



Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

ENCS3321

COMPUTER NETWORKS

SECOND SEMESTER PROJECT 2023-2024

Prepared by:

Jana Sawalmeh **1212467**

Lana Mahmoud Ali Musaffer **1210455**

Tariq Al-Atrash **1210122**

Instructor: **Dr. Abdalkarim Awad**

Section: **2**

Abstract

This project aims to create network applications using socket programming. The goals include constructing a UDP client-server setup for message broadcasting and developing a flexible web server capable of handling diverse content types and requests. In the first part, we explore network tools such as ping and nslookup, along with network testing using Wireshark, to comprehend network connectivity. The second part elaborates on setting up a UDP communication system, allowing clients within a network to exchange messages, showcasing real-time data transmission. In the third part, the focus shifts to building a web server that caters to various client requests, supports multiple content types, and effectively manages redirections and errors. Through this project, we enhance our practical grasp of network communications and establish a groundwork for more advanced network applications.

Theory

Computer networks:

A computer network is a set of computers sharing resources located on or provided by network nodes. Computers use common communication protocols over digital interconnections to communicate with each other. These interconnections are made up of telecommunication network technologies based on physically wired.

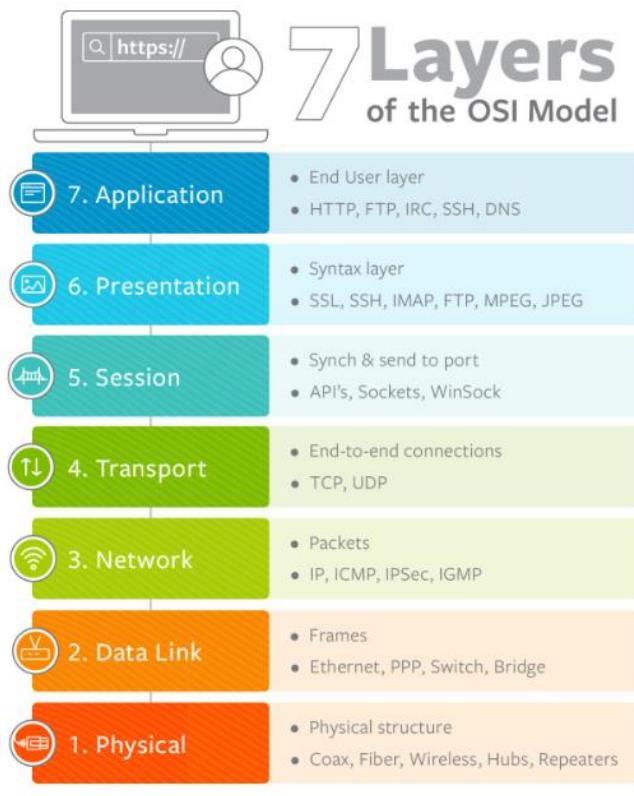


Figure 1.1 [1]

TCP and UDP

TCP ➔ The TCP protocol (Transmission Control Protocol) is the most widely used transport protocol in the world and is one of the core components of the Internet. For this reason IP stacks like emNet are often called TCP/IP stacks while they are not limited to TCP or the IP protocol only. [2]

UDP ➔ is one of the core communication protocols of the Internet protocol suite used to send messages (transported as datagrams in packets) to other hosts on an Internet Protocol (IP) network. Within an IP network, UDP does not require prior communication to set up communication channels or data paths.[2]

TCP vs UDP

TCP vs UDP

- | | |
|--|---|
| <ul style="list-style-type: none">• Connected• State Memory• Byte Stream• Ordered Data Delivery• Reliable• Error Free• Handshake• Flow Control• Relatively Slow• Point to Point• Security: SSL/TLS | <ul style="list-style-type: none">• Connectionless• Stateless• Packet/Datagram• No Sequence Guarantee• Lossy• Error Packets Discarded• No Handshake• No Flow Control• Relatively Fast• Supports Multicast• Security: DTLS |
|--|---|

Figure1.2[2]

Routers:

A router is a device that connects two or more packet-switched networks or subnetworks. It serves two primary functions: managing traffic between these networks by forwarding data packets to their intended IP addresses, and allowing multiple devices to use the same Internet connection. [3]

There are several types of routers, but most routers pass data between LANs (local area networks) and WANs (wide area networks). A LAN is a group of connected devices restricted to a specific geographic area. A LAN usually requires a single router.[3]

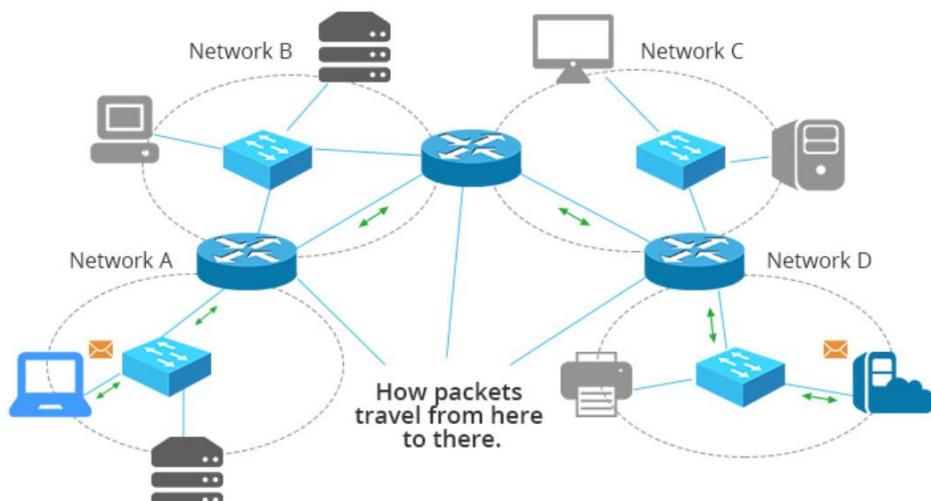


Figure 1.3 [3]

HTTP:

Hypertext Transfer Protocol (HTTP) is an application layer protocol in the Internet protocol suite model for distributed, collaborative, hypermedia information systems.^[1] HTTP is the foundation of data communication for the World Wide Web, where hypertext documents include hyperlinks to other resources that the user can easily access, for example by a mouse click or by tapping the screen in a web browser. Development of HTTP was initiated by Tim Berners-Lee at CERN in 1989 and summarized in a simple document describing the behavior of a client and a server using the first HTTP version, named 0.9.^[2] That version was subsequently developed, eventually becoming the public 1.0. Development of early HTTP Requests for Comments (RFCs) started a few years later in a coordinated effort by the Internet Engineering Task Force (IETF) and the World Wide Web Consortium (W3C), with work later moving to the IETF.^[4]

Socket:

A network socket is a software structure within a network node of a computer network that serves as an endpoint for sending and receiving data across the network. The structure and properties of a socket are defined by an application programming interface (API) for the networking architecture. Sockets are created only during the lifetime of a process of an application running in the node.

Because of the standardization of the TCP/IP protocols in the development of the Internet, the term *network socket* is most commonly used in the context of the Internet protocol suite, and is therefore often also referred to as Internet socket. In this context, a socket is externally identified to other hosts by its socket address, which is the triad of transport protocol, IP address, and port number.

The term *socket* is also used for the software endpoint of node-internal inter-process communication (IPC), which often uses the same API as a network socket.^[5]

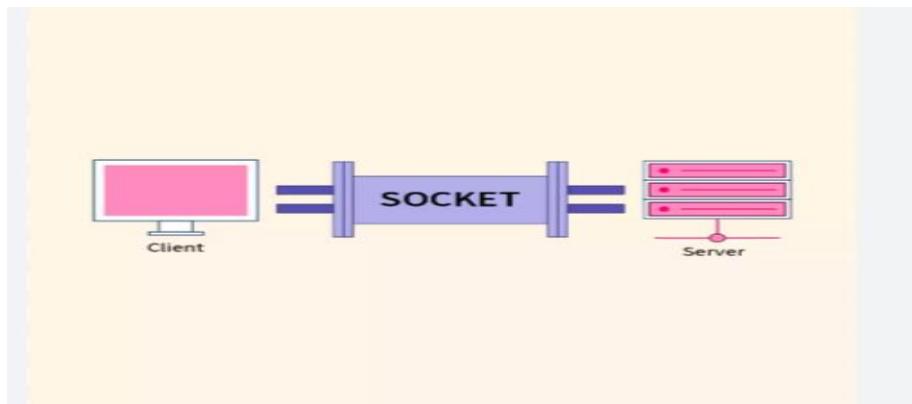


Figure 1.4

Table of Contents

1. Abstract

2. Theory

2.1 computer network

2.2 UDP and TCP

2.3 Routers

2.4 HTTP

2.5 Sockets

3. Part_1

3.1 network essential command

3.2 running the commands

3.3 using Wireshark for capturing some DNS messages

4. Part_2

4.1 construct a peer to peer connection through python

4.2 each peer sends an receive messages

5. Part_3

5.1 build a web page using HTML and css

5.2 transfer from one page to another

5.3 perform HTTP request on using terminal

6. conclusion

7. References

PART #1

1. A few network essential commands

Ping ➔ A tool for sending packets with a specific size and to set a timer to measure the time it takes for packets to travel to its destination, and before it sends the packets, it determine whether the IP address is accessible or not.

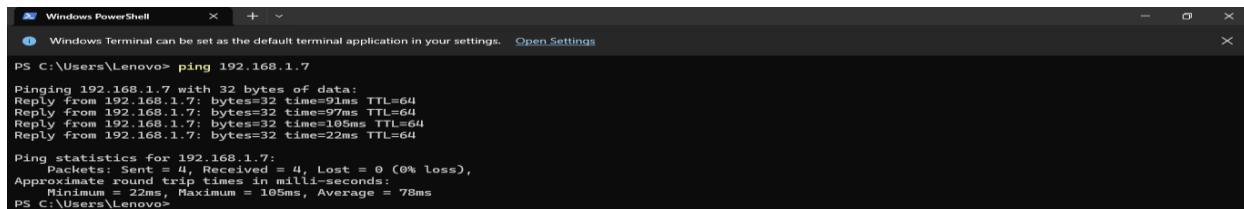
Tracert ➔ For displaying the visited routers and the taken path from the sender device to a specified destination along the packet's journey, by sending incremental TTL for each router until it reaches its destination.

Nslookup ➔ used to query the Domain Name System (DNS) to obtain domain name or IP address mapping, DNS records, and other DNS-related information. It's available on most operating systems, including Windows, macOS, and Linux.

Telnet ➔ used to establish a command-line connection between two computers over a network. It was one of the first Internet protocols and is still widely used for various purposes, although it has largely been replaced by more secure protocols like SSH (Secure Shell).

2. Running the commands

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



```
Windows PowerShell
PS C:\Users\Lenovo> ping 192.168.1.7

Pinging 192.168.1.7 with 32 bytes of data:
Reply from 192.168.1.7: bytes=32 time=91ms TTL=64
Reply from 192.168.1.7: bytes=32 time=97ms TTL=64
Reply from 192.168.1.7: bytes=32 time=105ms TTL=64
Reply from 192.168.1.7: bytes=32 time=22ms TTL=64

Ping statistics for 192.168.1.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 22ms, Maximum = 105ms, Average = 78ms
PS C:\Users\Lenovo>
```

The four reply messages show that the command "ping 192.168.1.7" was successfully sent to the target device and got responses.

The TTL (Time To Live) for each response received with a ping of 32 bytes is 64. These are normal TTLs for a local network, meaning that there is little network delay and good device connectivity.

The fact that the TTL value of 64 has not dramatically dropped from a maximum of 128 or 255 (typical starting values depending on the operating system) indicates that the devices are either on the same network or quite close to one another in terms of network.

The data (Sent = 4, Received = 4, Lost = 0) show that there was no packet loss during this test, which indicates

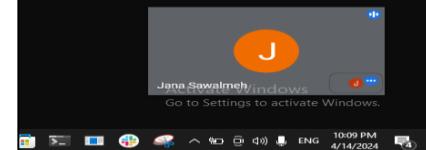
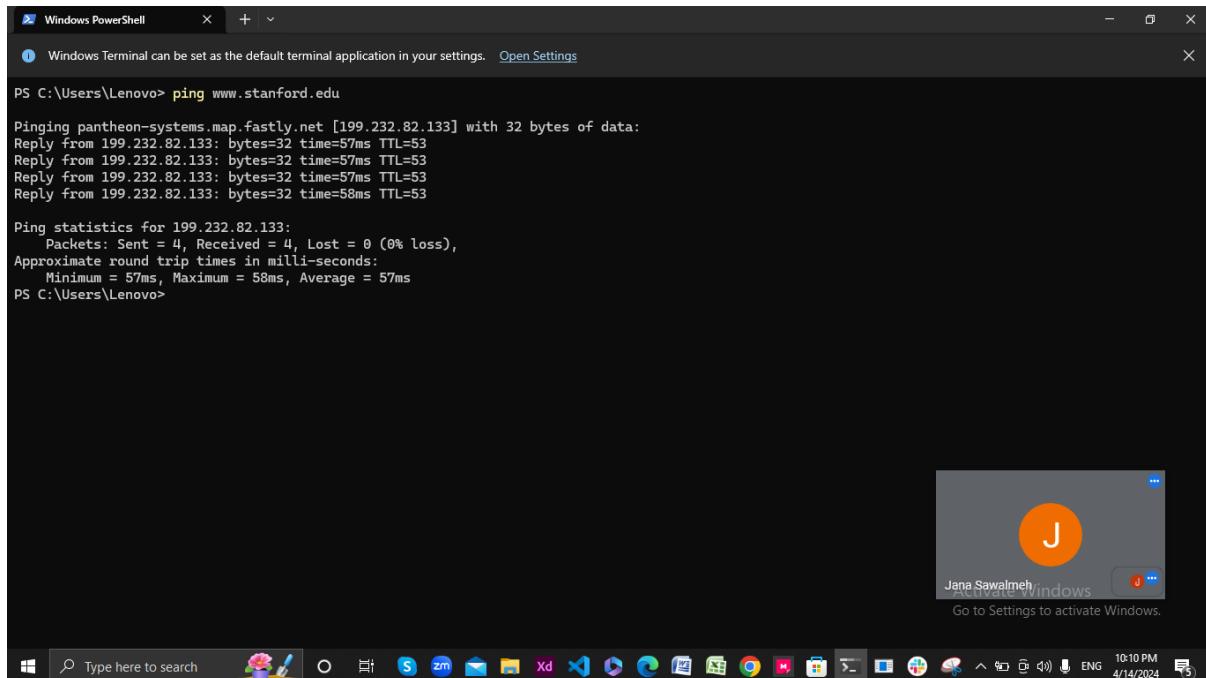


Figure 1.5

b. ping www.stanford.edu



```
PS C:\Users\Lenovo> 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=57ms TTL=53
Reply from 199.232.82.133: bytes=32 time=57ms TTL=53
Reply from 199.232.82.133: bytes=32 time=57ms TTL=53
Reply from 199.232.82.133: bytes=32 time=58ms TTL=53

Ping statistics for 199.232.82.133:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 57ms, Maximum = 58ms, Average = 57ms
PS C:\Users\Lenovo>
```

Figure 1.6

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

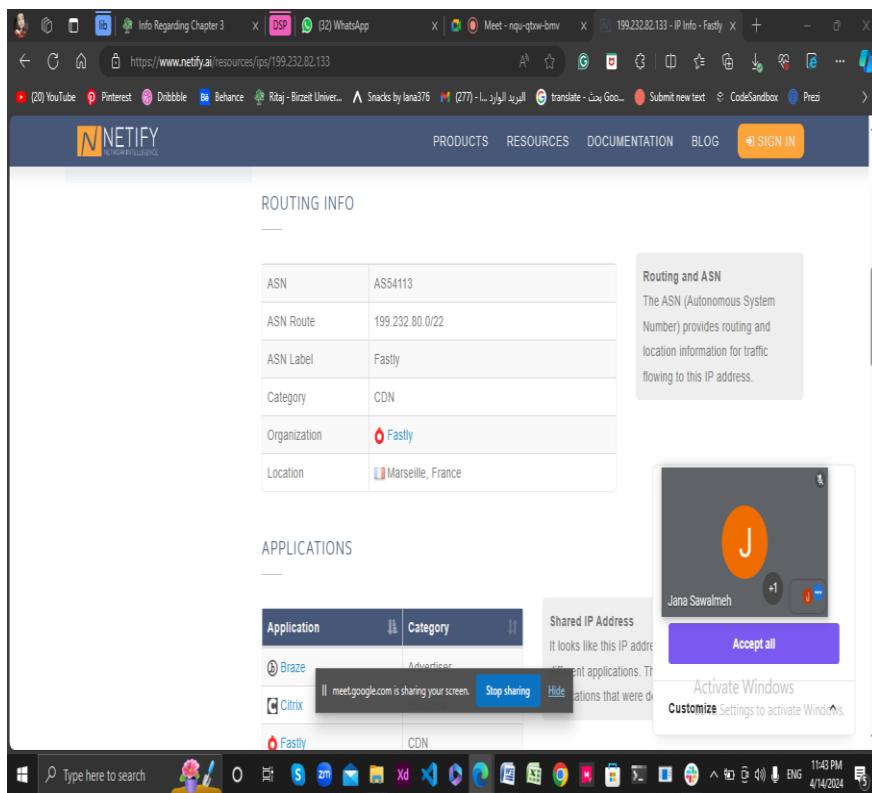


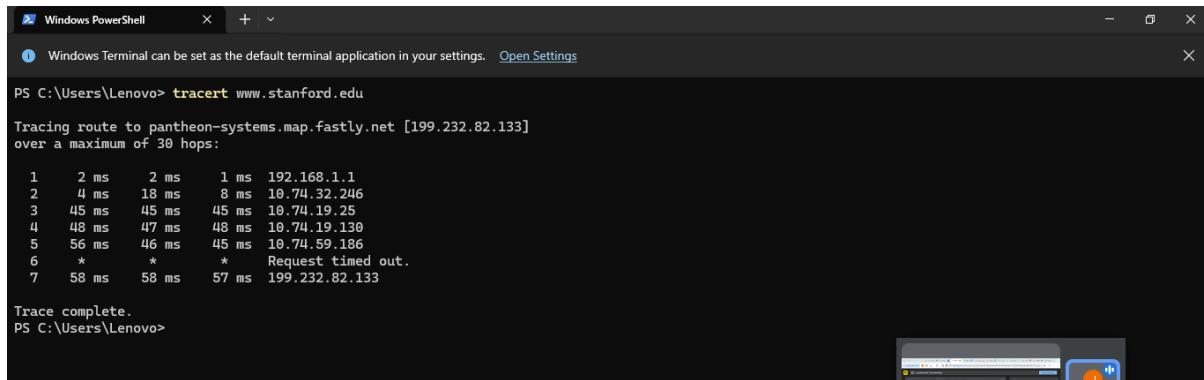
Figure 1.7

As shown in the above figure, at the first site we think that the response is from the USE because most of the servers are located there, but when we check Fastly website we see that the response location is from Marseille, France not USA.

[199.232.82.133 - IP Info - Fastly \(netify.ai\)](https://199.232.82.133-ip.info-fastly.netify.ai)

[Fastly network map | Fastly](#)

d. tracert www.stanford.edu



```
PS C:\Users\Lenovo> tracert www.stanford.edu
Tracing route to pantheon-systems.map.fastly.net [199.232.82.133]
over a maximum of 30 hops:
 1  2 ms    2 ms    1 ms  192.168.1.1
 2  4 ms    18 ms   8 ms  10.74.32.246
 3  45 ms   45 ms   45 ms  10.74.19.25
 4  48 ms   47 ms   48 ms  10.74.19.130
 5  56 ms   46 ms   45 ms  10.74.59.186
 6  *       *       *       Request timed out.
 7  58 ms   58 ms   57 ms  199.232.82.133

Trace complete.
PS C:\Users\Lenovo>
```

The traceroute command provides a detailed path from a local network to the IP address 199.232.82.133, which hosts Stanford University's website via Fastly's CDN. The trace details the journey through several network nodes.

Initial Hops: The trace begins within a local network, with the first hop at the local router (IP address 192.168.1.1) showing minimal latency of 1-2 ms, indicating immediate local network responsiveness.

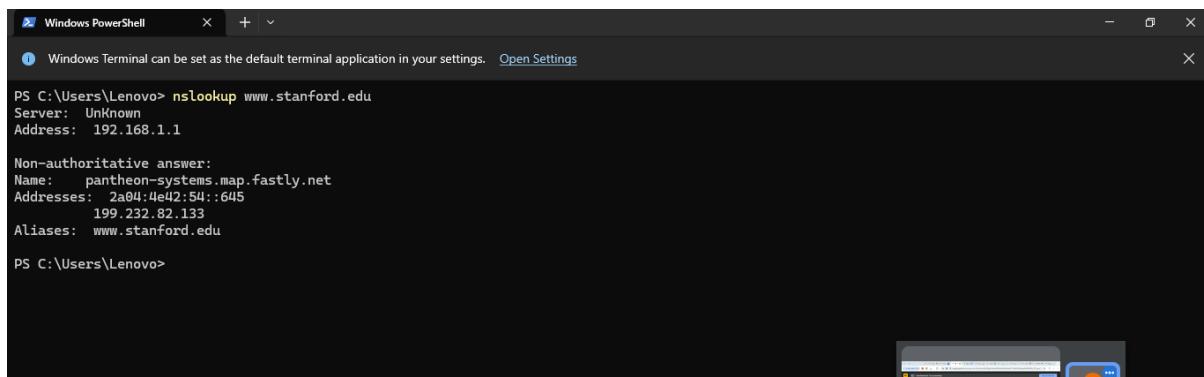


Activate Windows
Go to Settings to activate Windows.



Figure 1.8

e. nslookupwww.stanford.edu



```
PS C:\Users\Lenovo> 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

PS C:\Users\Lenovo>
```

Local DNS Server: The query was processed by the local DNS server at IP address 192.168.1.1, referred to as "Unknown" since its name could not be resolved.

DNS Resolution: The domain www.stanford.edu is linked to the CDN provider Fastly, as indicated by the resolved name pantheon-systems.map.fastly.net.

This demonstrates Stanford's use of a content delivery network to manage web traffic effectively.

IP Addresses: The lookup returned both IPv4 and IPv6 addresses:

IPv4: 199.232.82.133

IPv6: 2a04:4e42:54::645



Activate Windows
Go to Settings to activate Windows.

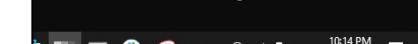


Figure 1.9

3. Using Wireshark for capturing some DNS messages

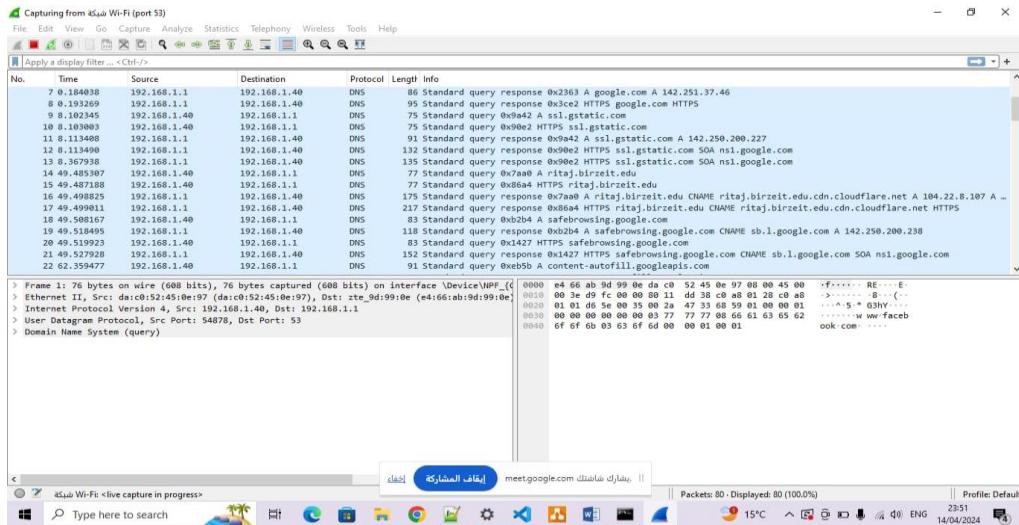
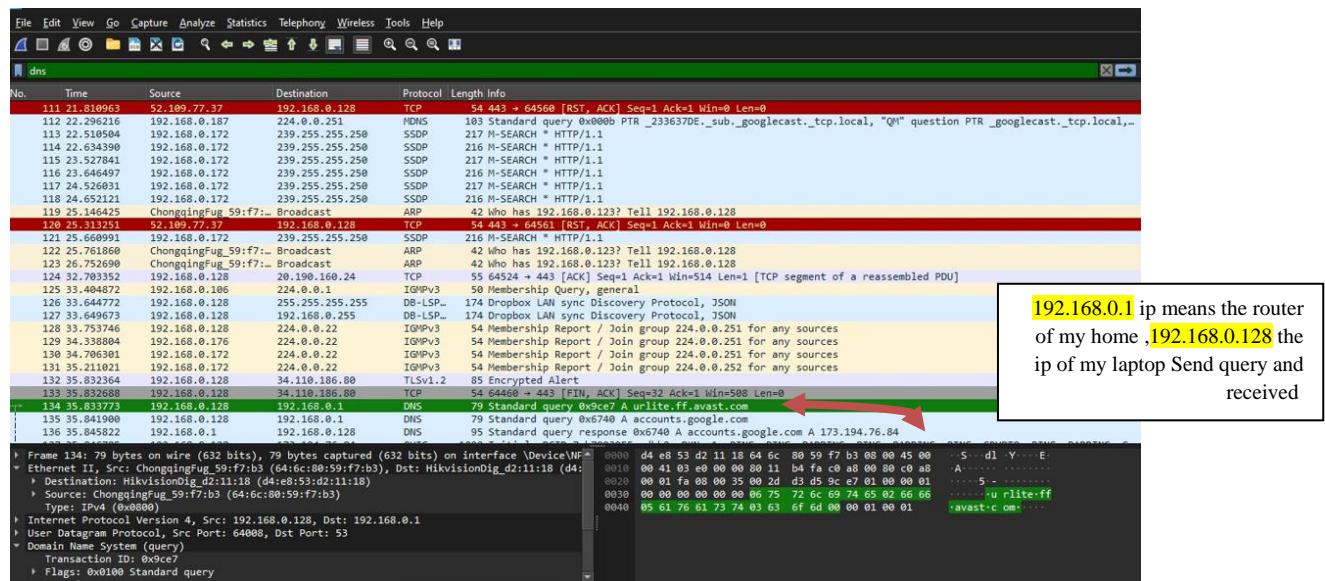


Figure 1.10

Did the operation from two different Laptops process {open wireshark wi_fi , cmd ipconfig / flushdns , open website , then write dns on wireshark}



Wireshark is an open-source network protocol analyzer widely used for capturing, analyzing, and troubleshooting network traffic. Developed by the Wireshark community, it provides a detailed and real-time view of data moving across a network, helping users understand the communication patterns between devices

Here we have to use the wireshark to capture some DNS messages, we chose Wi-Fi the network interface that corresponds to the network we want to monitor, to focus on DNS messages, we applied a display filter (DNS).

PART #2

code

```
# Part 2 Jana,Lana,Tariq
import socket
import threading
import datetime
|
PORT = 5051
# broadcast_ip we use Radmin vpn online network not the one from my laptop ipconfig
broadcast_ip = '26.255.255.255'
#this line retrieves the local IP address of the machine by getting the hostname,
#resolving it to IP addresses, and selecting the first IP address from the list.
local_ip = socket.gethostbyname_ex(socket.gethostname())[2][0]
first_name = input("Enter your first name: ")
last_name = input("Enter your last name: ")
full_name = f"{first_name} {last_name}"
buffer=1024 # max num of data will be received

def bring_time_of_now():
    # build in function that bring the time now
    return datetime.datetime.now().strftime("%H:%M:%S")
```

```
def mange_messeges(SOCKET_, received_messeges):
    while True:
        data, addr = SOCKET_.recvfrom(buffer)
        # check weather the messege is from peer itself or not
        if addr[0] != local_ip:
            messege = data.decode('utf-8')
            current_time = bring_time_of_now()
            received_messeges = f"received a message from {messege.split(':')[0]} at {current_time}"
            received_messeges.append(messege)
            print(f"from peer {full_name}")
            print(received_messeges)

def print_messages(messages):
    print("\nPeer {full_name}")
    # j represent the index of the tuple in messege list
    for j, messege in enumerate(messages, 1):
        print(f"{j}- {messege[0]}")

def print_full_messege(messages, command):
    try:
        l = int(command[:-1]) - 1
        print("\nFull message:")
        print(messages[l][1])
        #catches potential errors that may occur during the execution of the code inside the try block.
        #If there's either an IndexError (which means the index l is out of range for the messages list) or
        #a ValueError (which means the command couldn't be converted to an integer), it prints "Invalid command or index."
    except (IndexError, ValueError):
        print("Invalid command or index.")
```

```

def manage_sending_message(SOCKET_, received_messages):
    while True:
        com = input("\n please enter your message or [Digit_D] => (Message OR #D) ")

        # here we have two choices if u send Digit_D then the message is sent to the peer server
        if com.upper().endswith('D') and com[:-1].isdigit():
            print_full_message(received_messages, com)
        #else the message is sent to other peers in the broadcast
        else:
            full_message = f"{full_name}: {com}"
            SOCKET_.sendto(full_message.encode('utf-8'), (broadcast_ip, PORT))

def construct_peers():
    # creating a new socket
    SOCKET_ = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    #tells the socket to allow the reuse of addresses
    SOCKET_.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
    SOCKET_.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)
    SOCKET_.bind((local_ip, PORT))

    #to save the received messages
    Received_messages = []
    # to send messages for all peers at the same time
    #once the message sent from one peer it sends to other
    threading.Thread(target=manage_messages, args=(SOCKET_, Received_messages), daemon=True).start()
    manage_sending_message(SOCKET_, Received_messages)

if __name__ == "__main__":
    construct_peers()

```

Messages

```
> & C:/Users/user/AppData/Local/Programs/Python/Python31  
Enter your first name: JANA  
Enter your last name: SAWALMEH  
  
Type your message or command (Message or #D): from peer JANA SAWALMEH  
received a message from lana musaffer at 22:44:57  
  
Type your message or command (Message or #D): HI from peer JANA SAWALMEH  
received a message from tariq atrash at 22:45:15  
from peer JANA SAWALMEH  
received a message from tariq atrash at 22:45:17  
  
Type your message or command (Message or #D): from peer JANA SAWALMEH  
received a message from tariq atrash at 22:46:06  
from peer JANA SAWALMEH  
received a message from lana musaffer at 22:46:41  
from peer JANA SAWALMEH  
received a message from tariq atrash at 22:47:00  
  
Type your message or command (Message or #D): HELLO FREIND  
  
Type your message or command (Message or #D): from peer JANA SAWALMEH  
received a message from tariq atrash at 22:47:54  
from peer JANA SAWALMEH  
received a message from tariq atrash at 22:47:57  
from peer JANA SAWALMEH  
received a message from lana musaffer at 22:48:28  
from peer JANA SAWALMEH  
received a message from lana musaffer at 22:48:31  
|
```

PART #3

- ⊕ Have a look also on rfc2616 (<https://datatracker.ietf.org/doc/html/rfc2616>). From rfce2616, what is Entity Tag Cache Validators in the HTTP protocol and why do we need it?

1_Entity Tag (ETag) Cache Validators are a component of the RFC 2616-defined HTTP protocol. They give web servers a way to give online resources (such files or documents) distinct identities so that clients may use them to find out if the resource has changed since the last time it was requested.

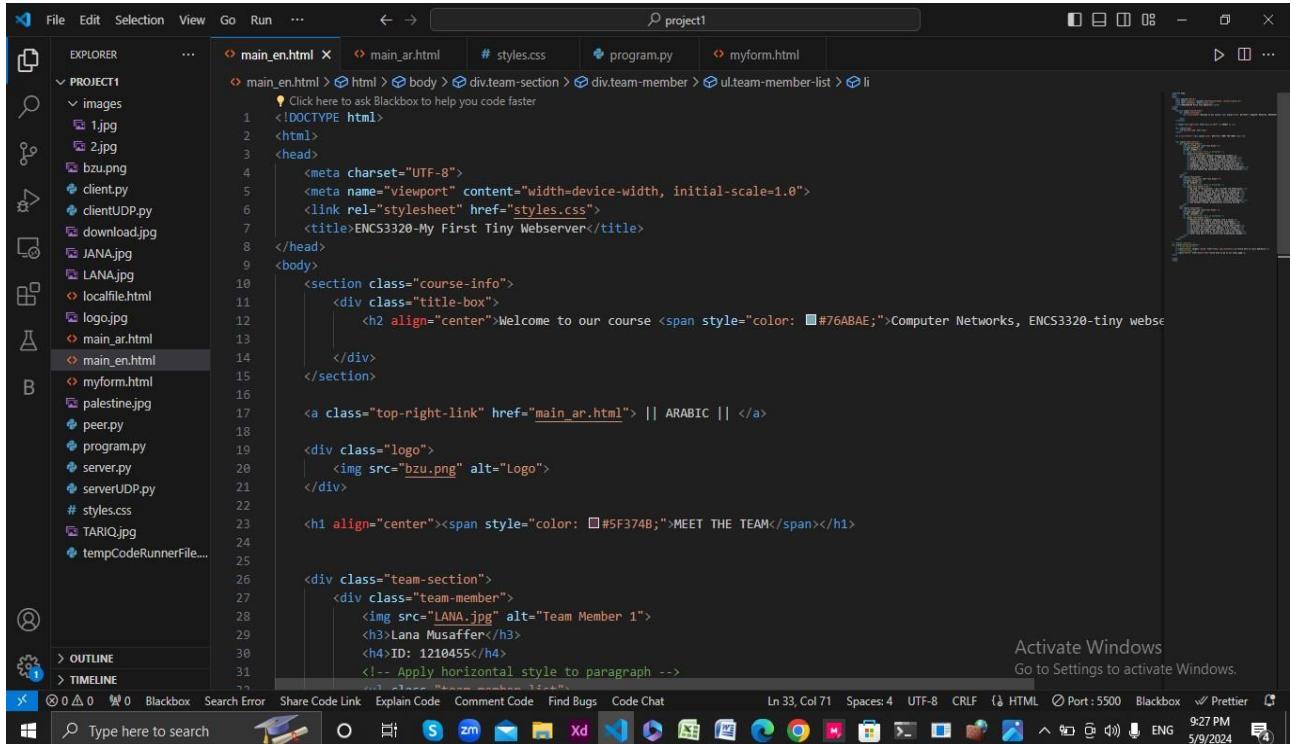
2_The server appends an ETag header with the resource's unique identification to a response sent by a client when it requests a resource. This identifier can then be stored by the client and used for upcoming requests. The client returns the ETag to the server in an If-None-Match header when requesting the same resource again. The server replies with a "304 Not Modified" response code, meaning that the client's cached copy of the resource is still valid, if it hasn't changed since the last request.

3_ETag Cache Validators minimize pointless data transport, which contributes to increased efficiency. They save bandwidth and enhance performance by enabling clients to cache resources locally and only request them from the server when they have been changed.

- We implemented a complete web server using python and socket programming, this server is listening on port 6060. The program checks:

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.

main_en.html code:



```

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="styles.css">
    <title>ENCS3320-My First Tiny Webserver</title>
</head>
<body>
    <section class="course-info">
        <div class="title-box">
            <h2 align="center">Welcome to our course <span style="color: #76ABAE;">Computer Networks, ENCS3320-tiny webse...
        </div>
    </section>

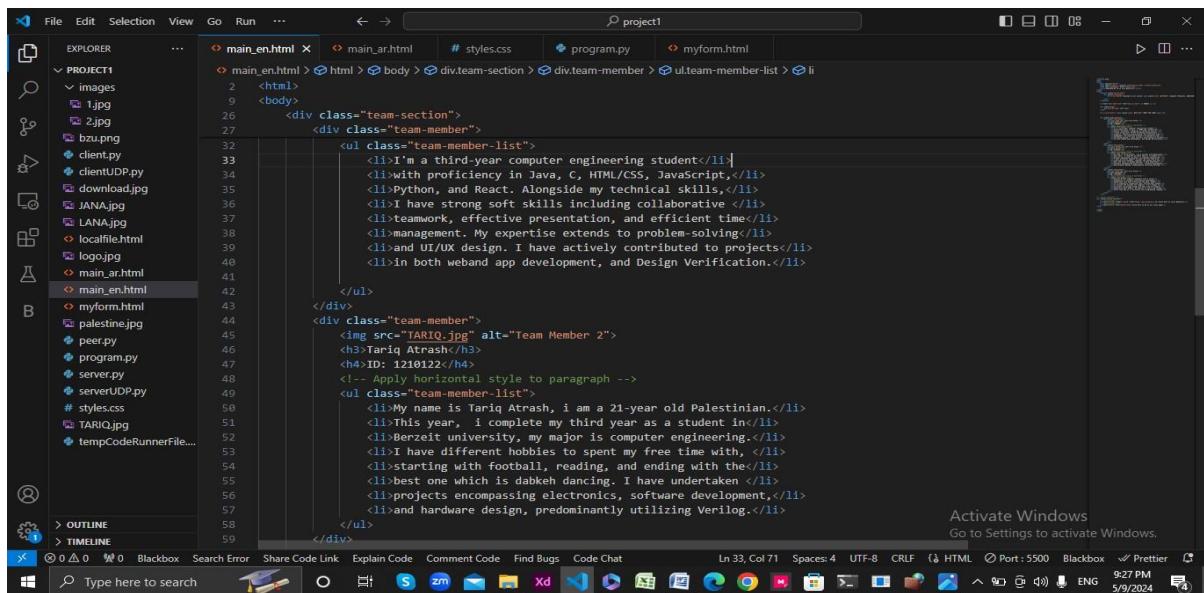
    <a class="top-right-link" href="main_ar.html"> || ARABIC || </a>

    <div class="logo">
        
    </div>

    <h1 align="center"><span style="color: #F5374B;">MEET THE TEAM</span></h1>

    <div class="team-section">
        <div class="team-member">
            
            <h3>Lana Musaffer</h3>
            <h4>ID: 1210455</h4>
            <!-- Apply horizontal style to paragraph -->
        </div>
    </div>

```



```

<div class="team-member">
    <ul class="team-member-list">
        <li>I'm a third-year computer engineering student.</li>
        <li>with proficiency in Java, C, HTML/CSS, JavaScript, and Python, and React. Alongside my technical skills, I have strong soft skills including collaborative teamwork, effective presentation, and efficient time management. My expertise extends to problem-solving and UI/UX design. I have actively contributed to projects both in web app development, and Design Verification.</li>
    </ul>
</div>
<div class="team-member">
    
    <h3>Tariq Atrash</h3>
    <h4>ID: 1210122</h4>
    <!-- Apply horizontal style to paragraph -->
    <ul class="team-member-list">
        <li>My name is Tariq Atrash, I am a 21-year old Palestinian.</li>
        <li>This year, I complete my third year as a student in Berzeit university, my major is computer engineering. I have different hobbies to spent my free time with, starting with football, reading, and ending with the best one which is dabkeh dancing. I have undertaken projects encompassing electronics, software development, and hardware design, predominantly utilizing Verilog.</li>
    </ul>
</div>

```

```
<!-- PROJECT1 -->
<!-- main_en.html -->
<html>
  <head>
    <title>Project1</title>
  </head>
  <body>
    <div class="team-section">
      <div class="team-member">
        
        <h3>Jana Sawalmeh</h3>
        <h4>ID: 12112467</h4>
        <!-- Apply horizontal style to paragraph -->
        <ul class="team-member-list">
          <li>Results-driven computer engineer with a solid</li>
          <li>foundation in software and hardware development.</li>
          <li>Experience with data structures and base, expert</li>
          <li>in os especially scheduling algo, testing best number</li>
          <li>of process and threads for computer, I'm a leader in</li>
          <li>any place can manage things and deal with all people</li>
          <li>especially deaf people, my passion to learn new things</li>
          <li>never stop and it is exclusive with programming language</li>
        </ul>
      </div>
    </div>
    <!-- Button container -->
    <div class="button-container">
      <!-- First button -->
      <a class="button" target="_blank" href="https://www.w3schools.com">Click here to visit W3Schools</a>
      <!-- Second button -->
      <a class="button" href="myform.html">Click here to go to our local page</a>
    </div>
  </body>
</html>
```

Activate Windows
Go to Settings to activate Windows.

Ln 33, Col 71 Spaces: 4 UTF-8 CRLF ⚙ HTML ⚙ Port: 5500 Blackbox ✎ Prettier

0 0 0 Blackbox Search Error Share Code Link Explain Code Comment Code Find Bugs Code Chat

Type here to search

main_en.html page:

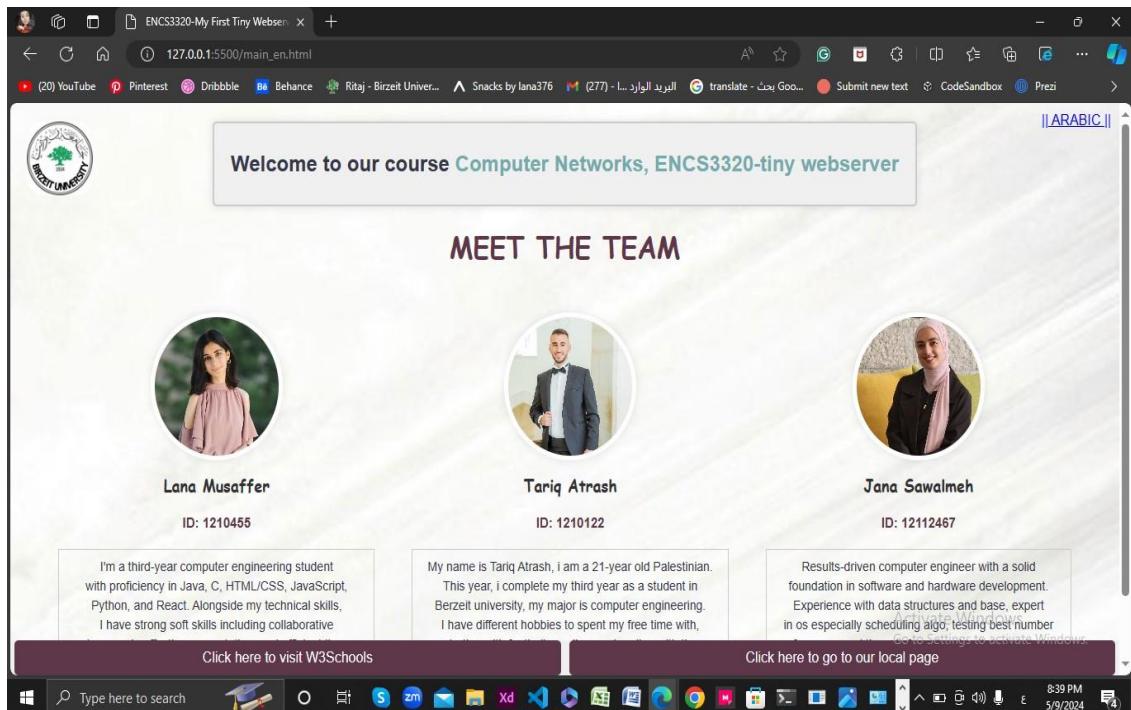


Figure 1.11

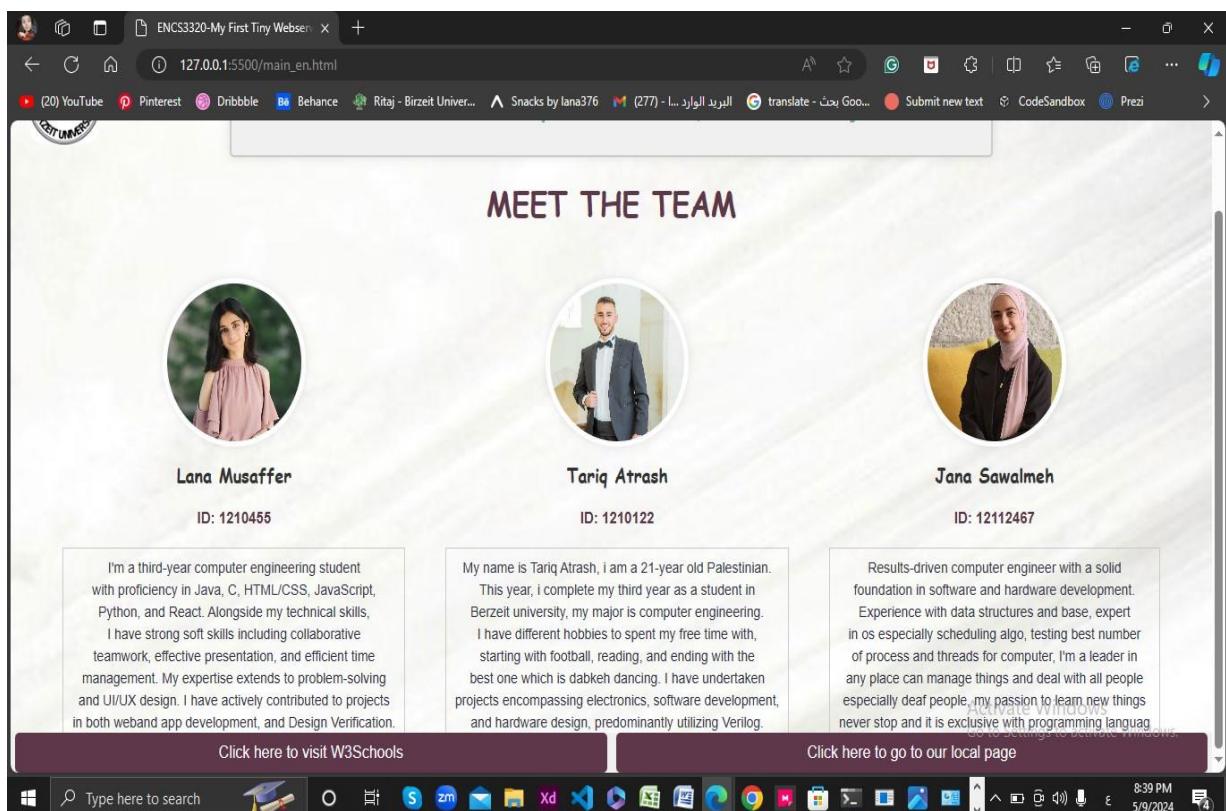


Figure 1.12

program.py code:

The screenshot shows a Windows desktop environment with a code editor window titled 'project1'. The code editor displays the 'program.py' file. The code is a Python script for a web server. It includes imports for socket, threading, and time, and defines a port number of 6060. It creates a TCP socket, binds it to the port, and listens for incoming connections. When a connection is accepted, it receives data from the client and prints the client's IP address and port. It then splits the request URL to determine the requested file. If the file is 'main_en.html' or 'index.html', it sends an HTTP response with the file content. If the file is '.html', it sends an HTTP response with the file content. The code also handles the case where the request is empty by closing the connection.

```

# Click here to ask Blackbox to help you code faster
1 # Lana Musaffer || 1210455
2 # Jana Sawalmeh || 1210122
3 # Tariq Atrash || 1210122
4
5 from socket import *
6
7 portNum=6060 #server port
8 serverSocket = socket(AF_INET,SOCK_STREAM) #creating a TCP socket for incoming request
9 serverSocket.bind(('',portNum)) #associate the server port number with this socket
10 serverSocket.listen(1) #the server listen for TCP connection requests from the client with i queued connections
11 print ("The server is ready to receive") #print a message to tell the client that the server is ready to receive
12
13 #start getting requests:
14 while True:
15     connectionSocket, address = serverSocket.accept() #when a client sends a TCP connection requests
16     sent=connectionSocket.recv(2048).decode() #create "connectionSocket" dedicated to this client
17     print(address)
18     IP= address[0]
19     port=address[1]
20     print("IP: "+ str(IP) +",Port: "+ str(port))
21     print("*****")
22     print(sent)
23     print("*****")
24
25     #if the sentence is not empty, the requested file is gotten from request header
26     if sent !='':
27         # Splitting the sentence by spaces and getting the second element (index 1)
28         # which represents the requested file in the URL
29         request_File=sent.split(' ')[1].replace('/','')
30         print("The request File is: "+request_File)
31
32     #if the request is empty the connection is closed
33     else:
34         connectionSocket.close()
35         continue
36
37     try:
38
39         #if the requested file is main.html or index.html or empty(default) or en
40         if request_File == '' or request_File=='main_en.html' or request_File== 'index.html' or request_File=='en':
41             connectionSocket.send(f"HTTP/1.1 200 OK\r\n".encode())
42             connectionSocket.send(f"Content-Type: text/html \r\n".encode())
43             connectionSocket.send(f"\r\n".encode())
44             mhtml=open('main_en.html' , 'rb')
45             connectionSocket.send(mhtml.read())
46             mhtml.close()
47
48         #If the request is /ar then the server response with main_ar.html which is an Arabic version of main_en.html
49         elif request_file== 'ar':
50             connectionSocket.send(f"HTTP/1.1 200 OK\r\n".encode())
51             connectionSocket.send(f"Content-Type: text/html \r\n".encode())
52             connectionSocket.send(f"\r\n".encode())
53             mhtml=open('main_ar.html' , 'rb')
54             connectionSocket.send(mhtml.read())
55             mhtml.close()
56
57         #if the request is an .html file
58         elif '.html' in request_file:
59             # then the server should send the requested html file with Content-Type: text/html.
60             connectionSocket.send(f"HTTP/1.1 200 OK\r\n".encode())
61             connectionSocket.send(f"Content-Type: text/html \r\n".encode())
62             connectionSocket.send(f"\r\n".encode())
63             print('response status: 200 OK\r\n')

```

This screenshot shows the same code editor window for 'program.py' with more of the code visible. The code continues from the previous snippet, handling the case where the request file is an .html file. It sends an HTTP response with the file content and the appropriate Content-Type header. The code also includes a print statement at the end of the file indicating a successful response status.

project1

```

File Edit Selection View Go Run ...
File Edit Selection View Go Run ...
EXPLORER ... main_en.html # styles.css program.py x myform.html 2.jpg
PROJECT1
images
1.jpg
2.jpg
bzu.png
client.py
clientUDP.py
download.jpg
JANA.jpg
localfile.html
main_ar.html
main_en.html
myform.html
palestine.jpg
peer.py
program.py
server.py
serverUDP.py
styles.css
TARIQ.jpg
tempCodeRunnerFile.py
OUTLINE
TIMELINE
B
Ln 4, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.2 64-bit Go Live Blackbox Prettier
Activate Windows
Go to Settings to activate Windows.
1:42 PM 5/9/2024
Type here to search
S zin XD
File Edit Selection View Go Run ...
File Edit Selection View Go Run ...
EXPLORER ... main_en.html # styles.css program.py x myform.html 2.jpg
PROJECT1
images
1.jpg
2.jpg
bzu.png
client.py
clientUDP.py
download.jpg
JANA.jpg
localfile.html
main_ar.html
main_en.html
myform.html
palestine.jpg
peer.py
program.py
server.py
serverUDP.py
styles.css
TARIQ.jpg
tempCodeRunnerFile.py
OUTLINE
TIMELINE
B
Ln 4, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.2 64-bit Go Live Blackbox Prettier
Activate Windows
Go to Settings to activate Windows.
1:42 PM 5/9/2024
Type here to search
S zin XD

```

Top Instance (Yellow Highlighted Lines):

```

57     #if the request is an .html file
58     elif '.html' in request_file:
59         # then the server should send the requested html file with Content-Type: text/html.
60         connectionSocket.send(("HTTP/1.1 200 OK\r\n".encode()))
61         connectionSocket.send(("Content-Type: text/html \r\n".encode()))
62         connectionSocket.send(("\".encode()")
63         print("response status: 200 OK\r\n")
64         f= open(str(request_file), 'rb')
65         connectionSocket.send(f.read())
66         f.close()
67
68     #if the request is a .css file
69     elif '.css' in request_file:
70         # then the server should send the requested css file with Content-Type: text/css.
71         connectionSocket.send(("HTTP/1.1 200 OK\r\n".encode()))
72         connectionSocket.send(("Content-Type: text/css \r\n".encode()))
73         connectionSocket.send(("\".encode()")
74         print("response status: 200 OK\r\n")
75         f= open(str(request_file), 'rb')
76         connectionSocket.send(f.read())
77         f.close()
78
79     #if the request is a .png
80     elif '.png' in request_file:
81         #then the server should send the png image with Content-Type: image/png.
82         connectionSocket.send(("HTTP/1.1 200 OK\r\n".encode()))
83         connectionSocket.send(("Content-Type: image/png \r\n".encode()))
84         connectionSocket.send(("\".encode()")
85         print("response status: 200 OK\r\n")
86         f= open(str(request_file), 'rb')
87         connectionSocket.send(f.read())
88         f.close()

```

Bottom Instance (Yellow Highlighted Lines):

```

89     #if the request is a .jpg
90     elif '.jpg' in request_file:
91         #then the server should send the jpg image with Content-Type: image/jpeg.
92         connectionSocket.send(("HTTP/1.1 200 OK\r\n".encode()))
93         connectionSocket.send(("Content-Type: image/jpeg \r\n".encode()))
94         connectionSocket.send(("\".encode()")
95         print("response status: 200 OK\r\n")
96         f= open(str(request_file), 'rb')
97         connectionSocket.send(f.read())
98         f.close()
99
100    #If the request is /so
101    elif request_file == '/so':
102        #then redirect to stackoverflow.com website
103        connectionSocket.send(("HTTP/1.1 307 Temporary Redirect\r\n".encode()))
104        connectionSocket.send(("Location: https://stackoverflow.com/\r\n".encode()))
105
106    #If the request is /itc
107    elif request_file == '/itc':
108        # then redirect to itc website
109        connectionSocket.send(("HTTP/1.1 307 Temporary Redirect\r\n".encode()))
110        connectionSocket.send(("Location: https://itc.birzeit.edu/\r\n".encode()))
111
112    #this is a handler only in order not to get not found error
113    elif 'favicon.ico' ==request_file:
114        print()
115    else:
116        raise Exception('Not found')
117
118    #if the file wrong or the file doesn't exist
119    except Exception as e:
120        #the server should return a simple HTML webpage that contains (Content-Type: text/html)

```

main_en.html HTTP request on the terminal window:

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The left sidebar contains icons for Explorer, Search, Problems, and other development tools. The main area has tabs for PROBLEMS, OUTPUT, TERMINAL, DEBUG CONSOLE, PORTS, and SEARCH ERROR. The TERMINAL tab is active, displaying the following log output:

```
PS C:\Users\Lenovo\Desktop\Networks\project1> python -u "c:\Users\Lenovo\Desktop\Networks\project1\program.py"
The server is ready to receive
('127.0.0.1', 51268)
IP: 127.0.0.1, Port: 51268
*****
GET /en HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
.safari/537.36 Edg/124.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image
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-US,en;q=0.9
```

The request File is: en
('127.0.0.1', 51269)
IP: 127.0.0.1, Port: 51269

```
*****  
GET /styles.css HTTP/1.1  
Host: localhost:6060  
Connection: keep-alive  
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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  
.0 Safari/537.36 Edg/124.0.0.0  
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/en  
Accept-Encoding: gzip, deflate, br, zstd  
Accept-Language: en-US,en;q=0.9
```

Activate Windows
Go to Settings to activate Windows.

Blackbox Prettier 12:03 AM 5/9/2024

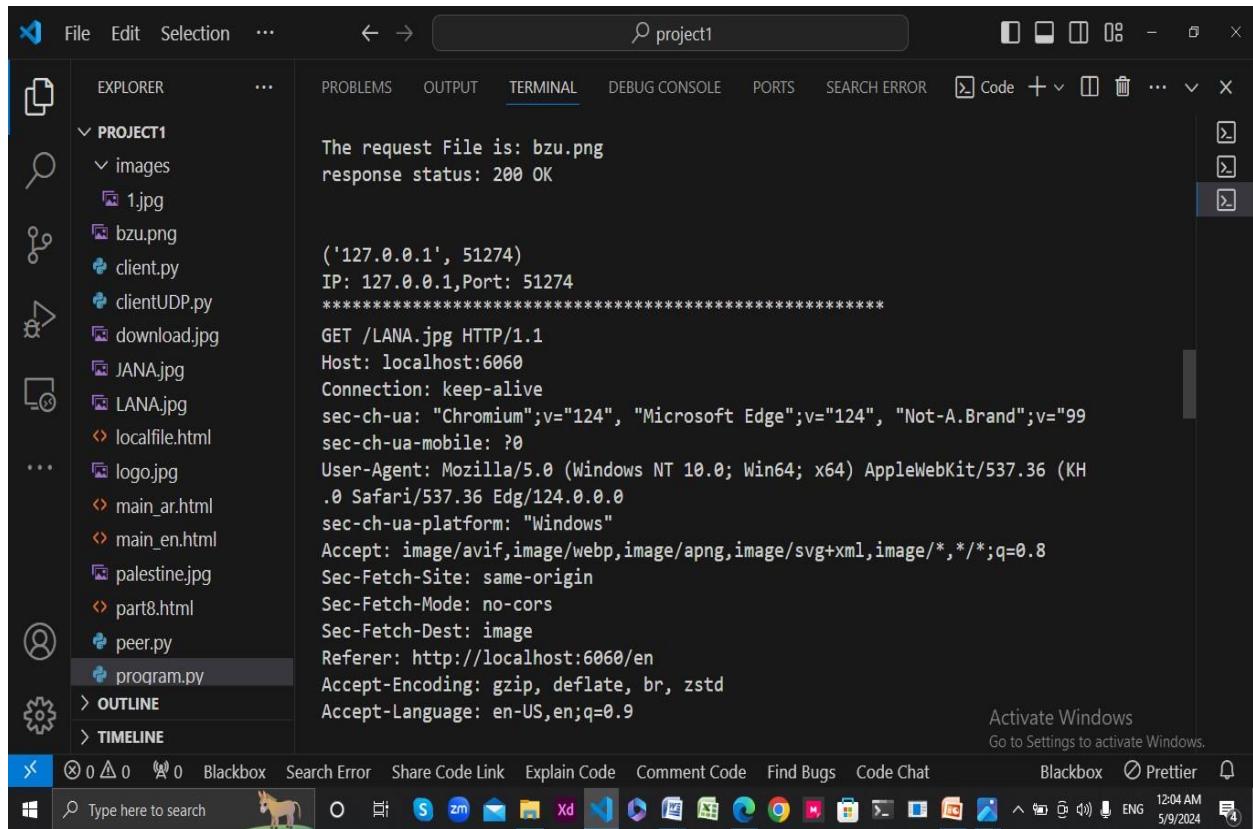
The request File is: styles.css
response status: 200 OK

```
*****  
('127.0.0.1', 51270)  
IP: 127.0.0.1, Port: 51270
```

```
GET /bzuz.png HTTP/1.1  
Host: localhost:6060  
Connection: keep-alive  
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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  
.0 Safari/537.36 Edg/124.0.0.0  
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/en  
Accept-Encoding: gzip, deflate, br, zstd  
Accept-Language: en-US,en;q=0.9
```

Activate Windows
Go to Settings to activate Windows.

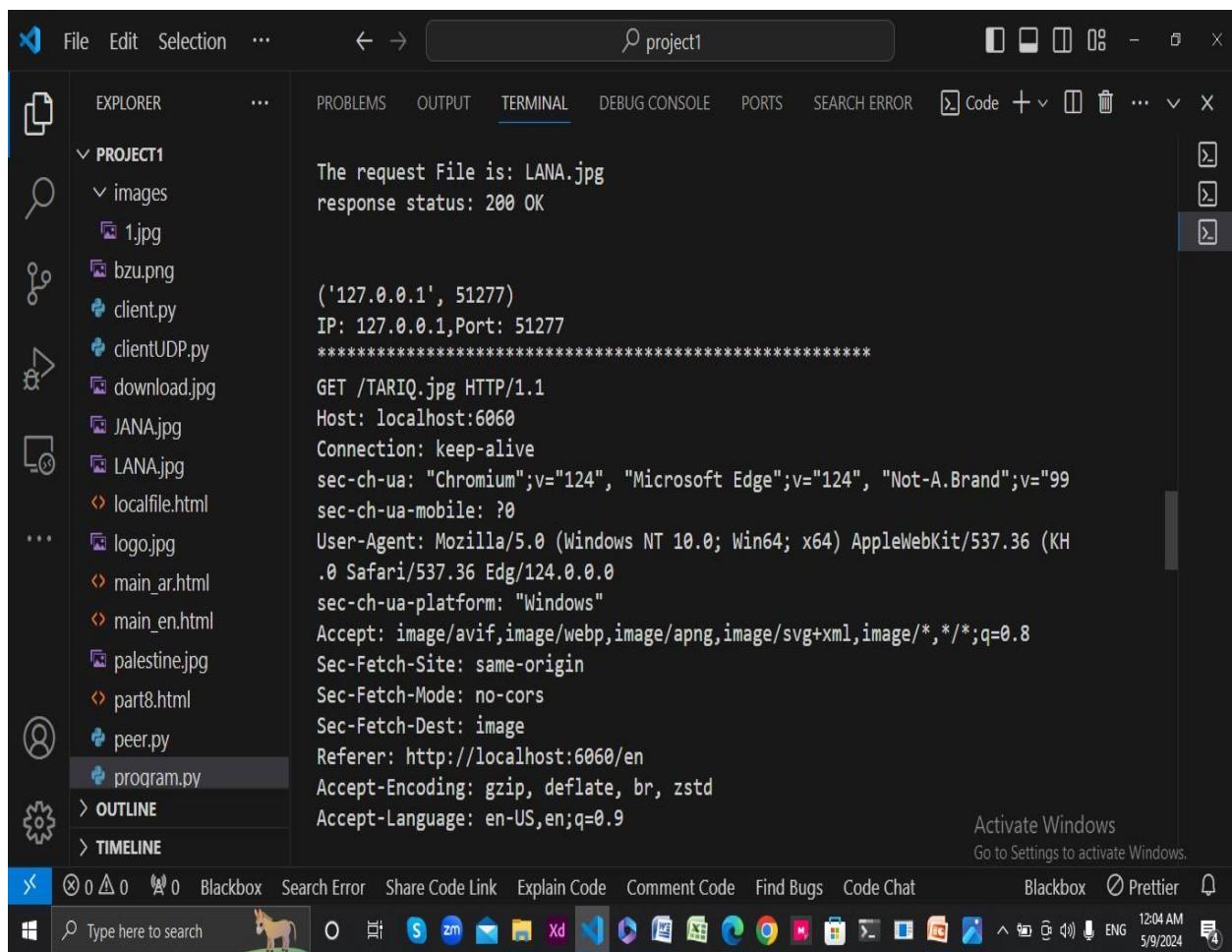
Blackbox Prettier 12:03 AM 5/9/2024



The request File is: bzu.png
response status: 200 OK

```
('127.0.0.1', 51274)
IP: 127.0.0.1, Port: 51274
*****
GET /LANA.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
.0 Safari/537.36 Edg/124.0.0.0
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/en
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9
```

Activate Windows
Go to Settings to activate Windows.



The request File is: LANA.jpg
response status: 200 OK

```
('127.0.0.1', 51277)
IP: 127.0.0.1, Port: 51277
*****
GET /TARIQ.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
.0 Safari/537.36 Edg/124.0.0.0
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/en
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9
```

Activate Windows
Go to Settings to activate Windows.

```
The request File is: TARIQ.jpg
response status: 200 OK

('127.0.0.1', 51278)
IP: 127.0.0.1, Port: 51278
*****
GET /JANA.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
.0 Safari/537.36 Edg/124.0.0.0
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/en
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9
```

Activate Windows
Go to Settings to activate Windows.

```
The request File is: JANA.jpg
response status: 200 OK

('127.0.0.1', 51275)
IP: 127.0.0.1, Port: 51275
*****
GET /download.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
.0 Safari/537.36 Edg/124.0.0.0
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/styles.css
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9
```

Activate Windows
Go to Settings to activate Windows.

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays two separate requests for files from the local host.

```
*****
The request File is: download.jpg
response status: 200 OK

('127.0.0.1', 51281)
IP: 127.0.0.1,Port: 51281
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";v="124", "Not-A.Brand";v="99
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/en
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 51282)
IP: 127.0.0.1,Port: 51282
*****
```

The terminal also includes a message about activating Windows:

Activate Windows
Go to Settings to activate Windows.

This screenshot shows the VS Code interface with the terminal tab selected. It displays a single request for the favicon.ico file.

```
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";v="124", "Not-A.Brand";v="99
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/en
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 51282)
IP: 127.0.0.1,Port: 51282
*****
```

The terminal also includes a message about activating Windows:

Activate Windows
Go to Settings to activate Windows.

On our web page, you can visit W3schools website by clicking on the button:

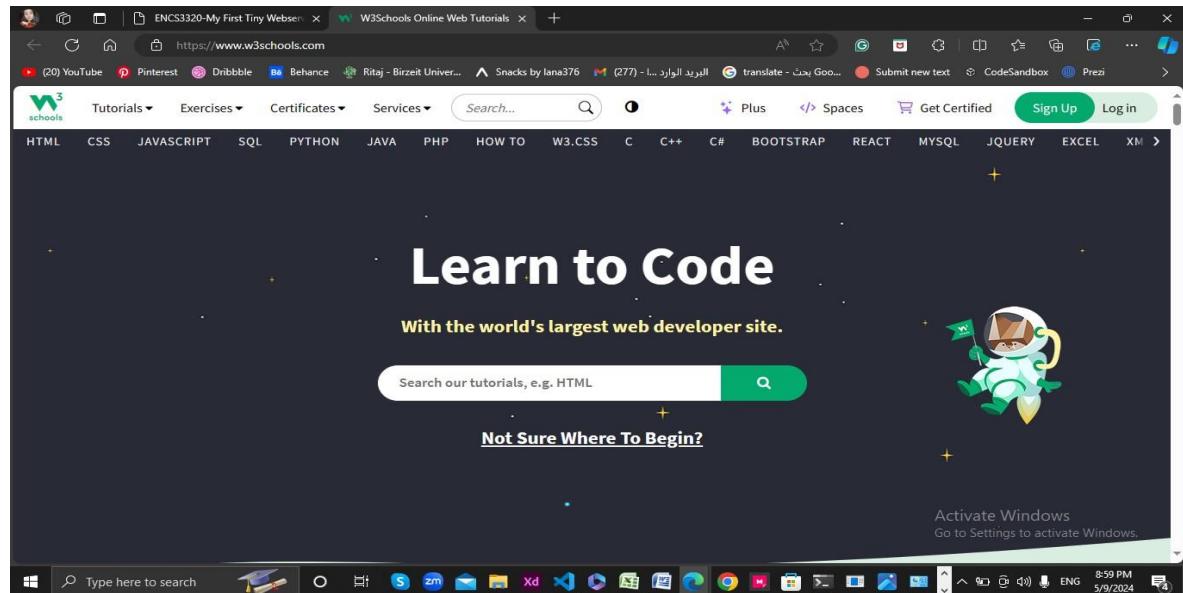


Figure 1.14

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

main_ar.html page:



Figure 1.15



Figure 1.16

main_ar.html HTTP request on the terminal window:

```

File Edit Selection View Go ... < - > project1
PROJECT1 PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS SEARCH ERROR
EXPLORER
images 1.jpg 2.jpg bzu.png client.py clientUDP.py download.jpg JANA.jpg LANA.jpg localfile.html logo.jpg main_ar.html main_en.html myform.html palestine.jpg peer.py program.py server.py serverUDP.py # styles.css
OUTLINE
TIMELINE
PS C:\Users\Lenovo\Desktop\Networks\project1> python -u "c:\Users\Lenovo\Desktop\Networks\project1\program.py"
The server is ready to receive
('127.0.0.1', 64016)
IP: 127.0.0.1, Port: 64016
*****
GET /main_ar.html HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,appl
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-US,en;q=0.9
*****
The request File is: main_ar.html
response status: 200 OK
('127.0.0.1', 64023)
IP: 127.0.0.1, Port: 64023
Activate Windows
Go to Settings to activate Windows.
Python 3.11.2 64-bit Go Live Blackbox Prettier
5:36 PM 5/9/2024 ENG

```

The screenshot shows a terminal window within a code editor interface. The terminal tab is active, displaying the following text:

```
*****
GET /styles.css HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
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-US,en;q=0.9

*****
The request File is: styles.css
response status: 200 OK

('127.0.0.1', 64025)
IP: 127.0.0.1,Port: 64025
*****
GET /bzu.png HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";v="124", "Not-A.Brand";v="99"
```

At the bottom right of the terminal window, there is a watermark: "Activate Windows Go to Settings to activate Windows."

The status bar at the bottom of the screen displays various system icons and the date/time: "5:36 PM 5/9/2024".

The screenshot shows a terminal window within a code editor interface, identical to the one above but with a different command history.

```
*****
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
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/main_ar.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: bzu.png
response status: 200 OK

('127.0.0.1', 64026)
IP: 127.0.0.1,Port: 64026
*****
GET /LANA.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
```

At the bottom right of the terminal window, there is a watermark: "Activate Windows Go to Settings to activate Windows."

The status bar at the bottom of the screen displays various system icons and the date/time: "5:37 PM 5/9/2024".

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the following log output:

```
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/main_ar.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: LANA.jpg
response status: 200 OK

('127.0.0.1', 64017)
IP: 127.0.0.1,Port: 64017
*****
GET /TARIQ.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
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
```

Activate Windows
Go to Settings to activate Windows.

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the following log output:

```
Sec-Fetch-Dest: image
Referer: http://localhost:6060/main_ar.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: TARIQ.jpg
response status: 200 OK

('127.0.0.1', 64029)
IP: 127.0.0.1,Port: 64029
*****
GET /JANA.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 S
afari/537.36 Edg/124.0.0.0
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/main_ar.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9
```

Activate Windows
Go to Settings to activate Windows.

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal output displays an HTTP request log:

```
('127.0.0.1', 64029)
IP: 127.0.0.1, Port: 64029
*****
GET /JANA.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 S
afari/537.36 Edg/124.0.0.0
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/main_ar.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: JANA.jpg
response status: 200 OK
```

The terminal also shows a message at the bottom right: "Activate Windows Go to Settings to activate Windows."

3. 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 file name)

we requested myform.html page, and this is the HTTP request on the terminal window:

File Edit Selection View Go ... ⏪ ⏩ ⏴ project1

EXPLORER

- PROJECT1
 - images
 - 1.jpg
 - 2.jpg
 - bzu.png
 - client.py
 - clientUDP.py
 - download.jpg
 - JANA.jpg
 - LANA.jpg
 - localfile.html
 - logo.jpg
 - main_ar.html
 - main_en.html
 - myform.html
 - palestine.jpg
 - peer.py
 - program.py**
 - server.py
 - serverUDP.py
 - styles.css

TERMINAL

```
The server is ready to receive
('127.0.0.1', 64228)
IP: 127.0.0.1,Port: 64228
*****
GET /myform.html HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 S
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8,application/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-US,en;q=0.9

*****
The request File is: myform.html
response status: 200 OK

('127.0.0.1', 64229)
IP: 127.0.0.1,Port: 64229
```

Activate Windows
Go to Settings to activate Windows.

0 0 Blackbox Search Error Share Code Link Explain Code Comment Code Find Bugs Code Chat Python 3.11.2 64-bit Go Live Blackbox Prettier 5:54 PM 5/9/2024

File Edit Selection View Go ... ⏪ ⏩ ⏴ project1

EXPLORER

- PROJECT1
 - images
 - 1.jpg
 - 2.jpg
 - bzu.png
 - client.py
 - clientUDP.py
 - download.jpg
 - JANA.jpg
 - LANA.jpg
 - localfile.html
 - logo.jpg
 - main_ar.html
 - main_en.html
 - myform.html
 - palestine.jpg
 - peer.py
 - program.py**
 - server.py
 - serverUDP.py
 - styles.css

TERMINAL

```
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 S
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/myform.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 64234)
IP: 127.0.0.1,Port: 64234
*****
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 S
afari/537.36 Edg/124.0.0.0
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Sec-Fetch-Site: same-origin
```

Activate Windows
Go to Settings to activate Windows.

0 0 Blackbox Search Error Share Code Link Explain Code Comment Code Find Bugs Code Chat Python 3.11.2 64-bit Go Live Blackbox Prettier 5:54 PM 5/9/2024

The request File is: favicon.ico

('127.0.0.1', 64234)
IP: 127.0.0.1, Port: 64234

Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 S
afari/537.36 Edg/124.0.0.0
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/myform.html
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

The request File is: favicon.ico

('127.0.0.1', 64235)
IP: 127.0.0.1, Port: 64235

4. 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 file name)

styles.css code:

```
# styles.css > top-right-link
    Click here to ask Blackbox to help you code faster
1 /* Lana Musaffer || 1210455
2 Jana Sawalmeh || 1210122
3 Tariq Atrash || 1210122 */
4
5 body {
6     background-image: url('download.jpg');
7     background-size: cover; /* Cover the entire background */
8     background-position: center; /* Center the background image */
9     font-family: Arial, sans-serif; /* Optional: Set the font family for the entire page */
10    color: #35374B; /* Optional: Set text color to contrast with the background */
11 }
12
13 .top-right-link {
14     position: absolute;
15     top: 10px; /* Adjust top position as needed */
16     right: 10px; /* Adjust right position as needed */
17     color: #0000FF; /* Link color */
18     text-decoration: none; /* Remove underline */
19     font-size: 16px; /* Adjust font size as needed */
20     text-decoration: underline;
21 }
22
23
24 .horizontal-paragraph {
25     display: inline-block; /* Display the paragraph inline */
26     white-space: nowrap; /* Prevent line breaks */
27     overflow: hidden; /* Hide overflowing content */
28     text-overflow: ellipsis; /* Add ellipsis for overflow */
29     width: 300px; /* Set a fixed width */
30     vertical-align: top; /* Align with the top of the image */
31     margin: 0 auto; /* Center the paragraph */
32 }
```

```
# styles.css > top-right-link
24 .horizontal-paragraph {
25   width: 300px; /* Set a fixed width */
26   vertical-align: top; /* Align with the top of the image */
27   margin: 0 auto; /* Center the paragraph */
28   text-align: center; /* Center-align the text */
29   font-size: 12px; /* Decrease font size */
30 }
31 /* Remove bullet points from list items */
32 .team-member-list {
33   list-style-type: none; /* Remove default bullet points */
34   border: 1px solid #ccc; /* Add a border around the list */
35   padding: 10px; /* Add padding inside the border */
36   margin-top: 10px; /* Add margin to separate from other elements */
37 }
38 /* Style list items */
39 .team-member-list li {
40   font-size: 14px; /* Increase font size for better readability */
41   margin-bottom: 5px; /* Add margin between list items */
42 }
43 /* Style for the buttons */
44 .button-container {
45   position: fixed;
46   left: 0;
47   bottom: 0;
48   width: 100%;
49   background-color: #f0f0f0; /* Background color for the button container */
50   padding: 0; /* Padding for the buttons */
51   text-align: center; /* Center align the buttons */
52   display: flex; /* Use flexbox */
53   justify-content: space-around; /* Distribute space between the buttons */
54 }
```

Activate Windows
Go to Settings to activate Windows.

```
# styles.css > top-right-link
52 .button-container {
53   width: 100%;
54   background-color: #f0f0f0; /* Background color for the button container */
55   padding: 0; /* Padding for the buttons */
56   text-align: center; /* Center align the buttons */
57   display: flex; /* Use flexbox */
58   justify-content: space-around; /* Distribute space between the buttons */
59 }
60 .button {
61   flex: 1; /* Each button takes up equal space */
62   padding: 10px 20px;
63   background-color: #5f3748;
64   color: white;
65   text-decoration: none;
66   border: none;
67   border-radius: 5px;
68   cursor: pointer;
69   transition: background-color 0.3s;
70   margin: 0 5px; /* Add a small margin between buttons */
71 }
72 .button:hover {
73   background-color: #764e64;
74   text-decoration: underline;
75 }
76 .logo img {
77   width: 80px; /* Adjust the width of the image */
78   height: auto; /* Maintain aspect ratio */
79 }
80 /* Adding position */
81 
```

Activate Windows
Go to Settings to activate Windows.

```
# styles.css > top-right-link
  /* Adjusting position */
  .logo {
    position: absolute;
    top: 20px;
    left: 20px;
  }
  header {
    background-color: #333;
    color: #fff;
    text-align: center;
    padding: 1em 0;
  }
  .team-section {
    text-align: center;
    margin-top: 50px;
  }
  .team-member {
    display: inline-block;
    margin: 0 20px;
  }
  .team-member img {
    border-radius: 50%;
    width: 150px;
    height: 150px;
    object-fit: cover;
    border: 5px solid #fff;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  }
```

Activate Windows
Go to Settings to activate Windows.

```
# styles.css > top-right-link
  .team-member h3 {
    margin-top: 10px;
    font-size: 18px;
    color: #333;
  }
  .team-member p {
    margin-top: 5px;
    font-size: 14px;
    color: #666;
  }
  .course-info, .summary-box, .student-info, .images, .links {
    margin: 20px;
  }
  .box {
    background-color: #8C6A5D;
    padding: 5px;
    border: 1px solid #ddd;
    margin-bottom: 5px;
  }
  .member-box {
    background-color: #f2f2f2;
    padding: 10px;
    border: 1px solid #ddd;
    margin-top: 10px;
  }
  img {
```

Activate Windows
Go to Settings to activate Windows.

```
# styles.css > top-right-link
151 img {
152   border-radius: 25%;
153 }
154
155
156 h4 {
157   font-size: 15px; /* Change the font size to your desired value */
158   font-family: Arial, sans-serif; /* Change the font family to your desired font */
159   color: #5F374B; /* Change the font color to your desired color */
160 }
161
162 .team-member p {
163   margin-top: 5px;
164   font-size: 14px;
165   color: #666;
166 }
167
168 .team-member p {
169   display: block; /* Make each paragraph appear on its own line */
170 }
171
172 h1 span {
173   font-family: cursive; /* Replace 'Your Font Name' with the name of the font you want to use */
174 }
175
176 h3 {
177   font-family: cursive; /* Change the font-family to your desired font */
178 }
179
180 .title-box {
181   background-color: #f0f0f0; /* Background color of the box */
182   padding: 10px 20px; /* Padding inside the box */
183 }
184
185
186
187
188
189 }
```

Activate Windows
Go to Settings to activate Windows.

```
# styles.css > top-right-link
156 h4 {
157 }
158
159 .team-member p {
160   margin-top: 5px;
161   font-size: 14px;
162   color: #666;
163 }
164
165 .team-member p {
166   display: block; /* Make each paragraph appear on its own line */
167 }
168
169 h1 span {
170   font-family: cursive; /* Replace 'Your Font Name' with the name of the font you want to use */
171 }
172
173 h3 {
174   font-family: cursive; /* Change the font-family to your desired font */
175 }
176
177 .title-box {
178   background-color: #f0f0f0; /* Background color of the box */
179   padding: 10px 20px; /* Padding inside the box */
180   border: 2px solid #ccc; /* Border style and color */
181   border-radius: 5px; /* Rounded corners */
182   box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); /* Box shadow for a 3D effect */
183   width: fit-content; /* Adjust width based on content */
184   margin: 0 auto; /* Center the box horizontally */
185   text-align: center; /* Center text horizontally */
186 }
187
188
189 }
```

Activate Windows
Go to Settings to activate Windows.

Styles.css HTTP request on the terminal window:

The server is ready to receive ('127.0.0.1', 64332)
IP: 127.0.0.1, Port: 64332

GET /styles.css HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 S
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,appli
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-US,en;q=0.9

The request File is: styles.css
response status: 200 OK

('127.0.0.1', 64335)
IP: 127.0.0.1, Port: 64335

The request File is: styles.css
response status: 200 OK

('127.0.0.1', 64335)
IP: 127.0.0.1, Port: 64335

GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
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/styles.css
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

The request File is: favicon.ico

5. 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 file name)

```
File Edit Selection View Go ⏪ ⏩ ⏴ project1
EXPLORER PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS SEARCH ERROR
PROJECT1
images
1.jpg
2.jpg
bzu.png
client.py
clientUDP.py
download.jpg
JANA.jpg
LANA.jpg
localfile.html
logo.jpg
main_ar.html
main_en.html
myform.html
palestine.jpg
peer.py
program.py
server.py
serverUDP.py
# styles.css
OUTLINE
TIMELINE
Code + - ×
PS C:\Users\Lenovo\Desktop\Networks\project1> python -u "c:/Users/Lenovo/Desktop/Networks/project1/prog
The server is ready to receive
('127.0.0.1', 59938)
IP: 127.0.0.1, Port: 59938
*****
GET /bzu.png HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.
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-US,en;q=0.9

*****
The request File is: bzu.png
response status: 200 OK
('127.0.0.1', 59938)
IP: 127.0.0.1, Port: 59938
Activate Windows
Go to Settings to activate Windows.
Python 3.11.2 64-bit
Port: 5500 Blackbox Prettier
8:56 PM 5/9/2024
Type here to search
```

The screenshot shows a Windows desktop environment with the Visual Studio Code (VS Code) application open. The title bar reads "project1". The left sidebar contains icons for Explorer, Problems, Output, Terminal, Debug Console, Ports, and Search Error. The Explorer view shows a project structure with files like "1.jpg", "2.jpg", "bzu.png", "client.py", "clientUDP.py", "download.jpg", "JANA.jpg", "LANA.jpg", "localfile.html", "logo.jpg", "main_ar.html", "main_en.html", "myform.html", "palestine.jpg", "peer.py", "program.py" (which is selected), "server.py", "serverUDP.py", and "# styles.css". Below the Explorer is an "OUTLINE" section. The main area is a terminal window displaying the following text:

```
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 Edg/124.0.0.0
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/bzu.png
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico
('127.0.0.1', 59943)
IP: 127.0.0.1, Port: 59943
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 Edg/124.0.0.0
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
```

The request File is: favicon.ico
('127.0.0.1', 59943)
IP: 127.0.0.1,Port: 59943

GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 Edg/124.0.0.0
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/bzu.png
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

The request File is: favicon.ico
('127.0.0.1', 59944)
IP: 127.0.0.1,Port: 59944

Activate Windows
Go to Settings to activate Windows.

6. 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 file name)

PS C:\Users\Lenovo\Desktop\Networks\project1> python -u "c:\Users\Lenovo\Desktop\Networks\project1\program.py"
The server is ready to receive
('127.0.0.1', 59863)
IP: 127.0.0.1,Port: 59863

GET /LANA.jpg HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/
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-US,en;q=0.9

The request File is: LANA.jpg
response status: 200 OK
('127.0.0.1', 59864)
IP: 127.0.0.1,Port: 59864

Activate Windows
Go to Settings to activate Windows.

```
File Edit Selection View Go ... ⏪ ⏩ ⏴ ⏵ project1
EXPLORER PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS SEARCH ERROR
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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/
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/LANA.jpg
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 59865)
IP: 127.0.0.1,Port: 59865
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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/
sec-ch-ua-platform: "Windows"
Activate Windows
Go to Settings to activate Windows.
# styles.css
> OUTLINE
> TIMELINE
0 0 Blackbox Search Error Share Code Link Explain Code Comment Code Find Bugs Code Chat Python 3.11.2 64-bit Port: 5500 Blackbox Prettier
Type here to search
O S zm xd Python 3.11.2 64-bit Port: 5500 Blackbox Prettier
8:48 PM 5/9/2024
```

```
File Edit Selection View Go ... ⏪ ⏩ ⏴ ⏵ project1
EXPLORER PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS SEARCH ERROR
*****
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 59865)
IP: 127.0.0.1,Port: 59865
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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/
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/LANA.jpg
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico
Activate Windows
Go to Settings to activate Windows.
# styles.css
> OUTLINE
> TIMELINE
0 0 Blackbox Search Error Share Code Link Explain Code Comment Code Find Bugs Code Chat Python 3.11.2 64-bit Port: 5500 Blackbox Prettier
Type here to search
O S zm xd Python 3.11.2 64-bit Port: 5500 Blackbox Prettier
8:49 PM 5/9/2024
```

7. You can also visit our local page for getting images from a folder and display them:

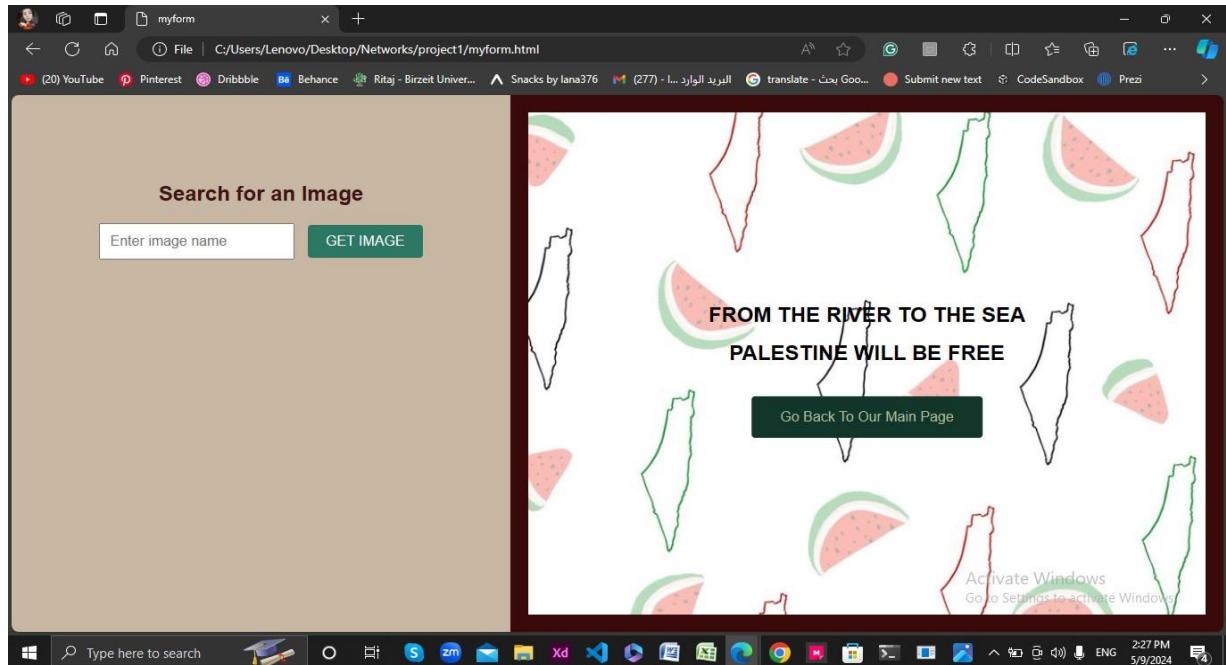


Figure 1.16

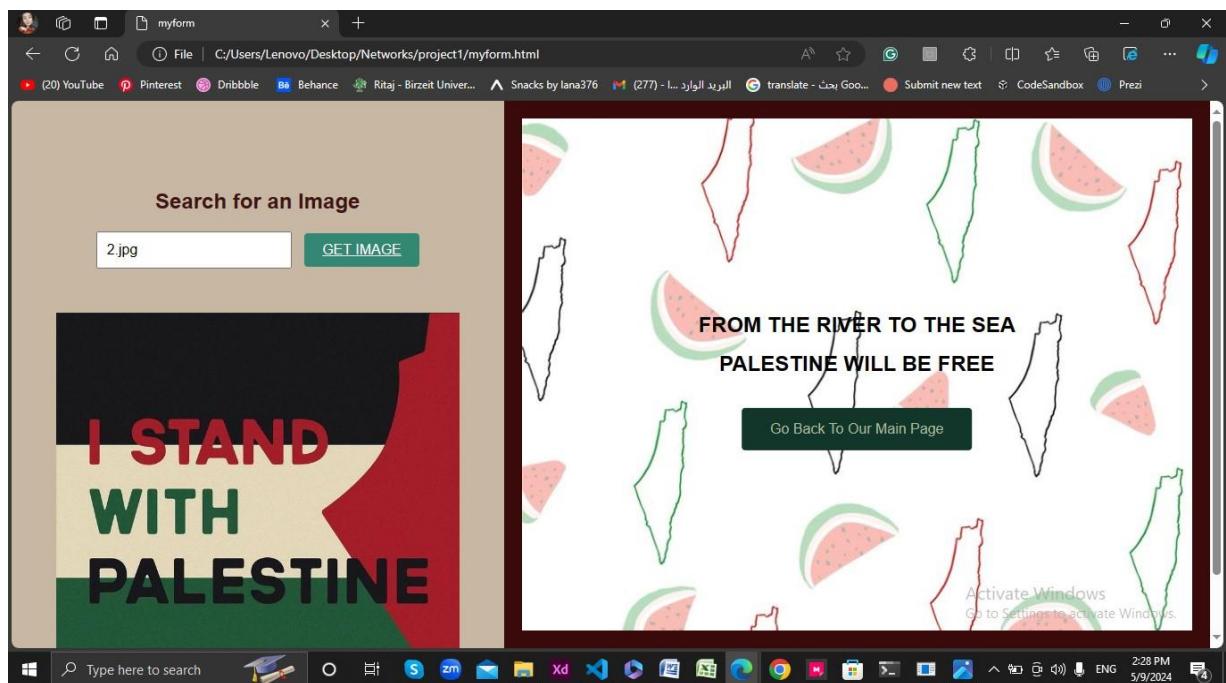


Figure 1.17

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

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

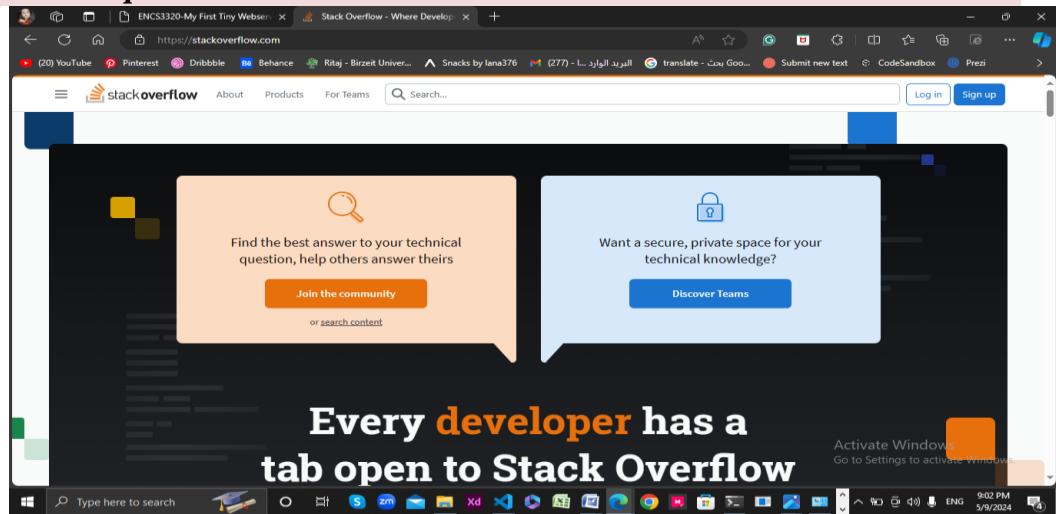


Figure 1.18

/so HTTP request on the terminal window:

A screenshot of a terminal window within a code editor interface. The terminal tab is active, showing the command "python -u "c:/Users/Lenovo/Desktop/Networks/project1/program.py"" being run. The output shows a server listening on port 59474 and then receives an HTTP GET request for "/so". The request details are printed, including the host, connection, user-agent, and other headers. The response "The request File is: so" is also shown. The code editor interface on the left shows a project structure with files like client.py, server.py, and program.py.

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

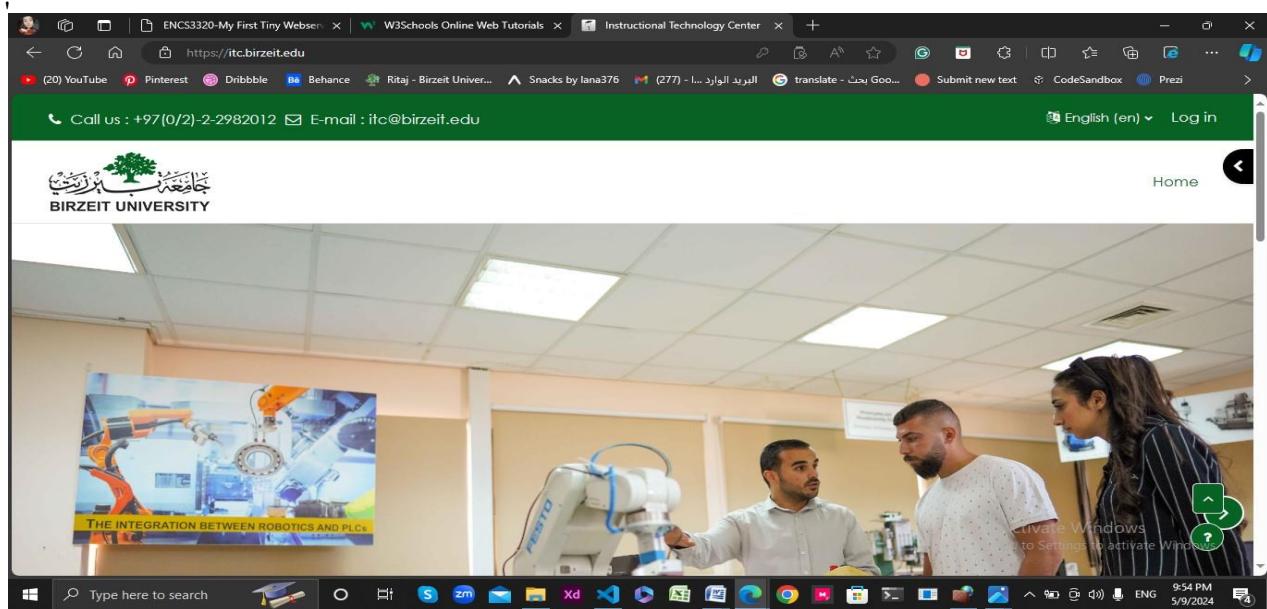


Figure 1.19

/itc HTTP request on the terminal window:

```

PS C:\Users\Lenovo\Desktop\Networks\project1> python -u "c:\Users\Lenovo\Desktop\Networks\project1\program.py"
The server is ready to receive
('127.0.0.1', 59424)
IP: 127.0.0.1,Port: 59424
*****
GET /itc HTTP/1.1
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 Edg/124.0.0.0
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-US,en;q=0.9

*****
The request File is: itc
('127.0.0.1', 59425)
IP: 127.0.0.1,Port: 59425
*****
('127.0.0.1', 59428)
IP: 127.0.0.1,Port: 59428
*****

```

9. 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 notfound**” in the body in red
- 4- Your names and IDs in Bold
- 5- The IP and port number of the client

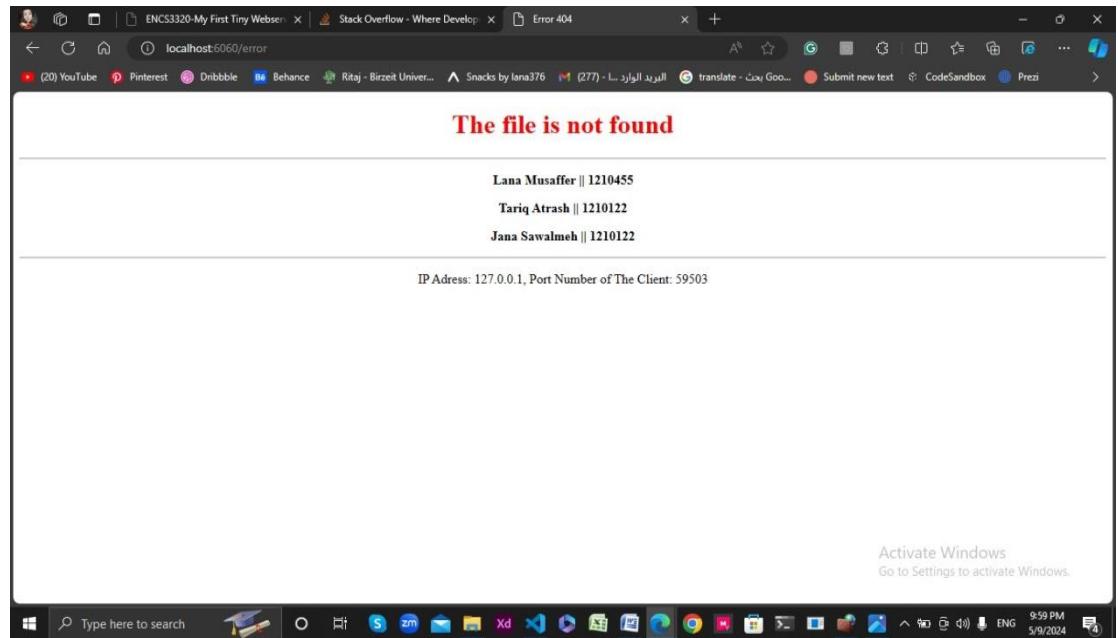


Figure 1.20

HTTP request on the terminal window:

```

File Edit Selection View Go Run ...
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE PORTS SEARCH ERROR
PS C:\Users\Lenovo\Desktop\Networks\project1> python -u "c:/Users/Lenovo/Desktop/Networks/project1/program.py"
The server is ready to receive
('127.0.0.1', 59526)
IP: 127.0.0.1, Port: 59526
*****
GET /error HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";v="124", "Not-A.Brand";v="99"
sec-ch-ua-mobile: ?
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 Edg/124.0.0.0
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: ?
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9
*****
The request File is: error
errortest
Response status: 404 Not Found
('127.0.0.1', 59527)
IP: 127.0.0.1, Port: 59527
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";v="124", "Not-A.Brand";v="99"
sec-ch-ua-mobile: ?
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 Edg/124.0.0.0
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8

```

```

        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
Referer: http://localhost:6060/error
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 59528)
IP: 127.0.0.1,Port: 59528
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 Edg/124.0.0.0
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/error
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 59531)
IP: 127.0.0.1,Port: 59531
*****

```

```

        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 Edg/124.0.0.0
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Referer: http://localhost:6060/error
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

('127.0.0.1', 59531)
IP: 127.0.0.1,Port: 59531
*****
GET /favicon.ico HTTP/1.1
Host: localhost:6060
Connection: keep-alive
sec-ch-ua: "Chromium";v="124", "Microsoft Edge";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 Edg/124.0.0.0
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/error
Accept-Encoding: gzip, deflate, br, zstd
Accept-Language: en-US,en;q=0.9

*****
The request File is: favicon.ico

```

We tested from different laptop:

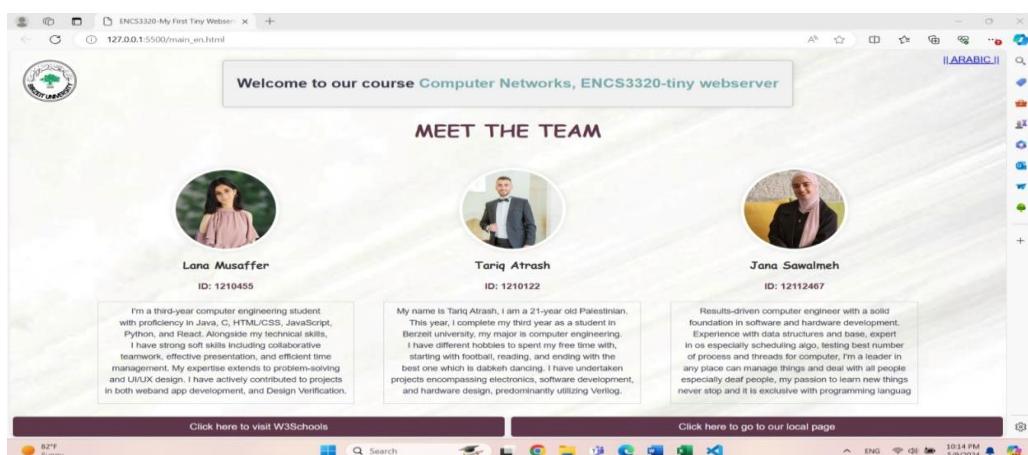


Figure 1.21

Conclusion

Through the use of socket programming, the project successfully created a flexible web server that can handle a variety of content kinds and client requests. It showcased effective file management for HTML, CSS, PNG, and JPG, reliable error handling with educational 404 pages, and dynamic content distribution via forms that users could fill out. The server's reliability and compatibility were validated by testing on various devices, underscoring its preparedness for practical uses. The practical understanding of network communications has been enhanced by this experience, providing a solid basis for the advancement of network applications in the future.

References

- [1] <https://www.bbc.co.uk/bitesize/guides/zr3yb82/revision/4>
- [2] <https://www.avast.com/c-tcp-vs-udp-difference>
- [3] <https://www.cloudflare.com/learning/network-layer/what-is-a-router/>
- [4] <https://en.wikipedia.org/wiki/HTTP>
- [5] https://en.wikipedia.org/wiki/Network_socket