



**Faculty of Engineering & Technology
Electrical & Computer Engineering Department
Computer Networks ENCS3320
Project 1 Report**

Prepared by:

Yara khattab 1210520

Miassar shamla 1210519

Taleen bayatneh 1211305

Instructor: Dr. Imad Tartir

Section: 3

Date: 5-5-2024

Contents

Part 1	4
Part 1.1	4
1- Ping	4
2- Tracert.....	4
3- Nslookup.....	4
4- Telnet.....	4
Part 1.2	5
Ping a device in the same network.....	5
ping www.stanford.edu	6
From the ping results, do you think the response you got is from USA?	7
tracert www.stanford.edu	8
nslookup www.stanford.edu	9
Part 1.3	10
Part 2	11
The python code:	11
Server code:.....	11
Peer code :	12
The run:	13
server:.....	13
Peer 1:	14
Peer 2:	15
Part 3	16
The codes:	19
main_en.html:	19
main_ar.html:	21
myformEN.html:.....	24
myformAR.html:.....	24
The html file is connected with a css file:.....	25
Test from the computer:.....	25
Test from the computer:	33

if the request is an .html file then the server should send the requested html file with Content-Type: text/html. You can use any html file. Make it general (not only for specific filename)	36
if the request is a .css file then the server should send the requested css file with Content-Type: text/css. You can use any CSS file. Make it general (not only for specific filename)	37
if the request is a .png then the server should send the png image with Content-Type: image/png. You can use any image. Make it general (not only for specific filename)	38
if the request is a .jpg then the server should send the jpg image with Content-Type: image/jpeg. You can use any image. Make it general (not only for specific filename)	39
Use myform.html to get image by typing the name of the image in a box	40
Use the status code 307 Temporary Redirect to redirect the following.....	43
If the request is wrong or the file doesn't exist the server should return a simple HTML webpage that contains (Content-Type: text/html).....	45
Test from another phone:	46

Part 1

Part 1.1

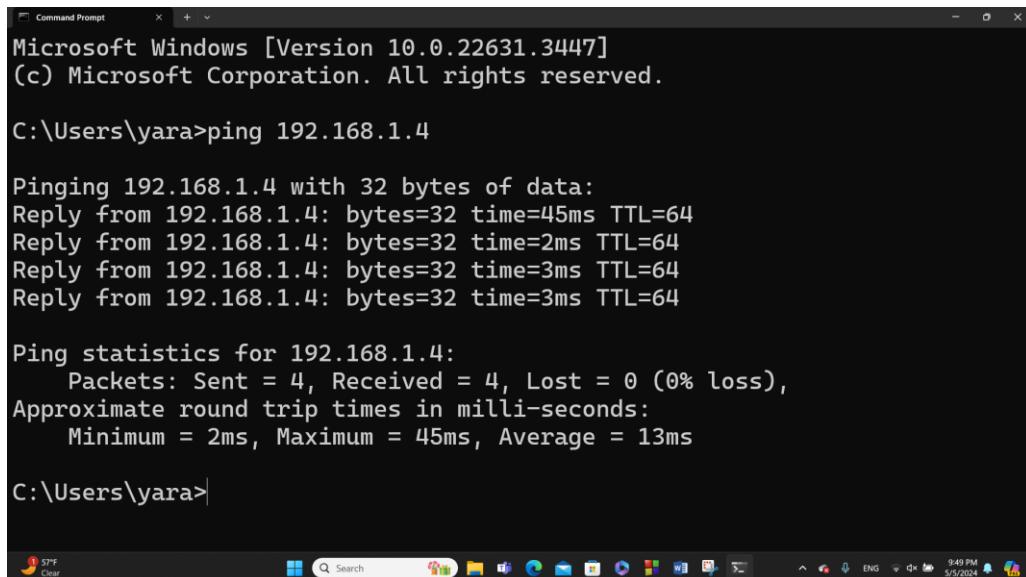
In your own words, what are ping, tracert, nslookup, and telnet (write one sentence for each one).

- 1- Ping:** a tool that sends small data packet to a specific IP address, and measure the round-trip time (RTT) it takes for the packet to reach the destination and come back.
- 2- Tracert:** command used to trace the network path and measure latency at each hop (routers and intermediate devices in the path).
- 3- Nslookup:** is a program that allows the user to enter the hostname and find the IP address or DNS, also by using a specific command can do reverse DNS lookup (find the hostname of the IP address).
- 4- Telnet:** a protocol and command-line tool that enables the user to establish a connection to a distant server or device.

Part 1.2

Make sure that your computer is connected to the internet and then run the following commands:

Ping a device in the same network, e.g., from a laptop to a smartphone



The screenshot shows a Windows Command Prompt window titled "Command Prompt". The title bar also displays "Microsoft Windows [Version 10.0.22631.3447]" and "(c) Microsoft Corporation. All rights reserved.". The window contains the following text:

```
Microsoft Windows [Version 10.0.22631.3447]
(c) Microsoft Corporation. All rights reserved.

C:\Users\yara>ping 192.168.1.4

Pinging 192.168.1.4 with 32 bytes of data:
Reply from 192.168.1.4: bytes=32 time=45ms TTL=64
Reply from 192.168.1.4: bytes=32 time=2ms TTL=64
Reply from 192.168.1.4: bytes=32 time=3ms TTL=64
Reply from 192.168.1.4: bytes=32 time=3ms TTL=64

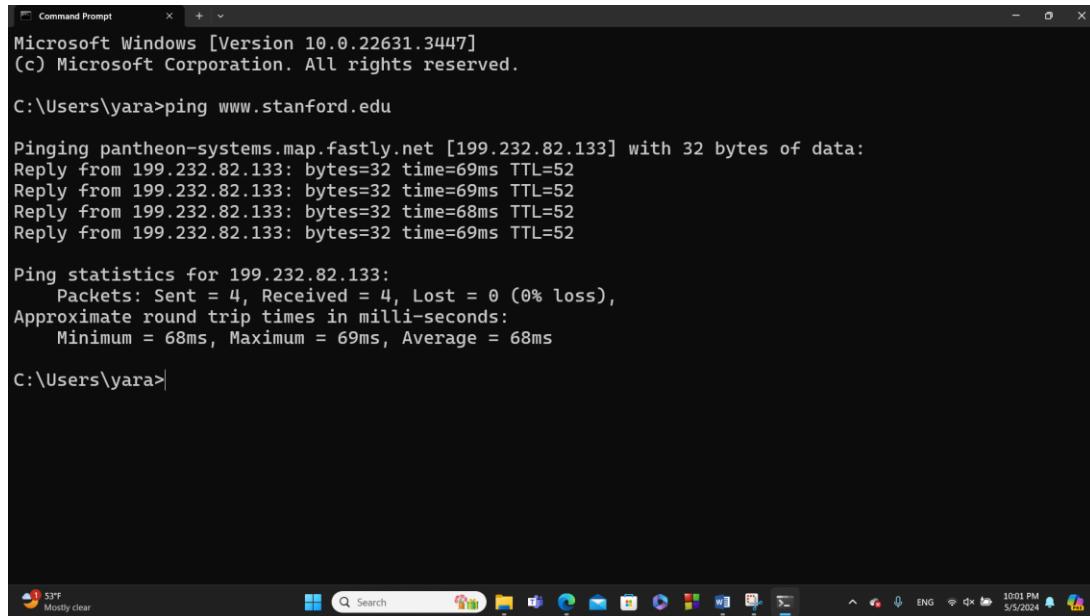
Ping statistics for 192.168.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 45ms, Average = 13ms

C:\Users\yara>
```

The taskbar at the bottom of the screen shows various pinned icons and the system tray indicating the date and time as 5/5/2024 at 9:49 PM.

From the output in the picture, we see that there are four requests sent to the IP address 192.168.1.4, we see the response to each of these requests, including the time taken for the device to respond in milliseconds. It also shows the TTL (Time To live) value which refers to the number of hops that a packet is allowed to pass through before it is discarded. The last section shows a summary of the ping results, it shows that all the packets were sent and received successfully. Some statistics on the round-trip time(RTT) of the packets are also shown, it is shown that four packets response were received from the IP address 192.168.1.4 with an average of 13ms.

ping www.stanford.edu



```
Microsoft Windows [Version 10.0.22631.3447]
(c) Microsoft Corporation. All rights reserved.

C:\Users\yara>ping www.stanford.edu

Pinging pantheon-systems.map.fastly.net [199.232.82.133] with 32 bytes of data:
Reply from 199.232.82.133: bytes=32 time=69ms TTL=52
Reply from 199.232.82.133: bytes=32 time=69ms TTL=52
Reply from 199.232.82.133: bytes=32 time=68ms TTL=52
Reply from 199.232.82.133: bytes=32 time=69ms TTL=52

Ping statistics for 199.232.82.133:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 68ms, Maximum = 69ms, Average = 68ms

C:\Users\yara>
```

This is sent to check if the device is existed or not, so the reply message is to inform that the device exists and connects to the network. Four packets response were sent by 199.232.82.133 with their TTL with an average of 68ms delay, the four packets were sent and received successfully without any loss.

TTL: time to live (numbers of hops the packet passes along the path)

From the ping results, do you think the response you got is from USA? Explain your answer briefly.

The screenshot shows a web browser window with the URL whatismyipaddress.com/ip/199.232.82.133. The page displays the following IP details:

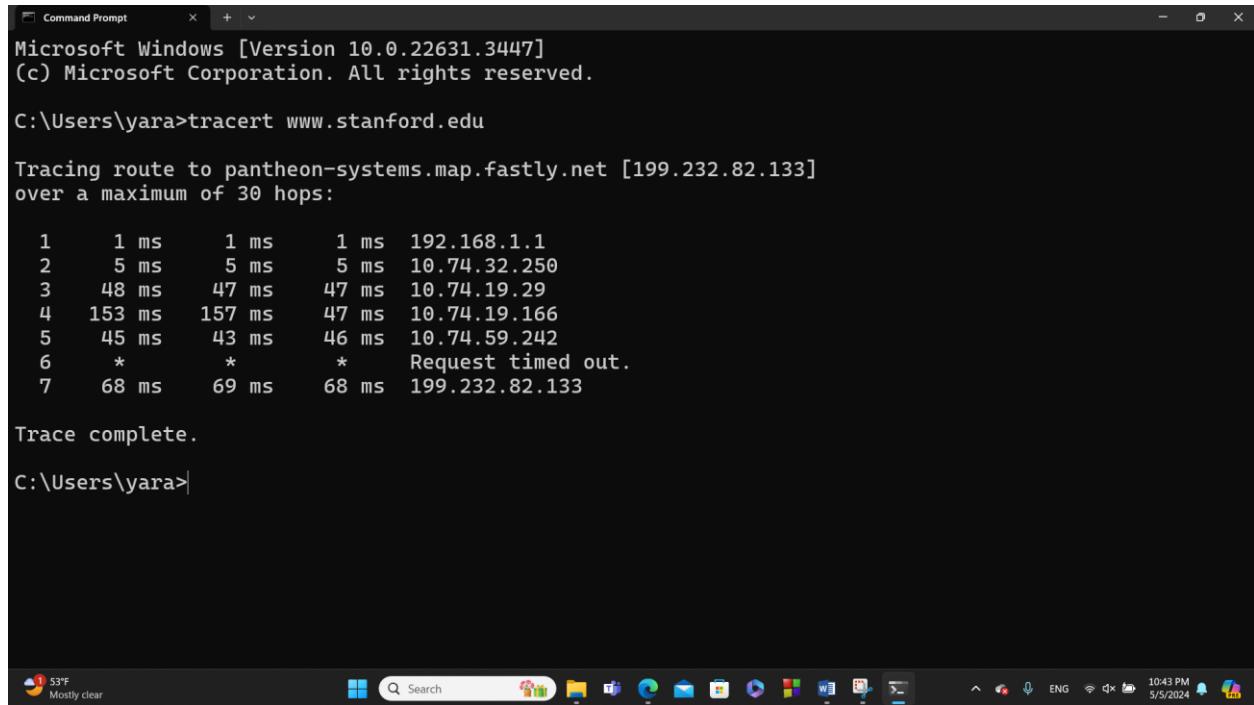
Detail	Value
Decimal	3353891461
Hostname	199.232.82.133
ASN	54113
ISP	Fastly Inc.
Services	Datacenter
Assignment	Unlikely Static IP
Country	United States
State/Region	California
City	San Francisco
Latitude	37.7757 (37° 46' 32.52" N)
Longitude	-122.3952 (122° 23' 42.73" W)

A map of California is shown with a red dot indicating the location of the IP address. Below the map is a note: "Latitude and Longitude are often near the center of population. These values are not precise enough to be used to identify a specific address, individual, or for legal purposes. IP data from [ip2location](#)". A red button at the bottom right of the map area says "CLICK TO CHECK BLACKLIST STATUS".

The browser's taskbar at the bottom shows various pinned icons and the system tray on the right indicates the date and time as 5/5/2024 at 10:53 PM, with a temperature of 12°C.

Yes, it appears that the responses are from USA. The IP address 199.232.82.133, which is associated with the domain "www.stanford.edu" in my ping command, is owned by (Fastly inc). Fastly's content delivery network (CDN) servers, is distributed globally. The IP address 199.232.82.133 is one of Fastly inc servers, and the low round-trip times (ranging from 68ms to 69ms) in the ping responses suggest a relatively low latency, which is typical for servers located in the United State.

tracert www.stanford.edu



```
Microsoft Windows [Version 10.0.22631.3447]
(c) Microsoft Corporation. All rights reserved.

C:\Users\yara>tracert www.stanford.edu

Tracing route to pantheon-systems.map.fastly.net [199.232.82.133]
over a maximum of 30 hops:

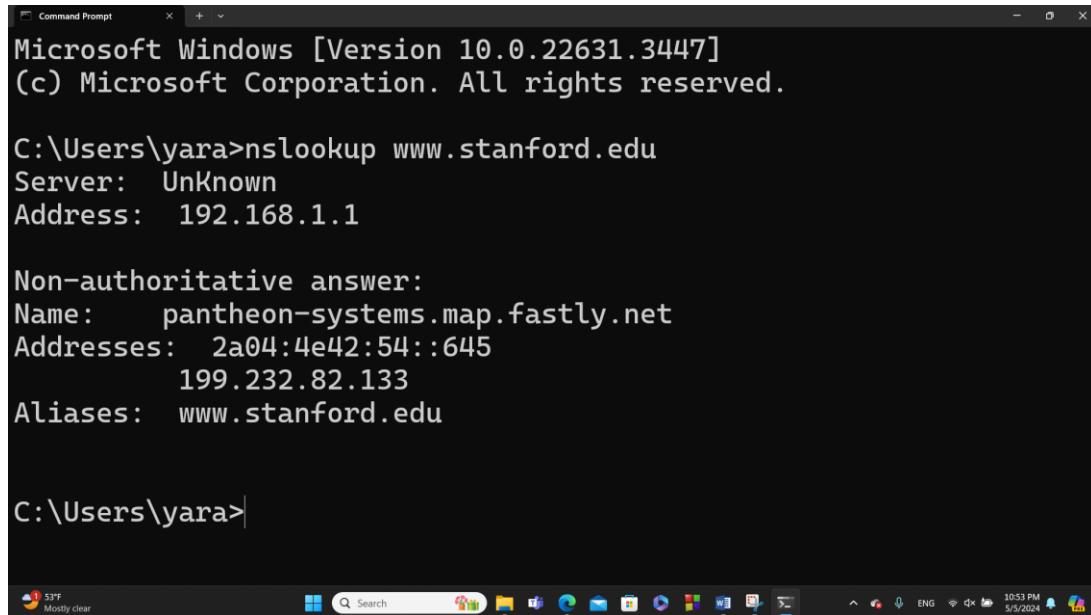
 1   1 ms    1 ms    1 ms  192.168.1.1
 2   5 ms    5 ms    5 ms  10.74.32.250
 3   48 ms   47 ms   47 ms  10.74.19.29
 4   153 ms  157 ms  47 ms  10.74.19.166
 5   45 ms   43 ms   46 ms  10.74.59.242
 6   *        *        *      Request timed out.
 7   68 ms   69 ms   68 ms  199.232.82.133

Trace complete.

C:\Users\yara>
```

Tracert shows the IP address of the destination (192.232.82.133) and lists all hops along the path, the first one is 192.168.1.1 then 10.74.32.250 then 10.74.19.29 then 10.74.19.166 then 10.74.59.422 then we have request timed out with measurements are *, then the packet is prevented from reaching the destination because there is a problem at the location or the router is incorrect , and the last IP address is the host.
When we go down in the lines , we will see the 3 measurements increases because the router go further.

nslookup www.stanford.edu



```
Microsoft Windows [Version 10.0.22631.3447]
(c) Microsoft Corporation. All rights reserved.

C:\Users\yara>nslookup www.stanford.edu
Server: Unknown
Address: 192.168.1.1

Non-authoritative answer:
Name: pantheon-systems.map.fastly.net
Addresses: 2a04:4e42:54::645
          199.232.82.133
Aliases: www.stanford.edu

C:\Users\yara>
```

Nslookup lets users enter a host name and find out the corresponding IP address or domain name system (DNS) record.

So, the output shows the DNS server used for the query is specified as "Unknown" with the IP address 192.168.1.1, label "Non-authoritative," means that the DNS server providing the information is not the primary source for the domain but has cached the information, while name was out as: pantheon-systems.map.fastly.net

And the corresponding addresses for this domain are also provided:

- Addresses: 2a04:4e42:54::645
- Addresses: 199.232.82.133

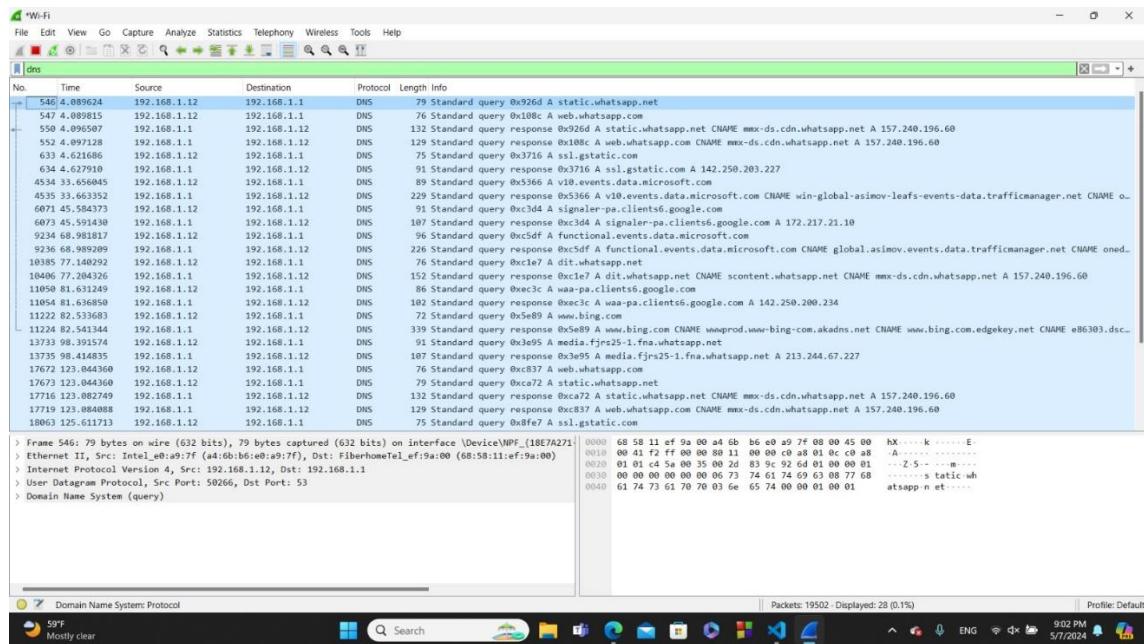
Aliases : Additional names associated with the resolved IP addresses. In this case, "www.stanford.edu" is an alias for the resolved IP addresses

Part 1.3

use wireshark to capture some DNS messages.

```
Command Prompt
Microsoft Windows [Version 10.0.22631.3997]
(C) Microsoft Corporation. All rights reserved.
C:\Users\yara\ipconfig
Windows IP Configuration

Ethernet adapter Ethernet:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . . . .
  Wireless LAN adapter Local Area Connection* 1:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . . . .
  Wireless LAN adapter Local Area Connection* 2:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . . . .
  Ethernet adapter VMware Network Adapter VMnet1:
  Connection-specific DNS Suffix . . . .
  Link-local IPv6 Address . . . . . : fe00::1:10ef:8021:7598%6
  IPv4 Address . . . . . : 192.168.1.12
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . .
  Ethernet adapter VMware Network Adapter VMnet8:
  Connection-specific DNS Suffix . . .
  Link-local IPv6 Address . . . . . : fe00::1:10e5:bace:d546:dbf7%15
  IPv4 Address . . . . . : 192.168.1.12
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . .
  Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . . .
  Link-local IPv6 Address . . . . . : fe00::21a3:250e:f4d5:913b%4
  IPv4 Address . . . . . : 192.168.1.12
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . .
  Ethernet adapter Bluetooth Network Connection:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . . .
  Ethernet adapter Ethernet (Default Switch):
  Connection-specific DNS Suffix . . .
  Link-local IPv6 Address . . . . . : fe00::822d:30w1:e2:deaa%20
  IPv4 Address . . . . . : 192.168.208.1
  Subnet Mask . . . . . : 255.255.248.0
  Default Gateway . . . . .
C:\Users\yara\ipconfig /flushdns
Windows IP Configuration
Successfully flushed the DNS Resolver Cache.
C:\Users\yara|
```



Part 2

Using socket programming, implement UDP client and server applications in go, python, java or C. The server should listen on port 5051.

All peers can send and receive messages. This way, a message sent by a peer will be received by all peers. The message should include first and last name as well as a message (e.g. "Hello"). Read the message from the keyboard. The server lists the last received message from a client. If 3 clients sent messages, the server should display 4 lines something like

A peer should display something like:

Peer First name Last name

- 1- received a message from **First name Last name** at **Time**
 - 2- received a message from **First name Last name** at **Time**
 - 3- received a message from **First name Last name** at **Time**
where **First name Last name** is the sender first and last names. **Time** is the last time
a message received from that sender.

Then the peer can display the content of the message received from a peer by typing a line number and the letter D. For example, 2D to display the message received from the peer in line 2.

The python code:

Server code:

The screenshot shows a Microsoft Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder structure including 'part2server.py'.
- Run and Debug:** Shows variables, watch, call stack, and breakpoints.
- Code Editor:** Displays the Python code for a network server. The code uses `socket` and `socket.SOCK_DGRAM` to handle broadcast messages. It binds to port 5051 and receives messages from clients. For each received message, it prints the message and the client's address, then sends the same message back to all other connected peers. It also adds the sender to the list of peers and handles keyboard interrupts.
- Terminal:** Shows the command-line interface with the following output:

```
PS C:\Users\user\OneDrive\Desktop\network> python part2server.py
[2024-05-10 18:37:08] Received message from yara khattab: HE LLO AT 2024-05-10 18:37:08
=====
[2024-05-10 18:37:17] Received message from miassar shamala: hi AT 2024-05-10 18:37:17
=====
[2024-05-10 18:37:34] Received message from yara khattab: Ho w are you? AT 2024-05-10 18:37:34
=====
[2024-05-10 18:38:03] Received message from miassar shamala: im fine,thx AT 2024-05-10 18:38:03
=====
```
- Status Bar:** Shows file path (part2server.py), line 23, column 26, spaces 4, UTF-8, Python 3.11.9 64-bit (Microsoft Store), and current time (6:41 PM, 5/10/2024).

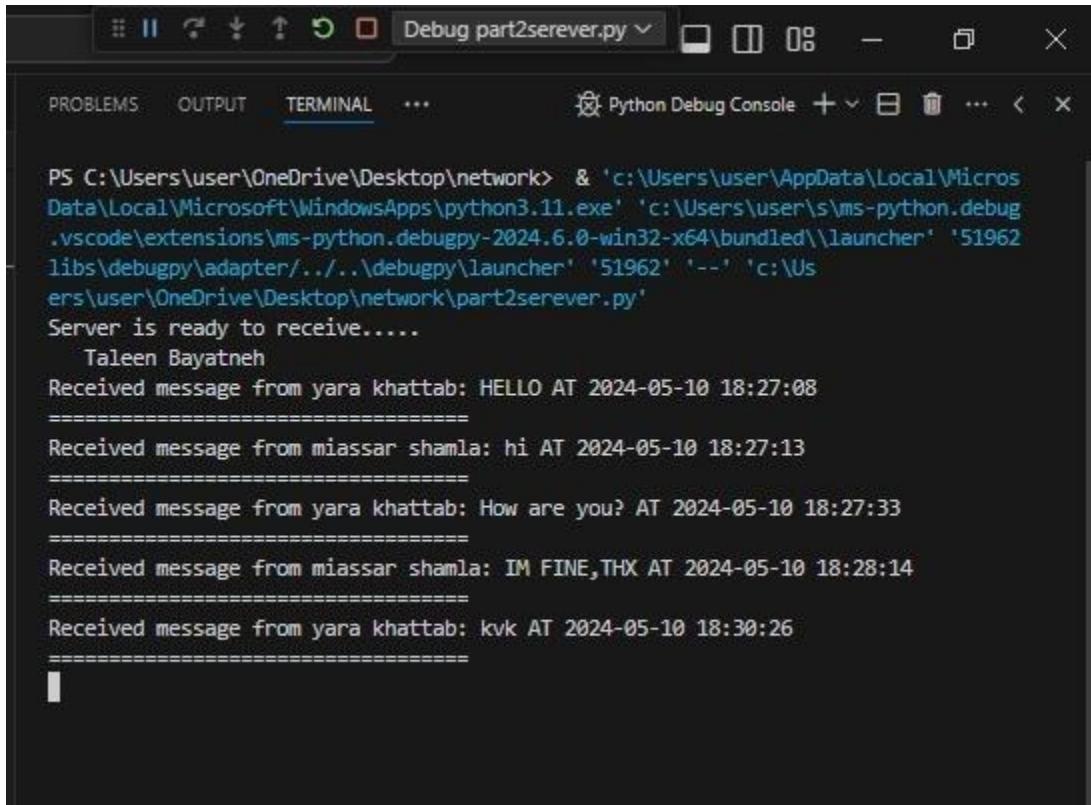
Peer code :

```
peercode.py X
C: > Users > yara > OneDrive > Desktop > network > peercode.py > ...
1  from socket import *
2  import threading
3  import datetime
4  port = 5051
5  server_ip = '192.168.1.107' #ip of the server laptop
6  messages = []
7  #receiving function
8  def receive_messages(socket1):
9      while True:
10         try:
11             data, _ = socket1.recvfrom(2048)
12             timestamp = datetime.datetime.now().strftime('%Y-%m-%d %H:%M:%S')
13             message = data.decode('utf-8')
14             line_number = len(messages) + 1
15             messages[line_number] = (message, timestamp)
16         except Exception as e:
17             print(f"An error occurred: {e}")
18     #thread
19 def main():
20     peer_socket = socket(AF_INET, SOCK_DGRAM)
21     peer_socket.bind('', port)
22     # threading for receiving message function
23     recv_thread = threading.Thread(target=receive_messages, args=(peer_socket,))
24     recv_thread.daemon = True # the thread will automatically exit when the main program exits even if it is still running
25     recv_thread.start() # start the thread
26
27     first_name = input("First name: ")
28     last_name = input("Last name: ")
29
30     try:
31         while True:
32             choice = input("Enter '1' to send a message, '2' to display a message: ")
33             #send a message by peer
34             if choice == '1':
35                 msg = input("Enter your message: ")
36                 full_mesge = f"{first_name} {last_name}: {msg}"
37                 peer_socket.sendto(full_mesge.encode('utf-8'), (server_ip, port))
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
```

```
peercode.py X
C: > Users > yara > OneDrive > Desktop > network > peercode.py > ...
19  def main():
20      peer_socket.sendto(full_mesge.encode('utf-8'), (server_ip, port))
21      print("Your message was sent.")
22      print("____")
23      #display a message that received to the peer by enter the line number +D
24      elif choice == '2':
25          line = input("Enter the line number and 'D' to display the message (e.g., '2D'): ")
26          if line.endswith('D'):
27              line_number = int(line[:-1])
28              if line_number in messages:
29                  print(f"Message from line {line_number}: {messages[line_number][0]}")
30                  print("____")
31              else:
32                  print("No message found of this line number !!!!")
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
```

The run:

server:



A screenshot of a terminal window titled "Debug part2serever.py". The window shows a command-line interface with several messages. The messages indicate that the server is listening for connections and has received messages from multiple peers. The messages are timestamped and show different users sending hello and status updates.

```
PS C:\Users\user\OneDrive\Desktop\network> & 'c:\Users\user\AppData\Local\Microsoft\WindowsApps\python3.11.exe' 'c:\Users\user\s\ms-python.debug\vscode\extensions\ms-python.debugpy-2024.6.0-win32-x64\bundled\launcher' '51962' 'libs\debugpy\adapter\..\..\debugpy\launcher' '51962' '--' 'c:\Users\user\OneDrive\Desktop\network\part2serever.py'
Server is ready to receive.....
Taleen Bayatneh
Received message from yara khattab: HELLO AT 2024-05-10 18:27:08
=====
Received message from miassar shamla: hi AT 2024-05-10 18:27:13
=====
Received message from yara khattab: How are you? AT 2024-05-10 18:27:33
=====
Received message from miassar shamla: IM FINE,THX AT 2024-05-10 18:28:14
=====
Received message from yara khattab: kvk AT 2024-05-10 18:30:26
=====
```

As we can see the server receive the message from the peer then if the other peers asked to see the message he will send it. So the server is an intermediary between peers.

Peer 1:

Peer 2:

```
PROBLEMS TERMINAL ... Python Debug Console + ×
```

```
PS C:\Users\A TO Z\Desktop> & 'c:\Users\A TO Z\AppData\Local\led\libs\debugpy\adapter/../..\debugpy\launcher' '51695' '--'
First name: miassar
Last name: shamla
Enter '1' to send a message, '2' to display a message: 1
Enter your message: hi
Your message was sent.

Enter '1' to send a message, '2' to display a message: 2
Enter the line number and 'd' to display the message (e.g., '2
Message from line 1: yara khattab: How are you?

Enter '1' to send a message, '2' to display a message: 1
isplay a message: 1
Enter '1' to send a message, '2' to d
isplay a messaaas sent.
Enter '1' to send a message, '2' to display a message:
isplayyour message: im fine,thx
Enter '1' to send a message, '2' to display a message: 1
Enter your message: im fine,thx
Your message was sent.

Enter '1' to send a message, '2' to display a message: 1

Activate Windows
Go to Settings to activate Windows.

Spaces: 4  UTF-8  { Python  3.11.9 64-bit (Microsoft Store)  🔔
15°C Mostly clear  ⏸  ☁  📡  ⚡  ENG  6:38 PM
5/10/2024  📈
```

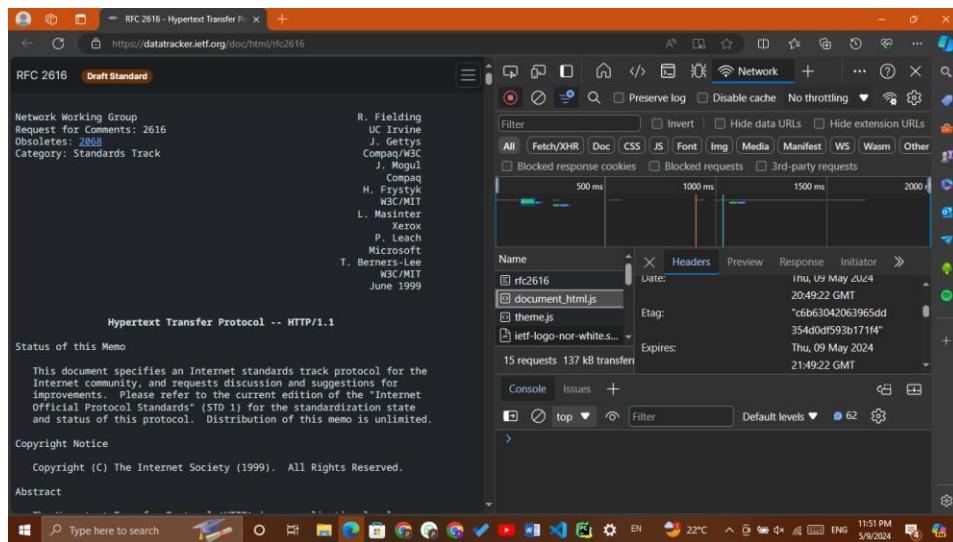
Here we ask the peers to enter there first and last name ,then enter 1 or 2 for send a message or display a message

Part 3

From rfce2616, what is Entity Tag Cache Validators in the HTTP protocol and why do we need it?

Entity Tag Cache Validators from rfce2616 :

Etag: "c6b63042063965dd354d0df593b171f4"



- What is Etag?

An ETag is an identifier that represents a specific version of a resource.

It allows caches (such as browser caches or proxy servers) to be more efficient by saving bandwidth. When a web server responds with an ETag, the client can use it to determine whether the content has changed since the last request.

If the content hasn't changed, the client can reuse its cached version, avoiding unnecessary data transfer.

- Why do we need Etag?

ETags play a crucial role in optimizing caching, preventing collisions during concurrent edits, and efficiently handling conditional requests in the HTTP protocol

Using socket programming, implement a simple but a complete web server in go, python, java or C that is listening on port 6060.

The code in python:

```
clientpart2.py
from socket import *
portnumber=6060
serversocket = socket(AF_INET,SOCK_STREAM)
serversocket.bind(("",portnumber))
serversocket.listen(1)
print ("The server is ready to listen on port 6060")
while True:
    connectionSocket, addr = serversocket.accept()
    message=connectionSocket.recv(2048).decode()
    print(addr)
    IP=addr[0]
    port=addr[1]
    print("IP: "+str(IP)+" Port: "+str(port))
    print(" ")
    print(message)
    print(" ")
    if message !='':
        requestedfile=message.split(' ')[1].replace('/','')
        print("The Requested File is: "+requestedfile)
    else:
        connectionSocket.close()
        continue
    try:
        if requestedfile == '' or requestedfile == 'index.html' or requestedfile=='main_en.html' or requestedfile == 'en':
            connectionSocket.send("HTTP/1.1 200 OK\r\n".encode())
            connectionSocket.send("Content-Type: text/html \r\n\r\n".encode())
            connectionSocket.send("\r\n".encode())
            main_html=open('main_en.html' , 'rb')
            connectionSocket.send(main_html.read())
            main_html.close()
        elif requestedfile=='ar':
            connectionSocket.send("HTTP/1.1 200 OK\r\n".encode())
            connectionSocket.send("Content-Type: text/html \r\n\r\n".encode())
            connectionSocket.send("\r\n".encode())
            main_html=open('main_ar.html' , 'rb')
            connectionSocket.send(main_html.read())
            main_html.close()
        elif '.html' in requestedfile:
            connectionSocket.send("HTTP/1.1 200 OK\r\n".encode())
            connectionSocket.send("Content-Type: text/html \r\n\r\n".encode())
        elif '.css' in requestedfile:
            connectionSocket.send("HTTP/1.1 200 OK\r\n".encode())
            connectionSocket.send("Content-Type: text/css \r\n\r\n".encode())
            connectionSocket.send("\r\n".encode())
            print('response status: 200 OK\r\n')
            requested_file= open(str(requestedfile), 'rb')
            connectionSocket.send(requested_file.read())
            requested_file.close()
        elif '.png' in requestedfile:
            connectionSocket.send("HTTP/1.1 200 OK\r\n".encode())
            connectionSocket.send("Content-Type: image/png \r\n\r\n".encode())
            connectionSocket.send("\r\n".encode())
            print('response status: 200 OK\r\n')
            requested_file= open(str(requestedfile), 'rb')
            connectionSocket.send(requested_file.read())
            requested_file.close()
        elif '.jpg' in requestedfile:
            connectionSocket.send("HTTP/1.1 200 OK\r\n".encode())
            connectionSocket.send("Content-Type: image/jpeg \r\n\r\n".encode())
            connectionSocket.send("\r\n".encode())
            print('response status: 200 OK\r\n')
            requested_file= open(str(requestedfile), 'rb')
            connectionSocket.send(requested_file.read())
            requested_file.close()
        elif requestedfile == 'so':
            connectionSocket.send("HTTP/1.1 307 Temporary Redirect\r\n".encode())
            connectionSocket.send("Location: https://stackoverflow.com/\r\n".encode())
        elif requestedfile == 'itc':
            connectionSocket.send("HTTP/1.1 307 Temporary Redirect\r\n".encode())
            connectionSocket.send("Location: https://itc.birzeit.edu/register/\r\n".encode())
    except:
        connectionSocket.close()
```

The screenshot shows a Microsoft Visual Studio Code interface with three tabs open: clientpart2.py, part3Finish.py (the active tab), and serverpart2.py. The code in part3Finish.py is as follows:

```
75
76
77     else:
78         raise Exception('Not Found')
79 except Exception as e:
80     connectionSocket.send(("HTTP/1.1 404 Not Found\r\n".encode()))
81     connectionSocket.send(("Content-Type: text/html \r\n\r\n".encode()))
82     connectionSocket.send(("\" not found\r\n".encode()))
83     print("Requested file - not found")
84     print("\n")
85     print("<html>\n<head>\n<title>Error 404</title>\n</head>\n<body>\n<div id="Error">\n<h1>The file is not found</h1>\n</div>\n<div id="name">\n<p>Missar Shama - 1210519</p>\n<p>Taleen Bayatneh - 1211305</p>\n<p>Yara Khattab - 1210528</p>\n</div>\n<div>\n<p>Ip Adress: " + str(IP) + ", Port Number: " + str(port) +\n</p> </div>\n</body></html>\n"
86
87     |     connectionSocket.send(requested_file.encode())
88
89     connectionSocket.close()
```

The Python Debug Console shows the command to run the script and the message "The server is ready to listen on port 6060".

The program should check

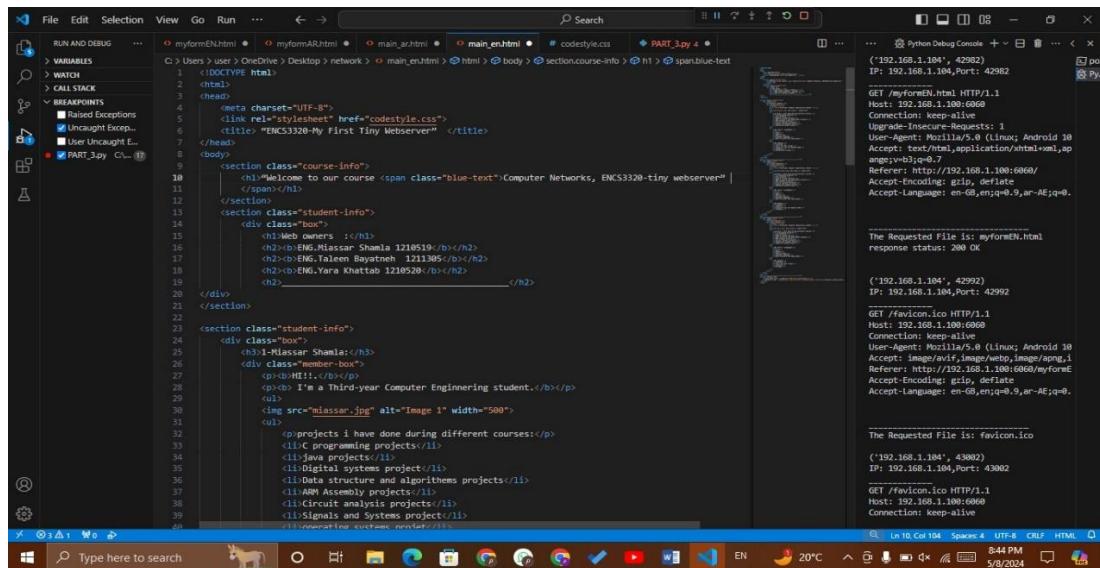
- 1- if the request is / or /index.html or /main_en.html or /en (for example localhost:6060/ or localhost:6060/en) then the server should send main_en.html file with Content-Type: text/html.

The main_en.html file should contain HTML webpage that contains

- a. "ENCS3320-My First Tiny Webserver" in the title
- b. "Welcome to our course Computer Networks, ENCS3320-tiny webserver" (part of the phrase is in Blue)
- c. Group members names and IDs
- d. Some information about the group members. For instance, projects you have done during different course (programming, electrical, math, etc), skills, hobbies, etc.
- e. Use CSS to make the page looks nice
- f. Divide the page in different boxes and put student's information in the different boxes
- g. Include CSS as a separate file
- h. The page should contain at least An image with extention.jpg and an image with extension .png
- i. A link to a local html file (myform.html) |
- i. a link to https://www.w3schools.com/python/python_syntax.asp

The codes:

main_en.html:



The screenshot shows a Windows desktop environment. On the left, there is a code editor window titled "main_en.html" containing the source code for the HTML file. The code includes a title, a section for group members with their names and IDs, and a section for projects. It also includes a CSS file reference and a few images. On the right, there is a terminal window titled "Python Debug Console" showing the server logs for requests to "/myformEN.html" and "/favicon.ico". The logs show the IP address (192.168.1.104), port (42982), and the content sent back to the client, which includes the requested file and its status (200 OK). The desktop taskbar at the bottom shows various icons for programs like File Explorer, Edge, and FileZilla.

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <link rel="stylesheet" href="codestyle.css">
        <title>ENCS3320-My First Tiny Webserver</title>
    </head>
    <body>
        <section class="course-info">
            <h1>Welcome to our course <span class="blue-text">Computer Networks, ENCS3320-tiny webserver</span></h1>
        </section>
        <section class="student-info">
            <div class="box">
                <h2>Web owners :</h2>
                <h2>ENG.Missar Shama 1210519</h2>
                <h2>ENG.Taleen Baytneh 1211305</h2>
                <h2>ENG.Yara Khattab 1210520</h2>
            </div>
        </section>
        <section class="student-info">
            <div class="member-box">
                <p>I'm a Third-year Computer Engineering student.</p>
                <ul>
                    <li></li>
                </ul>
                <p>Projects I have done during different courses:</p>
                <ul>
                    <li>C programming projects</li>
                    <li>Java projects</li>
                    <li>Digital systems project</li>
                    <li>Data structure and algorithms projects</li>
                    <li>ARM Assembly projects</li>
                    <li>Circuit analysis projects</li>
                    <li>Signals and Systems project</li>
                </ul>
            </div>
        </section>
    </body>
</html>
```

```
(*'192.168.1.104', 42982)
IP: 192.168.1.104, Port: 42982
GET /myformEN.html HTTP/1.1
Host: 192.168.1.104:6060
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Linux; Android 10
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/*,*/*;q=0.8
Referer: http://192.168.1.104:6060/myFormE
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,ar-AE;q=0.8

The Requested File is: myformEN.html
response status: 200 OK

(*'192.168.1.104', 42992)
IP: 192.168.1.104, Port: 42992
GET /favicon.ico HTTP/1.1
Host: 192.168.1.104:6060
Connection: keep-alive
User-Agent: Mozilla/5.0 (Linux; Android 10
Accept: image/webp,image/*,*/*;q=0.8
Referer: http://192.168.1.104:6060/myFormE
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,ar-AE;q=0.8

The Requested File is: favicon.ico
(*'192.168.1.104', 43002)
IP: 192.168.1.104, Port: 43002
GET /favicon.ico HTTP/1.1
Host: 192.168.1.104:6060
Connection: keep-alive
```

The screenshot shows a Windows desktop environment with several open windows. In the center, a browser window displays a local file named 'myFormEN.html'. The page content includes sections for 'student info' and 'Skills', listing various programming languages and hobbies. On the right side of the screen, a terminal window titled 'Python Debug Console' is open, showing the command line interface for a Python application. The terminal output includes several log messages indicating successful file reads and responses to requests from '192.168.1.104'. At the bottom of the screen, the taskbar is visible with icons for File Explorer, a search bar, and other system applications.

```
<html>
  <head>
    <title>My Form</title>
    <link href="codestyle.css" type="text/css" rel="stylesheet"/>
  </head>
  <body>
    <h1>Student Information</h1>
    <div>
      <h2>Telen Bayatneh</h2>
      <div class="member-box">
        <p>Hello!!</p>
        <p>I'm a third-year Computer Engineering student.</p>
        <ul>
          <li></li>
        </ul>
      </div>
    </div>
    <h2>Skills</h2>
    <div>
      <ul>
        <li>Signals and Systems project</li>
        <li>operating systems projet</li>
      </ul>
      <ul>
        <li>My skills: (languages)</li>
        <li>c</li>
        <li>Java</li>
        <li>Python</li>
        <li>good CSS</li>
        <li>Assembly language</li>
        <li>Data structure and algorithms</li>
      </ul>
      <ul>
        <li>My hobbies:</li>
        <li>relaxing</li>
        <li>chill out with frinds</li>
        <li>coding for fun not degrees hhhh</li>
        <li>sleeping</li>
        <li>eating with friends</li>
      </ul>
    </div>
  </body>
</html>
```

```
(192.168.1.104, 42982)
IP: 192.168.1.104, Port: 42982
GET /myFormEN.html HTTP/1.1
Host: 192.168.1.100:6060
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Linux; Android 10
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Referer: http://192.168.1.100:6060/
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8

The Requested File is: myFormEN.html
response status: 200 OK

(192.168.1.104, 42992)
IP: 192.168.1.104, Port: 42992
GET //favicon.ico HTTP/1.1
Host: 192.168.1.100:6060
Connection: keep-alive
User-Agent: Mozilla/5.0 (Linux; Android 10
Accept: image/*,*/*;q=0.8
Accept-Charset: image/png,image/jpeg,image/webp,image/apng,image/svg+xml,image/icon,image/x-icon,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8

The Requested File is: favicon.ico

(192.168.1.104, 43002)
IP: 192.168.1.104, Port: 43002
GET //favicon.ico HTTP/1.1
Host: 192.168.1.100:6060
Connection: keep-alive
```

The screenshot displays a Windows desktop environment with several open windows:

- Left Browser Window:** Shows a local file "myformEN.html" containing student information. The page includes sections for "student info" and "My skills", listing various projects and hobbies.
- Right Browser Window:** Shows a "Python Debug Console" with network traffic logs. It lists multiple GET requests for files like "myformEN.html", "favicon.ico", and "codestyle.css".
- Terminal Window:** Located at the bottom, showing command-line output related to the Python application.
- Taskbar:** At the bottom, showing standard Windows icons for file operations, search, and system status.

The screenshot shows a Windows desktop environment with two browser windows open and a Python debugger window.

Left Browser Window:

- Title: myformEN.html
- Content: Displays a form with fields for Name, Age, and Gender, and a button labeled "Click here".

Right Browser Window:

- Title: myformEN.html
- Content: Displays a page with sections for "Student Info", "Hobbies", and "External Link". The "Hobbies" section lists "relaxing", "hang out with friends", "swimming", and "driving".

Python Debugger (PART_3.py) Window:

- File Path: C:\Users\user\Desktop\network\main_en\html\main_enhtml\index.html
- Code View:

```
    <head>
        ...
        <body>
            ...
            <div class="member-box">
                ...
                <ul>
                    <li>My hobbies:</li>
                    <li>relaxing</li>
                    <li>hang out with friends</li>
                    <li>swimming</li>
                    <li>driving</li>
                </ul>
            </div>
        </div>
    </section>
<section>
    ...
</section>
<div class="link">
    ...
    <p>Local HTML File: <a href="myformEN.html">Click here</a></p>
    <p>External Link: <a href="https://www.w3schools.com/python/python_syntax.asp">w3schools python_syntax</a></p>
</div>
</body>
```
- Log View:

```
192.168.1.104 - - [29/Jul/2024:10:29:42 +0000] "GET / HTTP/1.1
Host: 192.168.1.104:42992
Connection: keep-alive
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win32; rv:109.0) Gecko/20100101 Firefox/109.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Referer: http://192.168.1.100:6060/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,ar-KE;q=0.8

The Requested File is: w3schools.html
response status: 200 OK

192.168.1.104 - - [29/Jul/2024:10:29:42 +0000] "GET /w3schools.html HTTP/1.1
Host: 192.168.1.104:42992
Connection: keep-alive
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win32; rv:109.0) Gecko/20100101 Firefox/109.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Referer: http://192.168.1.100:6060/w3schools/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9,ar-KE;q=0.8

The Requested File is: favicon.ico
response status: 200 OK

192.168.1.104 - - [29/Jul/2024:10:29:42 +0000] "GET /favicon.ico HTTP/1.1
Host: 192.168.1.104:42992
Connection: keep-alive
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win32; rv:109.0) Gecko/20100101 Firefox/109.0
Accept: image/png,*/*;q=0.8
Accept-Language: en-US,en;q=0.9,ar-KE;q=0.8
```

main_ar.html:

File Edit Selection View Go Run ... ← → ⌂ Search

RUN AND DEBUG ...

VARIABLES

WATCH

CALL STACK

BREAKPOINTS

- Raised Exceptions
- Uncaught Exceptions
- User Uncaught E...
- PART_3.py

```

C:\> Users > user > OneDrive > Desktop > network > main_en.html > http > body > section.course-info > h1 > span.blue-text
1 <html>
2   <body>
3     <section class="student-info">
4       <div class="member-box">
5         <ul>
6           <li>Signals and Systems project</li>
7           <li>operating systems proj</li>
8           <li></li>
9           <li>My skills: (languages)</p>
10          <ul>
11            <li>C</li>
12            <li>Java</li>
13            <li>Python</li>
14            <li>HTML and CSS</li>
15            <li>Assembly language</li>
16            <li>Data structure and algorithms</li>
17          </ul>
18        </ul>
19      </div>
20    </section>
21    <h2>Taleen Bayatneh</h2>
22    <div class="box">
23      <h3>I'm a Third-year Computer Engineering student.</h3>
24      <ul>
25        <li></li>
26      </ul>
27    </div>
28  </div>
29  </body>
30</html>

```

Python Debug Console

(192.168.1.104, 42982)

GET /myformEN.html HTTP/1.1

Host: 192.168.1.104:6060

Connection: keep-alive

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Linux; Android 10; Pixel 4 XL) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.82 Mobile Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8

Referer: http://192.168.1.104:6060/myFormE

Accept-Encoding: gzip, deflate

Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.

The Requested File is: myformEN.html

response status: 200 OK

(192.168.1.104, 42992)

GET /favicon.ico HTTP/1.1

Host: 192.168.1.104:6060

Connection: keep-alive

User-Agent: Mozilla/5.0 (Linux; Android 10; Pixel 4 XL) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.82 Mobile Safari/537.36

Accept: image/avif,image/webp,image/apng,*/*;q=0.8

Referer: http://192.168.1.104:6060/myFormE

Accept-Encoding: gzip, deflate

Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.

The Requested File is: favicon.ico

(192.168.1.104, 43002)

GET /favicon.ico HTTP/1.1

Host: 192.168.1.104:6060

Connection: keep-alive

User-Agent: Mozilla/5.0 (Linux; Android 10; Pixel 4 XL) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.82 Mobile Safari/537.36

Accept: image/avif,image/webp,image/apng,*/*;q=0.8

Referer: http://192.168.1.104:6060/myFormE

Accept-Encoding: gzip, deflate

Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.

The Requested File is: favicon.ico

8:44 PM 5/8/2024

File Edit Selection View Go Run ... ← → ⌂ Search

RUN AND DEBUG ...

VARIABLES

WATCH

CALL STACK

BREAKPOINTS

- Raised Exceptions
- Uncaught Exceptions
- User Uncaught E...
- PART_3.py

```

C:\> Users > user > OneDrive > Desktop > network > main_ar.html > http > body > section.course-info > h1 > span.blue-text
1 <html>
2   <body>
3     <section class="student-info">
4       <div class="member-box">
5         <ul>
6           <li>مختبر الاتصالات والرقمية</li>
7           <li>مشاريع البيانات وقواعد البيانات</li>
8           <li>الذكاء الاصطناعي</li>
9           <li>الشبكات والاتصالات</li>
10          <li>البرمجيات المترافقه</li>
11          <li>مقدمة في الاتصالات والتكنولوجيا</li>
12        </ul>
13      </div>
14    </section>
15    <h2>تالين بعثنه</h2>
16    <div class="box">
17      <h3>أنا طالبة منتدسة حاملي بحث تالين</h3>
18      <ul>
19        <li></li>
20      </ul>
21    </div>
22  </div>
23  </body>
24</html>

```

Python Debug Console

(192.168.1.104, 42982)

GET /myformEN.html HTTP/1.1

Host: 192.168.1.104:6060

Connection: keep-alive

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Linux; Android 10; Pixel 4 XL) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.82 Mobile Safari/537.36

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8

Referer: http://192.168.1.104:6060/myFormE

Accept-Encoding: gzip, deflate

Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.

The Requested File is: myformEN.html

response status: 200 OK

(192.168.1.104, 42992)

GET /favicon.ico HTTP/1.1

Host: 192.168.1.104:6060

Connection: keep-alive

User-Agent: Mozilla/5.0 (Linux; Android 10; Pixel 4 XL) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.82 Mobile Safari/537.36

Accept: image/avif,image/webp,image/apng,*/*;q=0.8

Referer: http://192.168.1.104:6060/myFormE

Accept-Encoding: gzip, deflate

Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.

The Requested File is: favicon.ico

(192.168.1.104, 43002)

GET /favicon.ico HTTP/1.1

Host: 192.168.1.104:6060

Connection: keep-alive

User-Agent: Mozilla/5.0 (Linux; Android 10; Pixel 4 XL) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/89.0.4389.82 Mobile Safari/537.36

Accept: image/avif,image/webp,image/apng,*/*;q=0.8

Referer: http://192.168.1.104:6060/myFormE

Accept-Encoding: gzip, deflate

Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.

The Requested File is: favicon.ico

8:44 PM 5/8/2024

The screenshot displays two separate instances of the PyCharm IDE running on a Windows operating system. Each instance has its own window title bar, menu bar, and toolbars. The left instance is focused on a file containing English text, while the right instance is focused on a file containing Arabic text. Both instances have their Python Debug Console tabs open, showing log entries for file requests and responses. The log entries include details such as the IP address, port number, HTTP method, connection type, user agent, accept headers, and accept language. The desktop background is visible at the bottom, showing the Windows taskbar with various pinned icons.

myformEN.html:



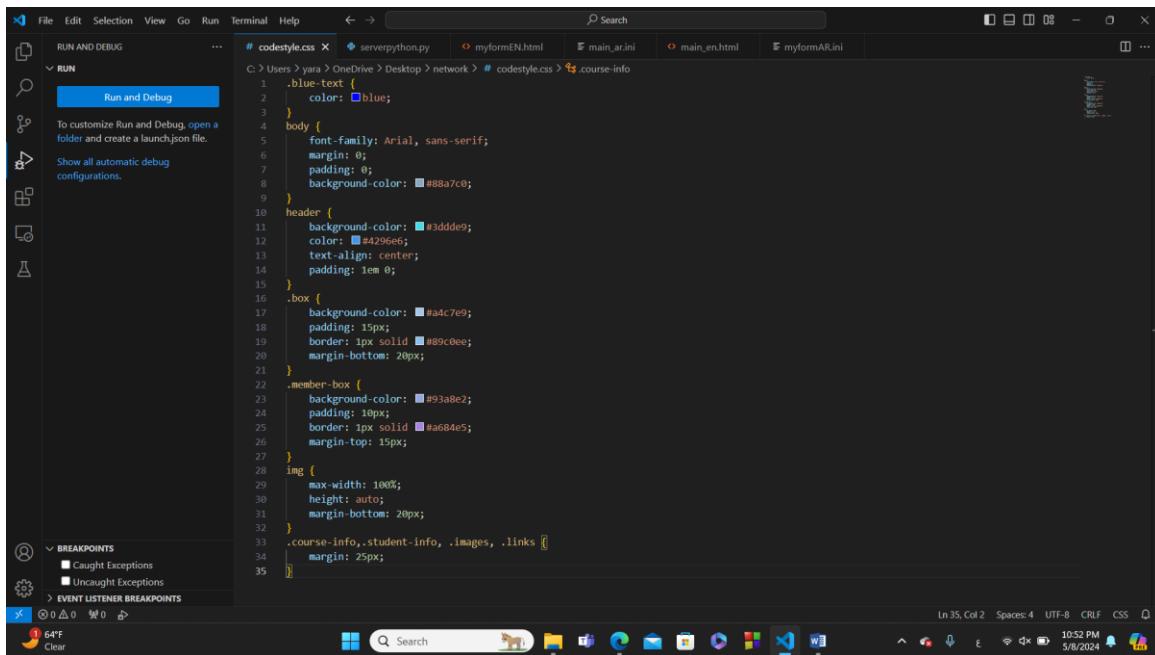
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>myform</title>
    <script>
      function displayImage() {
        var imageName = document.getElementById("imageName").value;
        document.getElementById("imageDisplay").src = imageName + ".png";
      }
    </script>
  </head>
  <style>
    .group {
      background-color: #f0f0f0;
      background-color: #f0f0f0;
      margin: 10px;
      border: 1px solid #d3d2c2;
      margin-bottom: 15px;
    }
    .box {
      background-color: #f0f0f0;
      padding: 15px;
      border: 1px solid #d3d2c2;
      margin-bottom: 15px;
    }
  </style>
  <body class="Group">
    <section class="box">
      <label for="imageName">Enter the name of the image:</label>
      <input type="text" id="imageName" name="imageName" onblur="displayImage()" value="Get Image"/>
      <br>
      <img id="imageDisplay" src="" alt="Image will be displayed here."/>
      <div>EMG.Missar Shemla 1116519</div>
      <div>EMG.Jaleen Bayatoh 1211455</div>
      <div>EMG.Yara Khattab 1116526</div>
      <div><a href="main_en.html" Click here!</a></div>
    </section>
  </body>
</html>
```

myformAR.html:



```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>مختبر المعرفة</title>
    <script>
      function displayImage() {
        var imageName = document.getElementById("imageName").value;
        document.getElementById("imageDisplay").src = imageName + ".png";
      }
    </script>
  </head>
  <style>
    .group {
      background-color: #f0f0f0;
      background-color: #f0f0f0;
      margin: 10px;
      border: 1px solid #d3d2c2;
      margin-bottom: 15px;
    }
    .box {
      background-color: #f0f0f0;
      padding: 15px;
      border: 1px solid #d3d2c2;
      margin-bottom: 15px;
    }
  </style>
  <body class="Group">
    <section class="box">
      <label for="imageName">ادخل اسم الصورة التي تريض عرضها :</label>
      <input type="text" id="imageName" name="imageName" onblur="displayImage()" value="عرض الصورة هنا">
      <div>صور تظهر المعرفة هنا</div>
      <div>صور تظهر المعرفة هنا</div>
      <div>صور تظهر المعرفة هنا</div>
      <div>صور تظهر المعرفة هنا</div>
      <div><a href="main_ar.html" ترجمة المحتوى هنا!</a></div>
    </section>
  </body>
</html>
```

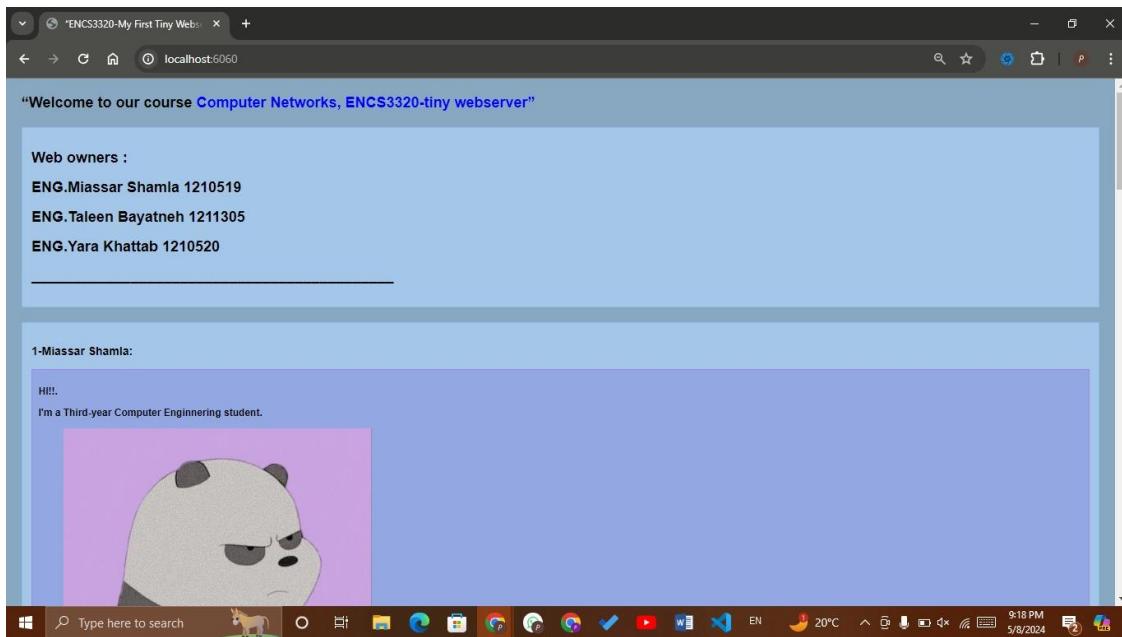
The html file is connected with a css file:



```
# codestyle.css
1 .blue-text {
2     color: blue;
3 }
4 body {
5     font-family: Arial, sans-serif;
6     margin: 0;
7     padding: 0;
8     background-color: #88a7c8;
9 }
10 header {
11     background-color: #3ddde9;
12     color: #496e6;
13     text-align: center;
14     padding: 1em 0;
15 }
16 .box {
17     background-color: #a4c7e9;
18     padding: 15px;
19     border: 1px solid #89c0ee;
20     margin-bottom: 20px;
21 }
22 .member-box {
23     background-color: #93a8e2;
24     padding: 10px;
25     border: 1px solid #a684e5;
26     margin-top: 15px;
27 }
28 img {
29     max-width: 100%;
30     height: auto;
31     margin-bottom: 20px;
32 }
33 .course-info,.student-info, .images, .links {
34     margin: 25px;
35 }
```

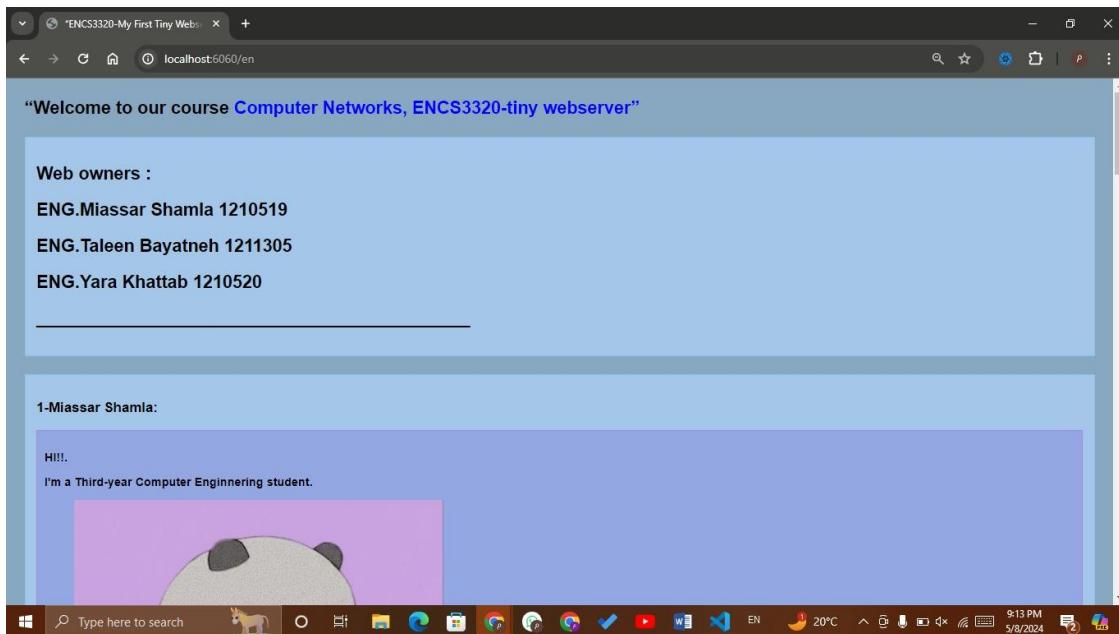
Test from the computer:

"/":



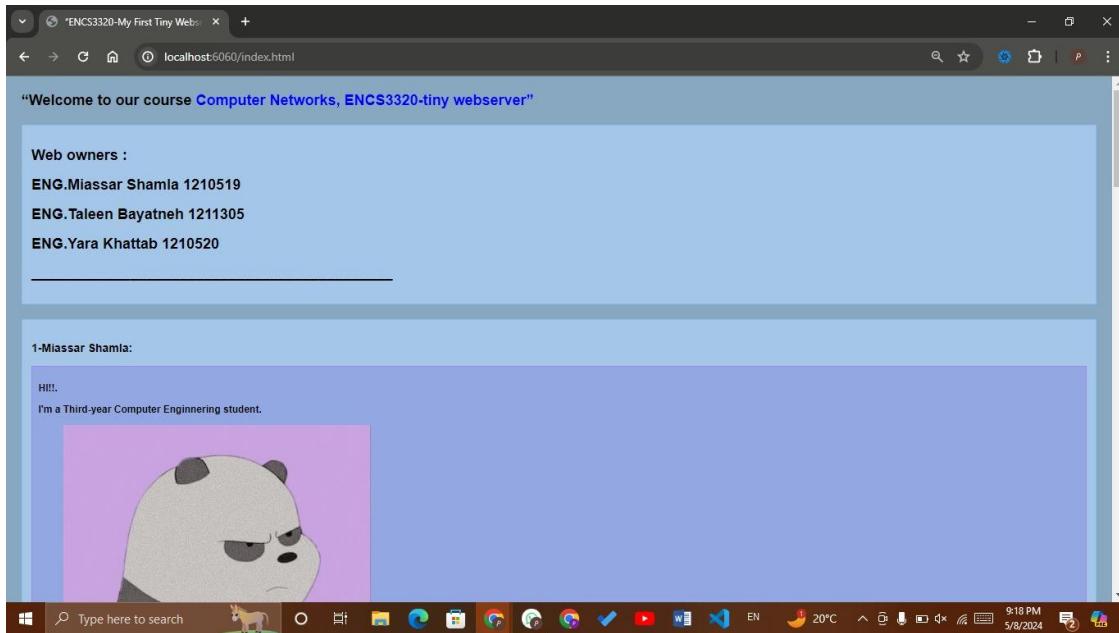
```
-----  
The Requested File is:  
('127.0.0.1', 54712)  
IP: 127.0.0.1, Port: 54712  
-----  
GET /codestyle.css HTTP/1.1  
Host: localhost:6060  
Connection: keep-alive  
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"  
sec-ch-ua-mobile: ?0  
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like  
Gecko) Chrome/124.0.0.0 Safari/537.36  
sec-ch-ua-platform: "Windows"  
Accept: text/css,*/*;q=0.1  
Sec-Fetch-Site: same-origin  
Sec-Fetch-Mode: no-cors  
Sec-Fetch-Dest: style  
Referer: http://localhost:6060/  
Accept-Encoding: gzip, deflate, br, zstd  
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

/en:



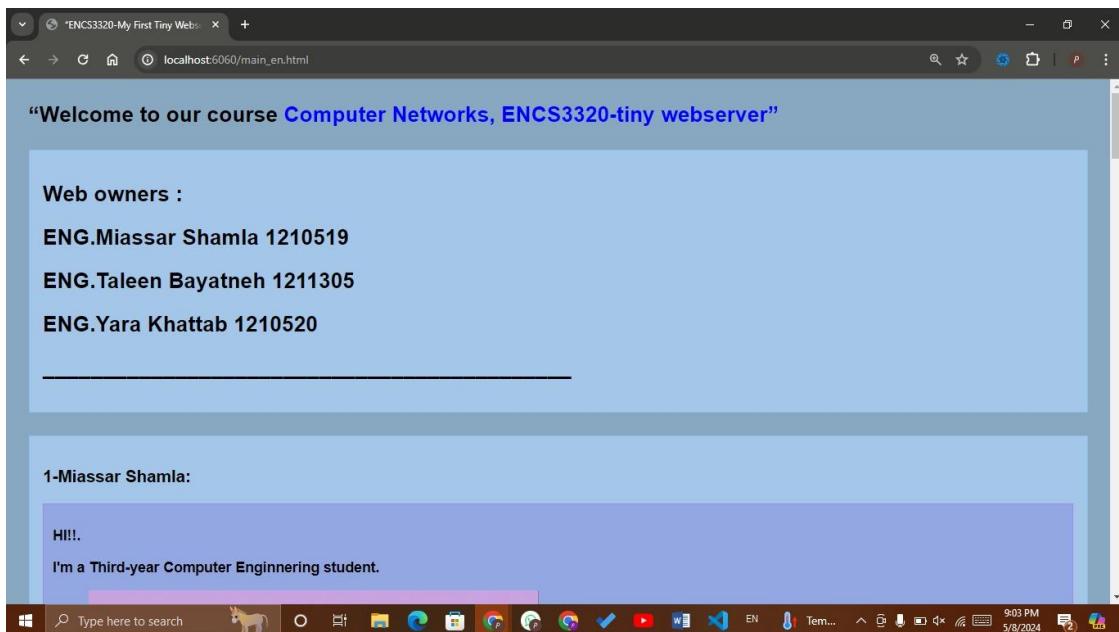
```
-----  
The Requested File is: en  
('127.0.0.1', 54967)  
IP: 127.0.0.1, Port: 54967  
-----  
GET /codestyle.css HTTP/1.1  
Host: localhost:6060  
Connection: keep-alive  
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"  
Sec-Purpose: prefetch;prerender  
sec-ch-ua-mobile: ?0  
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko  
o) Chrome/124.0.0.0 Safari/537.36  
sec-ch-ua-platform: "Windows"  
Accept: text/css,*/*;q=0.1  
Purpose: prefetch  
Sec-Fetch-Site: same-origin  
Sec-Fetch-Mode: no-cors  
Sec-Fetch-Dest: style  
Referer: http://localhost:6060/en  
Accept-Encoding: gzip, deflate, br, zstd  
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

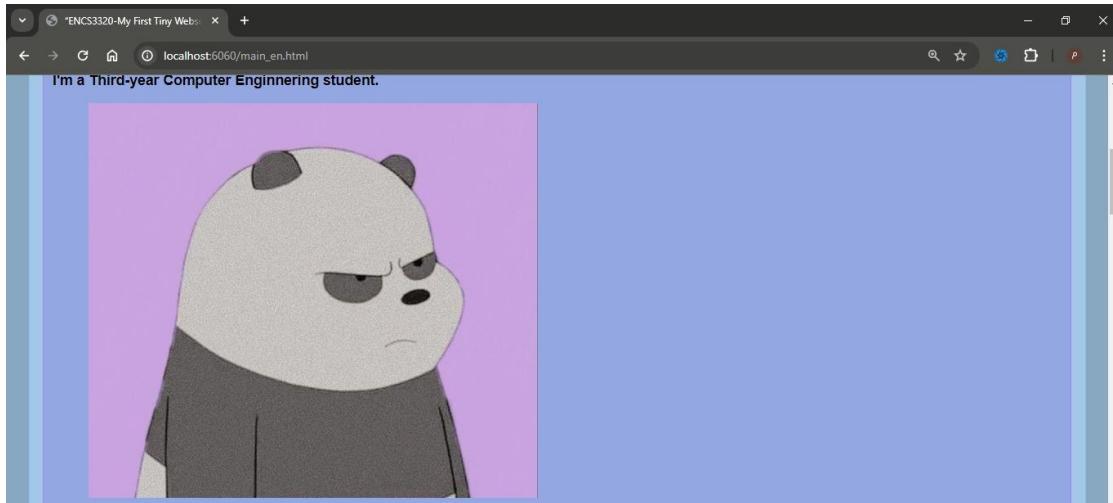
/index.html:



```
-----  
The Requested File is: index.html  
('127.0.0.1', 54924)  
IP: 127.0.0.1, Port: 54924  
  
GET /codestyle.css HTTP/1.1  
Host: localhost:6060  
Connection: keep-alive  
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"  
sec-ch-ua-mobile: ?0  
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)  
Chrome/124.0.0.0 Safari/537.36  
sec-ch-ua-platform: "Windows"  
Accept: text/css,*/*;q=0.1  
Sec-Fetch-Site: same-origin  
Sec-Fetch-Mode: no-cors  
Sec-Fetch-Dest: style  
Referer: http://localhost:6060/index.html  
Accept-Encoding: gzip, deflate, br, zstd  
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

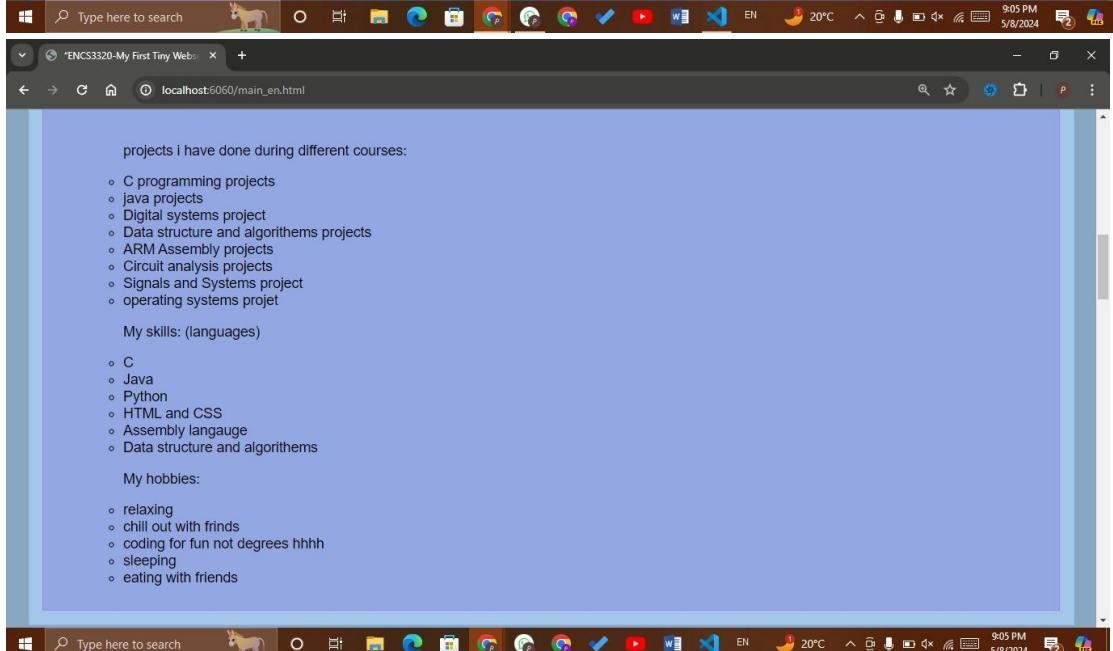
'main_en.html':

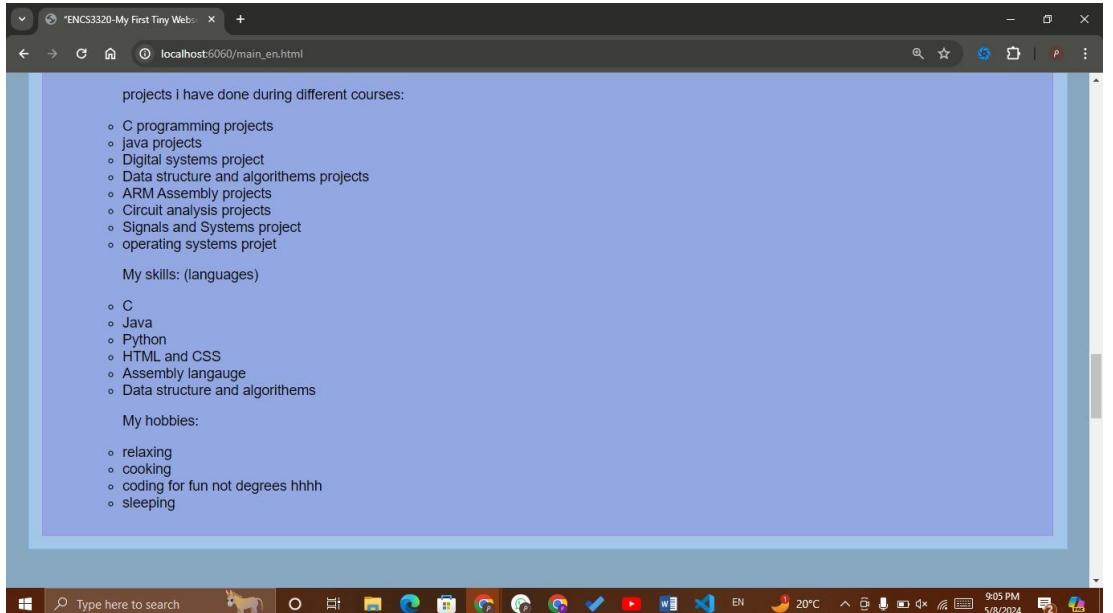
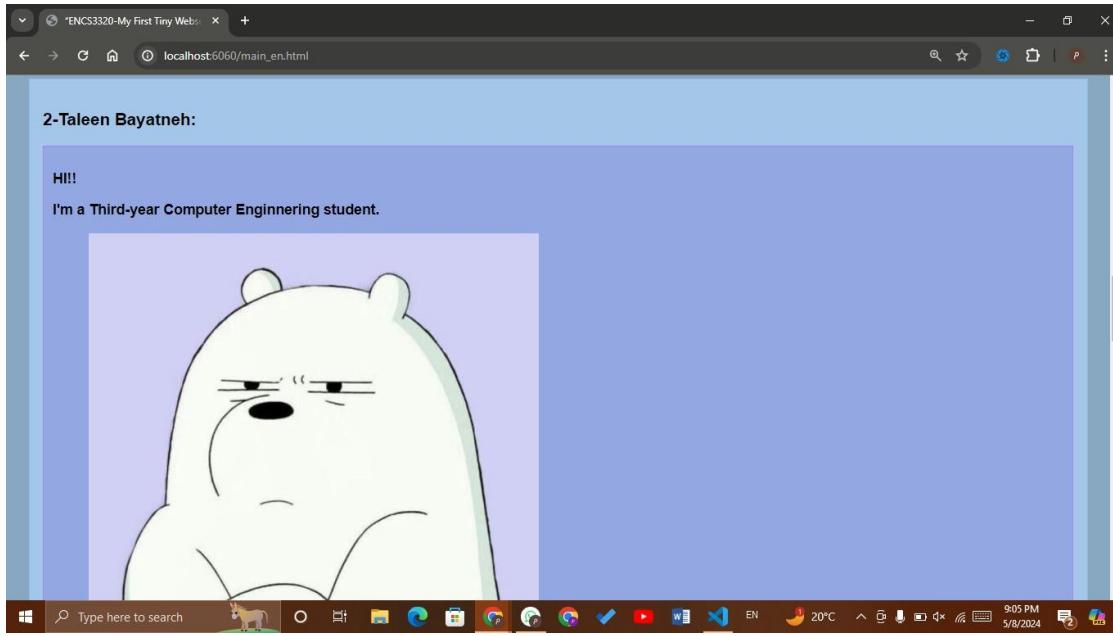


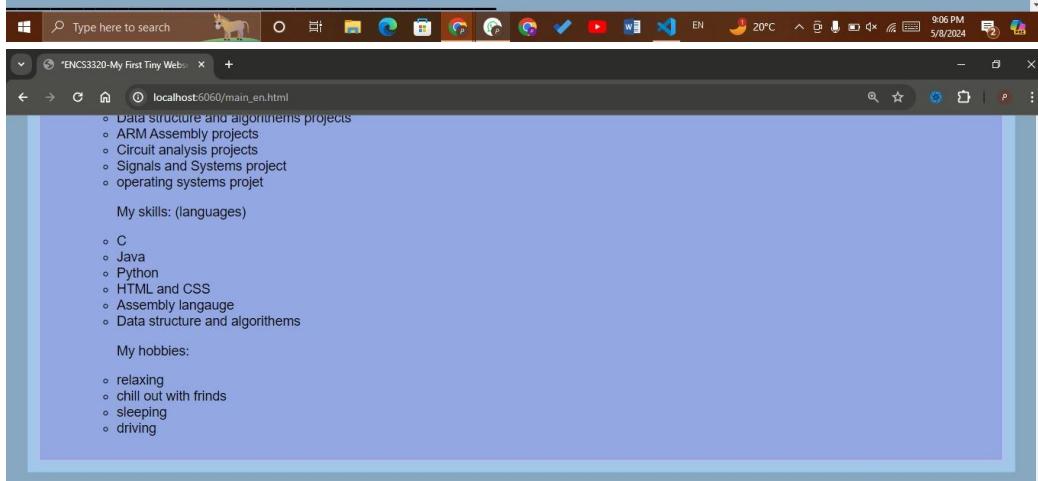
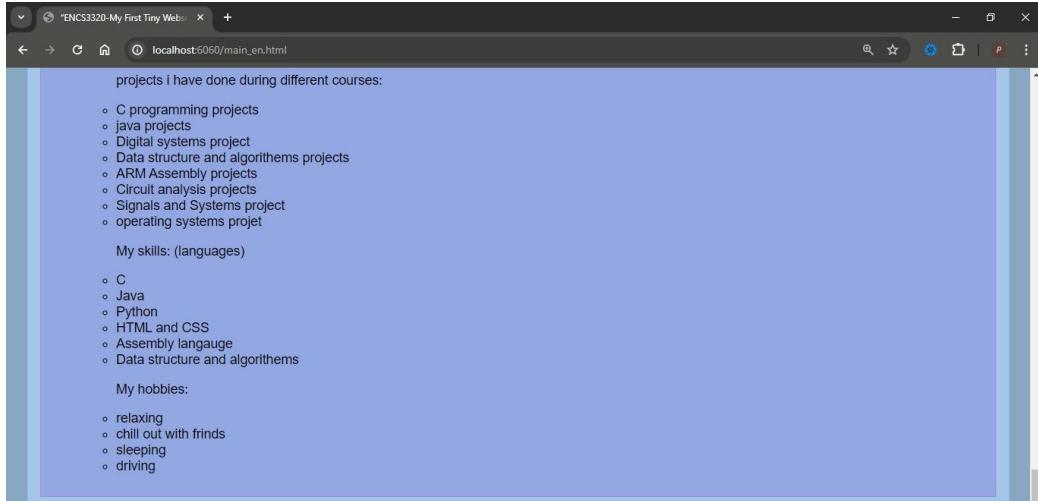
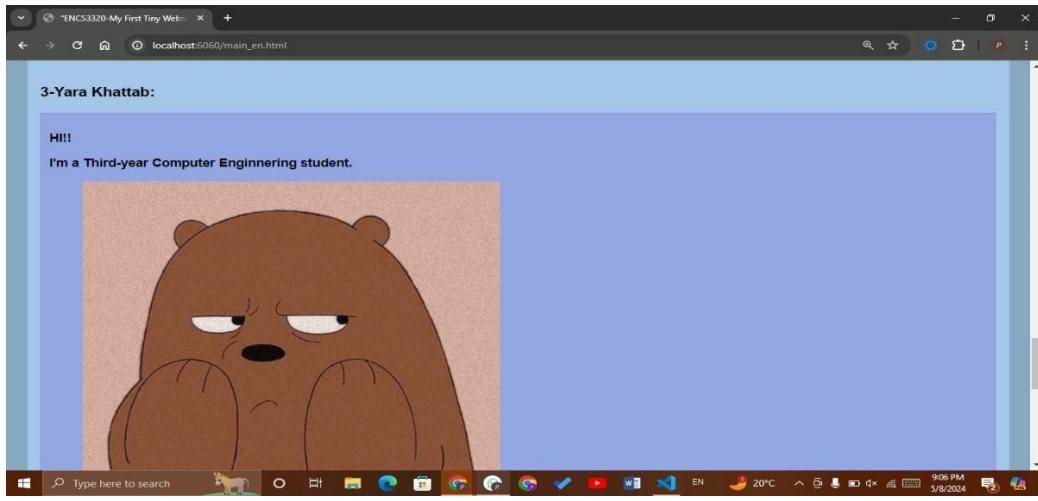


projects i have done during different courses:

- C programming projects
- java projects







Local HTML File: [Click here](#)

External Link: [w3schools python syntax](#)

```
The Requested File is: main_en.html
('127.0.0.1', 54596)
IP: 127.0.0.1, Port: 54596

-----
GET /codestyle.css HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: text/css,*/*;q=0.1
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: style
Referer: http://localhost:6060/main_en.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7
,en-US;q=0.6
```

When we click on External link:

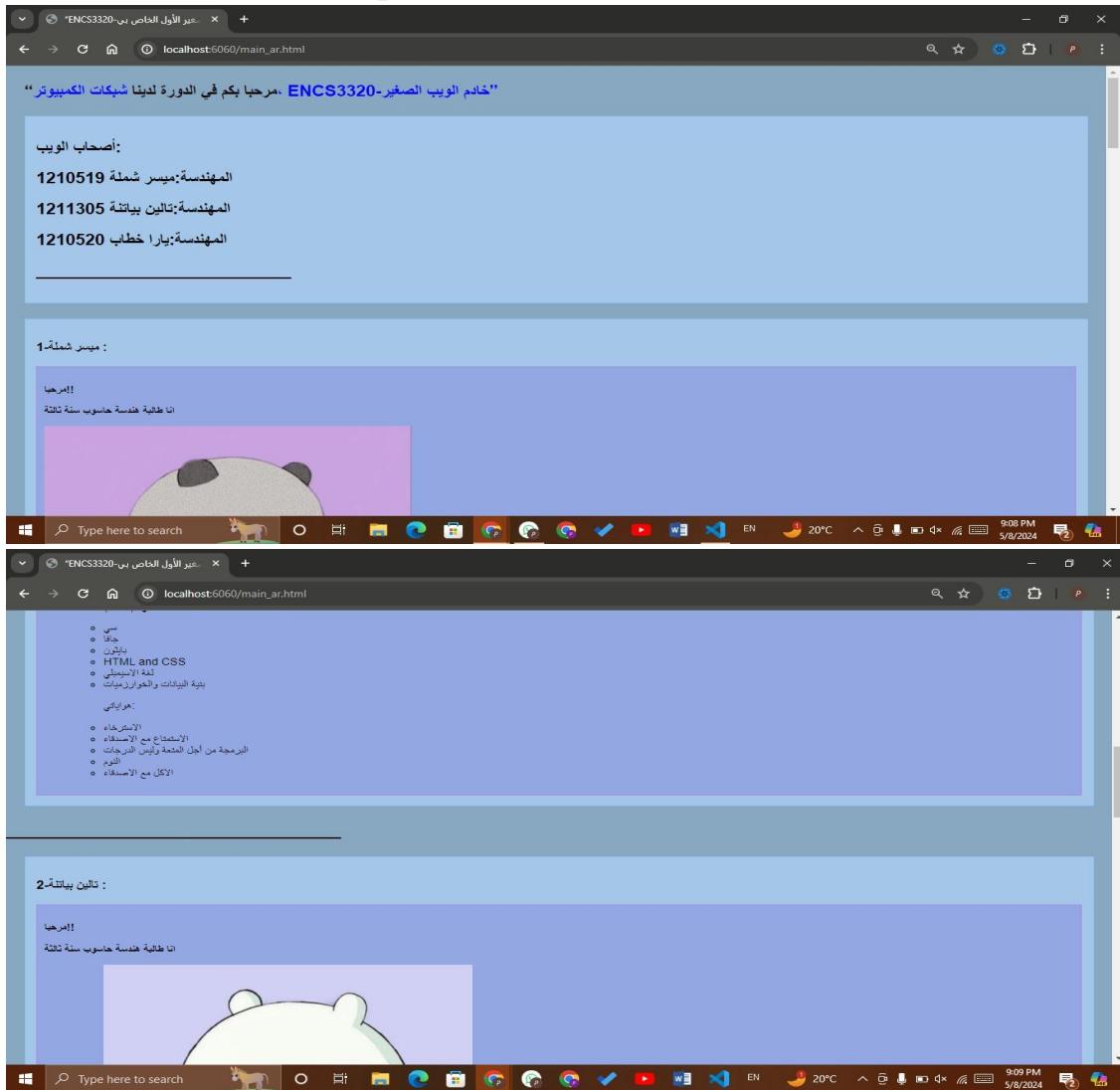
The screenshot shows a Microsoft Edge browser window displaying the 'Python Syntax' page from w3schools.com. The URL in the address bar is `w3schools.com/python/python_syntax.asp`. The browser's navigation bar includes icons for back, forward, search, and refresh. The top menu bar has links for Tutorials, Exercises, Certificates, Services, Search, Plus, Spaces, Get Certified, Sign Up, and Log In. The main content area features a sidebar on the left with a navigation tree for Python tutorials, including 'Python Tutorial', 'Python HOME', 'Python Intro', 'Python Get Started', and 'Python Syntax' (which is currently selected and highlighted in green). The main content area has a header 'Python Syntax' with 'Previous' and 'Next' buttons. Below the header is a section titled 'Execute Python Syntax' containing the text: 'As we learned in the previous page, Python syntax can be executed by writing directly in the Command Line:' followed by a code block:

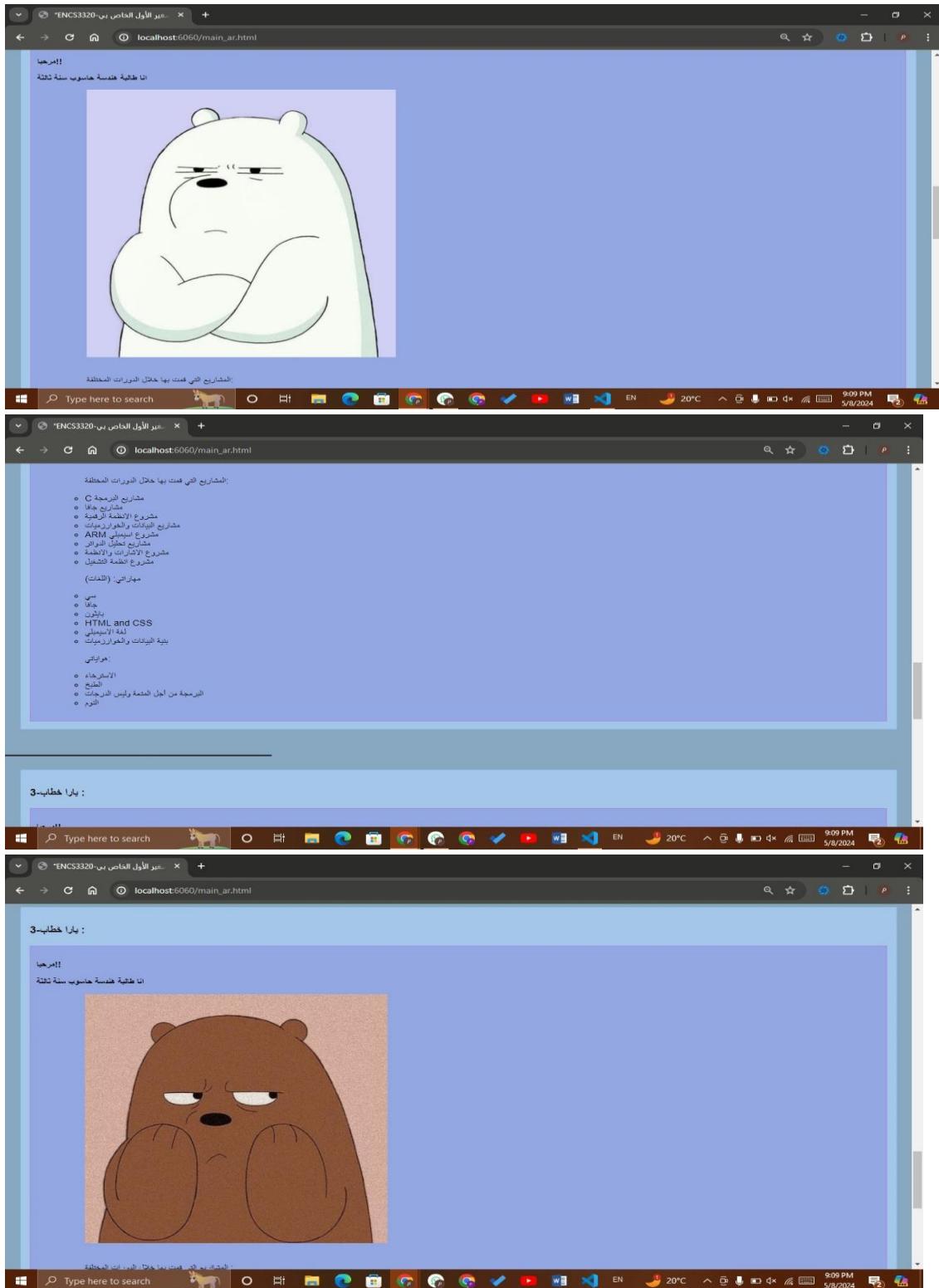
```
>>> print("Hello, World!")
Hello, World!
```

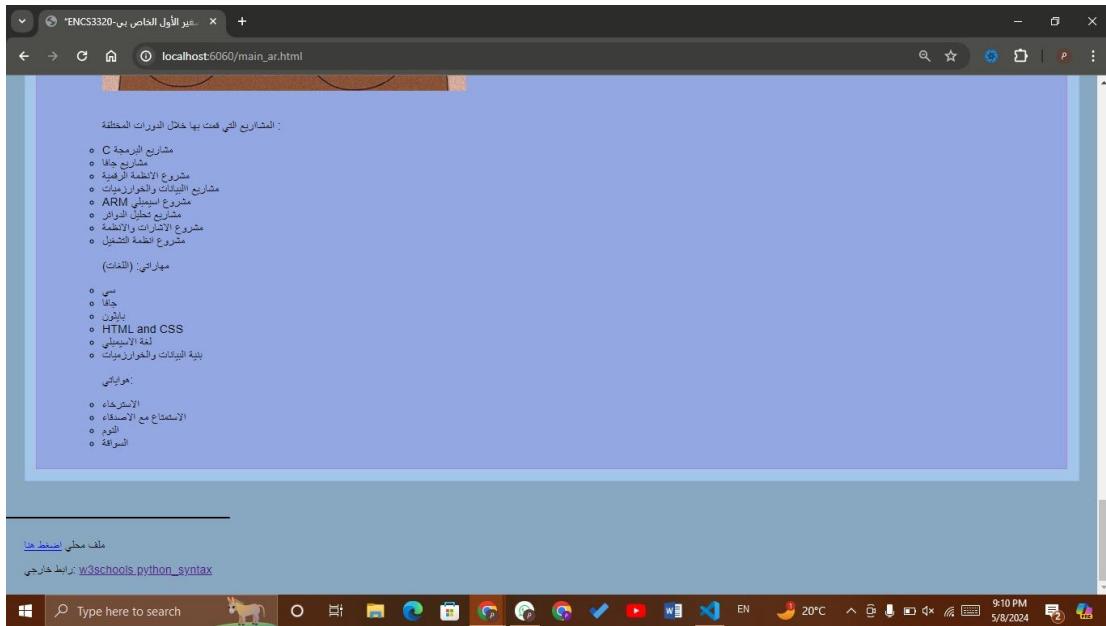
. To the right of the content area is a sidebar titled 'On this page' with links to 'Execute Python Syntax', 'Python Indentation', 'Python Variables', 'Python Comments', and 'Exercises'. A large advertisement banner on the right side of the page says 'BUILD YOUR CAREER. GET FULL ACCESS. SAVE 770\$' with a 'Start today' button. The Windows taskbar at the bottom shows various pinned icons and the system clock.

- 2- If the request is /ar then the server response with **main_ar.html** which is an Arabic version of **main_en.html**

Test from the computer:







```
The Requested File is: ar
('127.0.0.1', 54696)
IP: 127.0.0.1,Port: 54696
-----
GET /codestyle.css HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/124.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: text/css,*/*;q=0.1
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: style
Referer: http://localhost:6060/ar
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.8,es;q=0.7,es-ES;q=0.6
```

if the request is an .html file then the server should send the requested html file with Content-Type: text/html. You can use any html file. Make it general (not only for specific filename)

The screenshot shows a web browser window and a terminal window side-by-side.

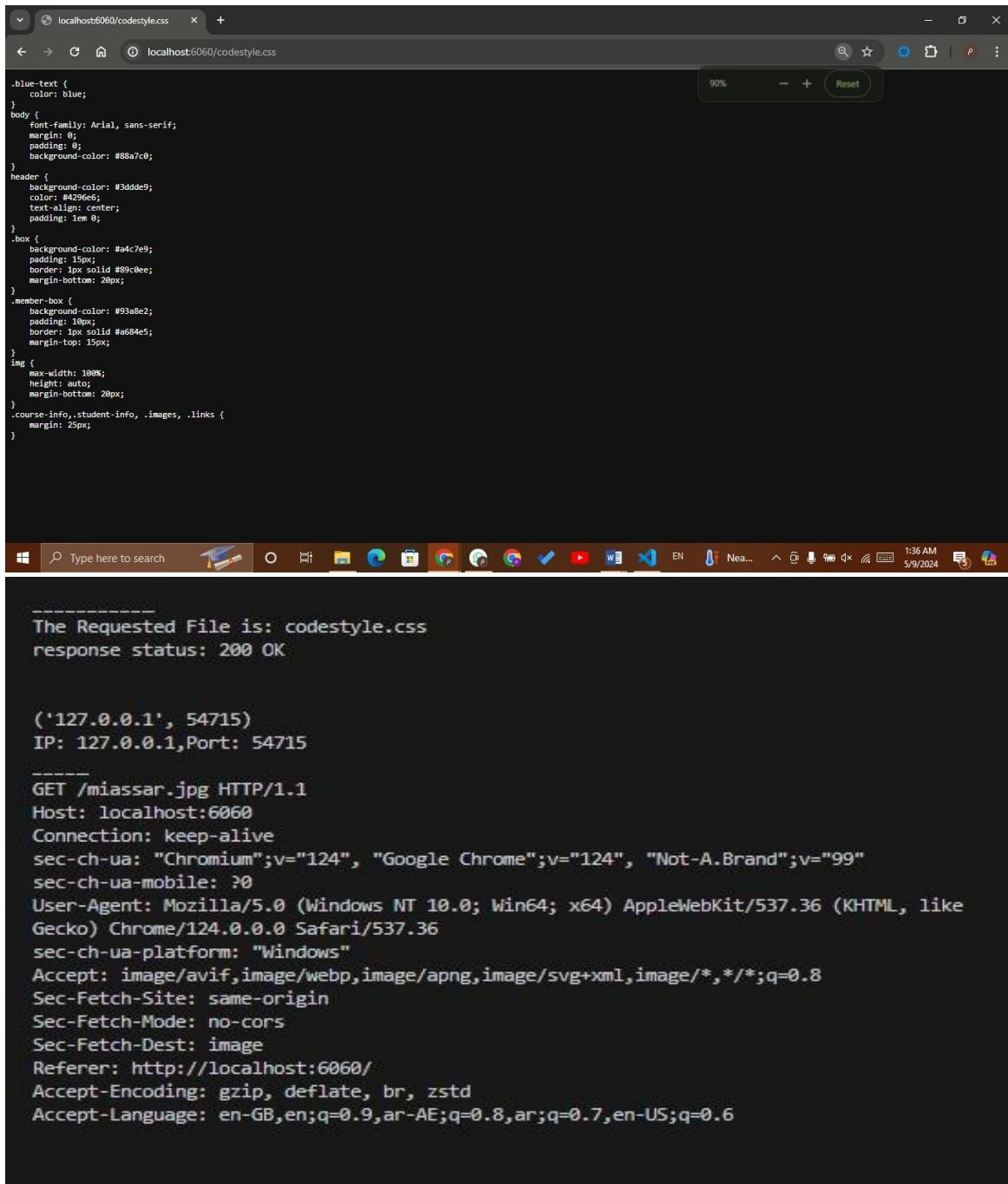
Web Browser Screenshot:

- Title bar: "ENCS3320- عبر الأول الخاص بي"
- Address bar: "localhost:6060/main_ar.html"
- Content:
 - "خادم الويب الصغير ENCS3320 ،مرحبا بكم في الدورة لدينا شبكات الكمبيوتر"
 - Section titled " أصحاب الويب":
 - المهندسة: ميسر شملة 1210519
 - المهندسة: تالين بياتنة 1211305
 - المهندسة: يارا خطاب 1210520

Terminal Screenshot:

```
-----  
The Requested File is: main_ar.html  
response status: 200 OK  
  
('127.0.0.1', 54876)  
IP: 127.0.0.1, Port: 54876  
  
----  
GET /codestyle.css HTTP/1.1  
Host: localhost:6060  
Connection: keep-alive  
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"  
sec-ch-ua-mobile: ?0  
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko  
) Chrome/124.0.0.0 Safari/537.36  
sec-ch-ua-platform: "Windows"  
Accept: text/css,*/*;q=0.1  
Sec-Fetch-Site: same-origin  
Sec-Fetch-Mode: no-cors  
Sec-Fetch-Dest: style  
Referer: http://localhost:6060/main_ar.html  
Accept-Encoding: gzip, deflate, br, zstd  
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

if the request is a .css file then the server should send the requested css file with Content-Type: text/css. You can use any CSS file. Make it general (not only for specific filename)



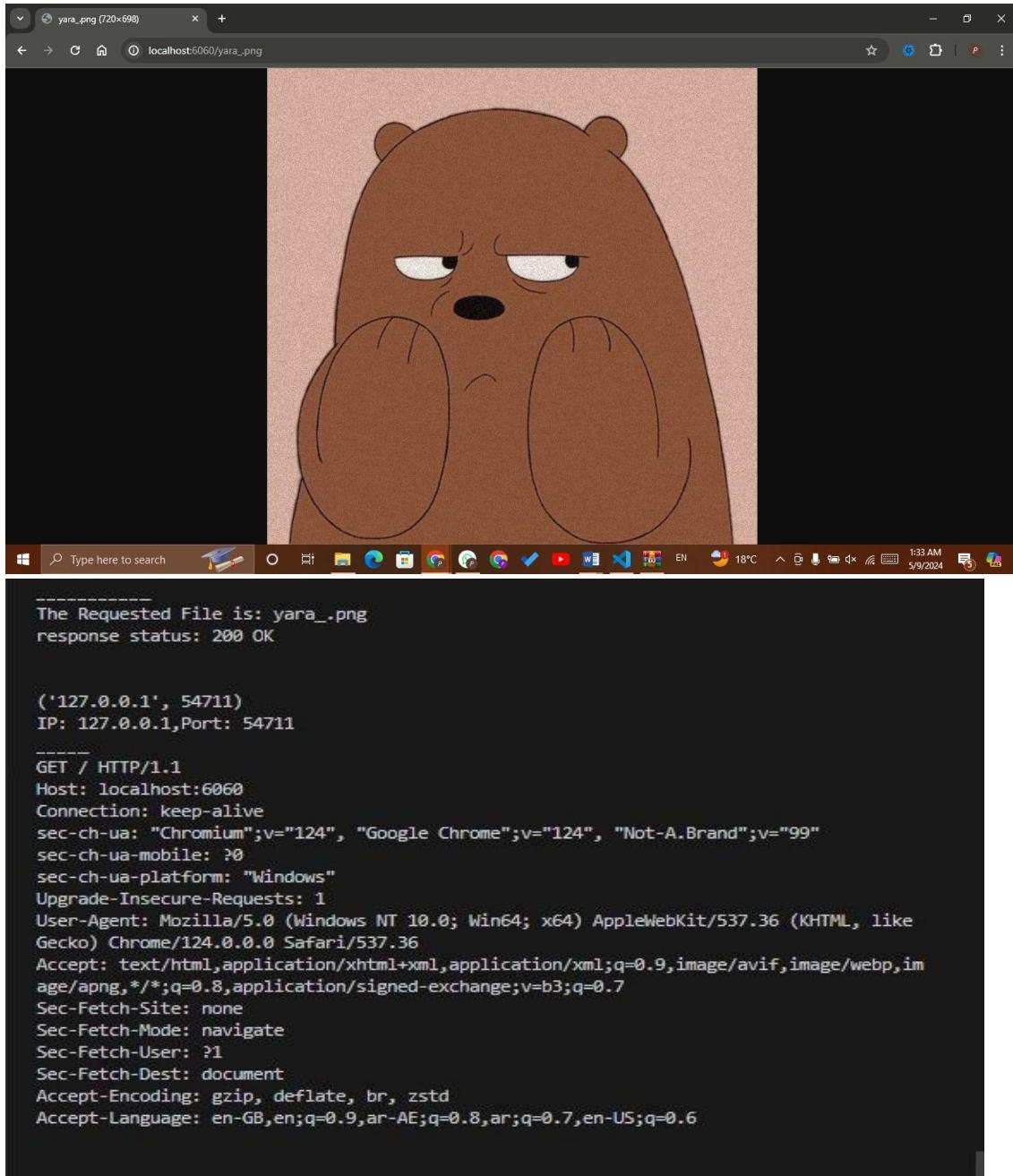
The Requested File is: codestyle.css
response status: 200 OK

```
('127.0.0.1', 54715)
IP: 127.0.0.1, Port: 54715

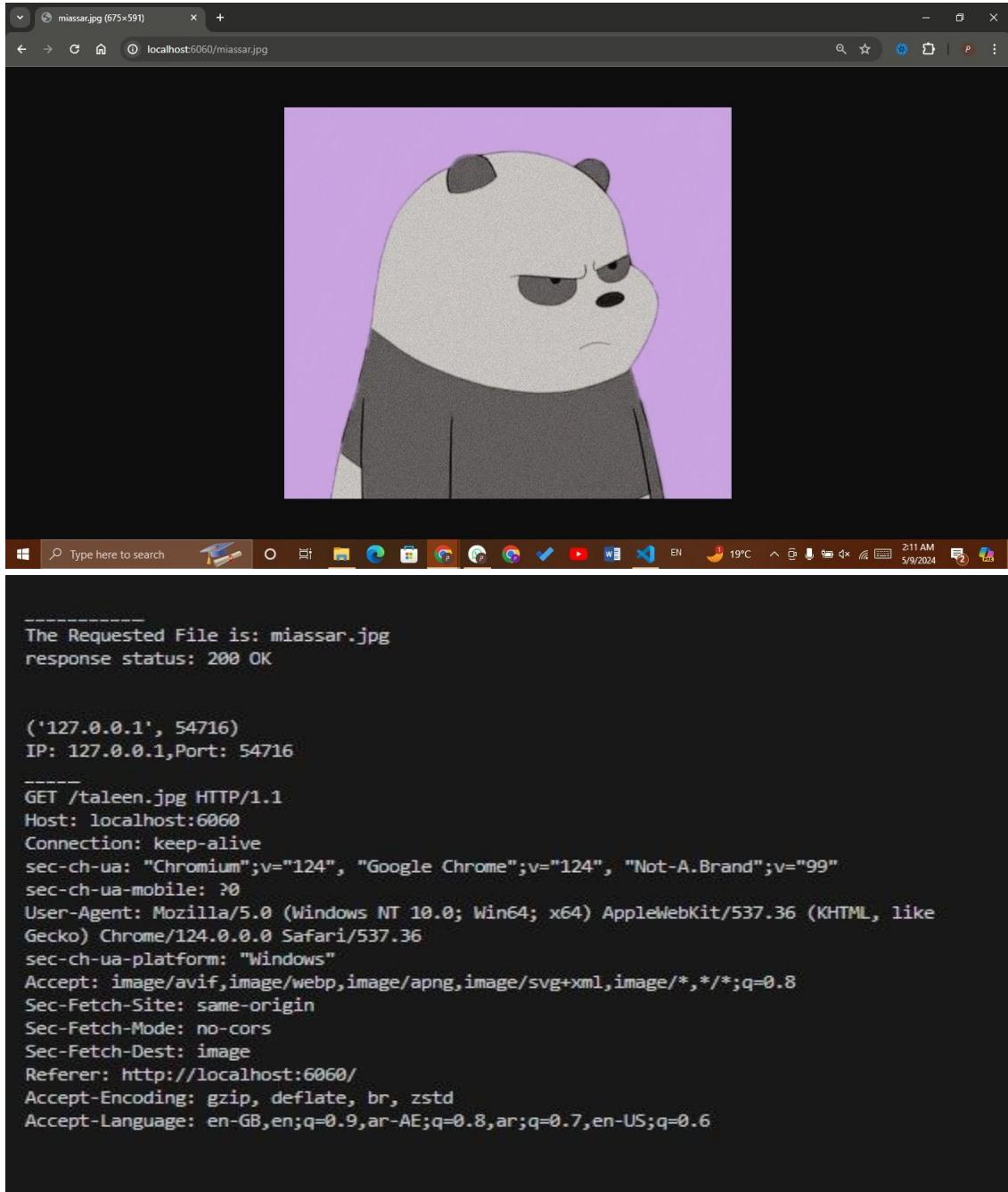
----  

GET /miassar.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:6060/
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

if the request is a .png then the server should send the png image with Content-Type: image/png. You can use any image. Make it general (not only for specific filename)

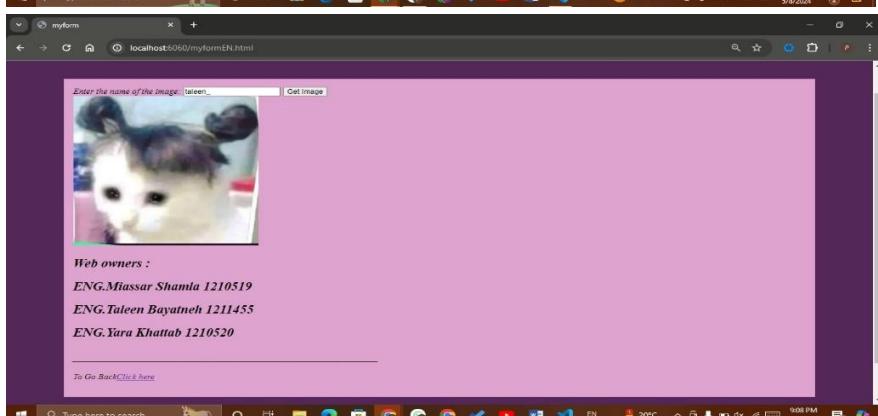
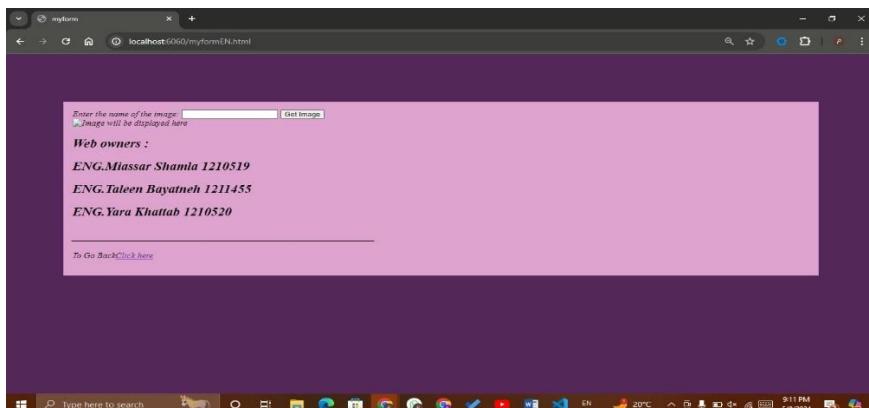


if the request is a .jpg then the server should send the jpg image with Content-Type: image/jpeg. You can use any image. Make it general (not only for specific filename)



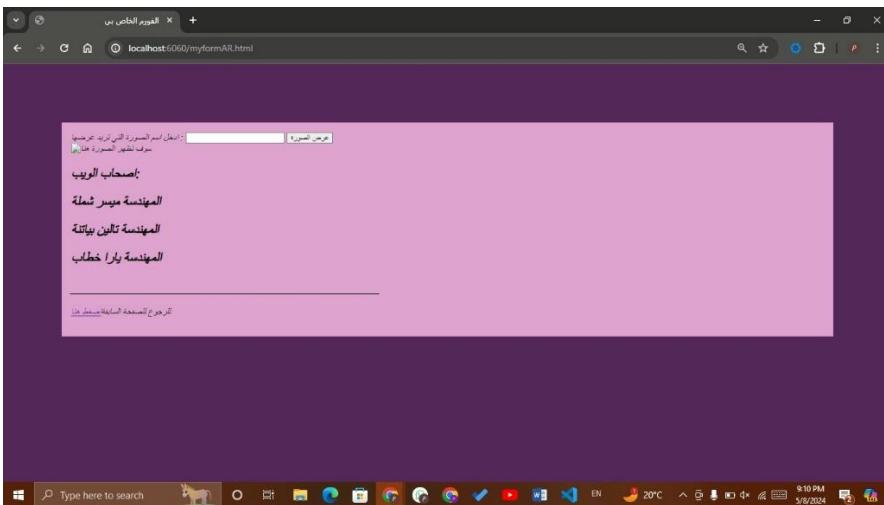
Use myform.html to get image by typing the name of the image in a box

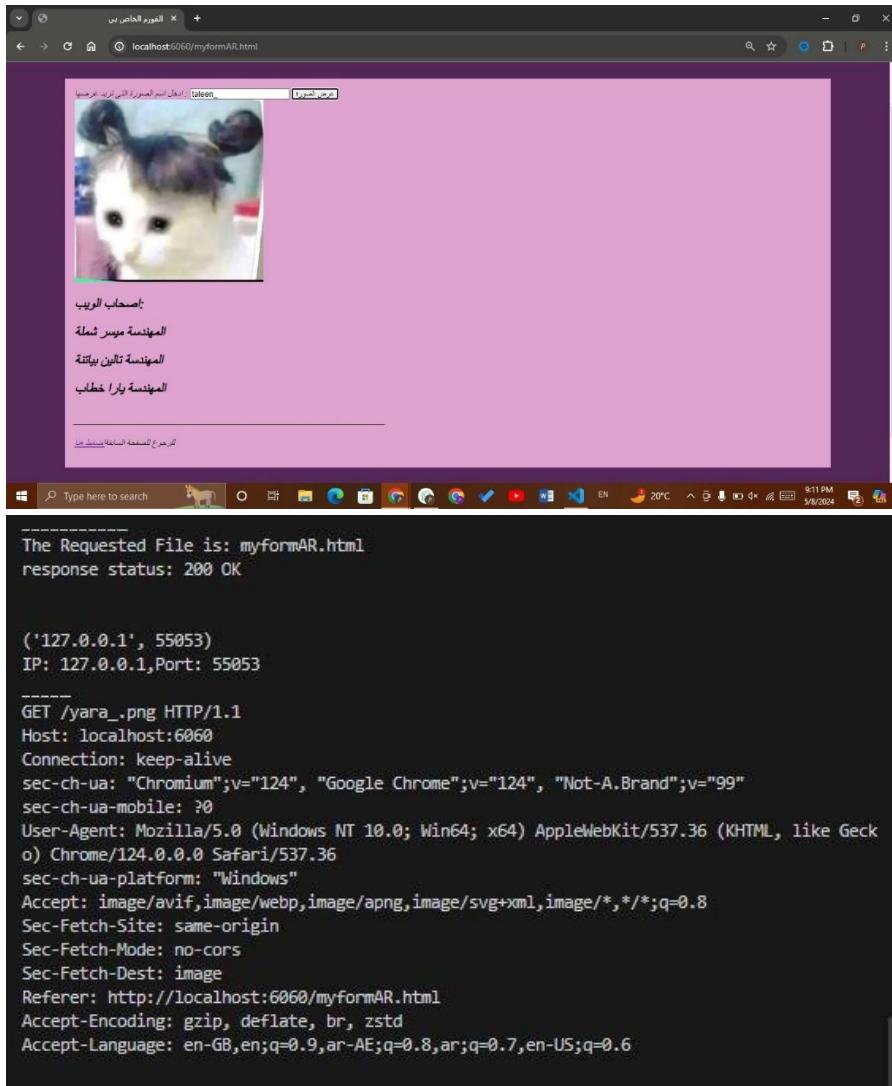
en:



```
-----  
The Requested File is: myformEN.html  
response status: 200 OK  
  
('127.0.0.1', 54978)  
IP: 127.0.0.1, Port: 54978  
  
-----  
GET /yara_.png HTTP/1.1  
Host: localhost:6060  
Connection: keep-alive  
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"  
sec-ch-ua-mobile: ?0  
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)  
Chrome/124.0.0.0 Safari/537.36  
sec-ch-ua-platform: "Windows"  
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8  
Sec-Fetch-Site: same-origin  
Sec-Fetch-Mode: no-cors  
Sec-Fetch-Dest: image  
Referer: http://localhost:6060/myformEN.html  
Accept-Encoding: gzip, deflate, br, zstd  
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

ar:

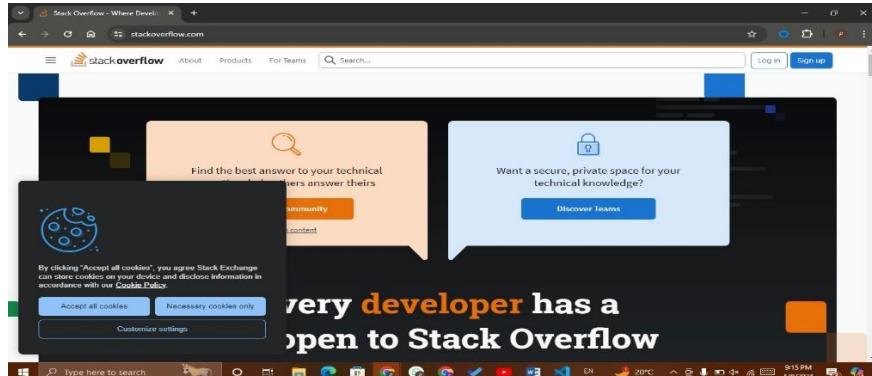
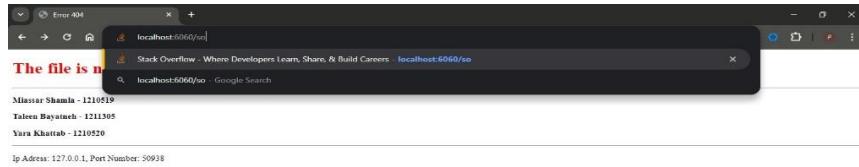




Use the status code **307 Temporary Redirect** to redirect the following

- If the request is **/so** then redirect to stackoverflow.com website
- If the request is **/itc** then redirect to itc.birzeit.edu website

a:



```
The Requested File is: so
('127.0.0.1', 54613)
IP: 127.0.0.1, Port: 54613

GET /itc HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"
sec-ch-ua-mobile: 0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

b:

The screenshot shows a Windows desktop environment with three main windows:

- Top Left Window:** A browser window titled "localhost:6060/itc" showing a list of student names and IDs:
 - المهندسة:ميسر شملة 1210519
 - المهندسة:بنالين بياتنة 12111305
 - المهندسة:نيراء خطاب 1210520
- Top Right Window:** A browser window titled "Instructional Technology Center localhost:6060/itc" showing a slide with Arabic text and a cartoon panda image.
- Bottom Window:** A browser window titled "itc.birzeit.edu/register/" showing a landing page for registration. It features the Birzeit University logo and a photograph of three people working with a robotic arm in a lab setting. The text on the slide reads: "THE INTEGRATION BETWEEN ROBOTICS AND PLCs".

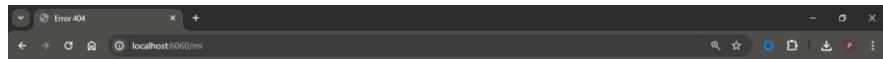
At the bottom of the screen, there is a terminal window displaying the following text:

```
The Requested File is: itc
('127.0.0.1', 54623)
IP: 127.0.0.1,Port: 54623

GET /mi HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

If the request is wrong or the file doesn't exist the server should return a simple HTML webpage that contains
(Content-Type: text/html)

- 1- "HTTP/1.1 404 Not Found" in the response status
- 2- "Error 404" in the title
- 3- "The file is not found" in the body in red
- 4- Your names and IDs in **Bold**
- 5- The IP and port number of the client



The file is not found

Miassar Shamla - 1210519

Taleen Bayatneh - 1211305

Yara Khattab - 1210520

Ip Adress: 127.0.0.1, Port Number: 51847

A screenshot of a Windows taskbar at the top. Below it is a terminal window with a black background. The terminal output shows the following:

```
The Requested File is: mi
mi not found
Response status:HTTP/1.1 404 Not Found
('127.0.0.1', 51847)
IP: 127.0.0.1,Port: 51847

-----
GET /mi HTTP/1.1
Host: localhost:6060
Connection: keep-alive
Cache-Control: max-age=0
sec-ch-ua: "Chromium";v="124", "Google Chrome";v="124", "Not-A.Brand";v="99"

sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```

Note: option 11 already done above

Test from another phone:

En/

2.168.1.103:6060 + ⌂ :

"Welcome to our course
Computer Networks, ENCS3320-
tiny webserver"

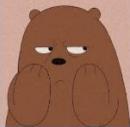
Web owners :
ENG.Miassar Shamla 1210519
ENG.Taleen Bayatneh 1211305
ENG.Yara Khattab 1210520

1-Miassar Shamla:
Hi!!!
I'm a Third-year Computer Engineering student.


projects i have done during different courses:
◦ C programming projects
◦ java projects
◦ Digital systems project
◦ Data structure and algorithms projects
◦ ARM Assembly projects
◦ Circuit analysis projects
◦ Signals and Systems project
◦ operating systems projet
My skills: (languages)
◦ C

2-Taleen Bayatneh:
Hi!!!
I'm a Third-year Computer Engineering student.


projects i have done during different courses:
◦ C programming projects
◦ java projects
◦ Digital systems project
◦ Data structure and algorithms projects
◦ ARM Assembly projects
◦ Circuit analysis projects
◦ Signals and Systems project
◦ operating systems projet

3-Yara Khattab:
Hi!!!
I'm a Third-year Computer Engineering student.


projects i have done during different courses:
◦ C programming projects
◦ java projects
◦ Digital systems project
◦ Data structure and algorithms projects
◦ ARM Assembly projects
◦ Circuit analysis projects
◦ Signals and Systems project
◦ operating systems projet

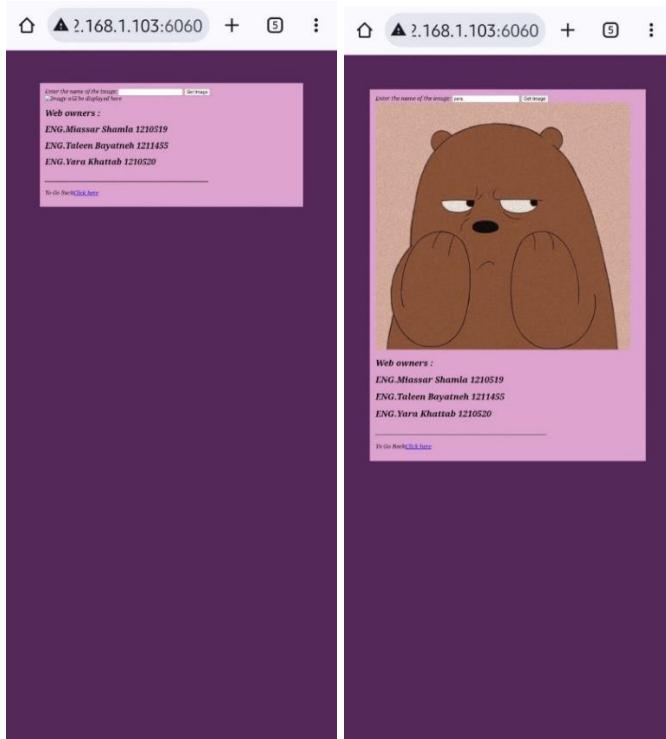
Hi!!
I'm a Third-year Computer Engineering student.


projects i have done during different courses:
◦ C programming projects
◦ java projects
◦ Digital systems project
◦ Data structure and algorithms projects
◦ ARM Assembly projects
◦ Circuit analysis projects
◦ Signals and Systems project
◦ operating systems projet
My skills: (languages)
◦ C
◦ Java
◦ Python
◦ HTML and CSS
◦ Assembly language
◦ Data structure and algorithms
My hobbies:
◦ relaxing
◦ chill out with frinds
◦ sleeping
◦ driving

Local HTML File: [Click here](#)
External Link: [w3schools](#)
[python syntax](#)

```
The Requested File is: main_en.html
('192.168.1.104', 33686)
IP: 192.168.1.104, Port: 33686

-----
GET /codestyle.css HTTP/1.1
Host: 192.168.1.103:6060
Connection: keep-alive
User-Agent: Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Mobile Safari/537.36
Accept: text/css,*/*;q=0.1
Referer: http://192.168.1.103:6060/main_en.html
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```



```
-----  
The Requested File is: myformEN.html  
response status: 200 OK  
  
( '192.168.1.104', 41822 )  
IP: 192.168.1.104, Port: 41822  
  
-----  
GET /myformEN.html HTTP/1.1  
Host: 192.168.1.103:6060  
Connection: keep-alive  
Upgrade-Insecure-Requests: 1  
User-Agent: Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Mobile Safari/537.36  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7  
Referer: http://192.168.1.103:6060/  
Accept-Encoding: gzip, deflate  
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6  
  
-----
```

1: مرحبا بك في الدورة لدينا شركات " خادم الويب-الكمبيوتر ENCS3320 ، المبرمجون الصغار"

أصحاب الويب :

- المهندسة: ميسر شملة 1210519
- المهندسة: تالين بياتنة 1211305
- المهندسة: يارا خطاب 1210520

ميسر شملة-1 :

مرحبا!!

انا طالبة هندسة حاسوب سنة ثالثة



المشاريع التي قمت بها خلال الدورات

المشاريع التي قمت بها خلال الدورات :

- مشاريع البرمجة C
- مشاريع جافا
- مشروع الانظمة الرقمية
- مشاريع البيانات والخوارزميات
- مشروع اسيمولي ARM
- مشاريع تحليل الدوائر
- مشروع الاشارات والانظمة
- مشروع انظمة التشغيل
- مهاراتي: (اللغات)
- سي
- جافا
- بایرون
- HTML and CSS
- لغة الاسيمولي
- بنية البيانات والخوارزميات
- هوبياتي
- الاسترخاء
- الطبع
- الاستمتاع مع الاصدقاء
- البرمجة من أجل المتعة وليس الدرجات
- النوم
- الأكل مع الاصدقاء

تالين بياتنة-2 :

مرحبا!!

انا طالبة هندسة حاسوب سنة ثالثة



المشاريع التي قمت بها خلال الدورات

المشاريع التي قمت بها خلال الدورات:

- مشاريع البرمجة C
- مشاريع جافا
- مشروع الانظمة الرقمية
- مشاريع البيانات والخوارزميات
- مشروع اسيمولي ARM
- مشاريع تحليل الدوائر
- مشروع الاشارات والانظمة
- مشروع انظمة التشغيل
- مهاراتي: (اللغات)
- سي
- جافا
- بایرون
- HTML and CSS
- لغة الاسيمولي
- بنية البيانات والخوارزميات
- هوبياتي
- الاسترخاء
- الطبع
- الاستمتاع مع الاصدقاء
- البرمجة من أجل المتعة وليس الدرجات
- النوم
- السواقة

يارا خطاب-3 :

مرحبا!!

انا طالبة هندسة حاسوب سنة ثالثة



The Requested File is: main_ar.html
response status: 200 OK

```

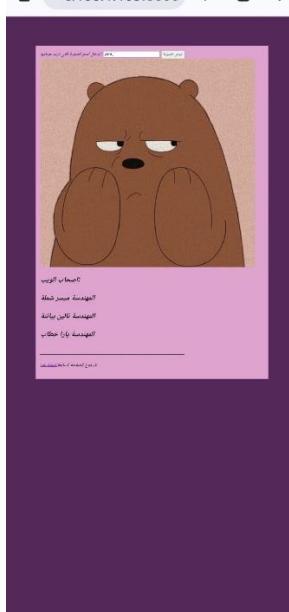
('192.168.1.104', 56480)
IP: 192.168.1.104,Port: 56480
-----
GET /codestyle.css HTTP/1.1
Host: 192.168.1.103:6060
Connection: keep-alive
User-Agent: Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Mobile Safari/537.36
Accept: text/css,*/*;q=0.1
Referer: http://192.168.1.103:6060/main_ar.html
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
-----
```



A screenshot of a web browser window titled "2.168.1.103:6060". The page displays a form with the following text in Arabic:

العنوان المطلوب
العنوان المطلوب
المقدمة ممدوح
المقدمة باسم
المقدمة بارا خلاب

لرجوع المقدمة لـ ممدوح



A screenshot of a web browser window titled "2.168.1.103:6060". The page displays a form with the following text in Arabic:

العنوان المطلوب
العنوان المطلوب
المقدمة ممدوح
المقدمة باسم
المقدمة بارا خلاب

لرجوع المقدمة لـ ممدوح

The Requested File is: myformAR.html
response status: 200 OK

```
('192.168.1.104', 37906)
IP: 192.168.1.104, Port: 37906
-----
('192.168.1.104', 54988)
IP: 192.168.1.104, Port: 54988
-----
GET / HTTP/1.1
Host: 192.168.1.103:6060
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.0.0 Mobile Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,ar-AE;q=0.8,ar;q=0.7,en-US;q=0.6
```