54 }
55
56 a:visited {
57   color: blue
```

Task 1

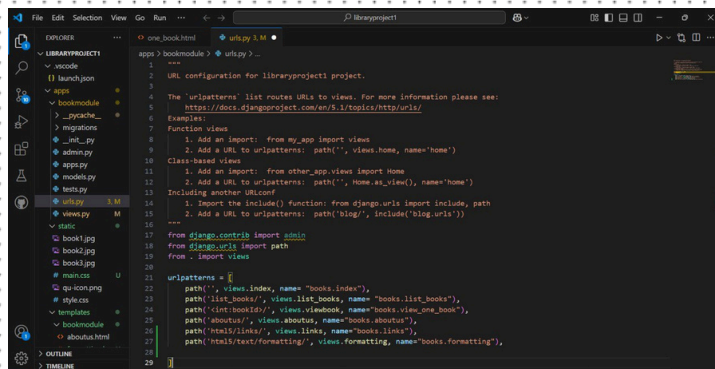
Output



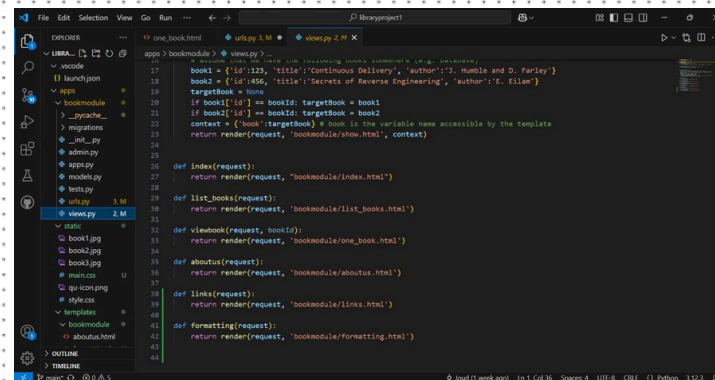
Task 1 Output



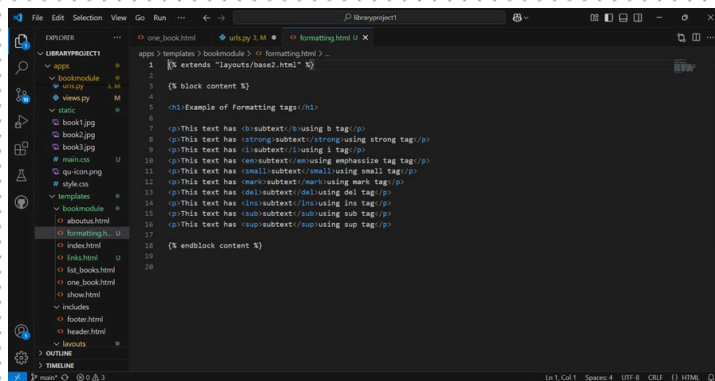
Task 2 Code



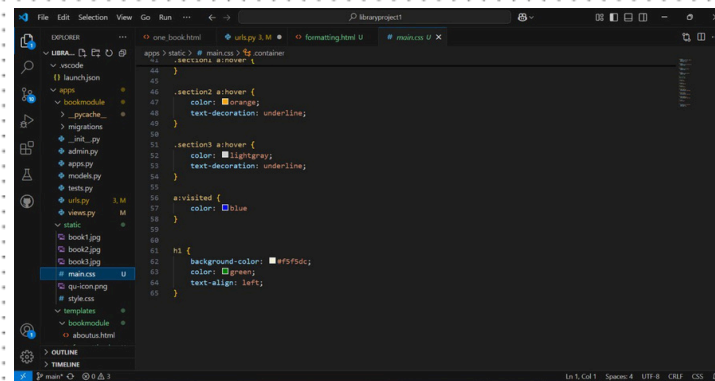
```
1 # -*- coding: utf-8 -*-
2 """
3 URL configuration for libraryproject1 project.
4
5 The 'urlpatterns' list routes URLs to views. For more information please see:
6 https://docs.djangoproject.com/en/5.1/topics/http/urls/
7 Examples:
8 1. Add an import: from my_app import views
9 2. Add a URL to urlpatterns: path('', views.home, name='home')
10 Class-based views:
11 1. Add an import: from other_app.views import home
12 2. Add a URL to urlpatterns: path('', home.as_view(), name='home')
13 Including another urlpatterns:
14 1. Import the include() function: from django.urls import include, path
15 2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
16 """
17 from django.contrib import admin
18 from django.urls import path
19 from . import views
20
21 urlpatterns = [
22     path('', views.index, name='books.index'),
23     path('list_books/', views.list_books, name='books.list_books'),
24     path('list_book/{id}', views.view_book, name='books.view_one_book'),
25     path('aboutus/', views.aboutus, name='books.aboutus'),
26     path('html/links/', views.links, name='books.links'),
27     path('html/text/formatting/', views.formatting, name='books.formatting'),
28 ]
```



```
1 # -*- coding: utf-8 -*-
2 """
3 Views for the libraryproject1 project.
4 """
5 from django.shortcuts import render
6
7 book1 = {'id':123, 'title':'Continuous Delivery', 'author':'J. Humble and D. Farley'}
8 book2 = {'id':456, 'title':'Secrets of Reverse Engineering', 'author':'E. Eilam'}
9
10 targetbook = None
11 if book1['id'] == bookid: targetbook = book1
12 if book2['id'] == bookid: targetbook = book2
13 context = {'book':targetbook} # book is the variable name accessible by the template
14 return render(request, 'bookmodule/show.html', context)
15
16 def index(request):
17     return render(request, 'bookmodule/index.html')
18
19 def list_books(request):
20     return render(request, 'bookmodule/list_books.html')
21
22 def viewbook(request, bookid):
23     return render(request, 'bookmodule/one_book.html')
24
25 def aboutus(request):
26     return render(request, 'bookmodule/aboutus.html')
27
28 def links(request):
29     return render(request, 'bookmodule/links.html')
30
31 def formatting(request):
32     return render(request, 'bookmodule/formatting.html')
```



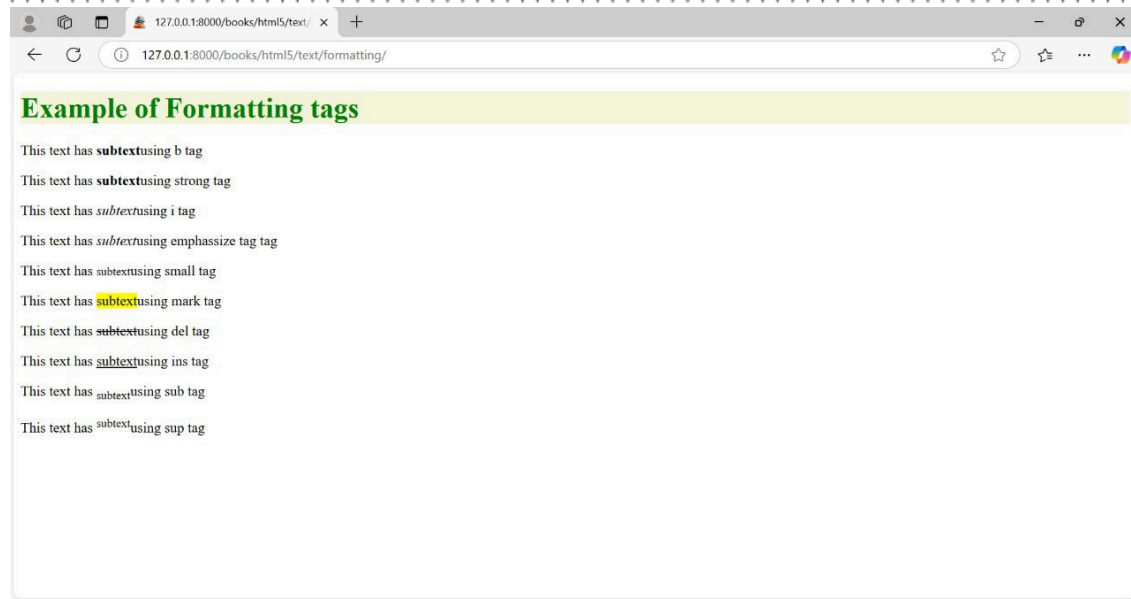
```
1 {% extends "layouts/base2.html" %}
2
3 {% block content %}
4
5 <h1>Example of formatting tags</h1>
6
7 <p>This text has <sub>subtext</sub> using b tag</p>
8 <p>This text has <strong>subtext</strong> using strong tag</p>
9 <p>This text has <i>subtext</i> using i tag</p>
10 <p>This text has <em>subtext</em> using emphasize tag</p>
11 <p>This text has <small>subtext</small> using small tag</p>
12 <p>This text has <mark>subtext</mark> using mark tag</p>
13 <p>This text has <del>subtext</del> using del tag</p>
14 <p>This text has <ins>subtext</ins> using ins tag</p>
15 <p>This text has <sub>subtext</sub> using sub tag</p>
16 <p>This text has <sup>subtext</sup> using sup tag</p>
17
18 {% endblock content %}
```



```
1
2
3 section1 {
4     color: #ff00ff;
5     text-decoration: underline;
6 }
7
8 section2 {
9     color: #00ff00;
10     text-decoration: underline;
11 }
12
13 section3 {
14     color: #0000ff;
15     text-decoration: underline;
16 }
17
18 a {
19     color: #0000ff;
20 }
21
22 h1 {
23     background-color: #ff00ff;
24     color: #00ff00;
25     text-align: left;
26 }
```

Task 2

Output



Task 3

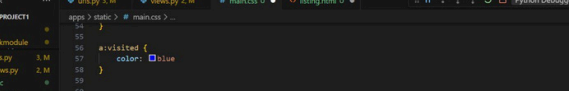
Code

The screenshot shows a VS Code editor with a Django project. The Explorer sidebar on the left displays the project structure, including folders like 'app1', 'bookmodels', 'bookviews', 'maincms', 'quicongpg', 'stylescss', 'templates', 'bookmodule', 'aboutushtml', 'formatting_h_u', 'indexhtml', 'enishtml', 'ita bookshtml', 'isinghtml_u', 'one_bookhtml', 'showhtml', 'includes', 'footer.html', 'OUTLINE', and 'README'. The main editor window shows the file 'app1/urls.py' with the following content:

```

1 from django.urls import path
2 from django.contrib.admin import admin
3 from django.urls import path
4 from . import views
5
6 urlpatterns = [
7     path("", views.index, name="books.index"),
8     path("list_books/", views.list_books, name="books.list_books"),
9     path("<int:book_id>/", views.view_book, name="books.view_one_book"),
10    path("enctus/", views.about_us, name="books.about_us"),
11    path("htmls/links/", views.links, name="books.links"),
12    path("htmls/text/formatting/", views.formatting, name="books.formatting"),
13    path("books/listing/", views.listing, name="books.listing"),
14 ]

```

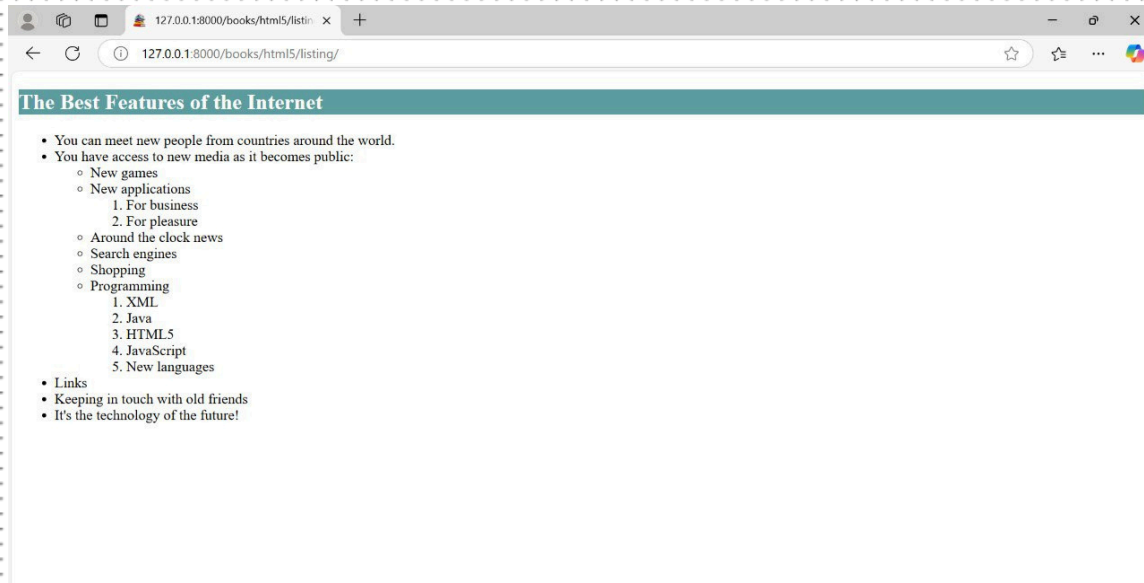
[illegible][illegible]

The screenshot shows the Visual Studio Code interface with the following components:

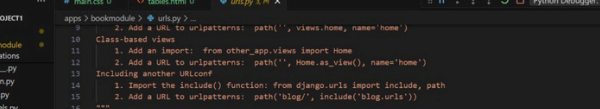
- Explorer Pane:** Displays a file tree for 'LIBRARYPROJECT1'. The 'static' directory is expanded, showing files like 'book1.jpg', 'book2.jpg', 'book3.jpg', 'main.css' (selected), 'qi-icon.png', 'style.css', and 'templates'.
- Run and Debug Pane:** Shows the 'static' directory, indicating the current working directory for the debugger.
- Editor Pane:** Displays the content of 'main.css'. The code defines a 'body' class with a 'background-color' of '#f5f5dc', 'color' of 'green', and 'text-align' of 'left'. It also defines a 'h1' class with a 'background-color' of '#f5f5dc', 'color' of 'green', and 'text-align' of 'left'. The 'h2' class is defined with a 'background-color' of 'cadetblue', 'color' of 'white', and 'text-align' of 'left'.

Task 3

Output



Task 4
Code



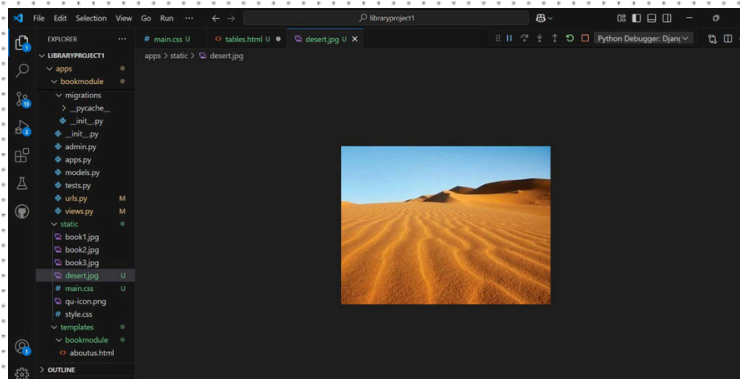
The screenshot shows a code editor with a file explorer on the left and a Python file named `urls.py` open in the main editor. The file explorer shows a Django project structure with folders like `migrations`, `admin.py`, `models.py`, and `views.py`. The main editor displays the contents of `urls.py`, which includes imports for Django's URL dispatcher and Django REST framework, and a list of URL patterns for a books application.

```

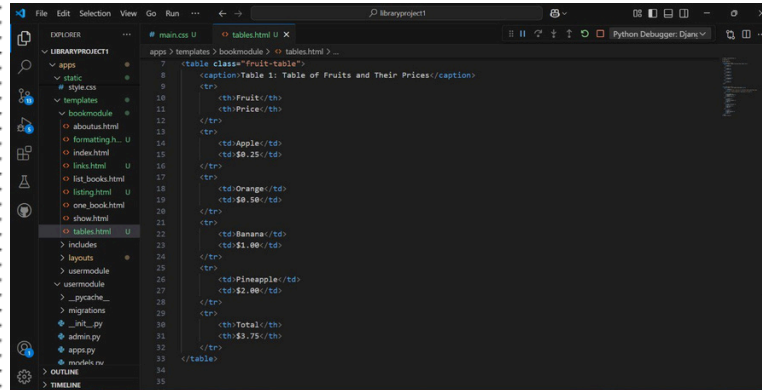
# maincms U
# books.html U
# urls.py 3 M X
# migrations U
# admin.py
# models.py
# tests.py
# urls.py 3 M
# views.py M
# static
# book1.jpg
# book2.jpg
# book3.jpg
# maincms U
# quicon.png
# styles.css
# templates
# bookmodule
# aboutus.html
# formatting_h... U
# index.html
# book.html U
# list-books.html

app = bookmodule
urlpatterns = [
    path('', views.home, name='home')
]

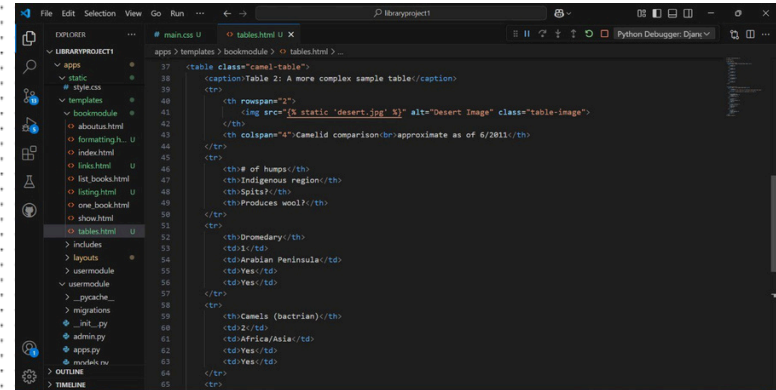
```



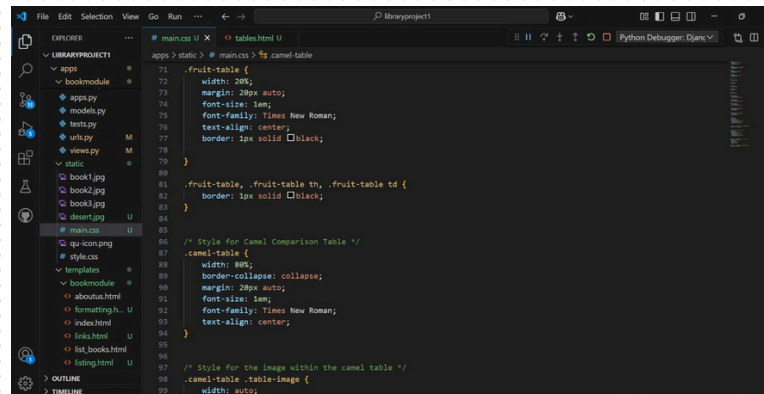
Task 4 Code



```
1 <table class="fruit-table">
2   <caption>Table 1: Table of Fruits and Their Prices/<caption>
3   <tr>
4     <th>Fruit</th>
5     <th>Price</th>
6     <th></th>
7   </tr>
8   <tr>
9     <td>Apple</td>
10    <td>$0.25</td>
11    <td></td>
12  </tr>
13  <tr>
14    <td>Orange</td>
15    <td>$0.50</td>
16    <td></td>
17  </tr>
18  <tr>
19    <td>Banana</td>
20    <td>$1.00</td>
21    <td></td>
22  </tr>
23  <tr>
24    <td>Pineapple</td>
25    <td>$2.00</td>
26    <td></td>
27  </tr>
28  <tr>
29    <th>Total</th>
30    <th>$3.75</th>
31    <th></th>
32  </tr>
33 </table>
34
```



```
1 <table class="camel-table">
2   <caption>Table 2: A more complex sample table/<caption>
3   <tr>
4     <th rowspan="2">Camel</th>
5     <th rowspan="2">Region</th>
6     <th rowspan="2">Spots</th>
7     <th rowspan="2">Produces</th>
8   </tr>
9   <tr>
10    <th>of humps</th>
11    <th>Indigenous region</th>
12    <th>Spots</th>
13    <th>Produces</th>
14  </tr>
15  <tr>
16    <td>Arabian Peninsula</td>
17    <td>Yes</td>
18    <td>1</td>
19    <td>No</td>
20  </tr>
21  <tr>
22    <td>Camels (Bactrian)</td>
23    <td>2</td>
24    <td>2</td>
25    <td>2</td>
26  </tr>
27  <tr>
28    <td>Africa/Asia</td>
29    <td>Yes</td>
30    <td>2</td>
31    <td>2</td>
32  </tr>
33 </table>
34
```



```
1 .fruit-table {
2   width: 200px;
3   margin: 20px auto;
4   font-size: 1em;
5   font-family: Times New Roman;
6   text-align: center;
7   border: 1px solid black;
8 }
9
10 .fruit-table, .fruit-table th, .fruit-table td {
11   border: 1px solid black;
12 }
13
14 /* Style for Camel Comparison Table */
15 .camel-table {
16   width: 80%;
17   border-collapse: collapse;
18   margin: 20px auto;
19   font-size: 1em;
20   font-family: Times New Roman;
21   text-align: center;
22 }
23
24 /* Style for the image within the camel table */
25 .camel-table .table-image {
26   width: auto;
27 }
28
```

Task 4 Output


127.0.0.1:8000/books/html5/tables/

127.0.0.1:8000/books/html5/tables/

Table 1: Table of Fruits and Their Prices

Fruit	Price
Apple	\$0.25
Orange	\$0.50
Banana	\$1.00
Pineapple	\$2.00
Total	\$3.75

Table 2: A more complex sample table

	<div>Camelid comparison approximate as of 6/2011</div>				
	# of humps	Indigenous region	Spits?	Produces wool?	
	Dromedary	1	Arabian Peninsula	Yes	Yes
	Camels (bactrian)	2	Africa/Asia	Yes	Yes
	Llamas	1	Andes Mountains	Yes	Yes