



الجمهورية العربية السورية

جامعة تشرين

كلية الهندسة الميكانيكية والكهربائية

قسم الاتصالات

تقديم

كاتيا عزيز نعمان 2857

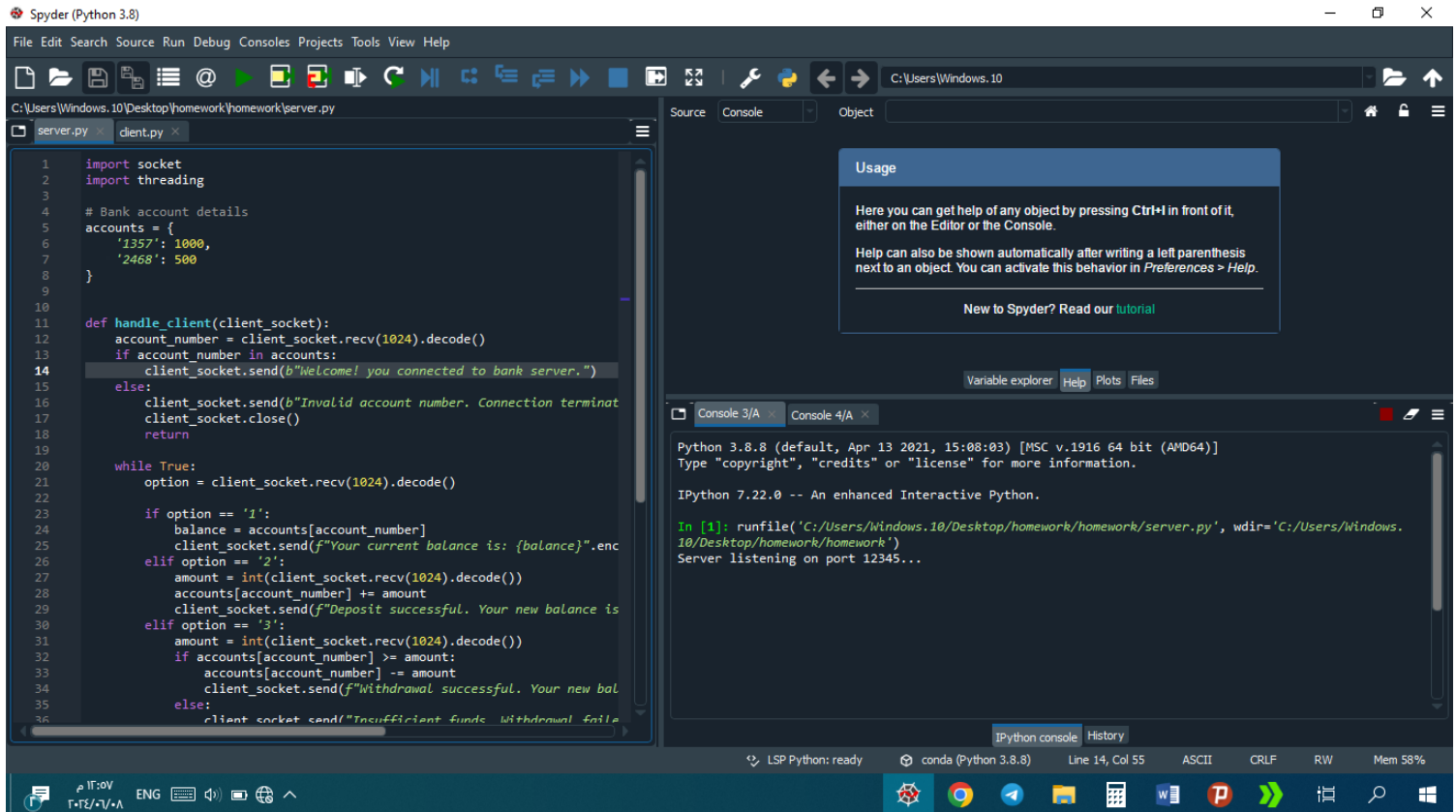
غوى عزيز عبدالله 2913

برمجة شبكات

Homework 2

# Question 1

## server code



The screenshot shows the Spyder Python IDE interface. The left pane displays the `server.py` file with the following code:

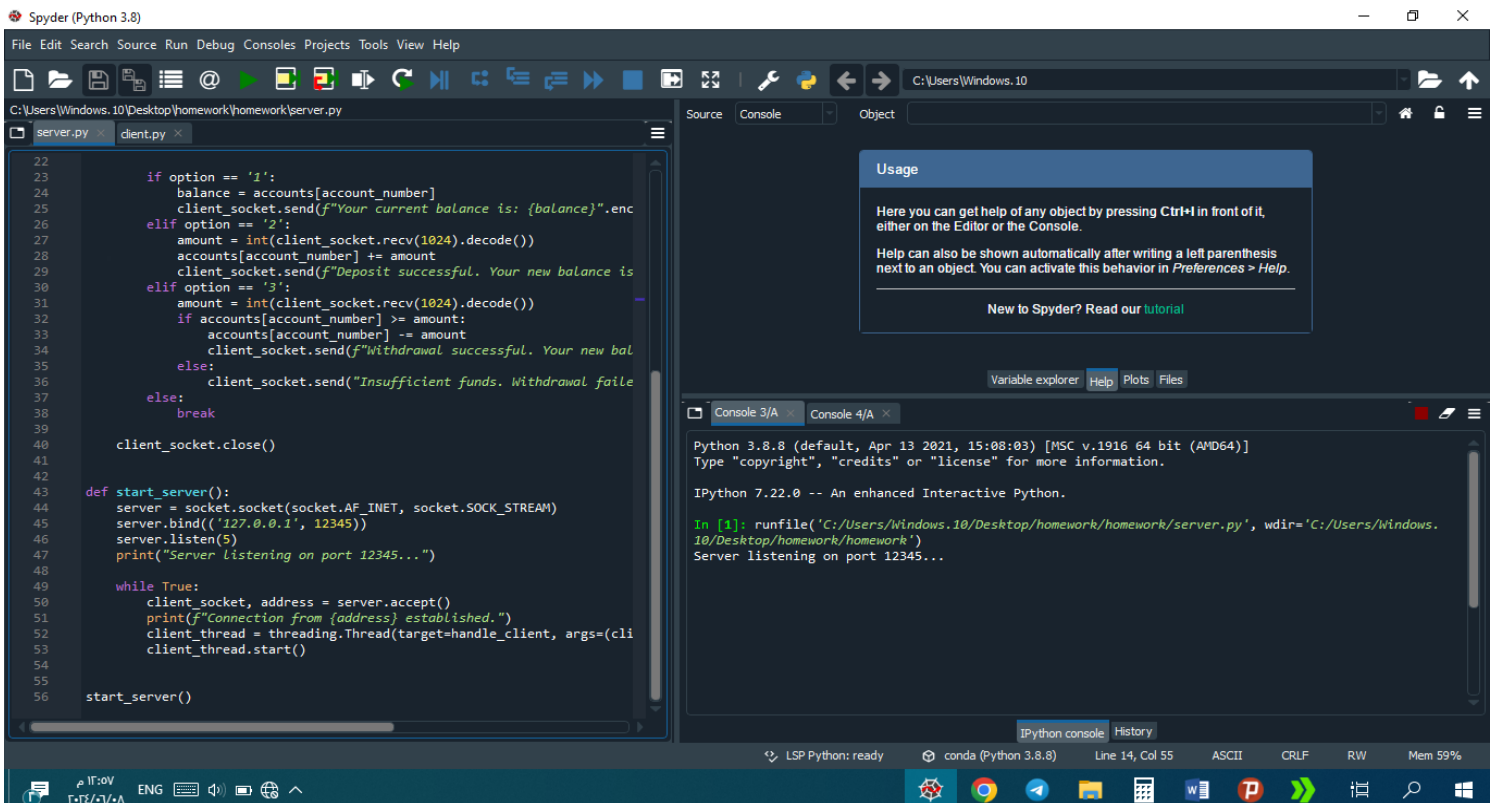
```
1 import socket
2 import threading
3
4 # Bank account details
5 accounts = {
6     '1357': 1000,
7     '2468': 500
8 }
9
10 def handle_client(client_socket):
11     account_number = client_socket.recv(1024).decode()
12     if account_number in accounts:
13         client_socket.send(b"Welcome! you connected to bank server.")
14     else:
15         client_socket.send(b"Invalid account number. Connection terminat")
16         client_socket.close()
17         return
18
19 while True:
20     option = client_socket.recv(1024).decode()
21
22     if option == '1':
23         balance = accounts[account_number]
24         client_socket.send(f"Your current balance is: {balance}".enc
25     elif option == '2':
26         amount = int(client_socket.recv(1024).decode())
27         accounts[account_number] += amount
28         client_socket.send(f"Deposit successful. Your new balance is
29     elif option == '3':
30         amount = int(client_socket.recv(1024).decode())
31         if accounts[account_number] >= amount:
32             accounts[account_number] -= amount
33             client_socket.send(f"Withdrawal successful. Your new bal
34         else:
35             client_socket.send("Insufficient funds. Withdrawal faile
36
```

The right pane shows the IPython console with the following output:

```
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.22.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/Windows.10/Desktop/homework/homework/server.py', wdir='C:/Users/Windows.10/Desktop/homework/homework')
Server listening on port 12345...
```



The screenshot shows the Spyder Python IDE interface. The left pane displays the `server.py` file with the following code:

```
22
23     if option == '1':
24         balance = accounts[account_number]
25         client_socket.send(f"Your current balance is: {balance}".enc
26     elif option == '2':
27         amount = int(client_socket.recv(1024).decode())
28         accounts[account_number] += amount
29         client_socket.send(f"Deposit successful. Your new balance is
30     elif option == '3':
31         amount = int(client_socket.recv(1024).decode())
32         if accounts[account_number] >= amount:
33             accounts[account_number] -= amount
34             client_socket.send(f"Withdrawal successful. Your new bal
35         else:
36             client_socket.send("Insufficient funds. Withdrawal faile
37     else:
38         break
39
40     client_socket.close()
41
42
43 def start_server():
44     server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
45     server.bind(('127.0.0.1', 12345))
46     server.listen(5)
47     print("Server listening on port 12345...")
48
49 while True:
50     client_socket, address = server.accept()
51     print(f"Connection from {address} established.")
52     client_thread = threading.Thread(target=handle_client, args=(cli
53     client_thread.start()
54
55
56 start_server()
```

The right pane shows the IPython console with the following output:

```
Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.22.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/Windows.10/Desktop/homework/homework/server.py', wdir='C:/Users/Windows.10/Desktop/homework/homework')
Server listening on port 12345...
```

# client code

The screenshot shows the Spyder Python IDE with the file `client.py` open. The code is a client for a bank server, using `socket` to connect to `127.0.0.1` on port `12345`. It prompts the user for an account number and then a menu of options: 1. Check Balance, 2. Deposit, 3. Withdraw, 4. Exit. The console output shows the program running successfully, with the user entering account number 2468 and selecting option 2 (Deposit).

```
1 import socket
2
3
4 def main():
5     client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
6     client.connect(('127.0.0.1', 12345))
7
8     account_number = input("Enter your account number: ")
9     client.send(account_number.encode())
10
11     print(client.recv(1024).decode())
12
13     while True:
14         print("\nOptions:")
15         print("1. Check Balance")
16         print("2. Deposit")
17         print("3. Withdraw")
18         print("4. Exit")
19         option = input("Enter option: ")
20
21         if option == '4':
22             break
23
24         client.send(option.encode())
25
26         if option == '1':
27             print(client.recv(1024).decode())
28         elif option == '2' or option == '3':
29             amount = input("Enter amount: ")
30             client.send(amount.encode())
31             print(client.recv(1024).decode())
32
33     client.close()
34
35 if __name__ == "__main__":
36     main()
```

Console 3/A x Console 4/A x

```
Python 3.8.8 (default, Apr 12 2021, 13:00:03) [MSC v.1916 64-bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.

IPython 7.22.0 -- An enhanced Interactive Python.

In [1]: runfile('C:/Users/Windows.10/Desktop/homework/homework/client.py', wdir='C:/Users/Windows.10/Desktop/homework/homework')

Enter your account number: 2468
Welcome! you connected to bank server.

Options:
1. Check Balance
2. Deposit
3. Withdraw
4. Exit

Enter option: |
```

The screenshot shows the same Spyder Python IDE with the `client.py` file. The console output now shows the user entering option 2, then option 3, performing a deposit and a withdrawal respectively. The console output shows the updated balance after each transaction.

```
1 import socket
2
3
4 def main():
5     client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
6     client.connect(('127.0.0.1', 12345))
7
8     account_number = input("Enter your account number: ")
9     client.send(account_number.encode())
10
11     print(client.recv(1024).decode())
12
13     while True:
14         print("\nOptions:")
15         print("1. Check Balance")
16         print("2. Deposit")
17         print("3. Withdraw")
18         print("4. Exit")
19         option = input("Enter option: ")
20
21         if option == '4':
22             break
23
24         client.send(option.encode())
25
26         if option == '1':
27             print(client.recv(1024).decode())
28         elif option == '2' or option == '3':
29             amount = input("Enter amount: ")
30             client.send(amount.encode())
31             print(client.recv(1024).decode())
32
33     client.close()
34
35 if __name__ == "__main__":
36     main()
```

Console 3/A x Console 4/A x

```
Enter option: 2

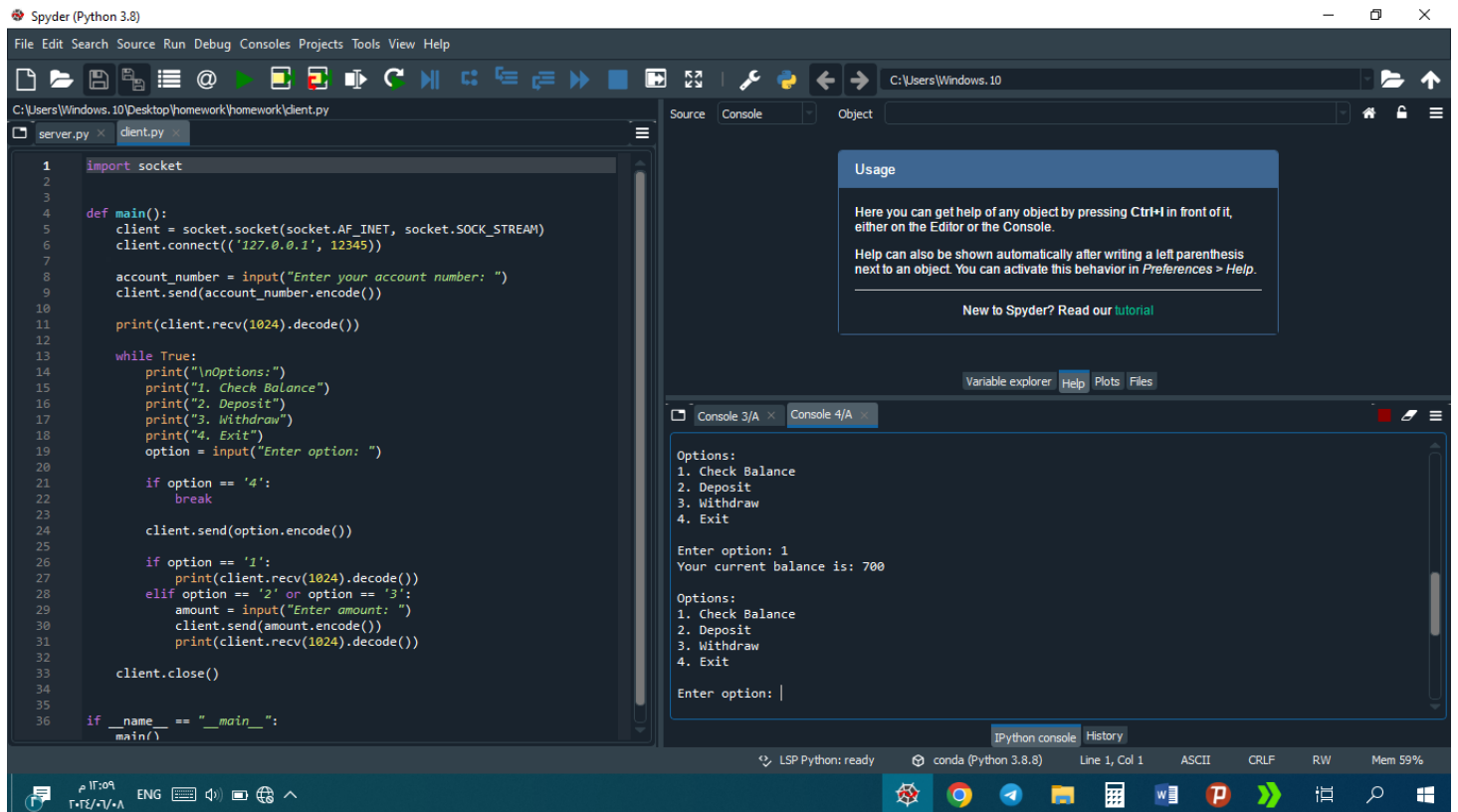
Enter amount: 500
Deposit successful. Your new balance is: 1000

Options:
1. Check Balance
2. Deposit
3. Withdraw
4. Exit

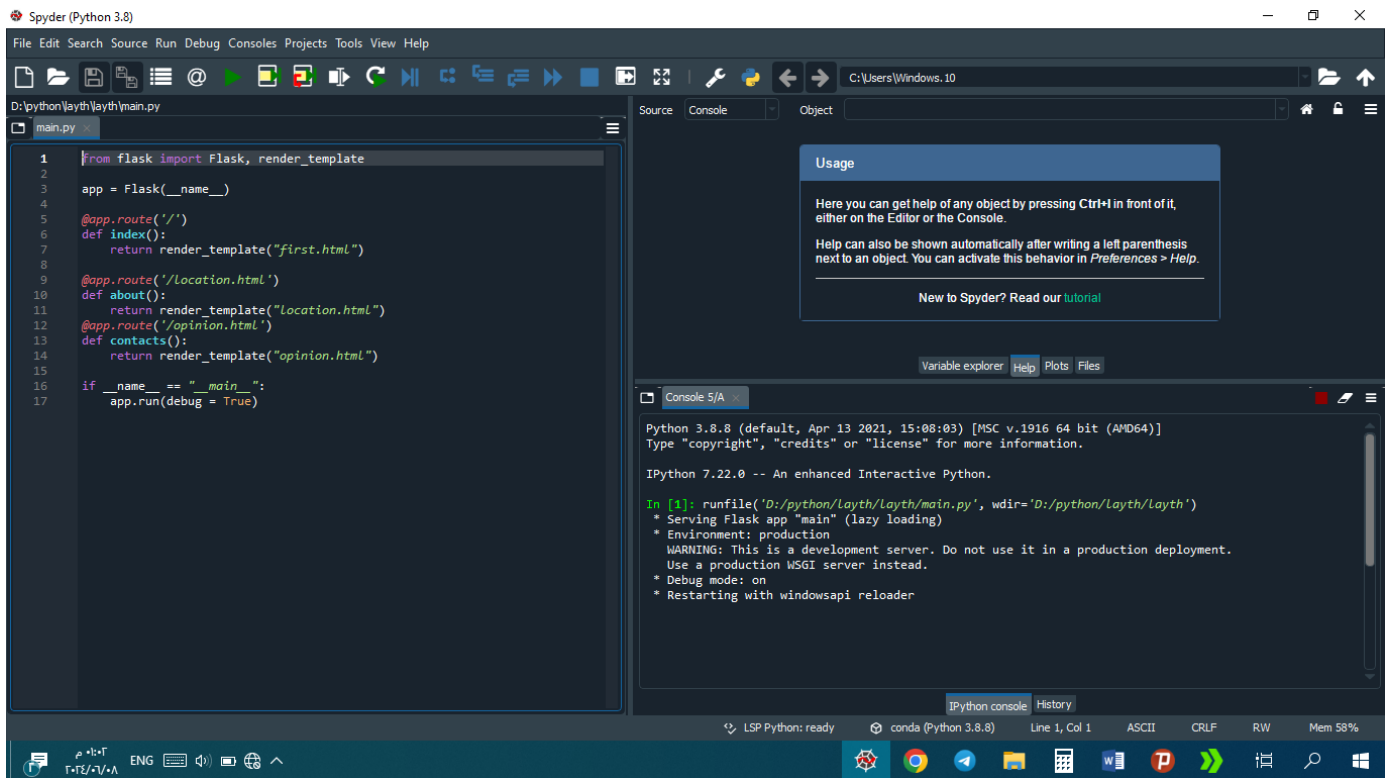
Enter option: 3

Enter amount: 300
Withdrawal successful. Your new balance is: 700

Options:
```



## Question 2



The screenshot shows the Spyder Python IDE interface. The main editor displays a Python script for a Flask web application. The code defines a Flask app with four routes: a root route returning 'first.html', a '/Location.html' route returning 'Location.html', a '/opinion.html' route returning 'opinion.html', and a '/contacts.html' route returning 'opinion.html'. The script is saved as 'main.py' in the directory 'D:\python\layth\layth\main.py'.

```
1 from flask import Flask, render_template
2
3 app = Flask(__name__)
4
5 @app.route('/')
6 def index():
7     return render_template("first.html")
8
9 @app.route('/Location.html')
10 def about():
11     return render_template("Location.html")
12 @app.route('/opinion.html')
13 def contacts():
14     return render_template("opinion.html")
15
16 if __name__ == "__main__":
17     app.run(debug = True)
```

The console output shows the IPython shell running the script. It displays the Flask version (3.8.8), the Python version (3.8.8), and the IPython version (7.22.0). It also shows the command to run the script and the output of the Flask application, including a warning about the development server and the debug mode being on.

Usage

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our [tutorial](#)

Variable explorer | Help | Plots | Files

Console 5/A

Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)]  
Type "copyright", "credits" or "license()" for more information.

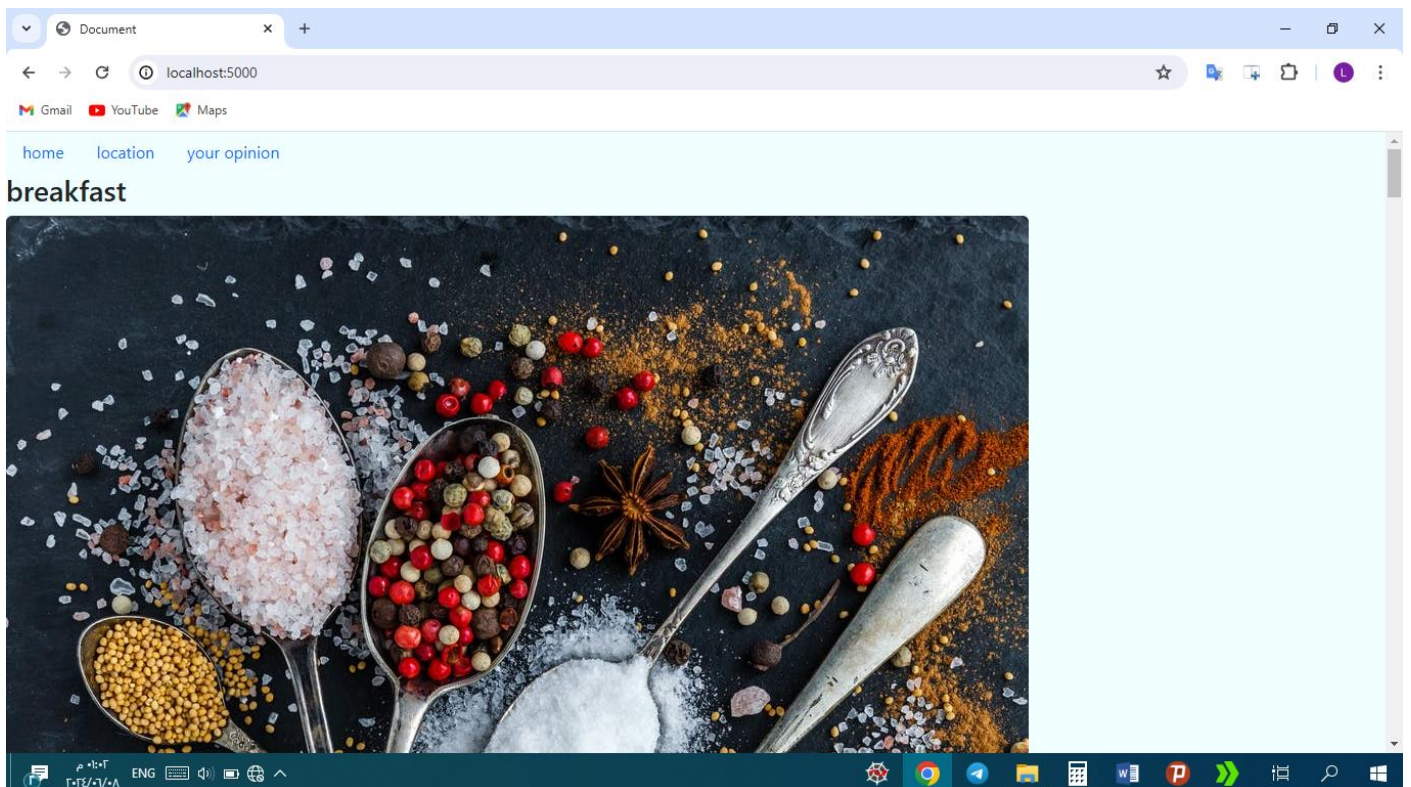
IPython 7.22.0 -- An enhanced Interactive Python.

In [1]: runfile('D:/python/Layth/Layth/main.py', wdir='D:/python/Layth/Layth')

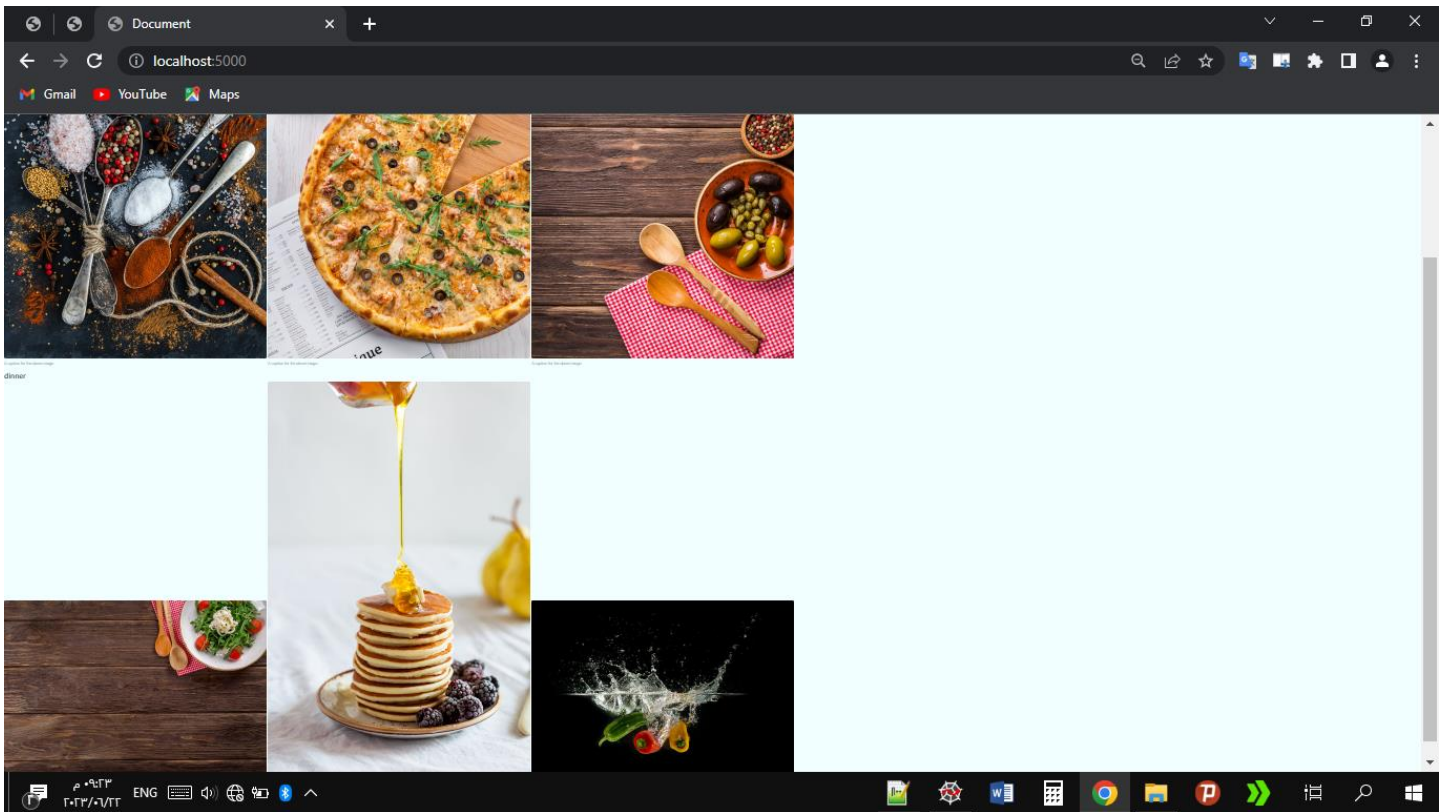
- \* Serving Flask app "main" (lazy loading)
- \* Environment: production
- WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
- \* Debug mode: on
- \* Restarting with windowsapi reloader

LSP Python: ready | conda (Python 3.8.8) | Line 1, Col 1 | ASCII | CRLF | RW | Mem 58%

عند تشغيل الكود والدخول الى المتصفح والدخول الى localhost:5000







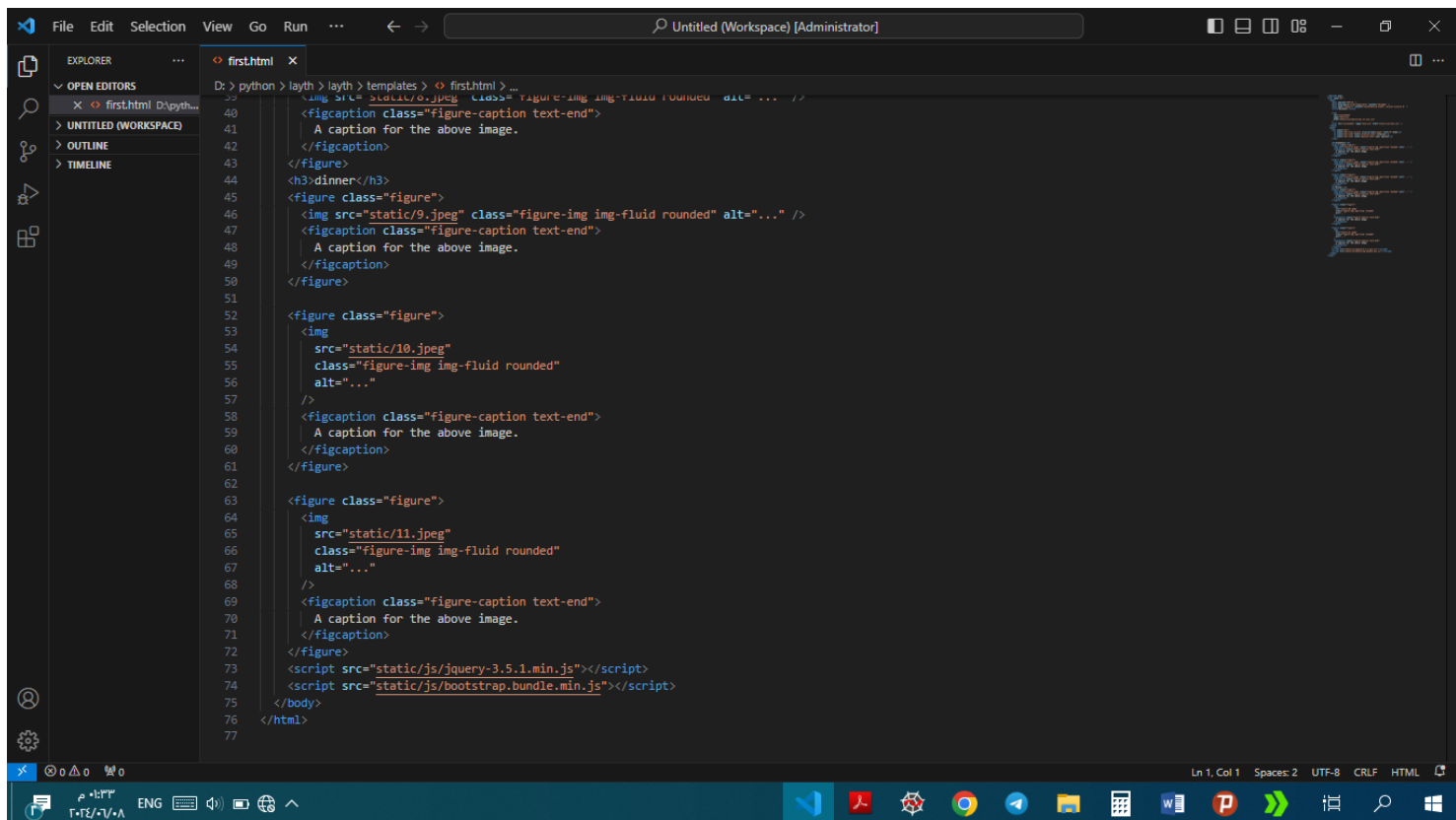
كود html الذي يظهر الصفحة الرئيسية للموقع:

```

File Edit Selection View Go Run ... Untitled (Workspace) [Administrator]
EXPLORER first.html
OPEN EDITORS
first.html D:\python...
UNTITLED (WORKSPACE)
OUTLINE
TIMELINE

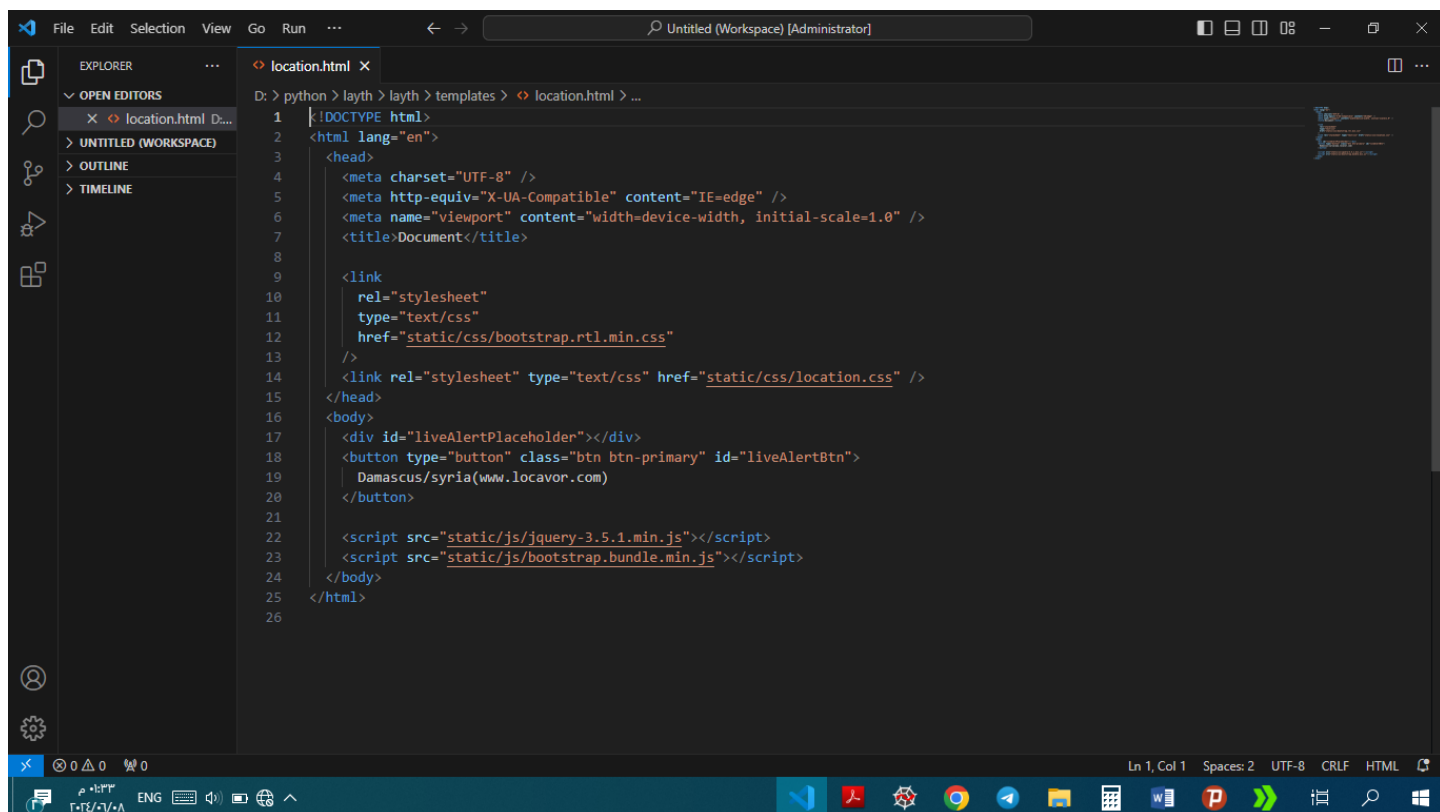
D:\> python > layth > layth > templates > > first.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8" />
5 <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6 <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7 <title>Document</title>
8
9
10 <link
11   rel="stylesheet"
12   type="text/css"
13   href="static/css/bootstrap.rtl.min.css"
14 />
15 <link rel="stylesheet" type="text/css" href="static/css/one.css" />
16 </head>
17 <body>
18 <nav class="nav">
19   <a class="nav-link active" aria-current="page" href="#">home</a>
20   <a class="nav-link" href="location.html">location</a>
21   <a class="nav-link" href="opinion.html">your opinion</a>
22 </nav>
23
24 <h3>breakfast</h3>
25 <figure class="figure">
26   
27   <figcaption class="figure-caption text-end">
28     A caption for the above image.
29   </figcaption>
30 </figure>
31
32 <figure class="figure">
33   
34   <figcaption class="figure-caption text-end">
35     A caption for the above image.
36   </figcaption>
37 </figure>
38
39 <figure class="figure">
40   
41   <figcaption class="figure-caption text-end">

```



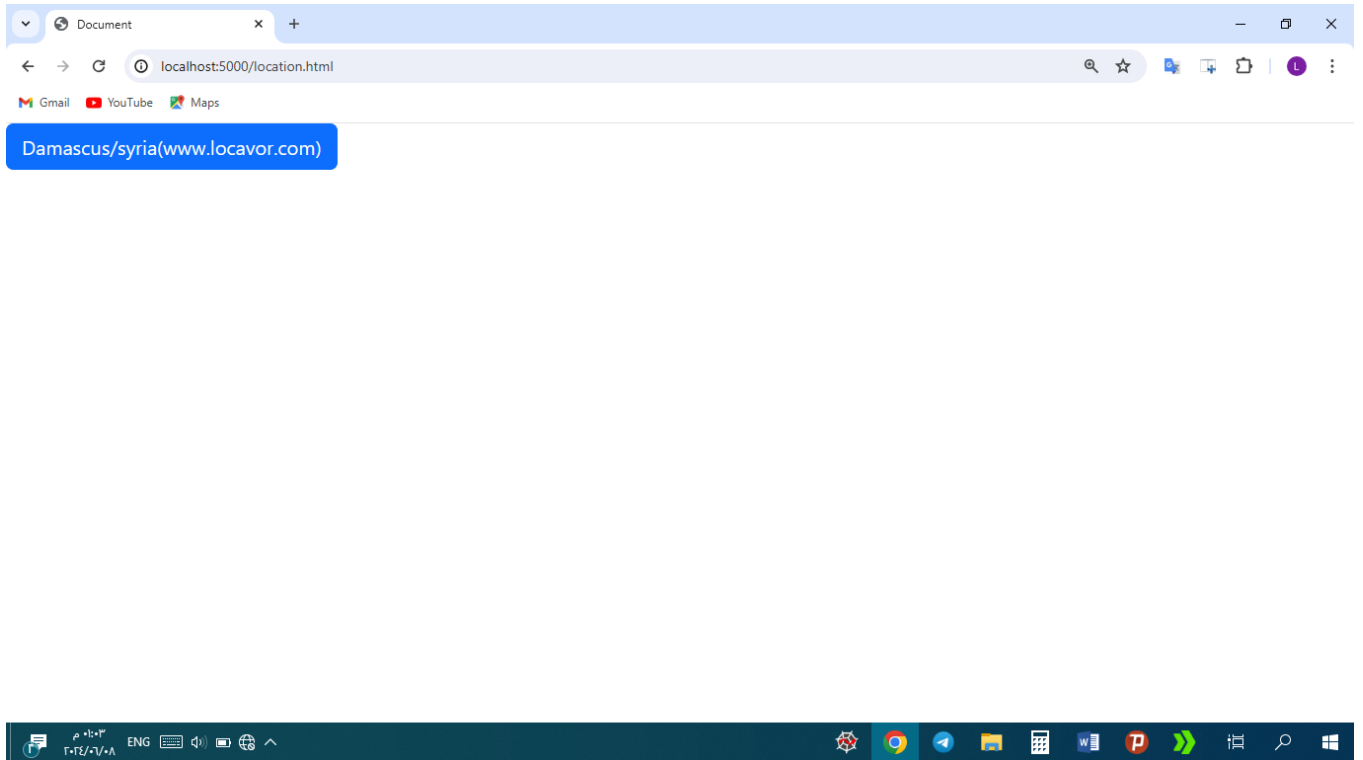
```
40 
41 <figcaption class="figure-caption text-end">
42   A caption for the above image.
43 </figcaption>
44 </figure>
45 <h3>dinner</h3>
46 <figure class="figure">
47   
48   <figcaption class="figure-caption text-end">
49     A caption for the above image.
50   </figcaption>
51 </figure>
52 <figure class="figure">
53   
58   <figcaption class="figure-caption text-end">
59     A caption for the above image.
60   </figcaption>
61 </figure>
62 <figure class="figure">
63   
68   <figcaption class="figure-caption text-end">
69     A caption for the above image.
70   </figcaption>
71 </figure>
72 </div>
73 <script src="static/js/jquery-3.5.1.min.js"></script>
74 <script src="static/js/bootstrap.bundle.min.js"></script>
75 </body>
76 </html>
77
```

كود html الذي يظهر صفحة الموقع:

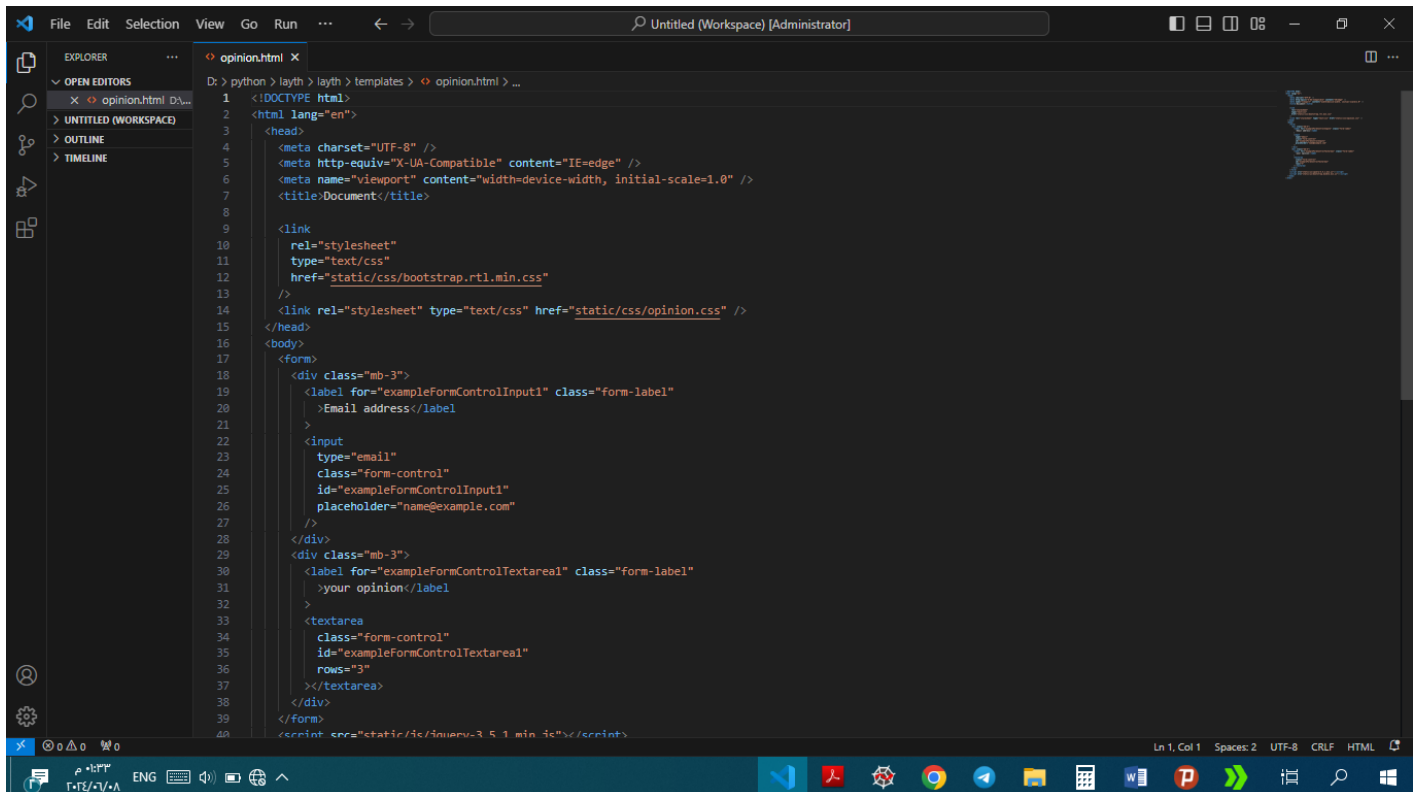


```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta http-equiv="X-UA-Compatible" content="IE=edge" />
6   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7   <title>Document</title>
8
9   <link
10     rel="stylesheet"
11     type="text/css"
12     href="static/css/bootstrap.rtl.min.css"
13   />
14   <link rel="stylesheet" type="text/css" href="static/css/location.css" />
15 </head>
16 <body>
17   <div id="liveAlertPlaceholder"></div>
18   <button type="button" class="btn btn-primary" id="liveAlertBtn">
19     Damascus/syria(www.locavor.com)
20   </button>
21
22   <script src="static/js/jquery-3.5.1.min.js"></script>
23   <script src="static/js/bootstrap.bundle.min.js"></script>
24 </body>
25 </html>
26
```

# الخرج على صفحة الويب:



# كود html الذي يظهر صفحة الرأي الشخصي للمستخدم





```
File Edit Selection View Go Run ...
opinion.html X
D:\> python > layth > templates > opinion.html > ...
<meta http-equiv="A-UN-Compatibility" content="ic=edge" //
6 <meta name="viewport" content="width=device-width, initial-scale=1.0" />
7 <title>Document</title>
8
9
10 <link
11   rel="stylesheet"
12   type="text/css"
13   href="static/css/bootstrap.rtl.min.css"
14 />
15 <link rel="stylesheet" type="text/css" href="static/css/opinion.css" />
16 </head>
17 <body>
18 <form>
19 <div class="mb-3">
20 <label for="exampleFormControlInput1" class="form-label">
21   Email address</label>
22 >
23 <input
24   type="email"
25   class="form-control"
26   id="exampleFormControlInput1"
27   placeholder="name@example.com"
28 />
29 </div>
30 <div class="mb-3">
31 <label for="exampleFormControlTextarea1" class="form-label">
32   your opinion</label>
33 >
34 <textarea
35   class="form-control"
36   id="exampleFormControlTextarea1"
37   rows="3"
38 />
39 </div>
40 </form>
41 <script src="static/js/jquery-3.5.1.min.js"></script>
42 <script src="static/js/bootstrap.bundle.min.js"></script>
43 </body>
44 </html>
```

الخرج على صفحة الويب:

