جامعة بيرزيت

BIRZEIT UNIVERSITY

Faculty of Engineering & Technology

Department of Electrical and Computer Engineering

ENCS3320

COMPUTER NETWORKS

Course project 1

Prepared by:	ID:
Rana Musa	1210007
Donia Alshiakh	1210517
Haneen Odeh	1210716

Instructor: Dr. Mohammad Jubran

Section 2

Date:15/12/2023

Abstract

This project consists of three parts. The first part is to run some commands on the terminal; ping a device in the same network, ping www.cornell.edu, tracert and nslookup it. The second part is implementing server and client application both for TCP. The server should listen on port 9955. The last part is to implement a simple but a complete web server in python that is listening on port 9966. Both html and CSS are used for the design of our website.

Table of Contents

Table of Figures

1. Part 1:

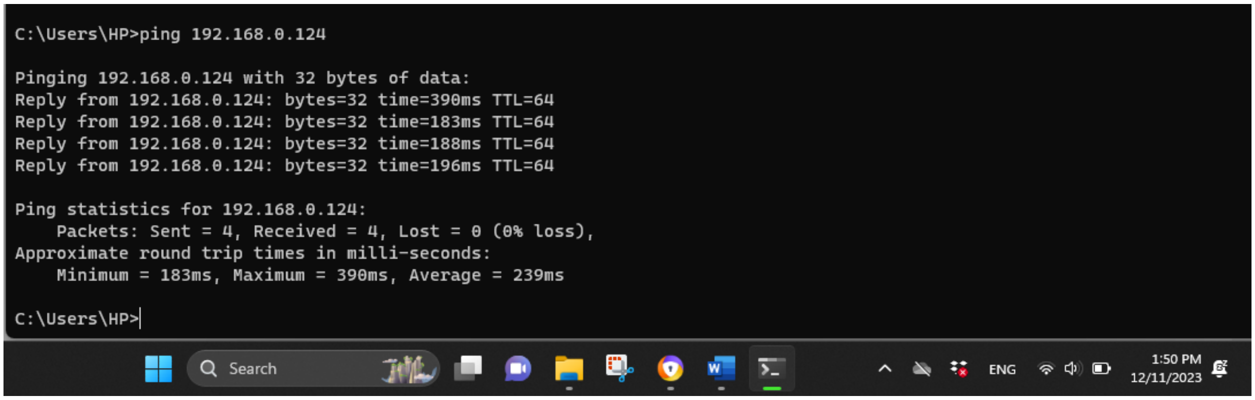
1.1. Ping command

Ping is a command that works across all operating systems. It can be used to determine how long it will take you to accomplish your goal and whether you can reach it. Ping transmits packets using the Internet Control Message Protocol (ICMP) to the intended address. It then awaits the echo response. It can display this request's statistics, faults, and packet loss.

This command will cause you to send a small number of echo requests—typically four. The results for each of them will then be sent to you, together with information on how much data was received, how long it took for a response, and TTL (Time to live). You can transmit a brief data packet to a specific IP address by using this command. After then, watch for a feedback packet. It can be used to ping a name resolution as well. A ping to an IP address that returns an answer but not to a name indicates that the two are inconsistent.

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

Note: 192.168.0.124 this is Haneen`s IP address Phone

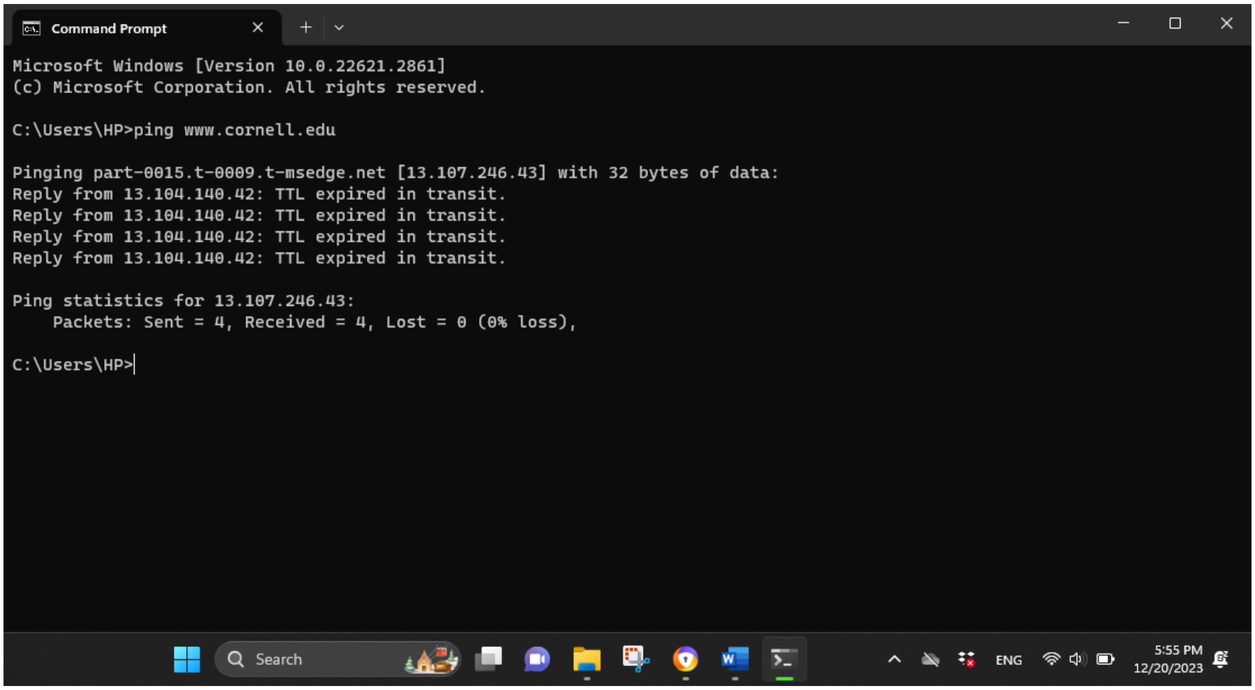


```
C:\Users\HP>ping 192.168.0.124

Pinging 192.168.0.124 with 32 bytes of data:
Reply from 192.168.0.124: bytes=32 time=390ms TTL=64
Reply from 192.168.0.124: bytes=32 time=183ms TTL=64
Reply from 192.168.0.124: bytes=32 time=188ms TTL=64
Reply from 192.168.0.124: bytes=32 time=196ms TTL=64

Ping statistics for 192.168.0.124:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 183ms, Maximum = 390ms, Average = 239ms

C:\Users\HP>
```

Figure 1:pinging a device in the same network

We sent out four packets to the destination, and the destination responds back with the same four packets, so we can see four responses is reply from 192.168.1 and 104 that is for router, Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), We can note that the time (time to reach destination and back) is very small because the two devices are in the same network and there is no need to go outside of the network, Approximate round trip times in mile-seconds: Minimum = 183ms, Maximum = 390ms, Average = 239ms.

1.1.2. Ping www.cornell.edu



```
Microsoft Windows [Version 10.0.22621.2861]
(c) Microsoft Corporation. All rights reserved.

C:\Users\HP>ping www.cornell.edu

Pinging part-0015.t-0009.t-msedge.net [13.107.246.43] with 32 bytes of data:
Reply from 13.104.140.42: TTL expired in transit.
Reply from 13.104.140.42: TTL expired in transit.
Reply from 13.104.140.42: TTL expired in transit.
Reply from 13.104.140.42: TTL expired in transit.

Ping statistics for 13.107.246.43:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Users\HP>
```

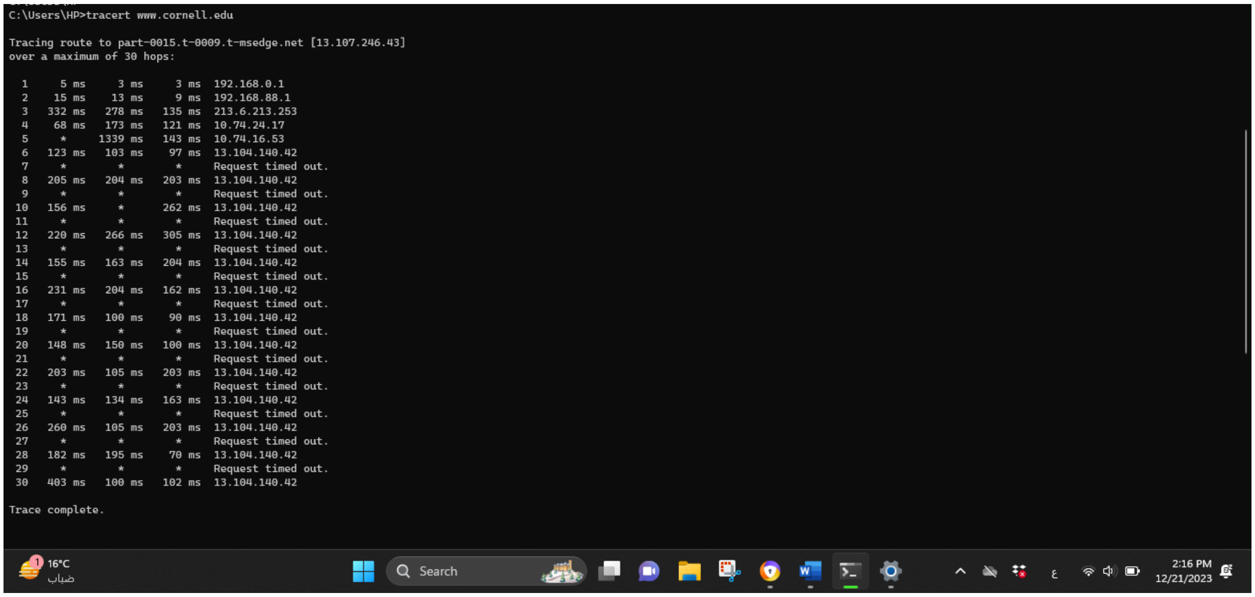
Figure 2:pinging www.cornell.edu

Cornell.edu is a site name, we sent out four packets to the destination, and the destination responds back with the same four packets, so we can see four responses is reply from this site, we sent out 32 bytes of data and we got back 32 byte of data. Packets: Sent = 4, Received = 4, Lost = 0 (0% loss), there is an important note that the time is more than the previous one because the server and client are not in the same network and there is need to go outside of the network to find the server and create connection with it, and as seen above (TTL expired in transit) it means that the Time-to-Live (TTL) value of the packet has reached zero during its journey through the network.

1.2. Tracert command

It’s command-line utility that you can use to trace the path that an Internet Protocol (IP) packet takes to its destination. The TRACERT diagnostic utility determines the route to a destination by sending Internet Control Message Protocol (ICMP) echo packets to the destination. In these packets, TRACERT uses varying IP Time-To-Live (TTL) values. Because each router along the path is required to decrement the packet's TTL by at least 1 before forwarding the packet, the TTL is effectively a hop counter. When the TTL on a packet reaches zero (0), the router sends an ICMP "Time Exceeded" message back to the source computer.

1.2.1 tracert www.cornell.edu



```
C:\Users\HP>tracert www.cornell.edu

Tracing route to part-0015.t-0009.t-msedge.net [13.107.246.43]
over a maximum of 30 hops:

  1     5 ms     3 ms     3 ms  192.168.0.1
  2    15 ms    13 ms     9 ms  192.168.88.1
  3   332 ms   278 ms   135 ms  213.6.213.253
  4    68 ms   173 ms   121 ms  10.74.24.17
  5     *     1339 ms   143 ms  10.74.16.53
  6   123 ms   103 ms    97 ms  13.104.140.42
  7     *        *        *     Request timed out.
  8   205 ms   204 ms   203 ms  13.104.140.42
  9     *        *        *     Request timed out.
 10   156 ms     *      262 ms  13.104.140.42
 11     *        *        *     Request timed out.
 12   220 ms   266 ms   305 ms  13.104.140.42
 13     *        *        *     Request timed out.
 14   155 ms   163 ms   204 ms  13.104.140.42
 15     *        *        *     Request timed out.
 16   231 ms   204 ms   162 ms  13.104.140.42
 17     *        *        *     Request timed out.
 18   171 ms   100 ms    90 ms  13.104.140.42
 19     *        *        *     Request timed out.
 20   148 ms   150 ms   100 ms  13.104.140.42
 21     *        *        *     Request timed out.
 22   203 ms   105 ms   203 ms  13.104.140.42
 23     *        *        *     Request timed out.
 24   143 ms   134 ms   163 ms  13.104.140.42
 25     *        *        *     Request timed out.
 26   260 ms   105 ms   203 ms  13.104.140.42
 27     *        *        *     Request timed out.
 28   182 ms   195 ms    70 ms  13.104.140.42
 29     *        *        *     Request timed out.
 30   403 ms   100 ms   102 ms  13.104.140.42

Trace complete.
```

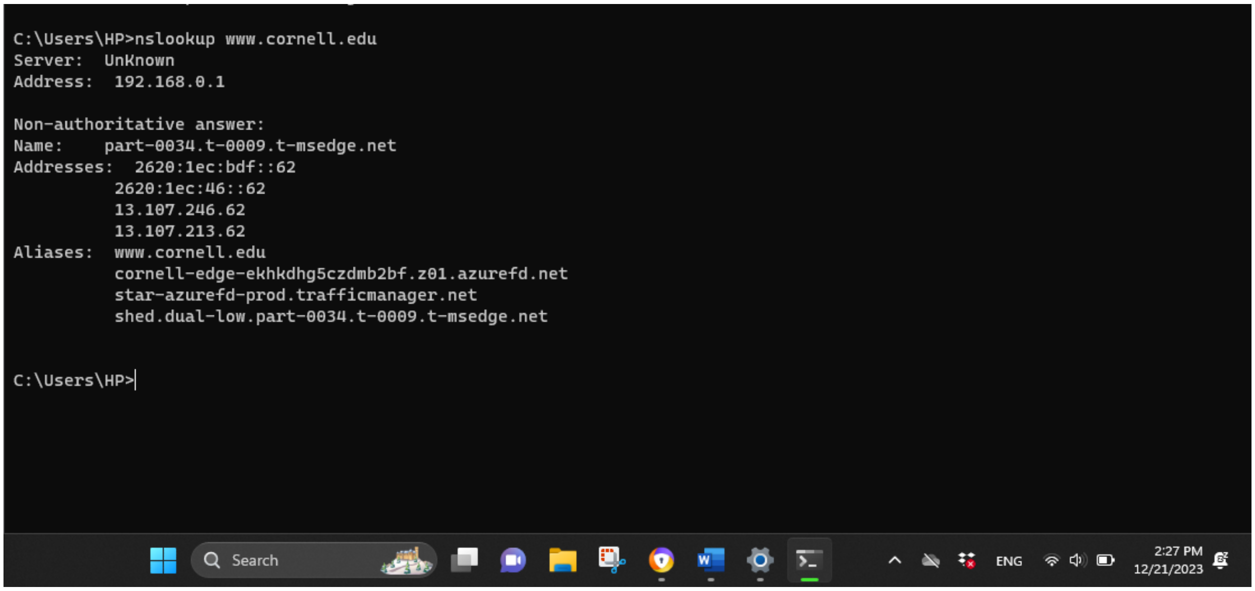
Figure 3:tracert www.cornell.edu

Here the 13.107.246.43 is the IP address for www.cornell, and as shown above it went step by step through 30 hops before reach www.cornell.edu, and if there is any problem with one of these hops it will be written as you see above in hops 7,11,13 and others.

1.3. Nslookup Command

This command will fetch the DNS records for a given domain name or IP address. The IP address and domain names are stores in DNS Servers, so the nslookup command lets you query the DNS Records to gather information. Using nslookup online is very simple. Enter a domain name in the search bar above and hit 'enter'. This will take you to an overview of DNS records for the domain name you specified. It allows you to view all the DNS records for a website. There are many situations where online nslookup can be a useful tool. For example, when you are configuring the DNS records of your own domain, you might want to check whether you have configured them correctly. You can do this by entering the domain name at the top of this page.

1.3.1. Nslookup www.cornell.edu



```
C:\Users\HP>nslookup www.cornell.edu
Server:  UnKnown
Address:  192.168.0.1

Non-authoritative answer:
Name:    part-0034.t-0009.t-msedge.net
Addresses:  2620:1ec:bdf::62
          2620:1ec:46::62
          13.107.246.62
          13.107.213.62
Aliases:  www.cornell.edu
          cornell-edge-ekhkdhg5czdmb2bf.z01.azurefd.net
          star-azurefd-prod.trafficmanager.net
          shed.dual-low.part-0034.t-0009.t-msedge.net


C:\Users\HP>
```

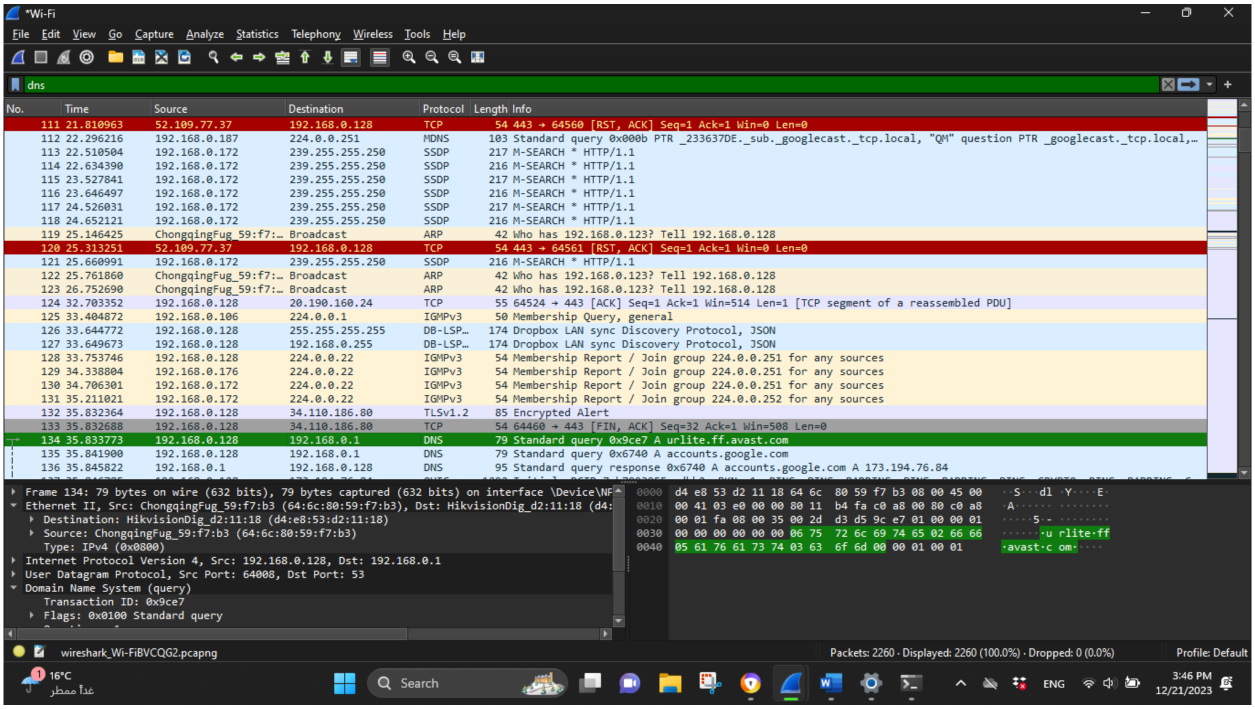
Figure 4:nslookup cornell website in the command line

Here the first two lines show which DNS server was used to get this result, DNS server happens to reside on router as can see 192.168.0.1 is the router used, there is also DNS server name but here is unknown because it maybe unnamed. And the "Non-authoritative answer" in an ‘nslookup’ response indicates that the DNS server providing the information is not the authoritative server for the domain in question. When we perform an ‘nslookup’, our request is typically directed to a DNS server, and that server may or may not be the authoritative server for the domain we're looking up.

1.4. use wireshark to capture some DNS messages.

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).


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
dns
```

```
No.   Time      Source               Destination      Protocol  Length Info
  111 21.810963 52.109.77.37         192.168.0.128    TCP        54 443 → 64560 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
  112 22.296216 192.168.0.187        224.0.0.251      MDNS      103 Standard query 0x000b PTR _233637DE._sub._googlecast._tcp.local, "QM" question PTR _googlecast._tcp.local,…
  113 22.510504 192.168.0.172        239.255.255.250  SSDP      217 M-SEARCH * HTTP/1.1
  114 22.634390 192.168.0.172        239.255.255.250  SSDP      216 M-SEARCH * HTTP/1.1
  115 23.527841 192.168.0.172        239.255.255.250  SSDP      217 M-SEARCH * HTTP/1.1
  116 23.646497 192.168.0.172        239.255.255.250  SSDP      216 M-SEARCH * HTTP/1.1
  117 24.526031 192.168.0.172        239.255.255.250  SSDP      217 M-SEARCH * HTTP/1.1
  118 24.652121 192.168.0.172        239.255.255.250  SSDP      216 M-SEARCH * HTTP/1.1
  119 25.146425 ChongqingFug_59:f7:… Broadcast        ARP        42 Who has 192.168.0.123? Tell 192.168.0.128
  120 25.313251 52.109.77.37         192.168.0.128    TCP        54 443 → 64561 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
  121 25.660991 192.168.0.172        239.255.255.250  SSDP      216 M-SEARCH * HTTP/1.1
  122 25.761860 ChongqingFug_59:f7:… Broadcast        ARP        42 Who has 192.168.0.123? Tell 192.168.0.128
  123 26.752690 ChongqingFug_59:f7:… Broadcast        ARP        42 Who has 192.168.0.123? Tell 192.168.0.128
  124 32.703352 192.168.0.128        20.190.160.24    TCP        55 64524 → 443 [ACK] Seq=1 Ack=1 Win=514 Len=1 [TCP segment of a reassembled PDU]
  125 33.404872 192.168.0.106        224.0.0.1        IGMPv3     50 Membership Query, general
  126 33.644772 192.168.0.128        255.255.255.255  DB-LSP…   174 Dropbox LAN sync Discovery Protocol, JSON
  127 33.649673 192.168.0.128        192.168.0.255    DB-LSP…   174 Dropbox LAN sync Discovery Protocol, JSON
  128 33.753746 192.168.0.128        224.0.0.22       IGMPv3     54 Membership Report / Join group 224.0.0.251 for any sources
  129 34.338804 192.168.0.176        224.0.0.22       IGMPv3     54 Membership Report / Join group 224.0.0.251 for any sources
  130 34.706301 192.168.0.172        224.0.0.22       IGMPv3     54 Membership Report / Join group 224.0.0.251 for any sources
  131 35.211021 192.168.0.172        224.0.0.22       IGMPv3     54 Membership Report / Join group 224.0.0.252 for any sources
  132 35.832364 192.168.0.128        34.110.186.80    TLSv1.2    85 Encrypted Alert
  133 35.832688 192.168.0.128        34.110.186.80    TCP        54 64460 → 443 [FIN, ACK] Seq=32 Ack=1 Win=508 Len=0
  134 35.833773 192.168.0.128        192.168.0.1      DNS        79 Standard query 0x9ce7 A urlite.ff.avast.com
  135 35.841900 192.168.0.128        192.168.0.1      DNS        79 Standard query 0x6740 A accounts.google.com
  136 35.845822 192.168.0.1          192.168.0.128    DNS        95 Standard query response 0x6740 A accounts.google.com A 173.194.76.84
```

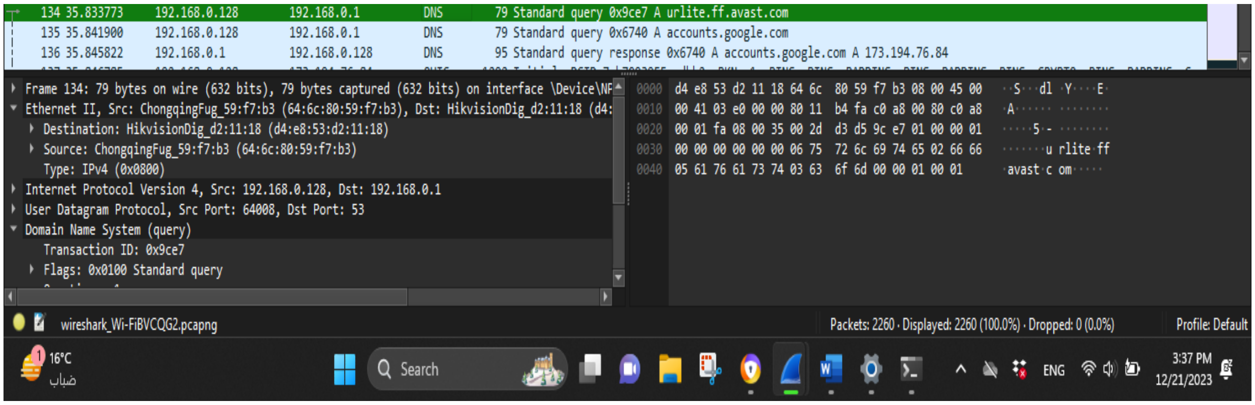
```
› Frame 134: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface \Device\NF
⌄ Ethernet II, Src: ChongqingFug_59:f7:b3 (64:6c:80:59:f7:b3), Dst: HikvisionDig_d2:11:18 (d4:
   › Destination: HikvisionDig_d2:11:18 (d4:e8:53:d2:11:18)
   › Source: ChongqingFug_59:f7:b3 (64:6c:80:59:f7:b3)
     Type: IPv4 (0x0800)
› Internet Protocol Version 4, Src: 192.168.0.128, Dst: 192.168.0.1
› User Datagram Protocol, Src Port: 64008, Dst Port: 53
⌄ Domain Name System (query)
     Transaction ID: 0x9ce7
   › Flags: 0x0100 Standard query
```

```
0000  d4 e8 53 d2 11 18 64 6c  80 59 f7 b3 08 00 45 00   ··S···dl ·Y····E·
0010  00 41 03 e0 00 00 80 11  b4 fa c0 a8 00 80 c0 a8   ·A······ ········
0020  00 01 fa 08 00 35 00 2d  d3 d5 9c e7 01 00 00 01   ·····5·- ········
0030  00 00 00 00 00 00 06 75  72 6c 69 74 65 02 66 66   ·······u rlite·ff
0040  05 61 76 61 73 74 03 63  6f 6d 00 00 01 00 01      ·avast·c om·····
```

wireshark_Wi-FiBVCQG2.pcapng

Packets: 2260 · Displayed: 2260 (100.0%) · Dropped: 0 (0.0%)

Profile: Default



```
  134 35.833773 192.168.0.128        192.168.0.1      DNS        79 Standard query 0x9ce7 A urlite.ff.avast.com
  135 35.841900 192.168.0.128        192.168.0.1      DNS        79 Standard query 0x6740 A accounts.google.com
  136 35.845822 192.168.0.1          192.168.0.128    DNS        95 Standard query response 0x6740 A accounts.google.com A 173.194.76.84
```

```
› Frame 134: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface \Device\NF
⌄ Ethernet II, Src: ChongqingFug_59:f7:b3 (64:6c:80:59:f7:b3), Dst: HikvisionDig_d2:11:18 (d4:
   › Destination: HikvisionDig_d2:11:18 (d4:e8:53:d2:11:18)
   › Source: ChongqingFug_59:f7:b3 (64:6c:80:59:f7:b3)
     Type: IPv4 (0x0800)
› Internet Protocol Version 4, Src: 192.168.0.128, Dst: 192.168.0.1
› User Datagram Protocol, Src Port: 64008, Dst Port: 53
⌄ Domain Name System (query)
     Transaction ID: 0x9ce7
   › Flags: 0x0100 Standard query
```

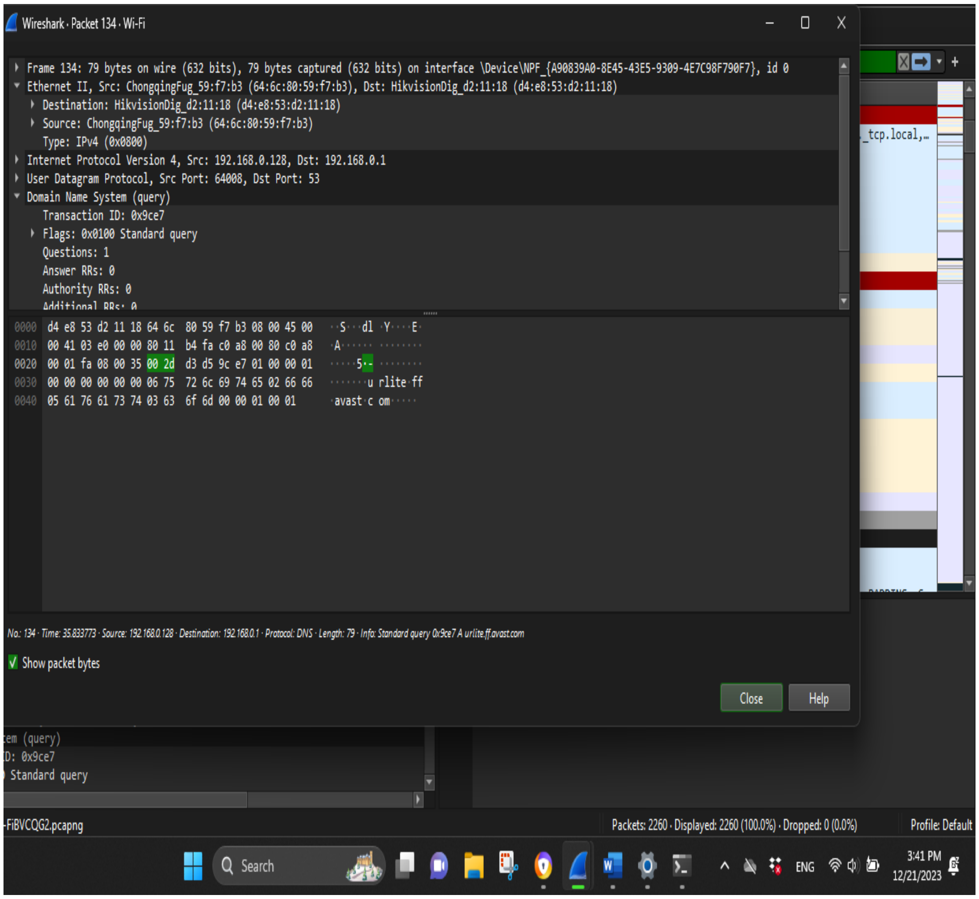
```
0000  d4 e8 53 d2 11 18 64 6c  80 59 f7 b3 08 00 45 00   ··S···dl ·Y····E·
0010  00 41 03 e0 00 00 80 11  b4 fa c0 a8 00 80 c0 a8   ·A······ ········
0020  00 01 fa 08 00 35 00 2d  d3 d5 9c e7 01 00 00 01   ·····5·- ········
0030  00 00 00 00 00 00 06 75  72 6c 69 74 65 02 66 66   ·······u rlite·ff
0040  05 61 76 61 73 74 03 63  6f 6d 00 00 01 00 01      ·avast·c om·····
```

wireshark_Wi-FiBVCQG2.pcapng

Packets: 2260 · Displayed: 2260 (100.0%) · Dropped: 0 (0.0%)

Profile: Default

Figure 5:displaying the DNS messages

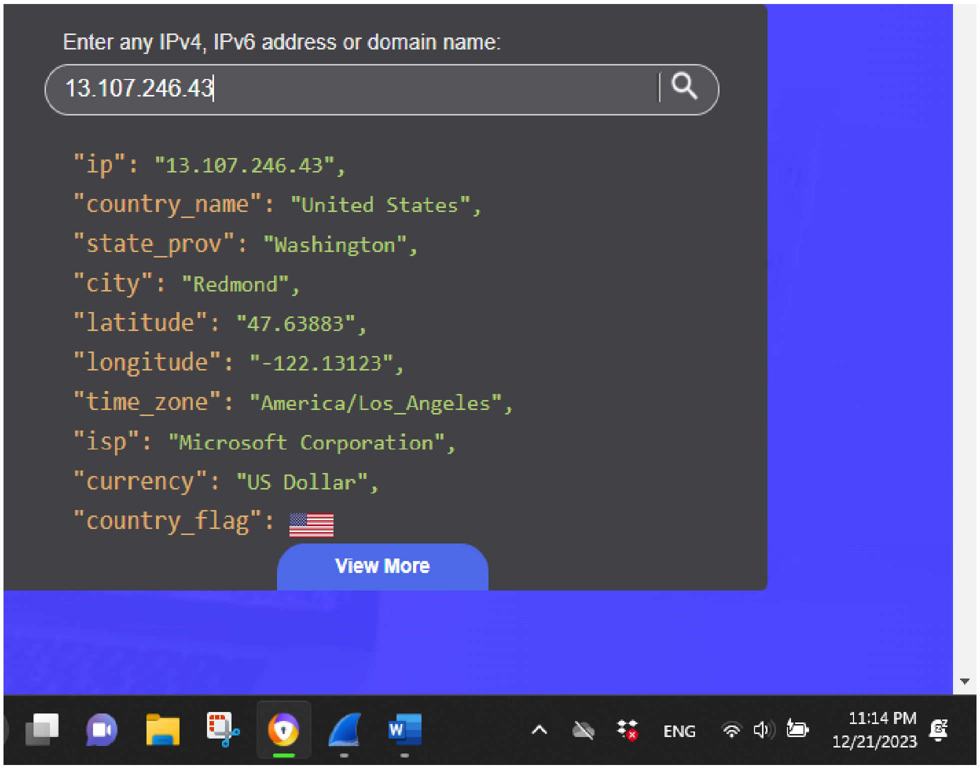

Wireshark · Packet 134 · Wi-Fi
Show packet bytes
Close
Help
-FiBVCQG2.pcapng
Packets: 2260 · Displayed: 2260 (100.0%) · Dropped: 0 (0.0%)
Profile: Default
Search
ENG
3:41 PM
12/21/2023

```
Frame 134: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface \Device\NPF_{A90839A0-8E45-43E5-9309-4E7C98F790F7}, id 0
Ethernet II, Src: ChongqingFug_59:f7:b3 (64:6c:80:59:f7:b3), Dst: HikvisionDig_d2:11:18 (d4:e8:53:d2:11:18)
    Destination: HikvisionDig_d2:11:18 (d4:e8:53:d2:11:18)
    Source: ChongqingFug_59:f7:b3 (64:6c:80:59:f7:b3)
    Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.0.128, Dst: 192.168.0.1
User Datagram Protocol, Src Port: 64008, Dst Port: 53
Domain Name System (query)
    Transaction ID: 0x9ce7
    Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0

0000  d4 e8 53 d2 11 18 64 6c  80 59 f7 b3 08 00 45 00   ··S···dl ·Y····E·
0010  00 41 03 e0 00 00 80 11  b4 fa c0 a8 00 80 c0 a8   ·A······ ········
0020  00 01 fa 08 00 35 00 2d  d3 d5 9c e7 01 00 00 01   ·····5·- ········
0030  00 00 00 00 00 00 06 75  72 6c 69 74 65 02 66 66   ·······u rlite·ff
0040  05 61 76 61 73 74 03 63  6f 6d 00 00 01 00 01      ·avast·c om·····
```

No.: 134 · Time: 35.833773 · Source: 192.168.0.128 · Destination: 192.168.0.1 · Protocol: DNS · Length: 79 · Info: Standard query 0x9ce7 A urlite.ff.avast.com

Figure 6:information about one of the DNS messages

As seen above, we have captured many DNS messages, each one of them has its own frame number that is a unique identifier for each captured packet, also there is a time that its the time elapsed since the start of the capture, and the source and destination IP addresses (or MAC addresses, depending on the layer) of the communication. The protocol Indicates the protocol used for the communication, such as TCP, UDP, or DNS, in addition the info column provides a brief summary of the packet content. For example, for DNS packets, you might see information about the DNS query or response.



Enter any IPv4, IPv6 address or domain name:

13.107.246.43

```
"ip": "13.107.246.43",
"country_name": "United States",
"state_prov": "Washington",
"city": "Redmond",
"latitude": "47.63883",
"longitude": "-122.13123",
"time_zone": "America/Los_Angeles",
"isp": "Microsoft Corporation",
"currency": "US Dollar",
"country_flag":
```

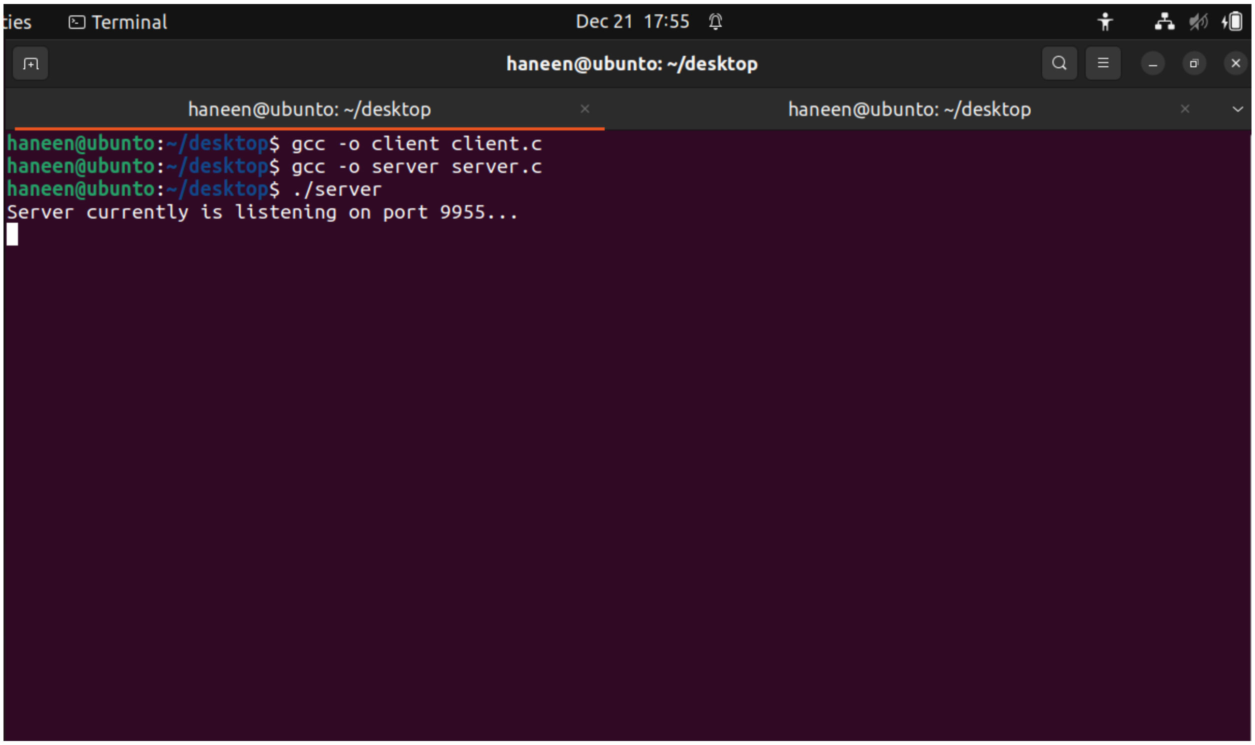
Figure 7:IP domain

As can be seen above, we used the IP address I got from pinging www.cornell.edu in an IP geolocation, to know in which country this IP address is associated with, so we discovered that the country is the United States, therefor the response we have got is from USA.

2. Part two: socket programming using TCP

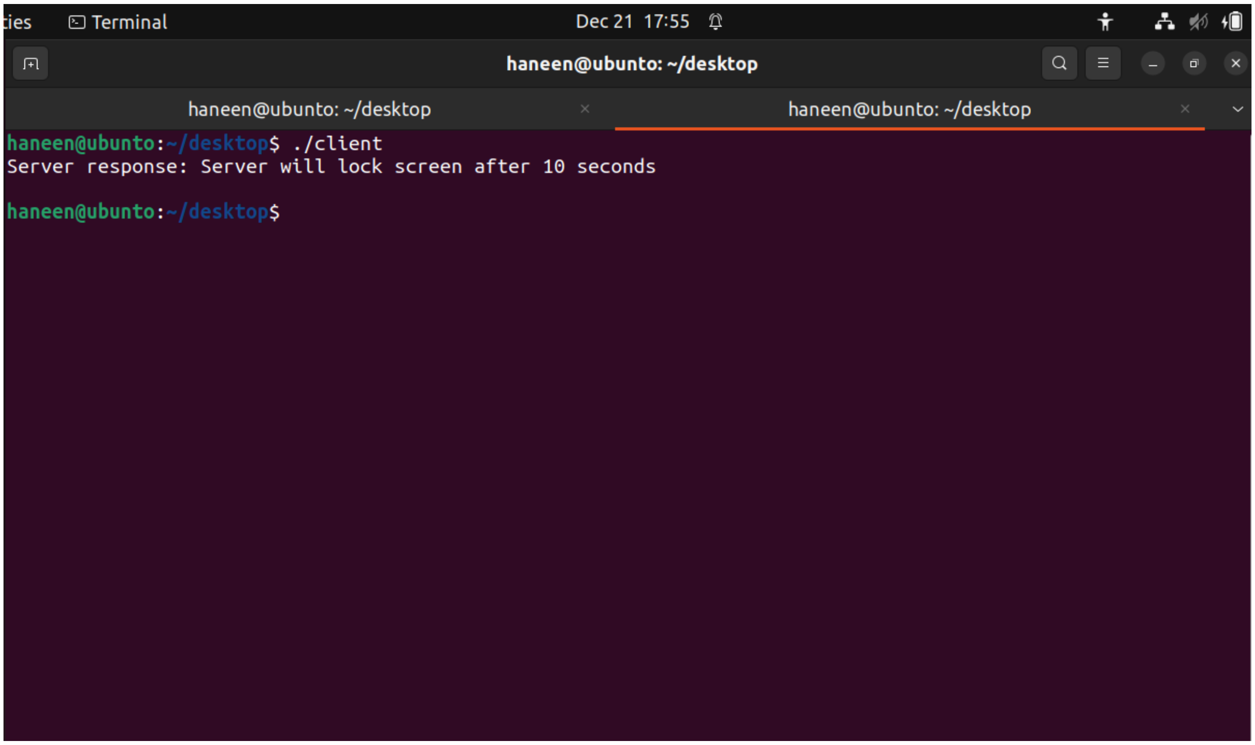
In this part socket programming was used to implement TCP client and server applications in C language. The server should listen on port 9955. The server waits for a message from a client. If the message is with one of the students ID the sever should do the following:

1. display a message on the server side that the OS will lock screen after 10 seconds
2. send a message to the client that the sever will lock screen after 10 seconds
3. then wait 10 seconds
4. then call a function lock the screen of the operating system (windows or Linux or MAC).



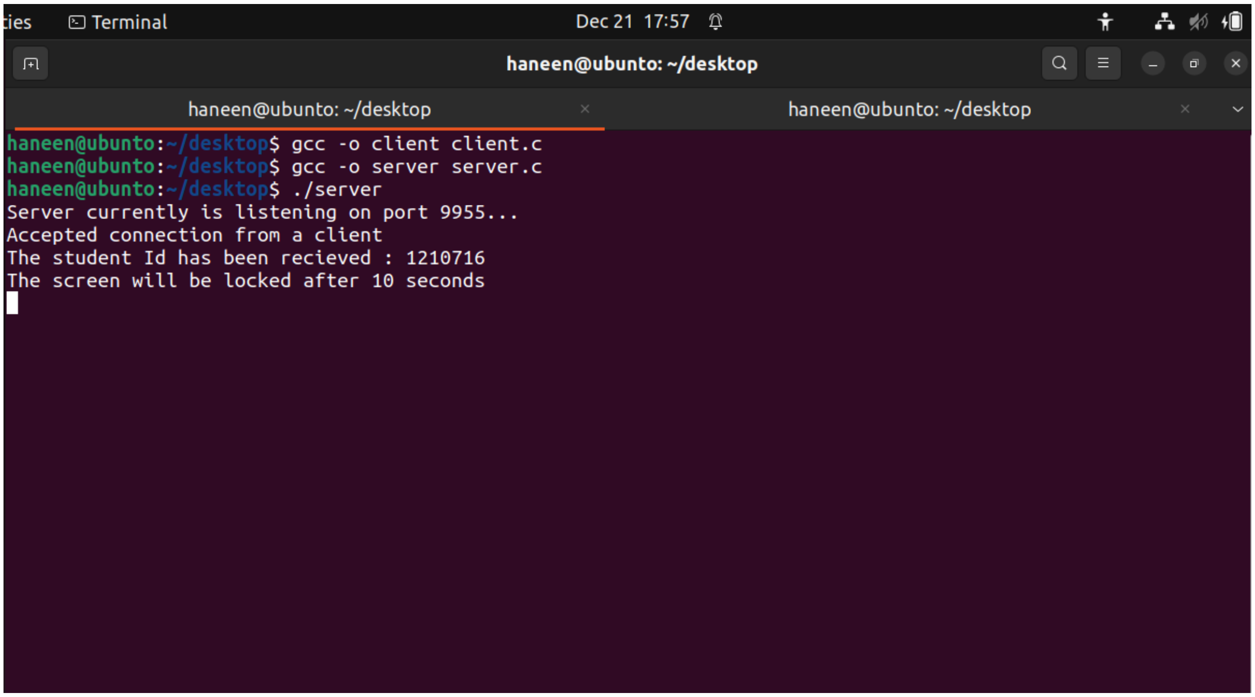
```
haneen@ubunto:~/desktop$ gcc -o client client.c
haneen@ubunto:~/desktop$ gcc -o server server.c
haneen@ubunto:~/desktop$ ./server
Server currently is listening on port 9955...
```

Figure 8:The server terminal



```
haneen@ubunto:~/desktop$ ./client
Server response: Server will lock screen after 10 seconds

haneen@ubunto:~/desktop$
```

Figure 10: the client terminal



```
haneen@ubunto:~/desktop$ gcc -o client client.c
haneen@ubunto:~/desktop$ gcc -o server server.c
haneen@ubunto:~/desktop$ ./server
Server currently is listening on port 9955...
Accepted connection from a client
The student Id has been recieved : 1210716
The screen will be locked after 10 seconds
```

Figure 9:the server terminal after running the program

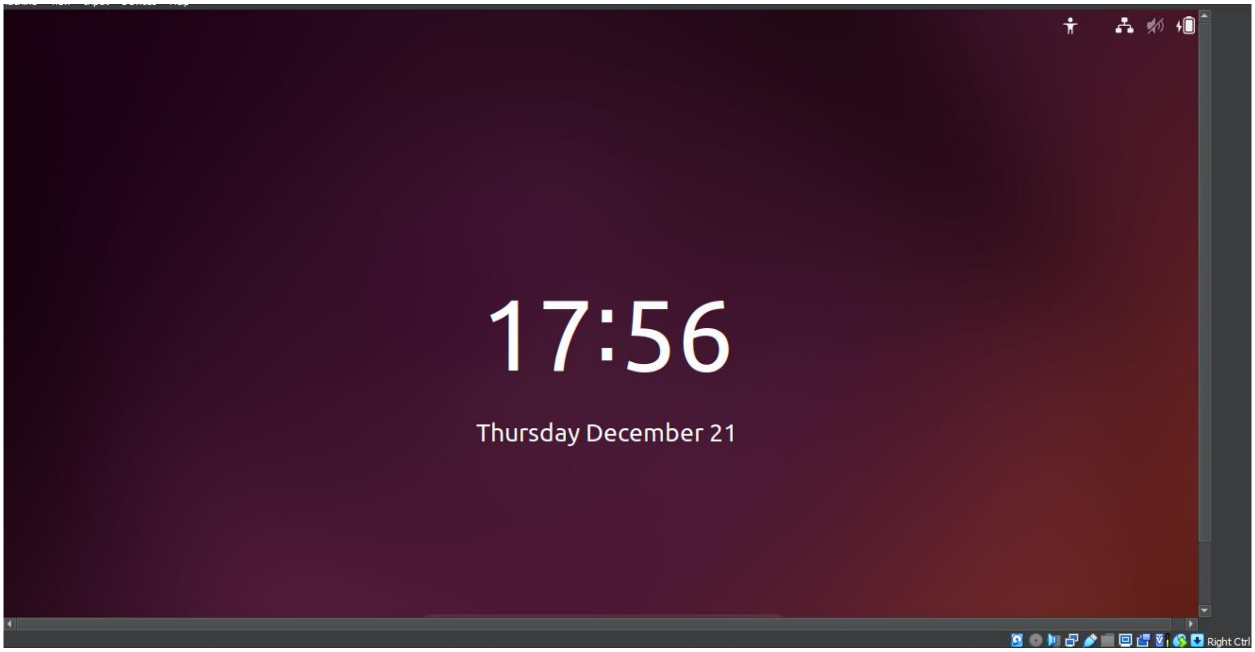
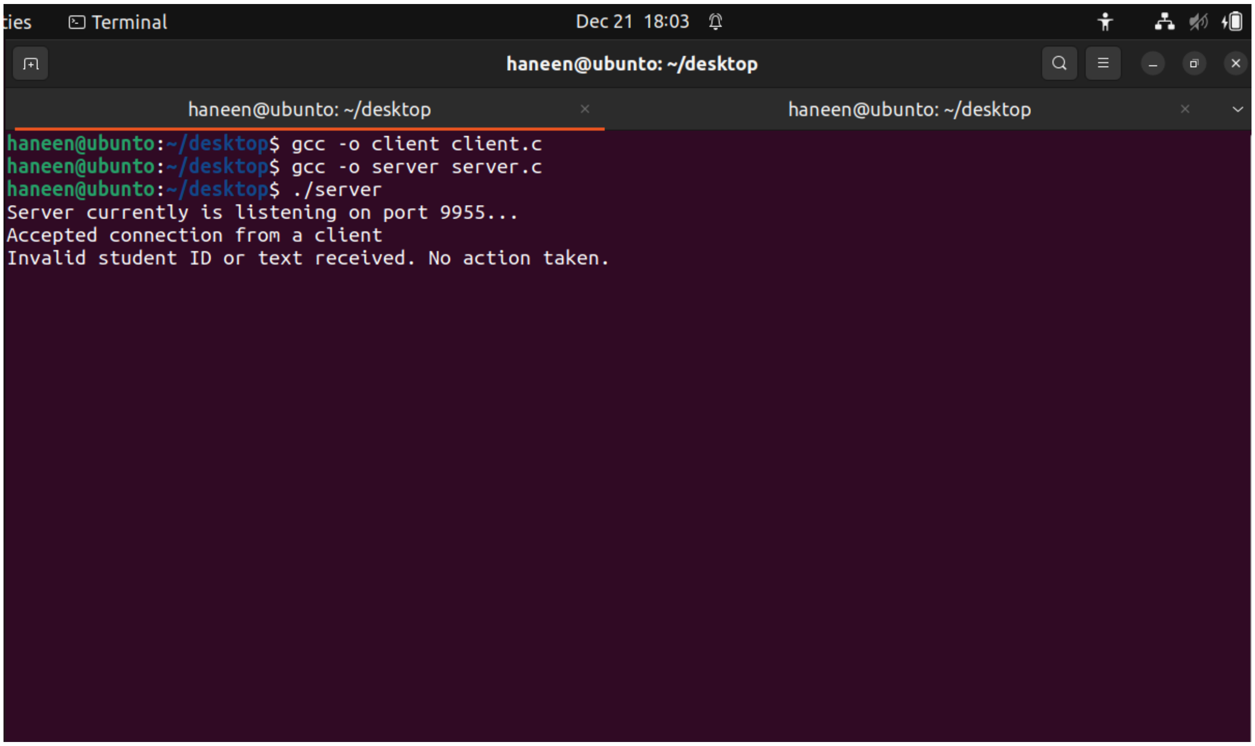


Figure 11:the screen is locked

As can be seen above, when we add one of our id number and then run the program, there was no errors, so it worked correctly and locked the screen after 10 seconds.

But if we change the ID number to an invalid number, we will get this result.



```
haneen@ubunto:~/desktop$ gcc -o client client.c
haneen@ubunto:~/desktop$ gcc -o server server.c
haneen@ubunto:~/desktop$ ./server
Server currently is listening on port 9955...
Accepted connection from a client
Invalid student ID or text received. No action taken.
```

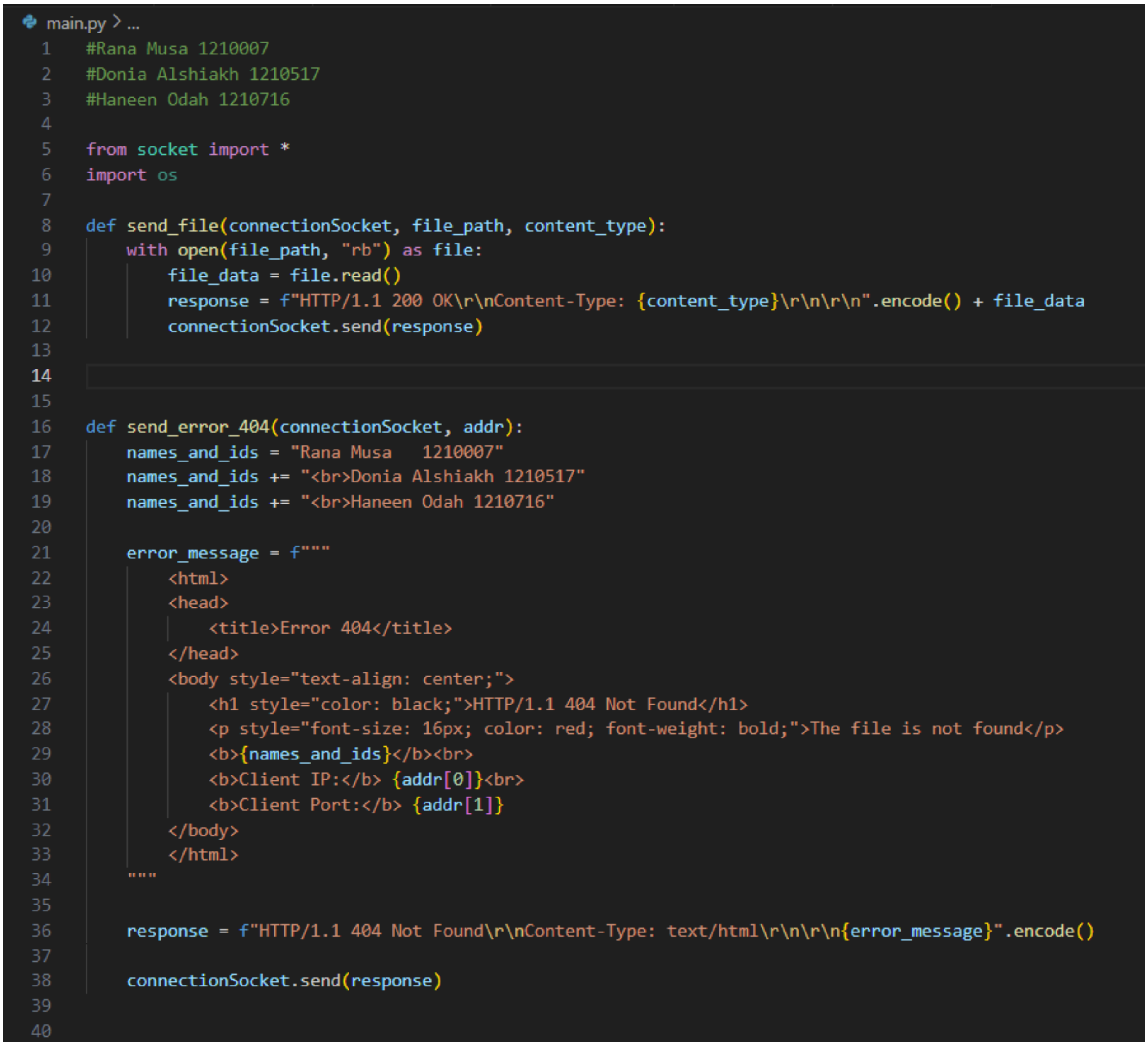
Figure 12:entering invalid id

Part 3. Web Server Application

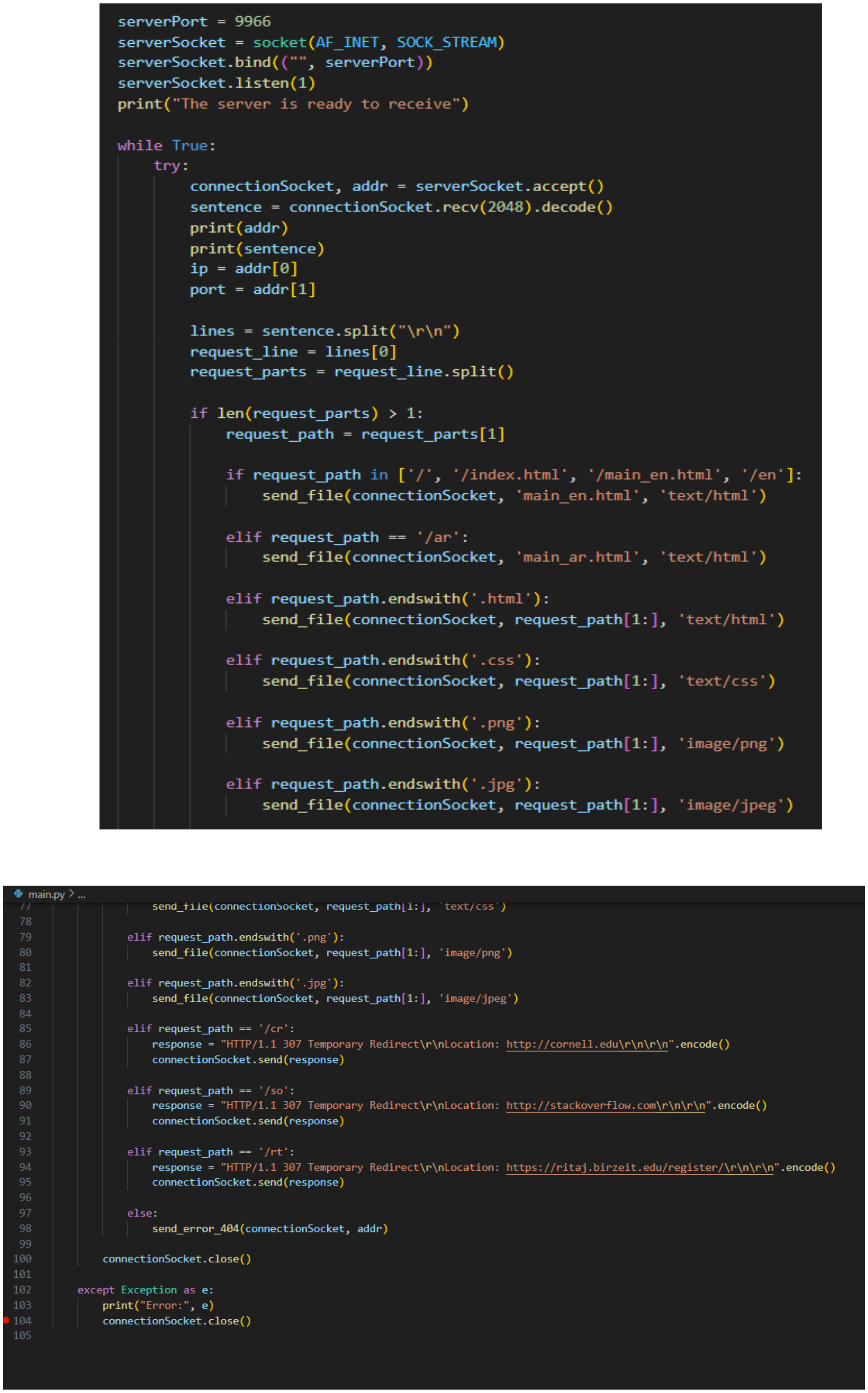
In this part socket programming was used to implement a simple but a complete web server listening on port 9966 in python language

3.1. OUR CODE:

3.1.1. Main.py code:



```python
#Rana Musa 1210007
#Donia Alshiakh 1210517
#Haneen Odah 1210716

from socket import *
import os

def send_file(connectionSocket, file_path, content_type):
    with open(file_path, "rb") as file:
        file_data = file.read()
        response = f"HTTP/1.1 200 OK\r\nContent-Type: {content_type}\r\n\r\n".encode() + file_data
        connectionSocket.send(response)



def send_error_404(connectionSocket, addr):
    names_and_ids = "Rana Musa   1210007"
    names_and_ids += "<br>Donia Alshiakh 1210517"
    names_and_ids += "<br>Haneen Odah 1210716"

    error_message = f"""
        <html>
        <head>
            <title>Error 404</title>
        </head>
        <body style="text-align: center;">
            <h1 style="color: black;">HTTP/1.1 404 Not Found</h1>
            <p style="font-size: 16px; color: red; font-weight: bold;">The file is not found</p>
            <b>{names_and_ids}</b><br>
            <b>Client IP:</b> {addr[0]}<br>
            <b>Client Port:</b> {addr[1]}
        </body>
        </html>
    """

    response = f"HTTP/1.1 404 Not Found\r\nContent-Type: text/html\r\n\r\n{error_message}".encode()

    connectionSocket.send(response)


```



```python
serverPort = 9966
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind(("", serverPort))
serverSocket.listen(1)
print("The server is ready to receive")

while True:
    try:
        connectionSocket, addr = serverSocket.accept()
        sentence = connectionSocket.recv(2048).decode()
        print(addr)
        print(sentence)
        ip = addr[0]
        port = addr[1]

        lines = sentence.split("\r\n")
        request_line = lines[0]
        request_parts = request_line.split()

        if len(request_parts) > 1:
            request_path = request_parts[1]

            if request_path in ['/', '/index.html', '/main_en.html', '/en']:
                send_file(connectionSocket, 'main_en.html', 'text/html')

            elif request_path == '/ar':
                send_file(connectionSocket, 'main_ar.html', 'text/html')

            elif request_path.endswith('.html'):
                send_file(connectionSocket, request_path[1:], 'text/html')

            elif request_path.endswith('.css'):
                send_file(connectionSocket, request_path[1:], 'text/css')

            elif request_path.endswith('.png'):
                send_file(connectionSocket, request_path[1:], 'image/png')

            elif request_path.endswith('.jpg'):
                send_file(connectionSocket, request_path[1:], 'image/jpeg')
```

main.py > ...

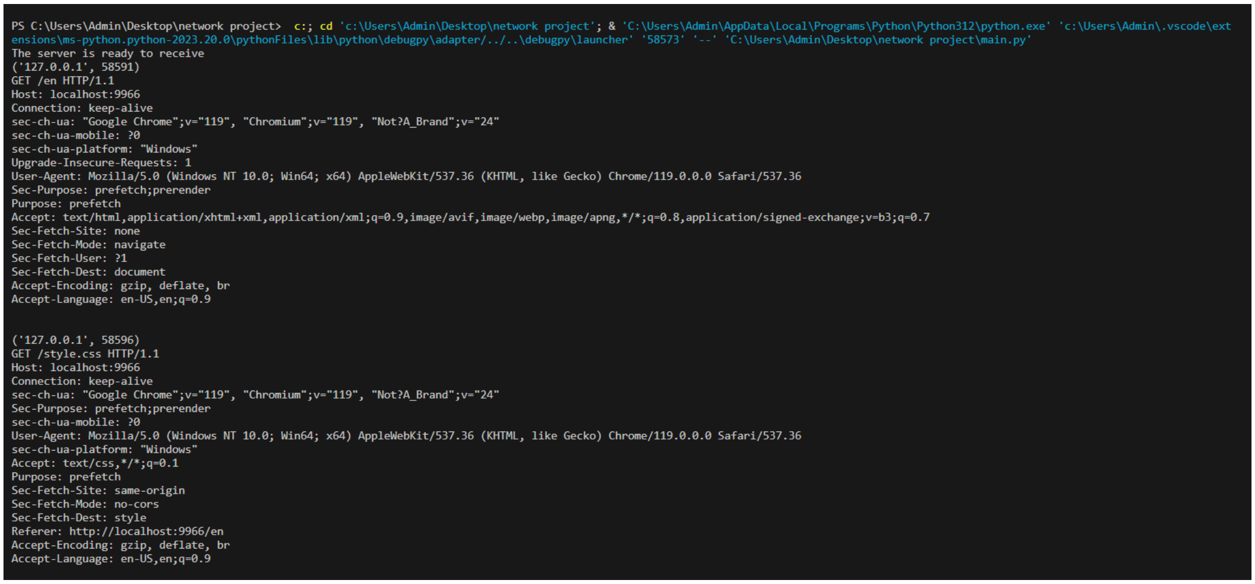
```python
                send_file(connectionSocket, request_path[1:], 'text/css')

            elif request_path.endswith('.png'):
                send_file(connectionSocket, request_path[1:], 'image/png')

            elif request_path.endswith('.jpg'):
                send_file(connectionSocket, request_path[1:], 'image/jpeg')

            elif request_path == '/cr':
                response = "HTTP/1.1 307 Temporary Redirect\r\nLocation: http://cornell.edu\r\n\r\n".encode()
                connectionSocket.send(response)

            elif request_path == '/so':
                response = "HTTP/1.1 307 Temporary Redirect\r\nLocation: http://stackoverflow.com\r\n\r\n".encode()
                connectionSocket.send(response)

            elif request_path == '/rt':
                response = "HTTP/1.1 307 Temporary Redirect\r\nLocation: https://ritaj.birzeit.edu/register/\r\n\r\n".encode()
                connectionSocket.send(response)

            else:
                send_error_404(connectionSocket, addr)

        connectionSocket.close()

    except Exception as e:
        print("Error:", e)
        connectionSocket.close()
```

Figure 13:our code in python for web server

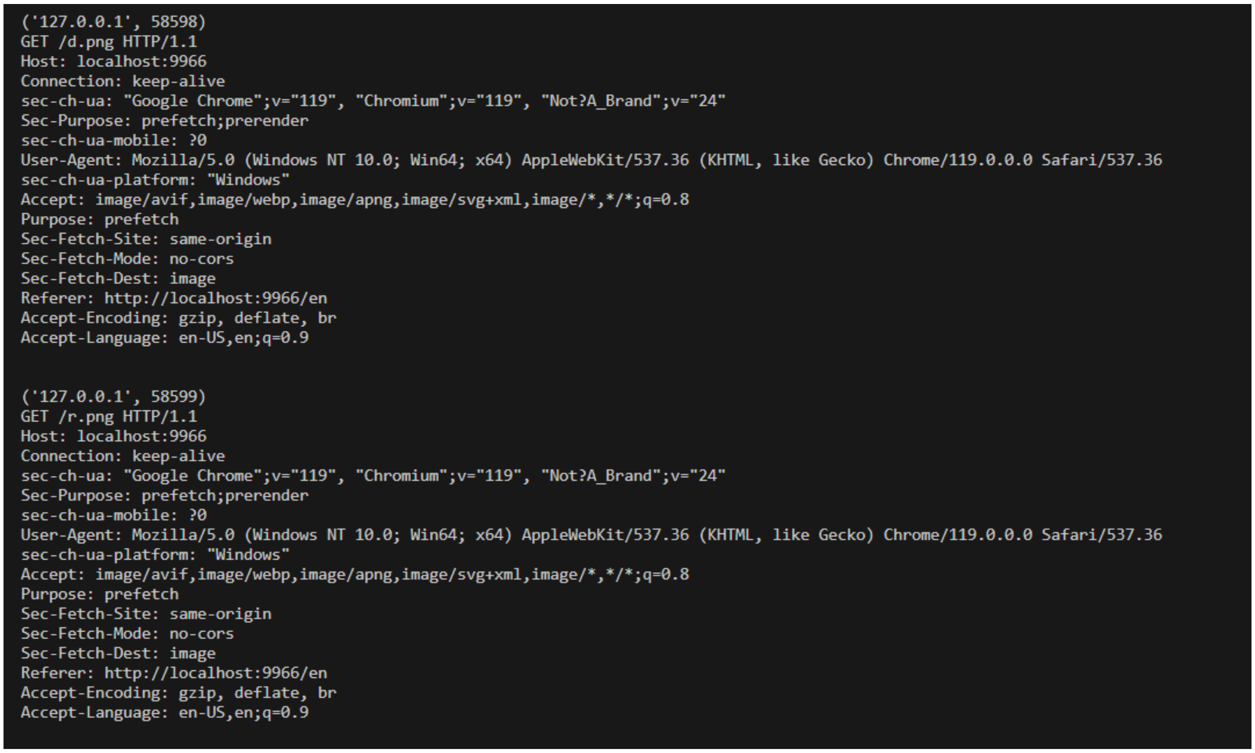
3.2. Testing the application on the same computer

- if the request is **/ or /index.html or /main_en.html or /en (for example localhost:9966/ or localhost:9966/en)** then the server should send main_en.html file with Content-Type: text/html.

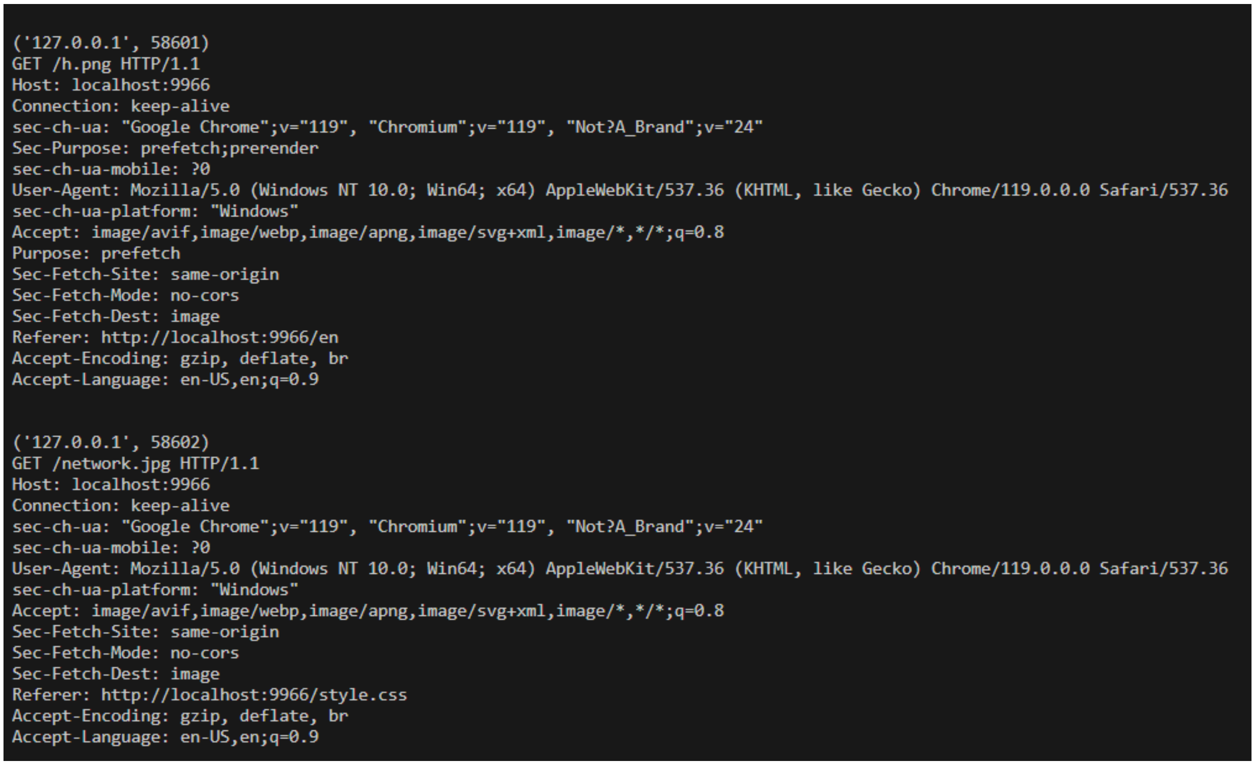
3.2.1. on command line (en):



```
PS C:\Users\Admin\Desktop\network project>  c:; cd 'c:\Users\Admin\Desktop\network project'; & 'C:\Users\Admin\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\Admin\.vscode\ext
ensions\ms-python.python-2023.20.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '58573' '--' 'C:\Users\Admin\Desktop\network project\main.py'
The server is ready to receive
('127.0.0.1', 58591)
GET /en HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 Safari/537.36
Sec-Purpose: prefetch;prerender
Purpose: prefetch
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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 58596)
GET /style.css HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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) Chrome/119.0.0.0 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:9966/en
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```



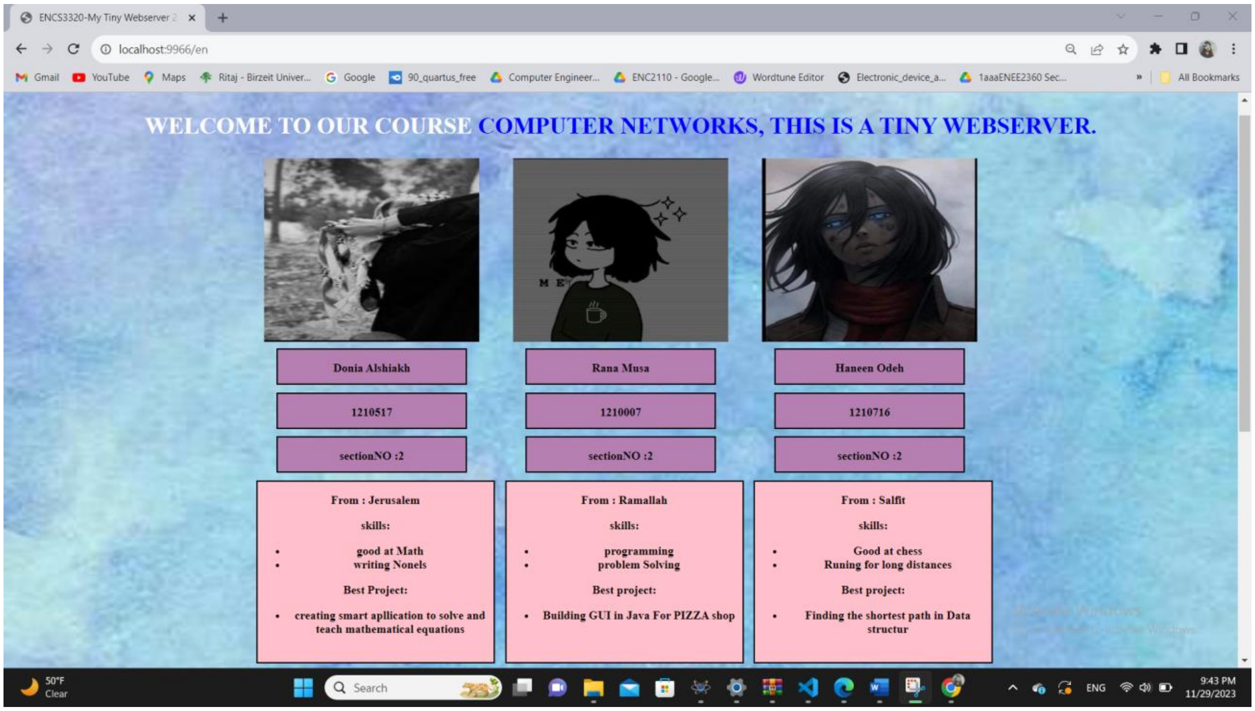
```
('127.0.0.1', 58598)
GET /d.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
Sec-Purpose: prefetch;prerender
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Purpose: prefetch
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:9966/en
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 58599)
GET /r.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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) Chrome/119.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Purpose: prefetch
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:9966/en
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```



```
('127.0.0.1', 58601)
GET /h.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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) Chrome/119.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Purpose: prefetch
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:9966/en
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 58602)
GET /network.jpg HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/style.css
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```

Figure 14:the request messages for / or /index.html or /main_en.html or /en

3.2.2. On the browser (en)



WELCOME TO OUR COURSE COMPUTER NETWORKS, THIS IS A TINY WEBSERVER.

Donia Alshiakh

1210517

sectionNO :2

From : Jerusalem

skills:

- good at Math
- writing Nonels

Best Project:

- creating smart apllication to solve and teach mathematical equations

Rana Musa

1210007

sectionNO :2

From : Ramallah

skills:

- programming
- problem Solving

Best project:

- Building GUI in Java For PIZZA shop

Haneen Odeh

1210716

sectionNO :2

From : Salfit

skills:

- Good at chess
- Runing for long distances

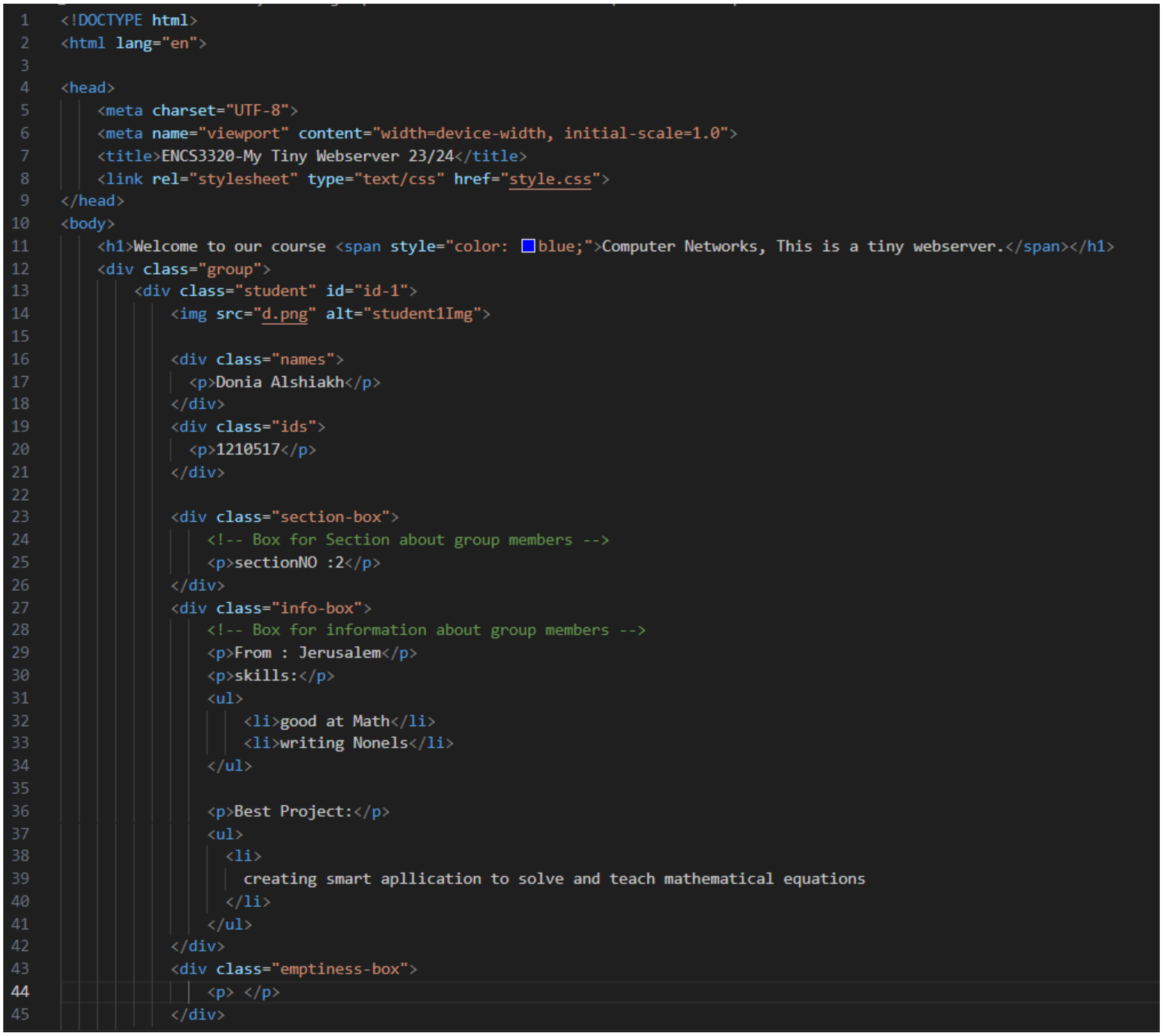
Best project:

- Finding the shortest path in Data structur

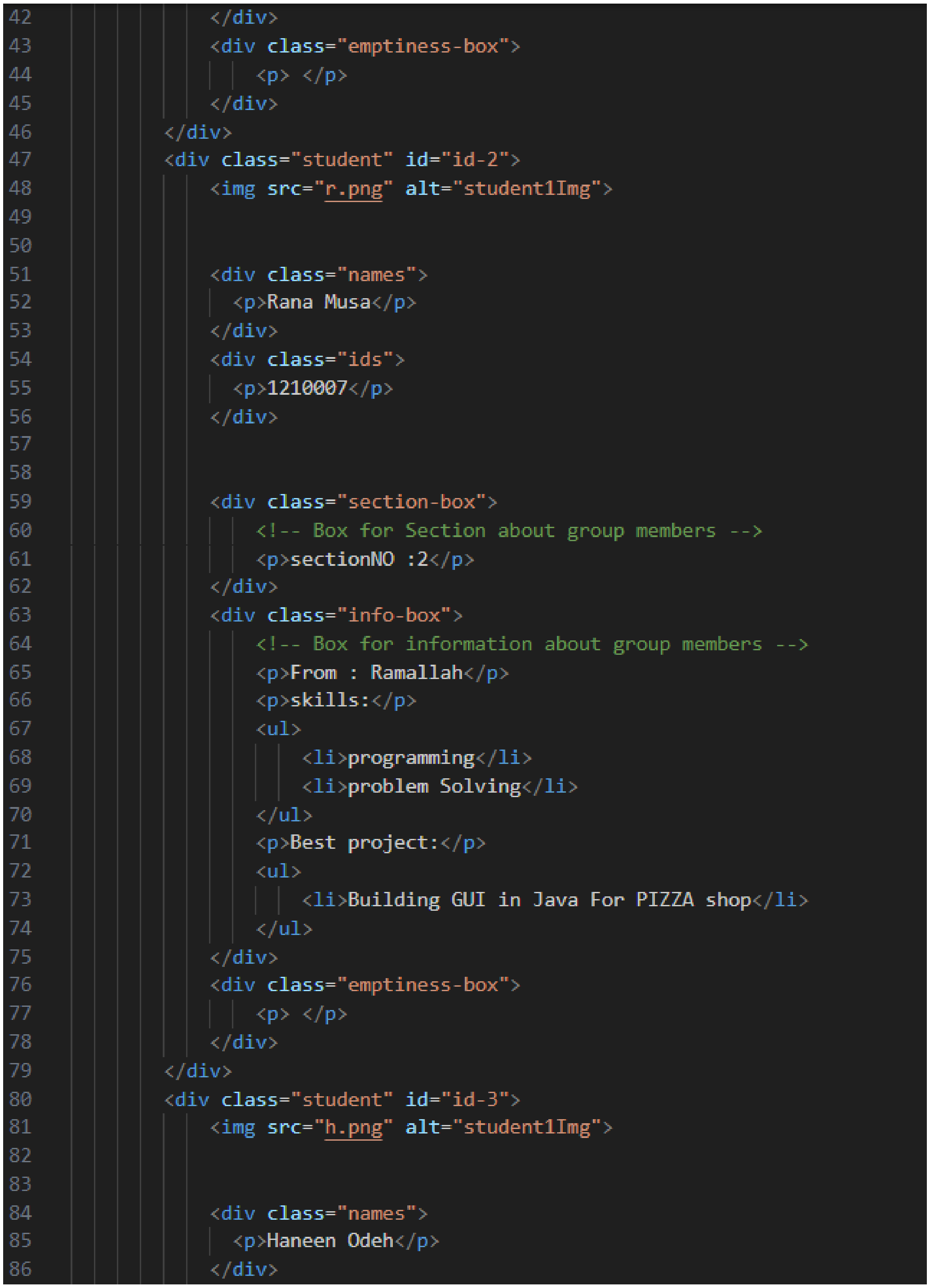
Figure 15:the result of the request in the web page in English

For the HTML code, “ENCS3320-My Tiny Webserver” in the title, “Welcome to our course computer networks” and our names and IDs were printed on the page of our website. And some information about us. This is the page of the English version of the website. All these texts were styled in the CSS code.

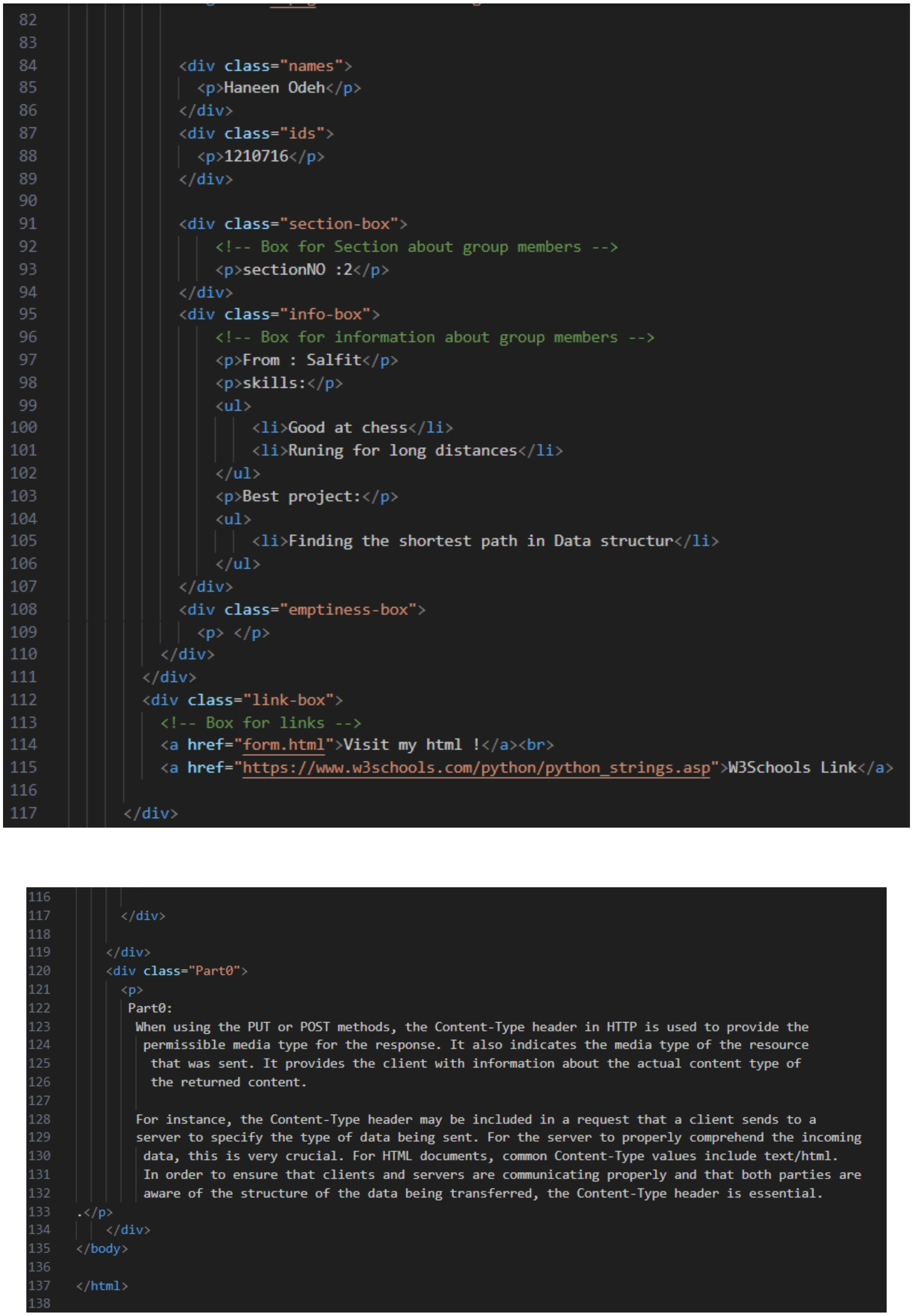
3.2.3. main_en.html code:



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

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>ENCS3320-My Tiny Webserver 23/24</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
    <h1>Welcome to our course <span style="color: blue;">Computer Networks, This is a tiny webserver.</span></h1>
    <div class="group">
        <div class="student" id="id-1">
            <img src="d.png" alt="student1Img">

            <div class="names">
              <p>Donia Alshiakh</p>
            </div>
            <div class="ids">
              <p>1210517</p>
            </div>

            <div class="section-box">
                <!-- Box for Section about group members -->
                <p>sectionNO :2</p>
            </div>
            <div class="info-box">
                <!-- Box for information about group members -->
                <p>From : Jerusalem</p>
                <p>skills:</p>
                <ul>
                    <li>good at Math</li>
                    <li>writing Nonels</li>
                </ul>

                <p>Best Project:</p>
                <ul>
                  <li>
                    creating smart apllication to solve and teach mathematical equations
                  </li>
                </ul>
            </div>
            <div class="emptiness-box">
                <p> </p>
            </div>
```



```
              </div>
              <div class="emptiness-box">
                  <p> </p>
              </div>
          </div>
          <div class="student" id="id-2">
              <img src="r.png" alt="student1Img">


              <div class="names">
                <p>Rana Musa</p>
              </div>
              <div class="ids">
                <p>1210007</p>
              </div>


              <div class="section-box">
                  <!-- Box for Section about group members -->
                  <p>sectionNO :2</p>
              </div>
              <div class="info-box">
                  <!-- Box for information about group members -->
                  <p>From : Ramallah</p>
                  <p>skills:</p>
                  <ul>
                      <li>programming</li>
                      <li>problem Solving</li>
                  </ul>
                  <p>Best project:</p>
                  <ul>
                      <li>Building GUI in Java For PIZZA shop</li>
                  </ul>
              </div>
              <div class="emptiness-box">
                  <p> </p>
              </div>
          </div>
          <div class="student" id="id-3">
              <img src="h.png" alt="student1Img">


              <div class="names">
                <p>Haneen Odeh</p>
              </div>
```

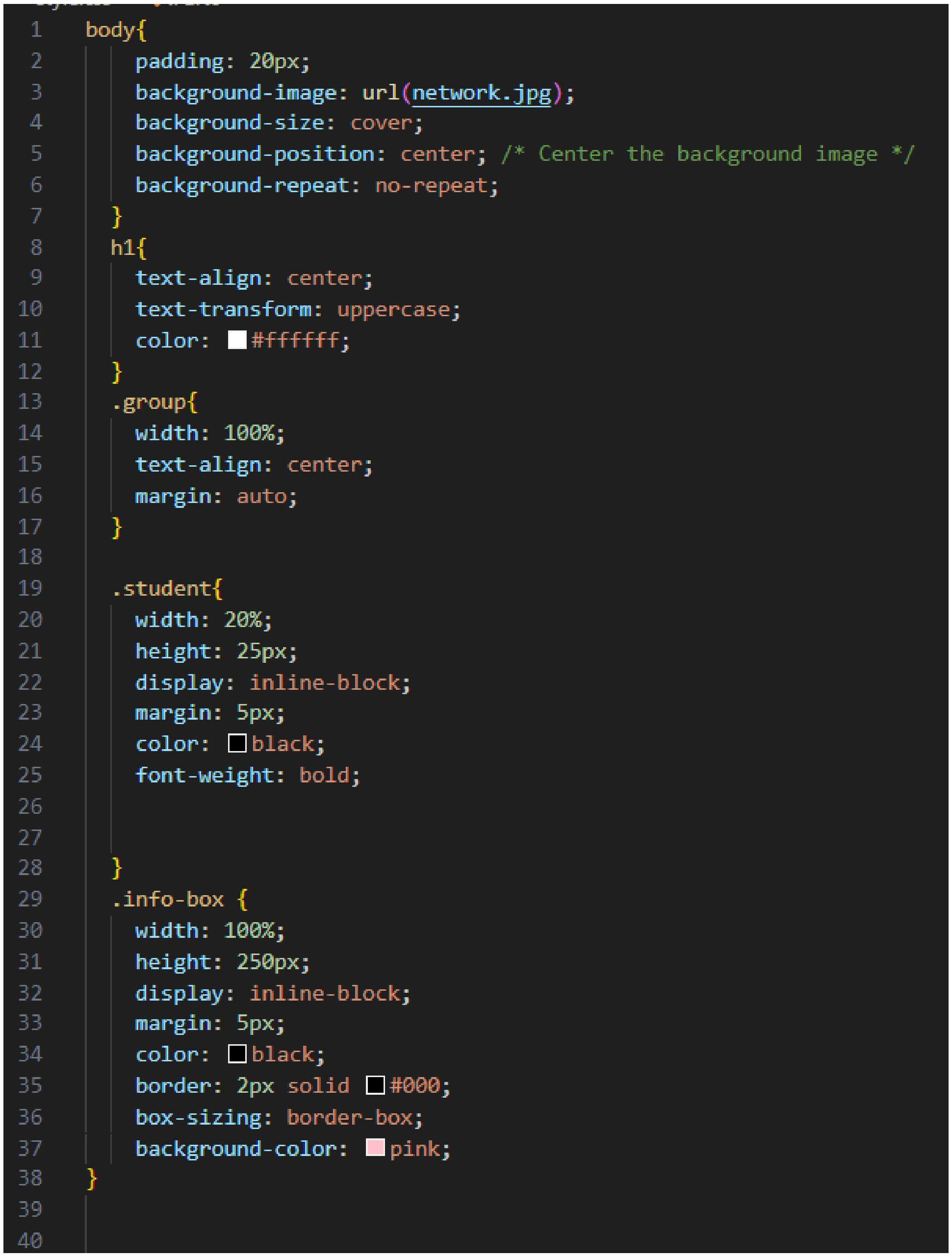


```html


            <div class="names">
              <p>Haneen Odeh</p>
            </div>
            <div class="ids">
              <p>1210716</p>
            </div>

            <div class="section-box">
                <!-- Box for Section about group members -->
                <p>sectionNO :2</p>
            </div>
            <div class="info-box">
                <!-- Box for information about group members -->
                <p>From : Salfit</p>
                <p>skills:</p>
                <ul>
                    <li>Good at chess</li>
                    <li>Runing for long distances</li>
                </ul>
                <p>Best project:</p>
                <ul>
                    <li>Finding the shortest path in Data structur</li>
                </ul>
            </div>
            <div class="emptiness-box">
              <p> </p>
          </div>
        </div>
        <div class="link-box">
          <!-- Box for links -->
          <a href="form.html">Visit my html !</a><br>
          <a href="https://www.w3schools.com/python/python_strings.asp">W3Schools Link</a>

      </div>
```

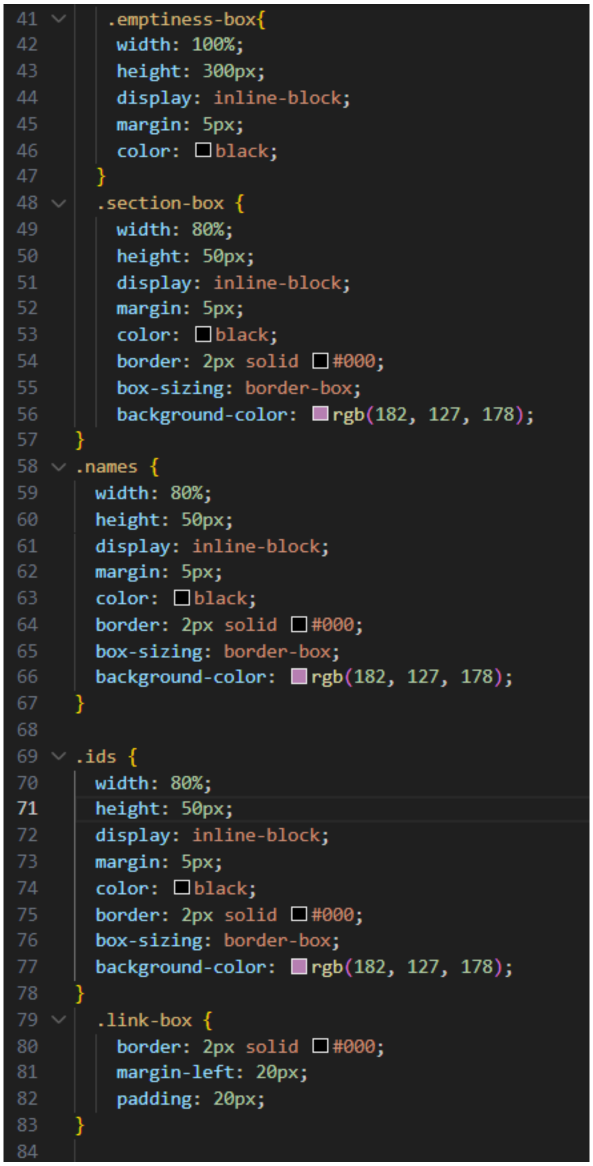
```html

        </div>

      </div>
      <div class="Part0">
        <p>
         Part0:
          When using the PUT or POST methods, the Content-Type header in HTTP is used to provide the
           permissible media type for the response. It also indicates the media type of the resource
            that was sent. It provides the client with information about the actual content type of
            the returned content.

          For instance, the Content-Type header may be included in a request that a client sends to a
          server to specify the type of data being sent. For the server to properly comprehend the incoming
           data, this is very crucial. For HTML documents, common Content-Type values include text/html.
           In order to ensure that clients and servers are communicating properly and that both parties are
           aware of the structure of the data being transferred, the Content-Type header is essential.
  .</p>
      </div>
  </body>

  </html>

```

Figure 16: main_en.html code

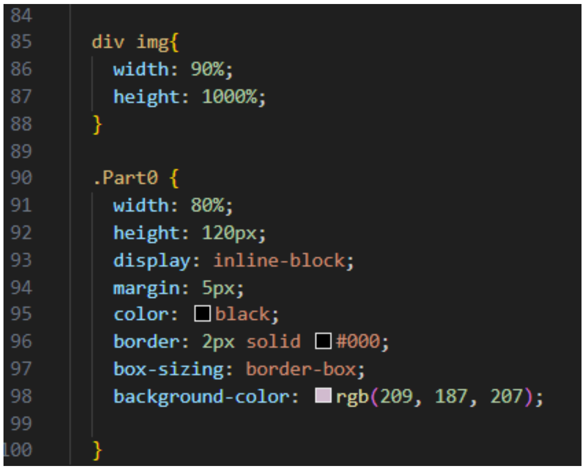
3.2.4. style.css code:



```css
body{
    padding: 20px;
    background-image: url(network.jpg);
    background-size: cover;
    background-position: center; /* Center the background image */
    background-repeat: no-repeat;
  }
  h1{
    text-align: center;
    text-transform: uppercase;
    color: #ffffff;
  }
  .group{
    width: 100%;
    text-align: center;
    margin: auto;
  }

  .student{
    width: 20%;
    height: 25px;
    display: inline-block;
    margin: 5px;
    color: black;
    font-weight: bold;


  }
  .info-box {
    width: 100%;
    height: 250px;
    display: inline-block;
    margin: 5px;
    color: black;
    border: 2px solid #000;
    box-sizing: border-box;
    background-color: pink;
}


```



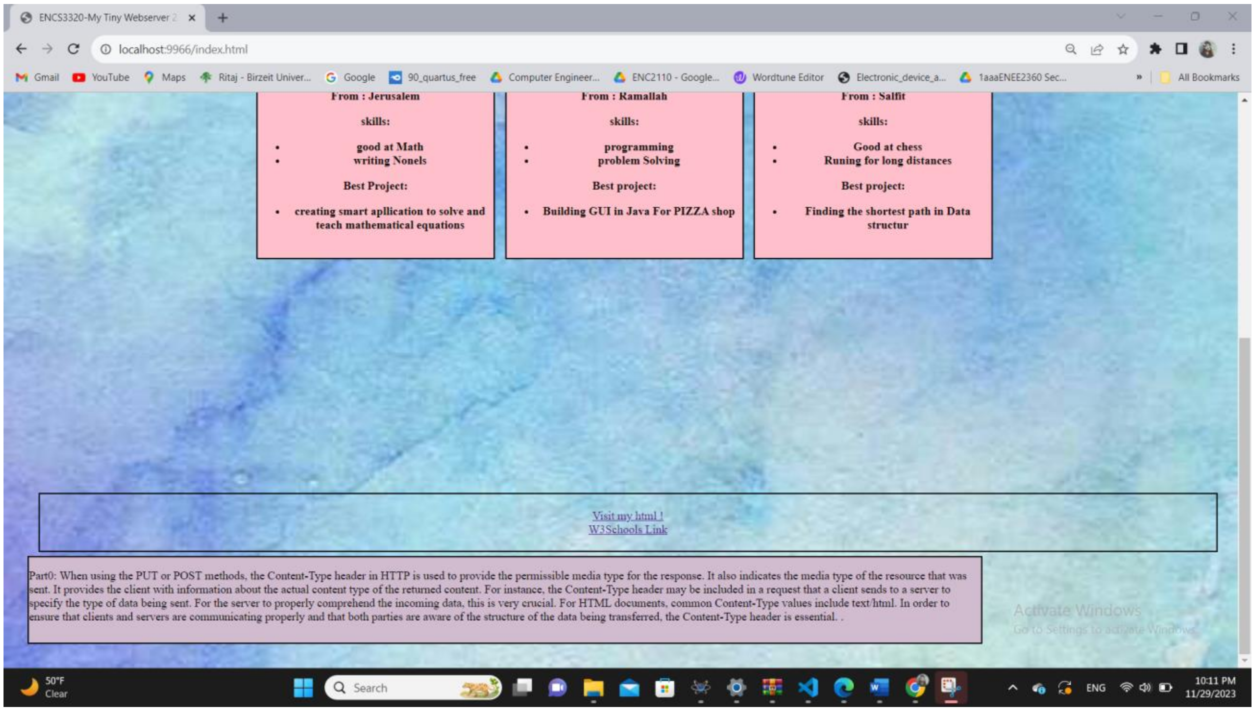
```
   .emptiness-box{
    width: 100%;
    height: 300px;
    display: inline-block;
    margin: 5px;
    color: black;
  }
  .section-box {
    width: 80%;
    height: 50px;
    display: inline-block;
    margin: 5px;
    color: black;
    border: 2px solid #000;
    box-sizing: border-box;
    background-color: rgb(182, 127, 178);
}
.names {
  width: 80%;
  height: 50px;
  display: inline-block;
  margin: 5px;
  color: black;
  border: 2px solid #000;
  box-sizing: border-box;
  background-color: rgb(182, 127, 178);
}

.ids {
  width: 80%;
  height: 50px;
  display: inline-block;
  margin: 5px;
  color: black;
  border: 2px solid #000;
  box-sizing: border-box;
  background-color: rgb(182, 127, 178);
}
  .link-box {
    border: 2px solid #000;
    margin-left: 20px;
    padding: 20px;
}

```



```

  div img{
    width: 90%;
    height: 1000%;
  }

  .Part0 {
    width: 80%;
    height: 120px;
    display: inline-block;
    margin: 5px;
    color: black;
    border: 2px solid #000;
    box-sizing: border-box;
    background-color: rgb(209, 187, 207);

  }
```

Figure 17:style.css code

3.2.5. Summarize point 0 in a box



From : Jerusalem

skills:

- good at Math
- writing Nonels

Best Project:

- creating smart apllication to solve and teach mathematical equations

From : Ramallah

skills:

- programming
- problem Solving

Best project:

- Building GUI in Java For PIZZA shop

From : Salfit

skills:

- Good at chess
- Runing for long distances

Best project:

- Finding the shortest path in Data structur

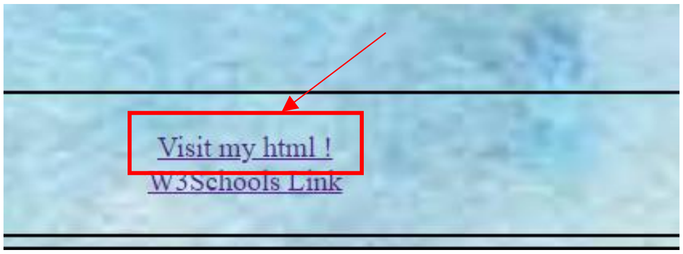
Visit my html !
W3Schools Link

Part0: When using the PUT or POST methods, the Content-Type header in HTTP is used to provide the permissible media type for the response. It also indicates the media type of the resource that was sent. It provides the client with information about the actual content type of the returned content. For instance, the Content-Type header may be included in a request that a client sends to a server to specify the type of data being sent. For the server to properly comprehend the incoming data, this is very crucial. For HTML documents, common Content-Type values include text/html. In order to ensure that clients and servers are communicating properly and that both parties are aware of the structure of the data being transferred, the Content-Type header is essential. .

Figure 18: part0 in webpage

3.2.6. When "visit my html" clicked

When "visit my html" clicked, then a local html file will appear.


Visit my html !
W3Schools Link

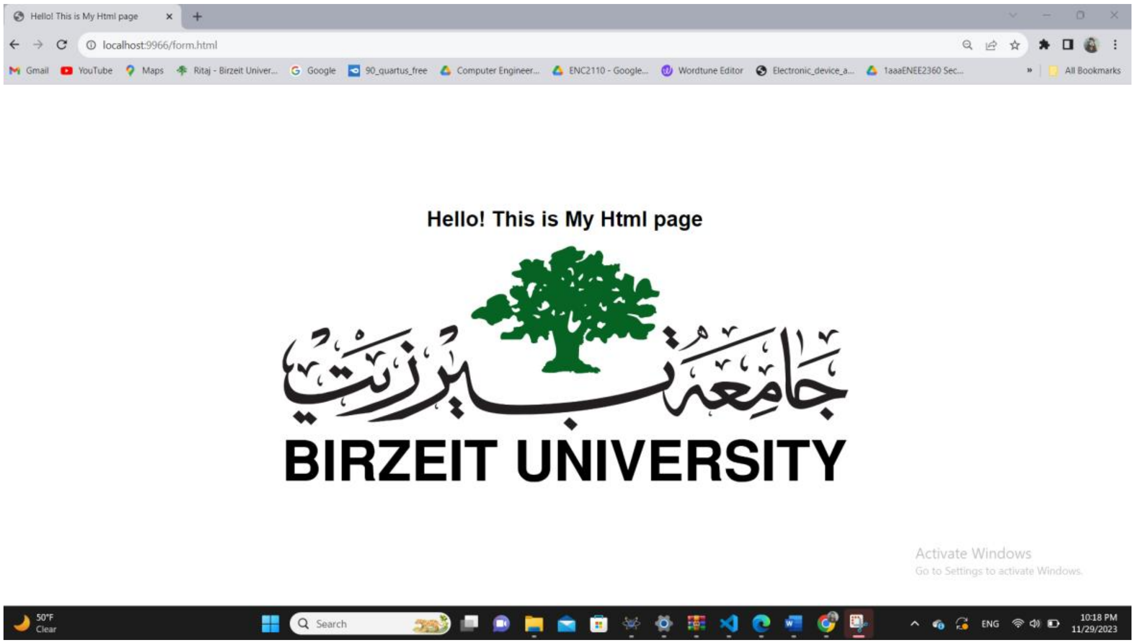
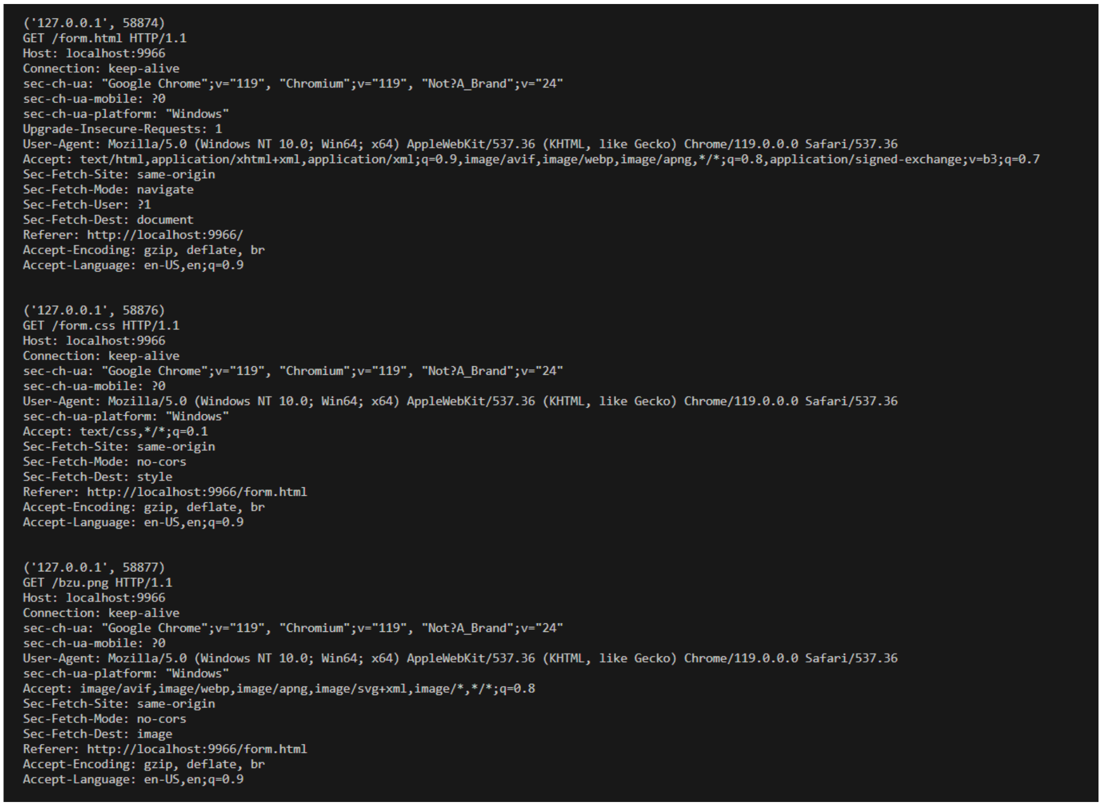

Hello! This is My Html page
localhost:9966/form.html
Hello! This is My Html page
جامعة بيرزيت
BIRZEIT UNIVERSITY
Activate Windows
Go to Settings to activate Windows.
10:18 PM
11/29/2023

Figure 19:visit my html

3.2.7. The request message on the command line:

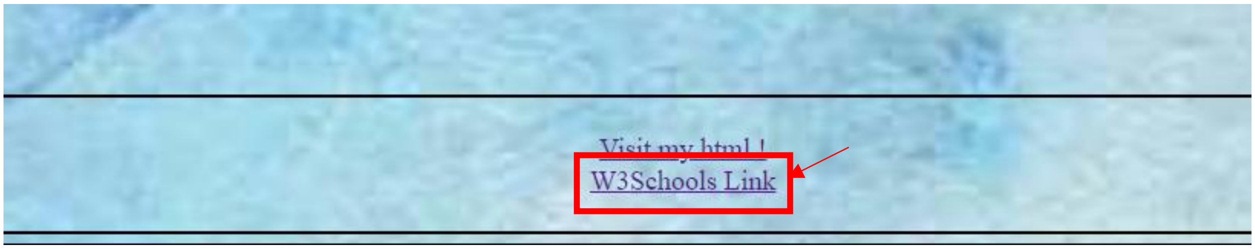


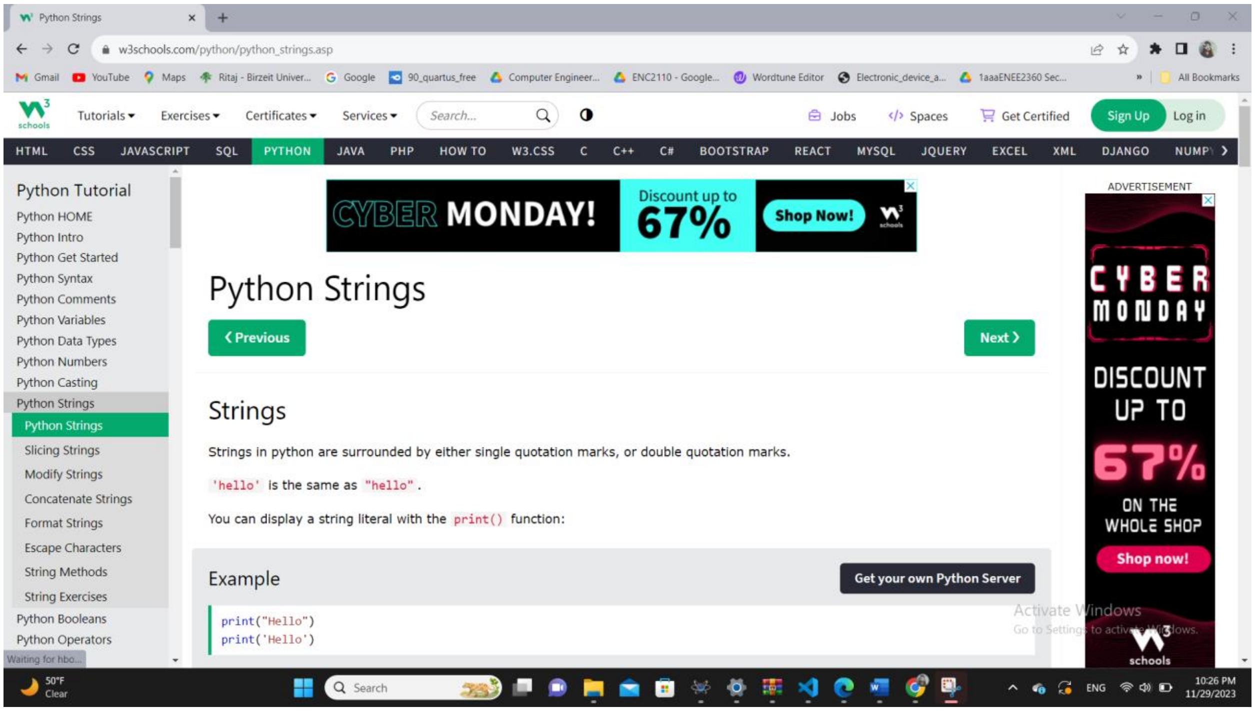
```
('127.0.0.1', 58874)
GET /form.html HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://localhost:9966/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 58876)
GET /form.css HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/form.html
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 58877)
GET /bzu.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/form.html
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```

Figure 20:request message for visit my html

3.2.8. W3schools link is clicked

If this link is clicked then the following page will appear





Python Strings

Strings

Strings in python are surrounded by either single quotation marks, or double quotation marks.

'hello' is the same as "hello".

You can display a string literal with the print() function:

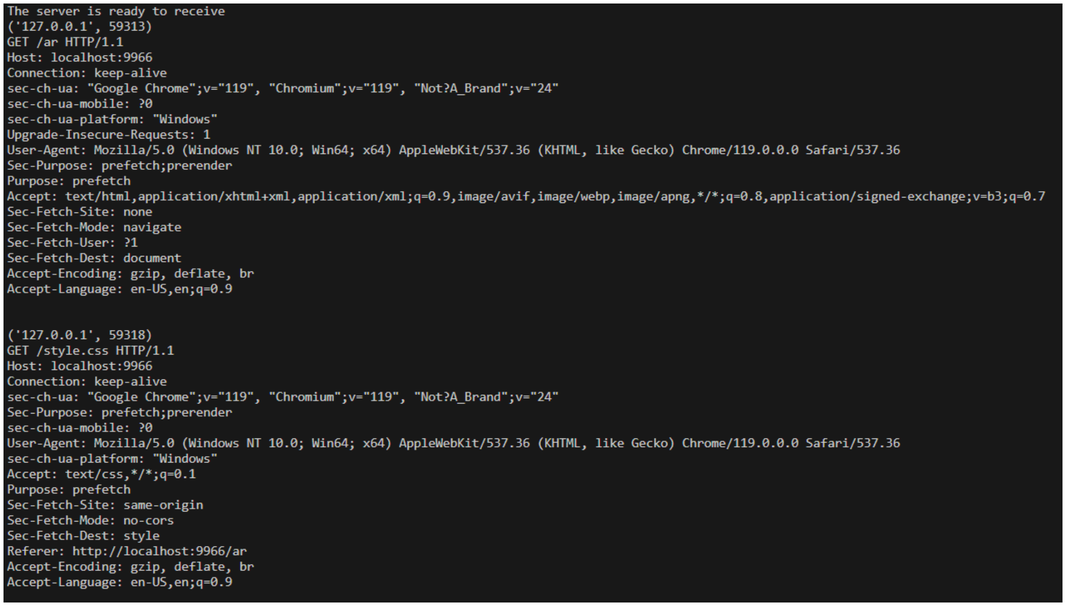
Example

```
print("Hello")
print('Hello')
```

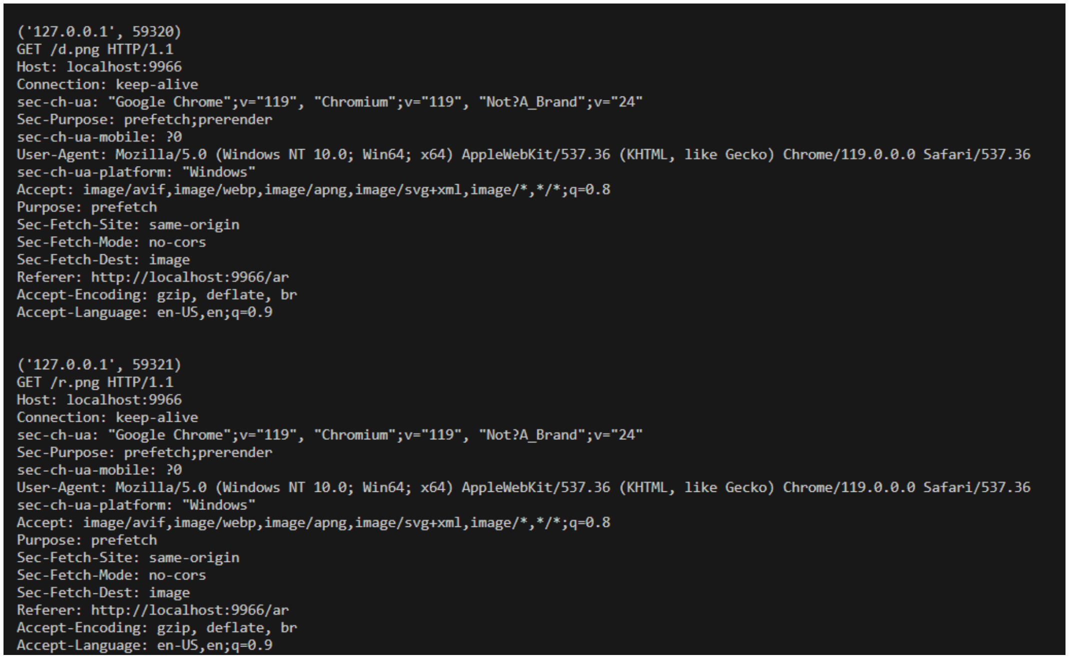
Figure 21:w3Schools page

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

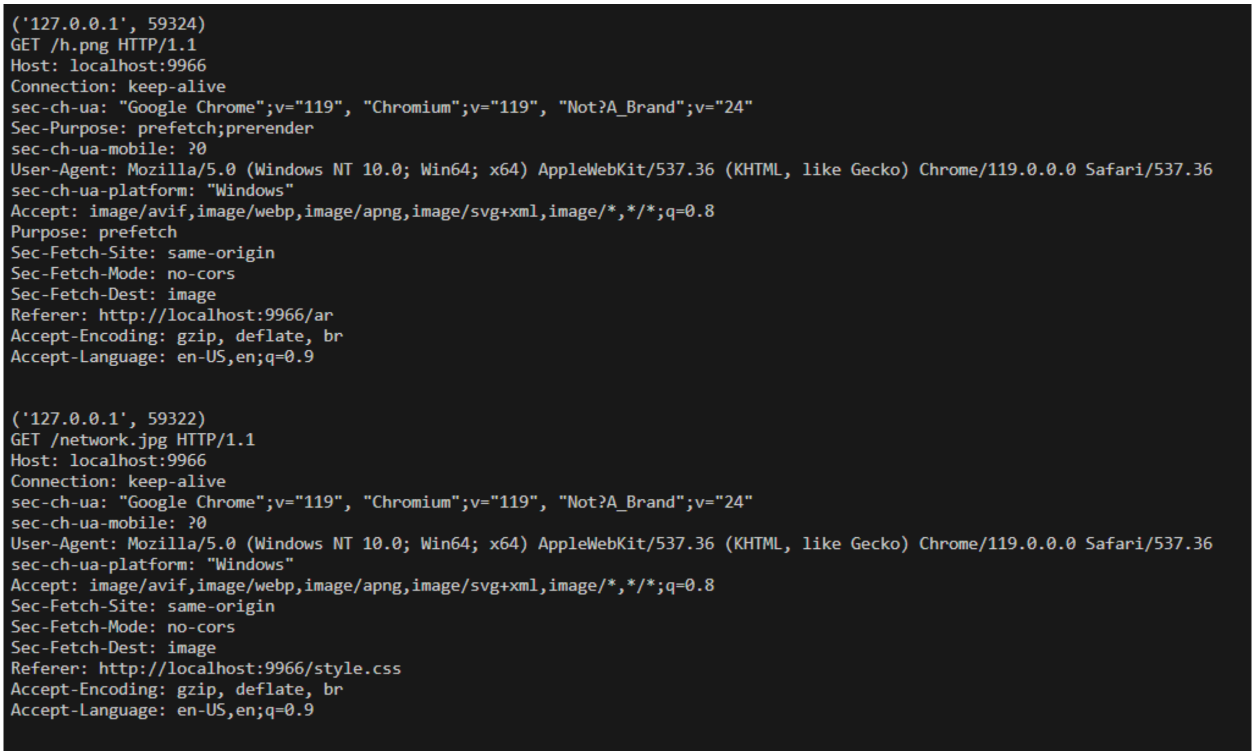
3.2.9. on command line(ar):



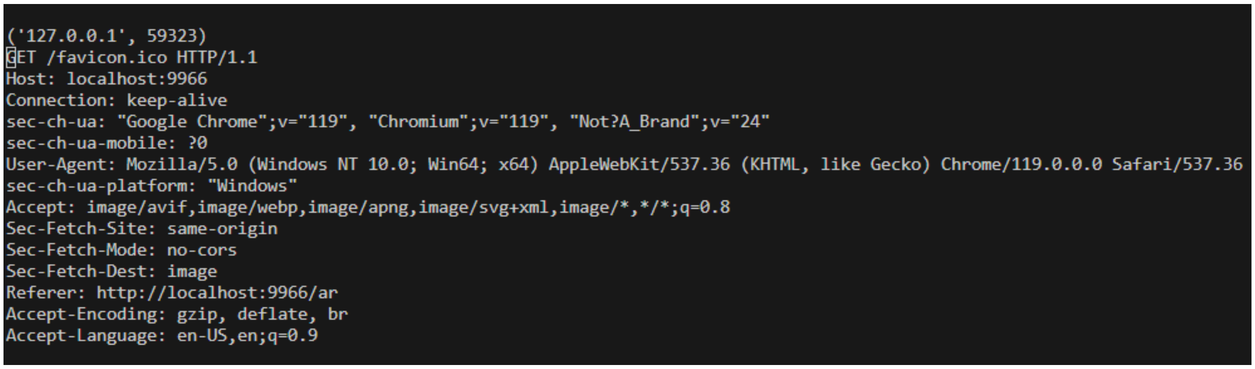
```
The server is ready to receive
('127.0.0.1', 59313)
GET /ar HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36
Sec-Purpose: prefetch;prerender
Purpose: prefetch
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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 59318)
GET /style.css HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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) Chrome/119.0.0.0 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:9966/ar
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```



```
('127.0.0.1', 59320)
GET /d.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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) Chrome/119.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Purpose: prefetch
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:9966/ar
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 59321)
GET /r.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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) Chrome/119.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Purpose: prefetch
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:9966/ar
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```



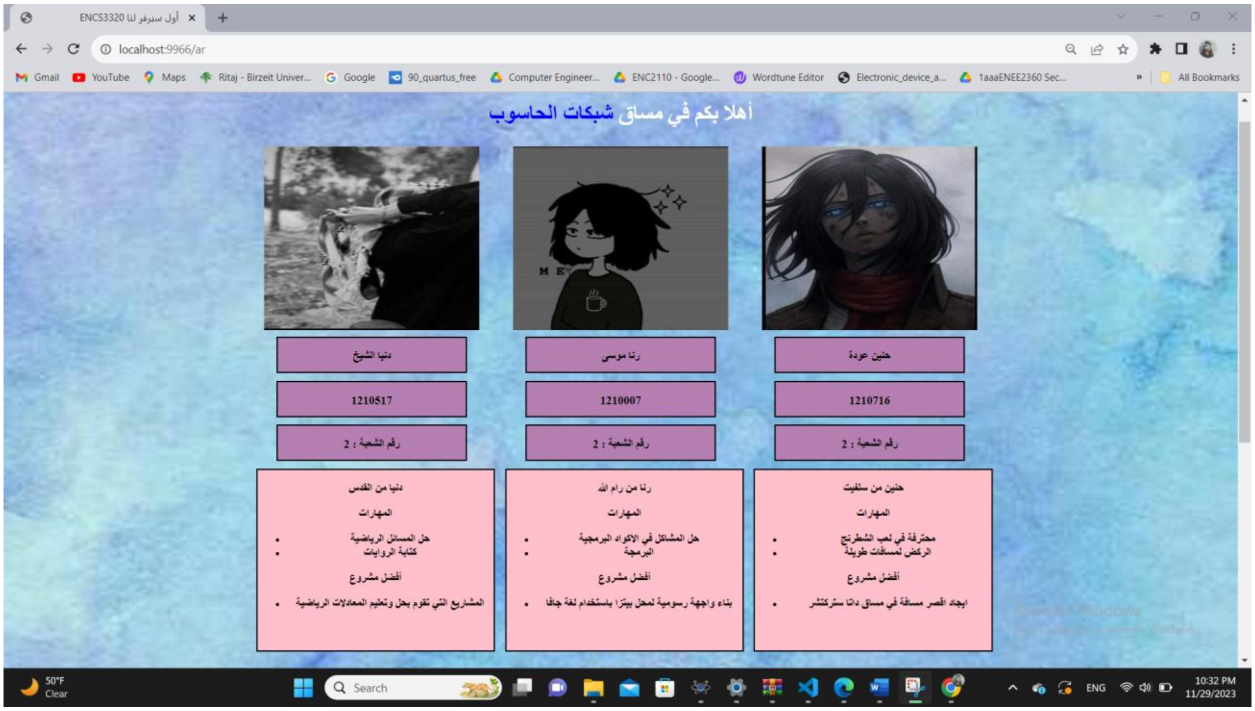
```
('127.0.0.1', 59324)
GET /h.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
Sec-Purpose: prefetch;prerender
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36
sec-ch-ua-platform: "Windows"
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Purpose: prefetch
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: image
Referer: http://localhost:9966/ar
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 59322)
GET /network.jpg HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/style.css
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```



```
('127.0.0.1', 59323)
GET /favicon.ico HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/ar
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```

Figure 22: the request message when we enter /ar

3.2.10. on the browser(ar):



أهلا بكم في مساق شبكات الحاسوب

حنين عودة

1210716

رقم الشعبة : 2

حنين من سلفيت

المهارات

- محترفة في لعب الشطرنج
- الركض لمسافات طويلة

أفضل مشروع

- ايجاد اقصر مسافة في مساق داتا ستركتشر

رنا موسى

1210007

رقم الشعبة : 2

رنا من رام الله

المهارات

- حل المشاكل في الاكواد البرمجية
- البرمجة

أفضل مشروع

- بناء واجهة رسومية لمحل بيتزا باستخدام لغة جافا

دنيا الشيخ

1210517

رقم الشعبة : 2

دنيا من القدس

المهارات

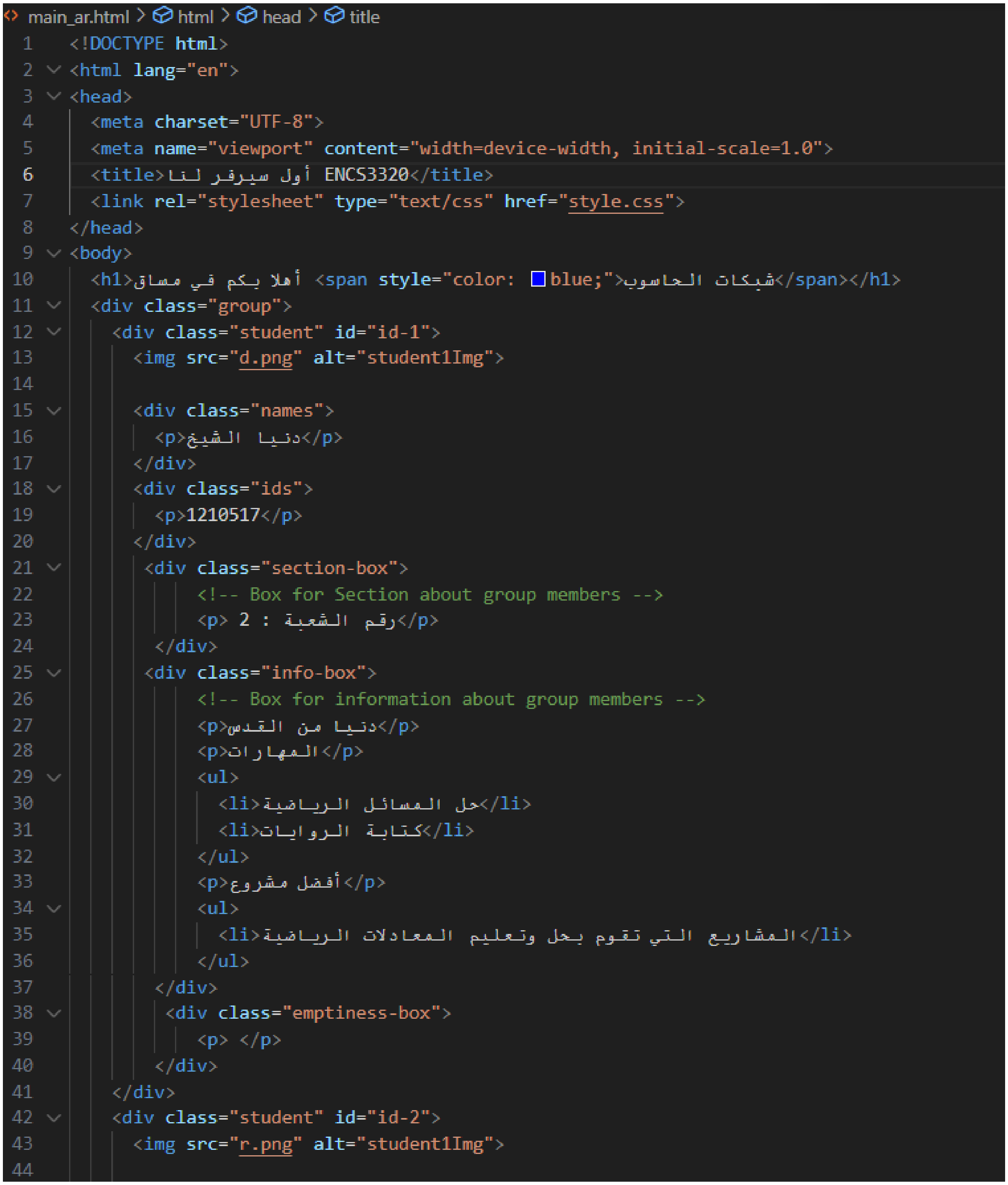
- حل المسائل الرياضية
- كتابة الروايات

أفضل مشروع

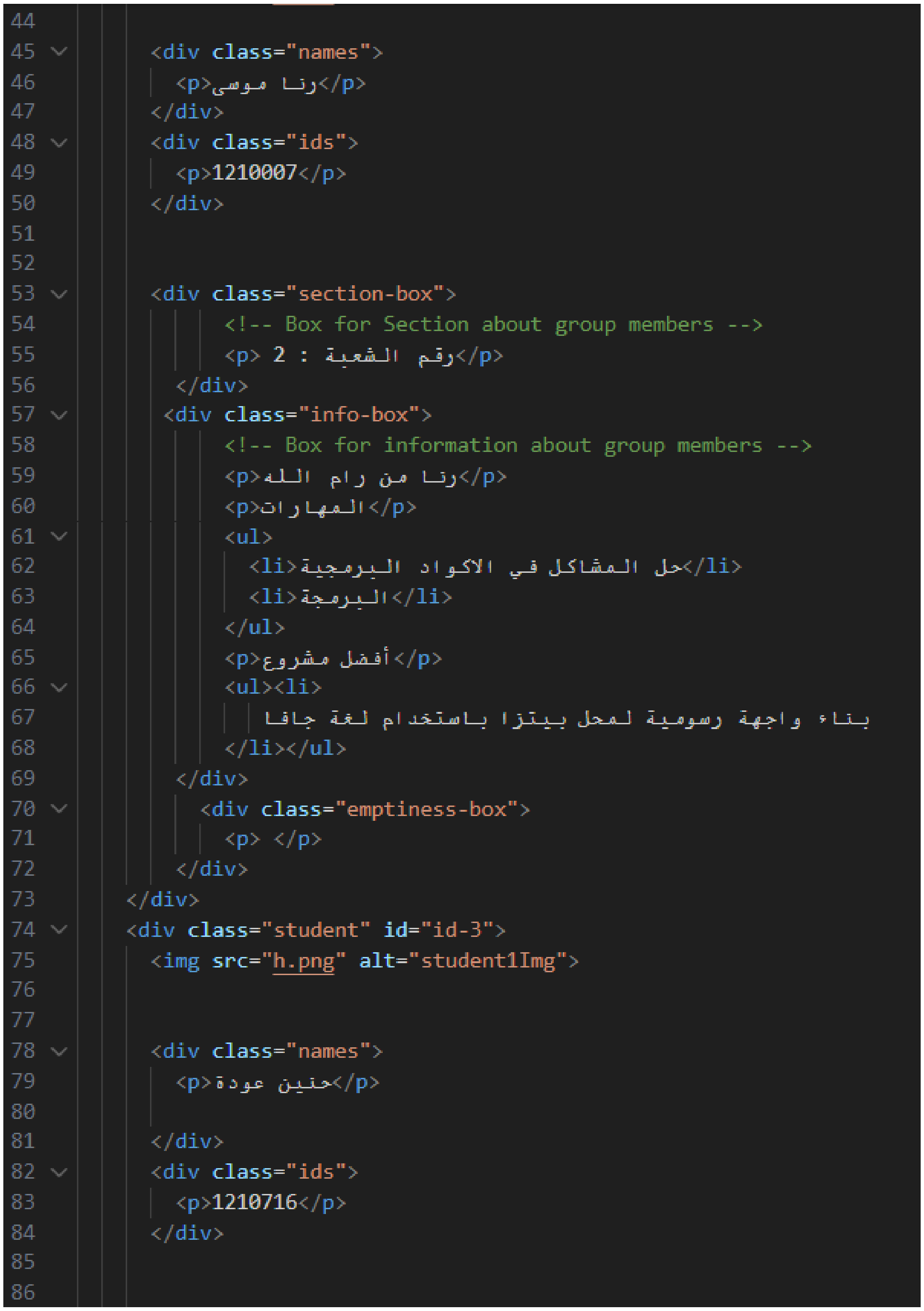
- المشاريع التي تقوم بحل وتعليم المعادلات الرياضية

Figure 23: the result of the request in the web page in Arabic

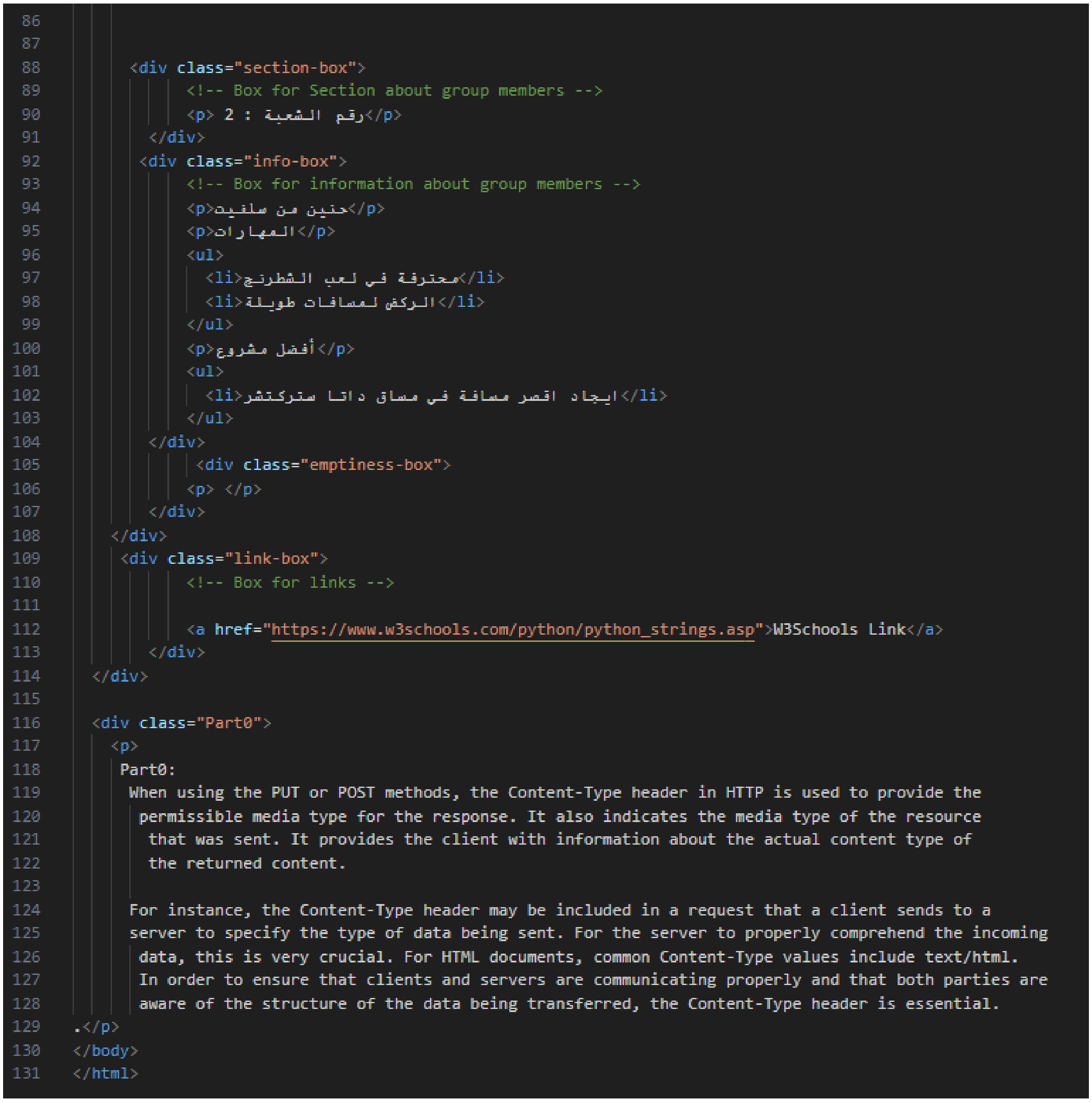
3.2.11. main_ar.html code:



```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>أول سيرفر لنا ENCS3320</title>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  <h1>أهلا بكم في مساق <span style="color: blue;">شبكات الحاسوب</span></h1>
  <div class="group">
    <div class="student" id="id-1">
      <img src="d.png" alt="student1Img">

      <div class="names">
        <p>دنيا الشيخ</p>
      </div>
      <div class="ids">
        <p>1210517</p>
      </div>
       <div class="section-box">
            <!-- Box for Section about group members -->
            <p> رقم الشعبة : 2</p>
        </div>
       <div class="info-box">
            <!-- Box for information about group members -->
            <p>دنيا من القدس</p>
            <p>المهارات</p>
            <ul>
              <li>حل المسائل الرياضية</li>
              <li>كتابة الروايات</li>
            </ul>
            <p>أفضل مشروع</p>
            <ul>
              <li>المشاريع التي تقوم بحل وتعليم المعادلات الرياضية</li>
            </ul>
        </div>
         <div class="emptiness-box">
            <p> </p>
        </div>
    </div>
    <div class="student" id="id-2">
      <img src="r.png" alt="student1Img">

```



```
  <div class="names">
    <p>رنا موسى</p>
  </div>
  <div class="ids">
    <p>1210007</p>
  </div>


  <div class="section-box">
      <!-- Box for Section about group members -->
      <p> 2 : رقم الشعبة</p>
    </div>
   <div class="info-box">
      <!-- Box for information about group members -->
      <p>رنا من رام الله</p>
      <p>المهارات</p>
      <ul>
        <li>حل المشاكل في الاكواد البرمجية</li>
        <li>البرمجة</li>
      </ul>
      <p>أفضل مشروع</p>
      <ul><li>
         بناء واجهة رسومية لمحل بيتزا باستخدام لغة جافا
      </li></ul>
    </div>
      <div class="emptiness-box">
      <p> </p>
    </div>
</div>
<div class="student" id="id-3">
  <img src="h.png" alt="student1Img">


  <div class="names">
    <p>حنين عودة</p>

  </div>
  <div class="ids">
    <p>1210716</p>
  </div>

```

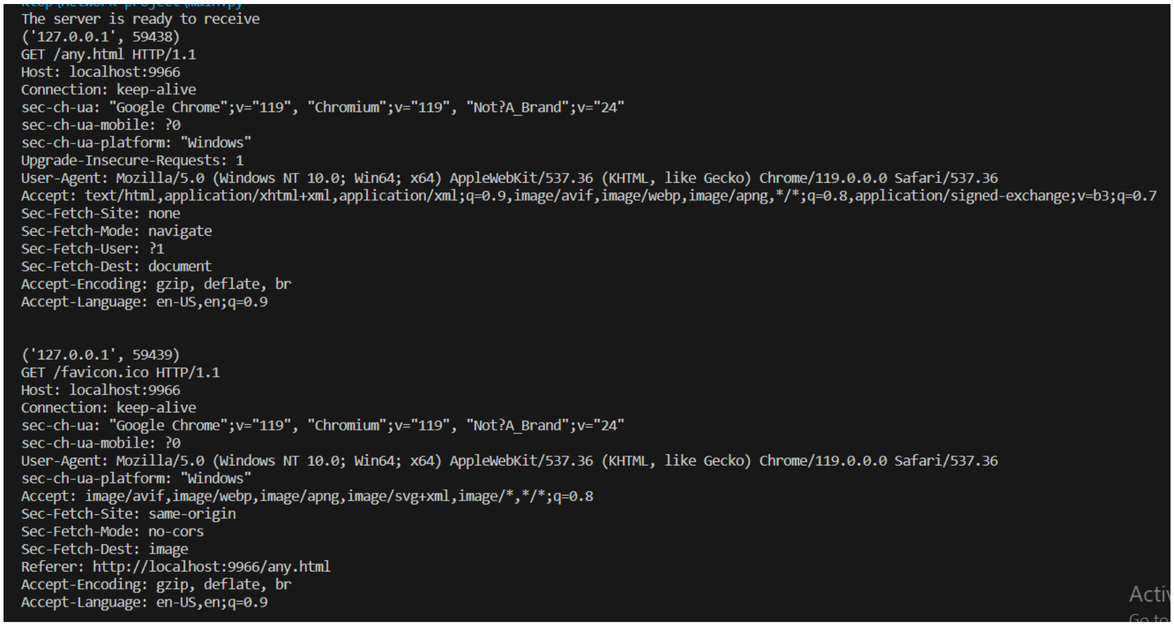


```


      <div class="section-box">
            <!-- Box for Section about group members -->
            <p> 2 : رقم الشعبة</p>
        </div>
       <div class="info-box">
            <!-- Box for information about group members -->
            <p>حنين من سلفيت</p>
            <p>المهارات</p>
            <ul>
              <li>محترفة في لعب الشطرنج</li>
              <li>الركض لمسافات طويلة</li>
            </ul>
            <p>أفضل مشروع</p>
            <ul>
              <li>ايجاد اقصر مسافة في مساق داتا ستركتشر</li>
            </ul>
        </div>
             <div class="emptiness-box">
            <p> </p>
        </div>
    </div>
     <div class="link-box">
            <!-- Box for links -->

            <a href="https://www.w3schools.com/python/python_strings.asp">W3Schools Link</a>
        </div>
  </div>

  <div class="Part0">
    <p>
     Part0:
      When using the PUT or POST methods, the Content-Type header in HTTP is used to provide the
       permissible media type for the response. It also indicates the media type of the resource
        that was sent. It provides the client with information about the actual content type of
        the returned content.

      For instance, the Content-Type header may be included in a request that a client sends to a
      server to specify the type of data being sent. For the server to properly comprehend the incoming
       data, this is very crucial. For HTML documents, common Content-Type values include text/html.
       In order to ensure that clients and servers are communicating properly and that both parties are
       aware of the structure of the data being transferred, the Content-Type header is essential.
.</p>
</body>
</html>
```

Figure 24:main_ar.html code

- 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.

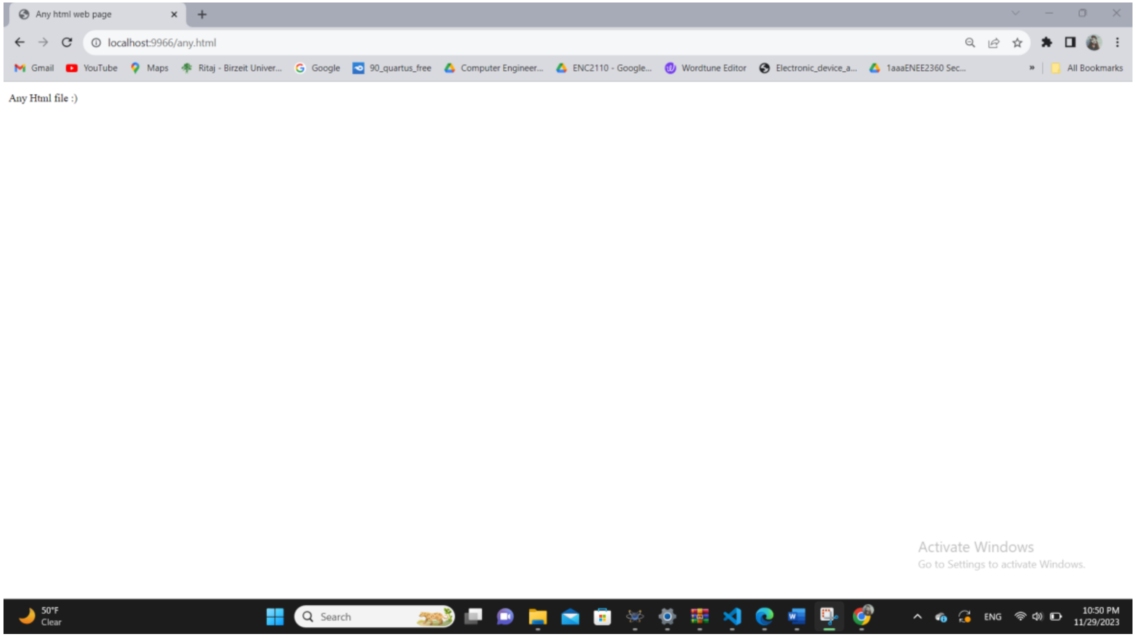
3.2.12. on the command line(any html file)



```
The server is ready to receive
('127.0.0.1', 59438)
GET /any.html HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 59439)
GET /favicon.ico HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/any.html
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```

Figure 25:any html file request

3.2.13. on the browser (any html file):

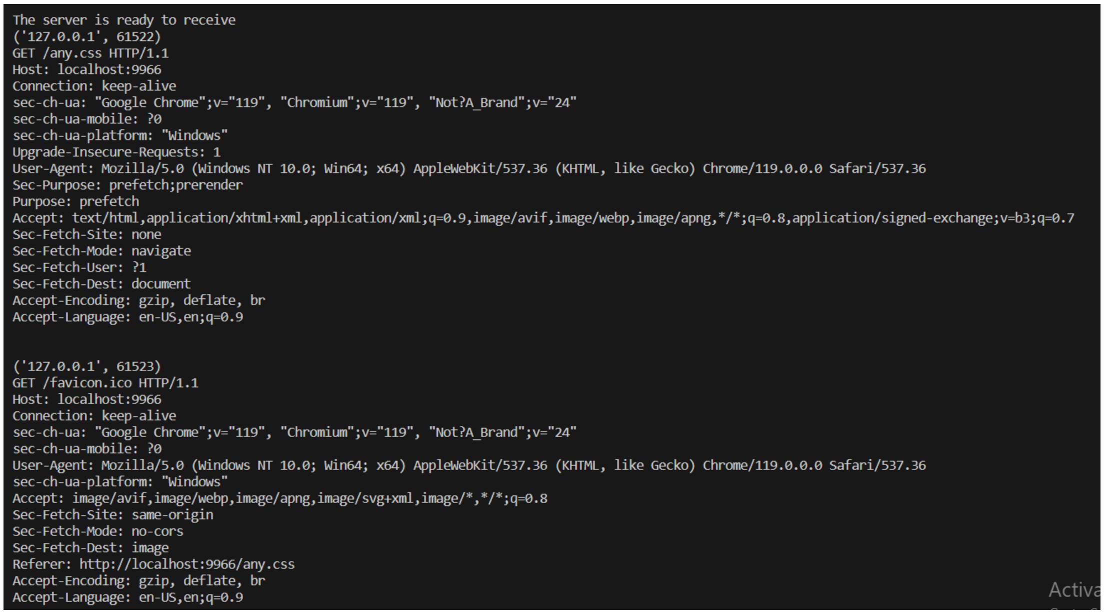


Any Html file :)

Figure 26:the any html file in website

- **if the request is a .css file then the server should send the requested css file with Content-Type: text/css.**

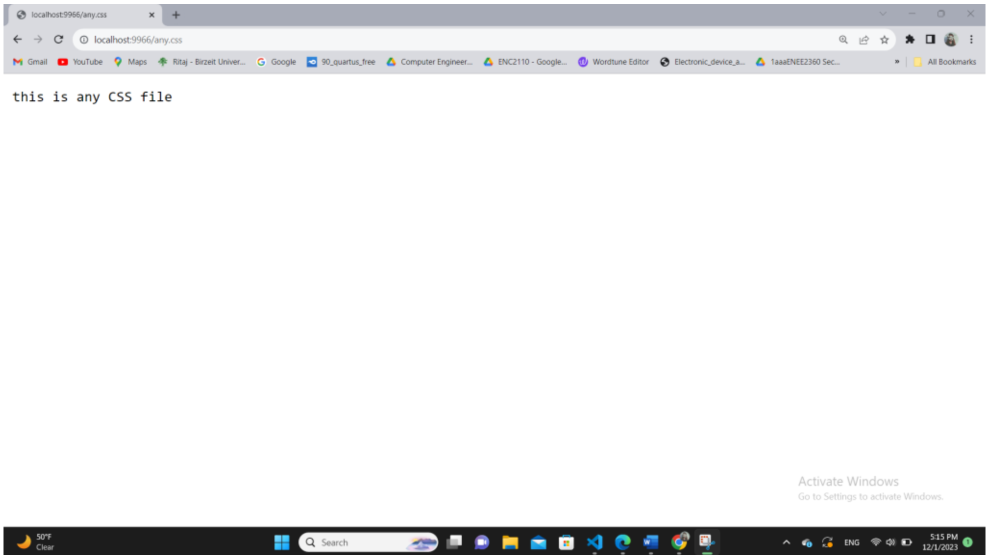
3.2.14. on the command line(any CSS file)



```
The server is ready to receive
('127.0.0.1', 61522)
GET /any.css HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 Safari/537.36
Sec-Purpose: prefetch;prerender
Purpose: prefetch
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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 61523)
GET /favicon.ico HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/any.css
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```

Figure 27:the request message for a css file

3.2.15. on the browser(any CSS file):

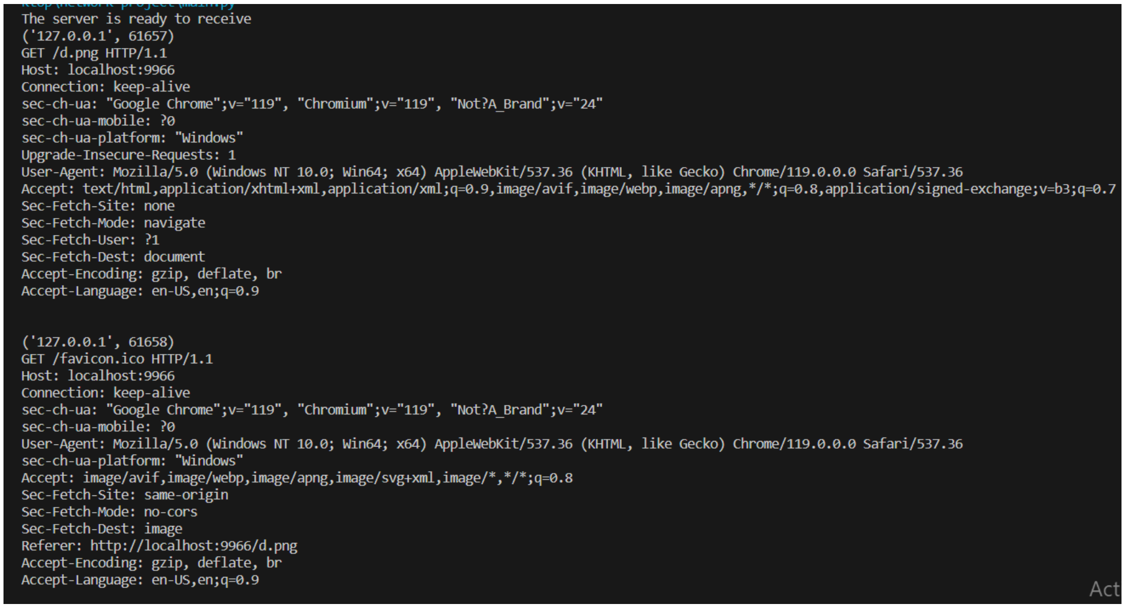


```
this is any CSS file
```

Figure 28:the css file in website

- ❖ **if the request is a .png then the server should send the png image with Content-Type: image/png.**

3.2.16. on the command line(any png file):



```
ktop\network project\main.py
The server is ready to receive
('127.0.0.1', 61657)
GET /d.png HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 61658)
GET /favicon.ico HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/d.png
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```

Figure 29:the request message for a png file

3.2.17. on the browser(any png file):

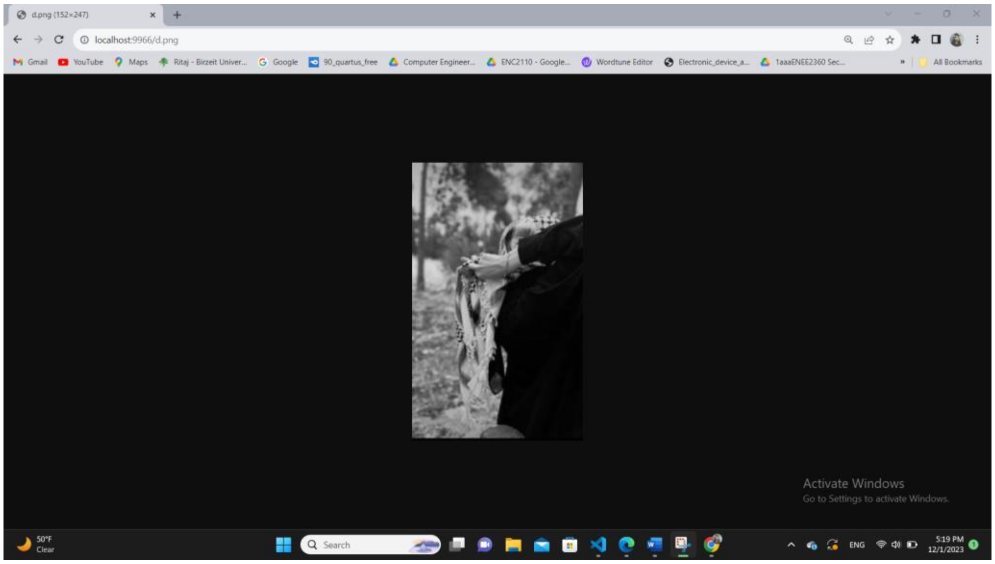
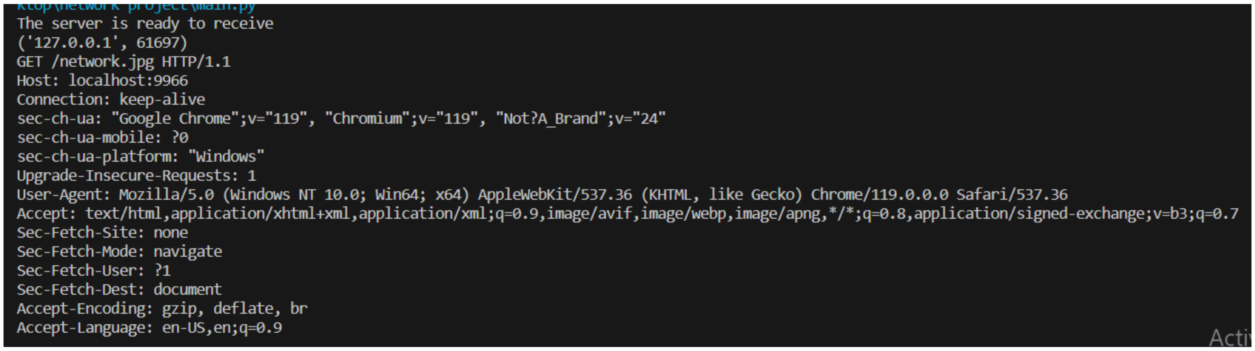


Figure 30:the png file in website

- **if the request is a .jpg then the server should send the jpg image with Content-Type: image/jpeg.**

3.2.18. on the command line (any jpg file):



```
ktop\network project\main.py
The server is ready to receive
('127.0.0.1', 61697)
GET /network.jpg HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 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
Accept-Language: en-US,en;q=0.9
```

Figure 31: the request message for a jpg file

3.2.19. on the browser (any jpg file):

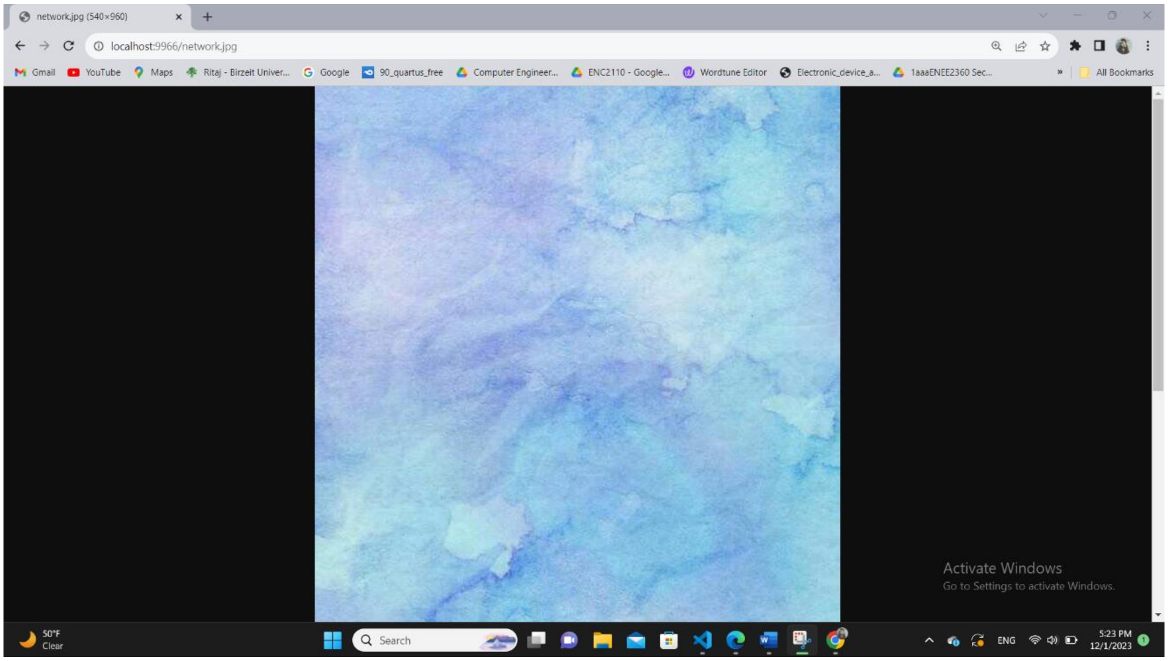
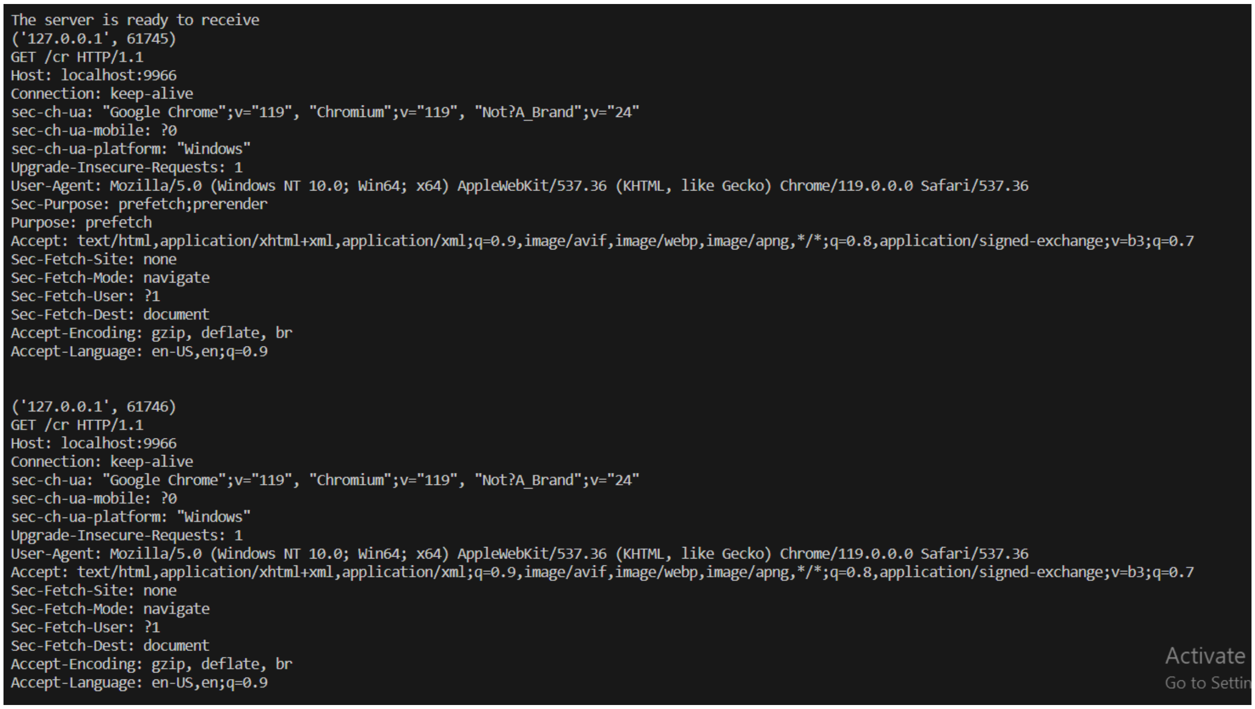


Figure 32:the jpg file on website

❖ **If the request is /cr then redirect to cornell.edu website :**



```
The server is ready to receive
('127.0.0.1', 61745)
GET /cr HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 Safari/537.36
Sec-Purpose: prefetch;prerender
Purpose: prefetch
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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 61746)
GET /cr HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 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
Accept-Language: en-US,en;q=0.9
```

Figure 33: Cornell website request message

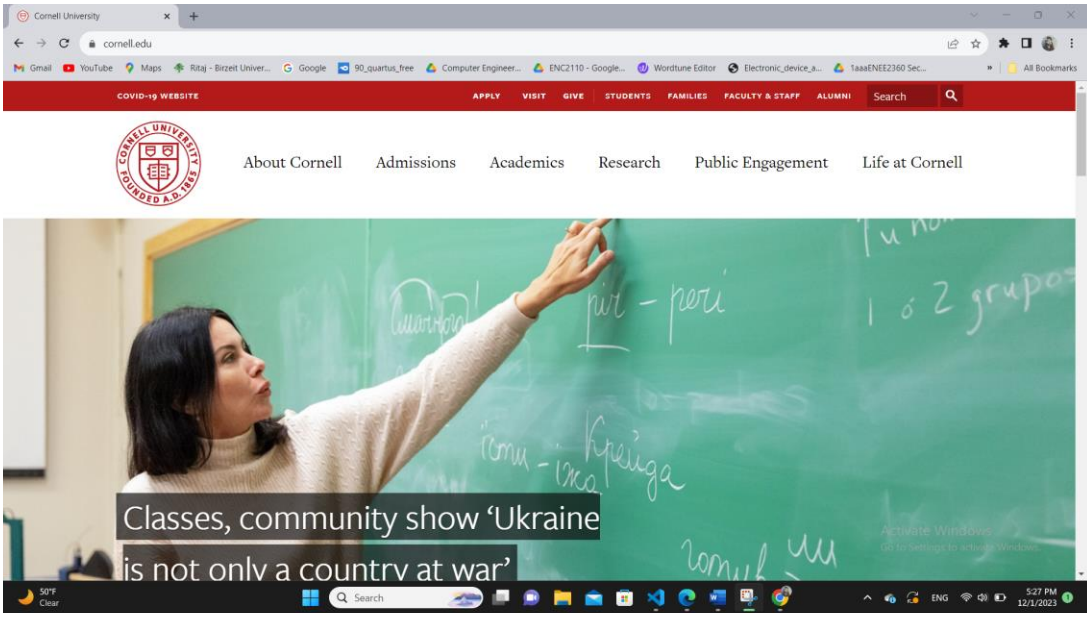
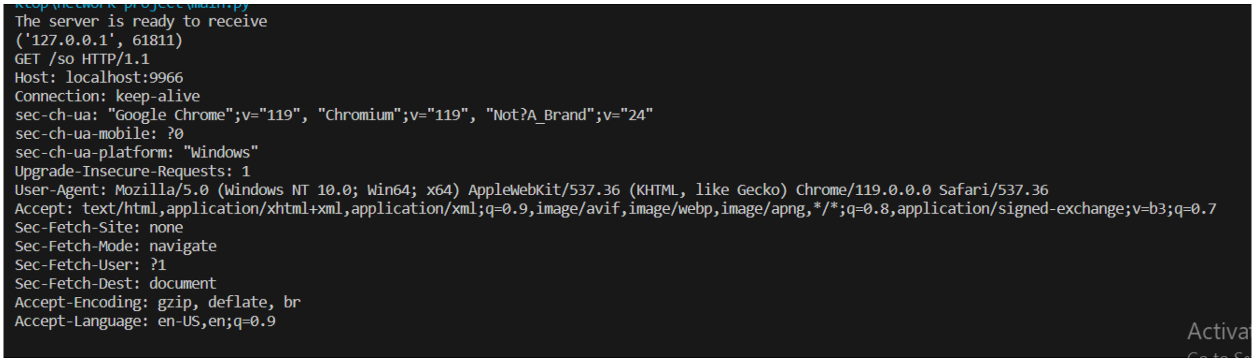


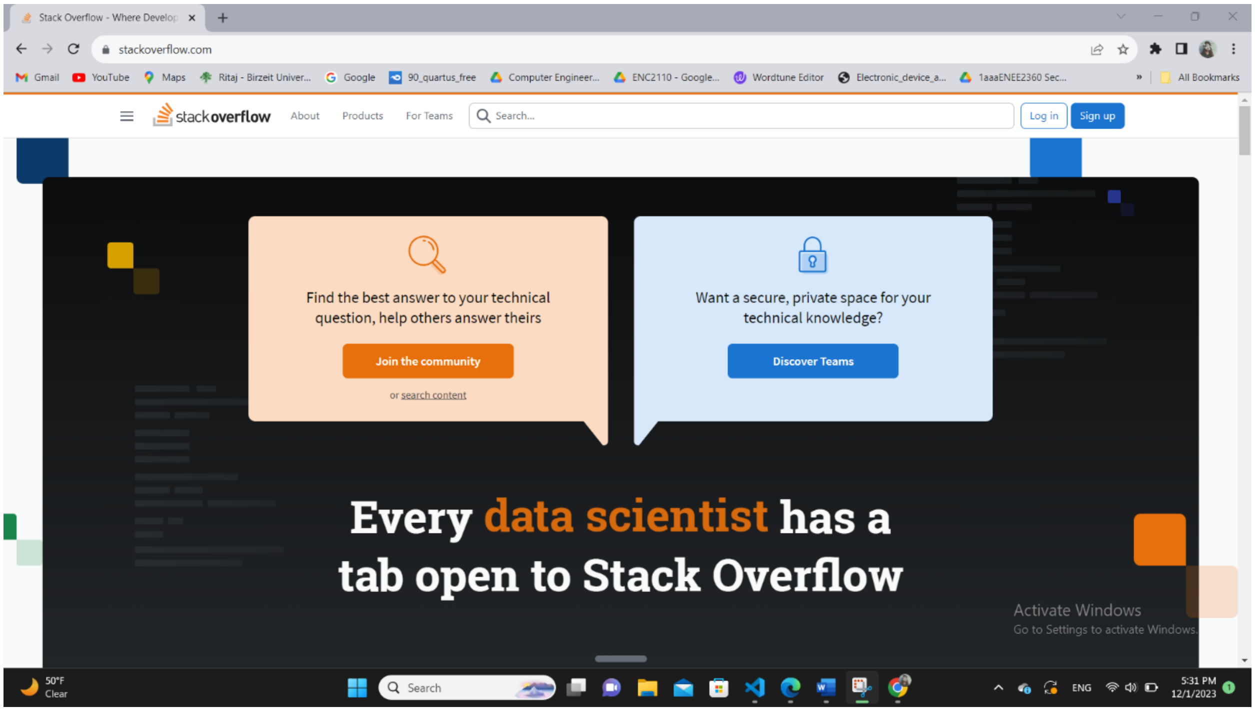
Figure 34:cornell.edu website

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



```
The server is ready to receive
('127.0.0.1', 61811)
GET /so HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 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
Accept-Language: en-US,en;q=0.9
```

Figure 35: stackoverflow website request message



stackoverflow.com

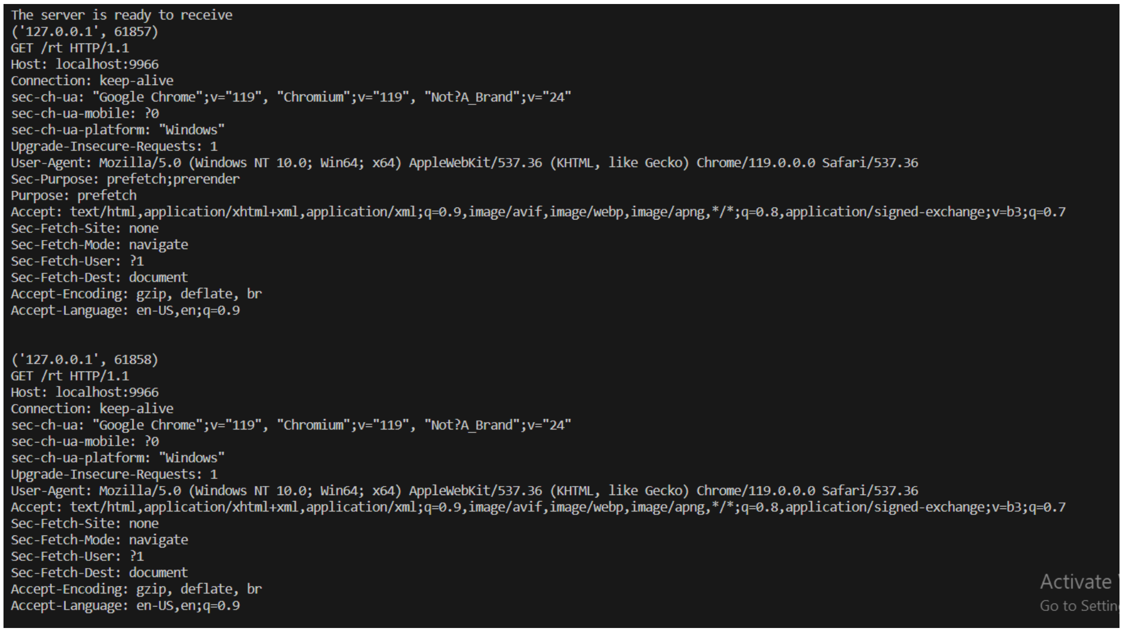
Find the best answer to your technical question, help others answer theirs

Want a secure, private space for your technical knowledge?

Every data scientist has a tab open to Stack Overflow

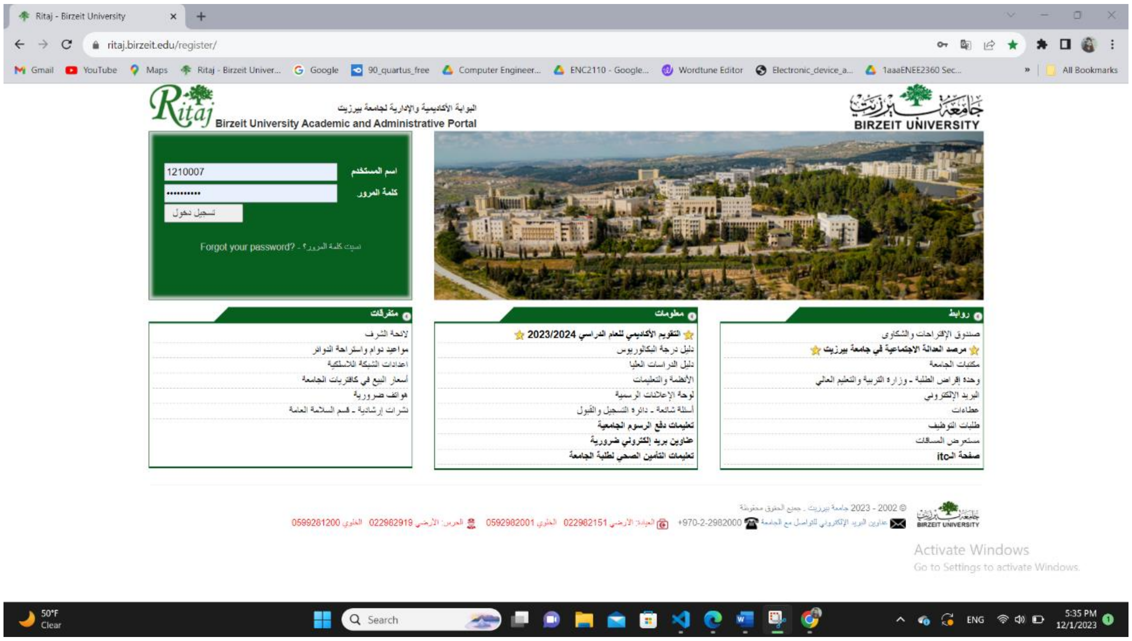
Figure 36: stackoverflow website

❖ If the request is /rt then redirect to ritaj website



```
The server is ready to receive
('127.0.0.1', 61857)
GET /rt HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 Safari/537.36
Sec-Purpose: prefetch;prerender
Purpose: prefetch
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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 61858)
GET /rt HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 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
Accept-Language: en-US,en;q=0.9
```

Figure 37: birzeit university website request message



ritaj.birzeit.edu/register/

البوابة الأكاديمية والإدارية لجامعة بيرزيت
Birzeit University Academic and Administrative Portal

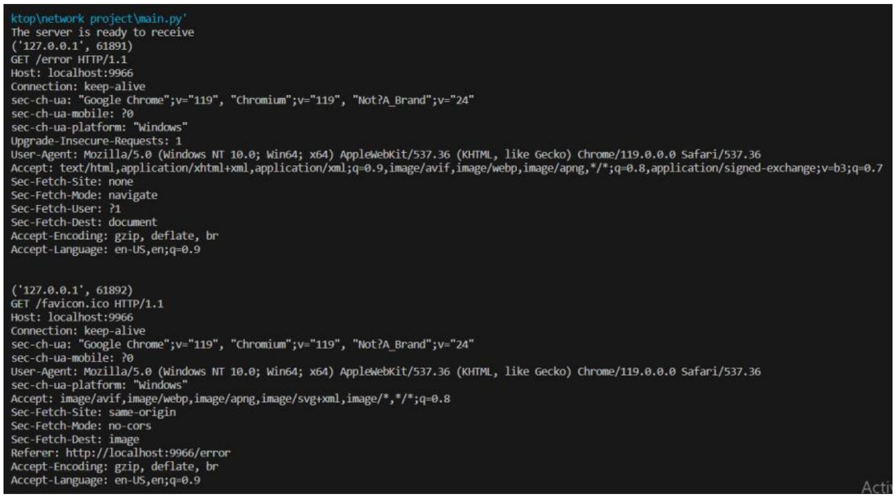
اسم المستخدم 1210007
كلمة المرور ••••••••••

Ritaj - Birzeit University
Gmail
YouTube
Maps
Ritaj - Birzeit Univer...
Google
90_quartus_free
Computer Engineer...
ENC2110 - Google...
Wordtune Editor
Electronic_device_a...
1aaaENEE2360 Sec...
All Bookmarks
Ritaj
جامعة بيرزيت
BIRZEIT UNIVERSITY
تسجيل دخول
Forgot your password? - نسيت كلمة المرور؟
متفرقات
لائحة الشرف
مواعيد دوام واستراحة الدوائر
اعدادات الشبكة اللاسلكية
أسعار البيع في كافتريات الجامعة
هواتف ضرورية
نشرات إرشادية - قسم السلامة العامة
معلومات
التقويم الأكاديمي للعام الدراسي 2023/2024
دليل درجة البكالوريوس
دليل الدراسات العليا
الأنظمة والتعليمات
لوحة الإعلانات الرسمية
أسئلة شائعة - دائرة التسجيل والقبول
تعليمات دفع الرسوم الجامعية
عناوين بريد إلكتروني ضرورية
تعليمات التأمين الصحي لطلبة الجامعة
روابط
صندوق الإقتراحات والشكاوى
مرصد العدالة الاجتماعية في جامعة بيرزيت
مكتبات الجامعة
وحدة إقراض الطلبة - وزارة التربية والتعليم العالي
البريد الإلكتروني
عطاءات
طلبات التوظيف
مستعرض المساقات
صفحة الـitc
© 2002 - 2023 جامعة بيرزيت . جميع الحقوق محفوظة
عناوين البريد الإلكتروني للتواصل مع الجامعة +970-2-2982000 العيادة: الأرضي 022982151 الخلوي 0592982001 الحرس: الأرضي 022982919 الخلوي 0599281200
Activate Windows
Go to Settings to activate Windows.
50°F
Clear
Search
ENG
5:35 PM
12/1/2023

Figure 38: birzeit university website

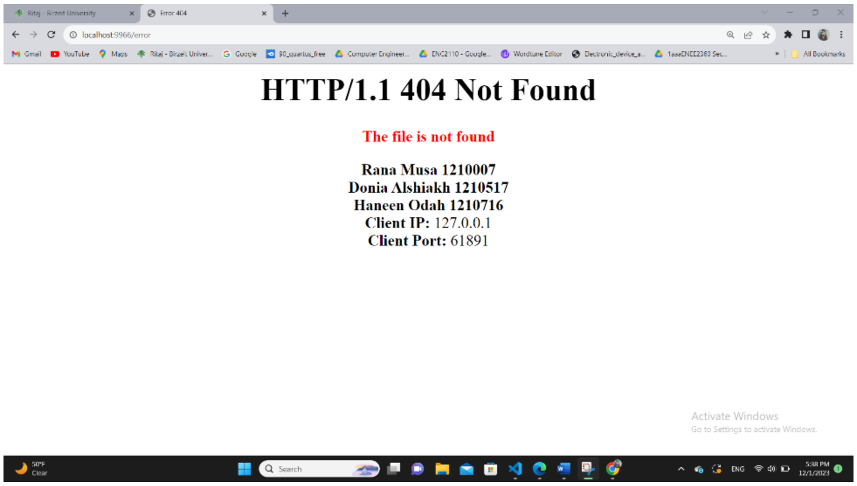
3.2.20. Error message:

- 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



```
ktop\network project\main.py'
The server is ready to receive
('127.0.0.1', 61891)
GET /error HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
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/119.0.0.0 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
Accept-Language: en-US,en;q=0.9


('127.0.0.1', 61892)
GET /favicon.ico HTTP/1.1
Host: localhost:9966
Connection: keep-alive
sec-ch-ua: "Google Chrome";v="119", "Chromium";v="119", "Not?A_Brand";v="24"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 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:9966/error
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
```

Figure 39: error webpage request message



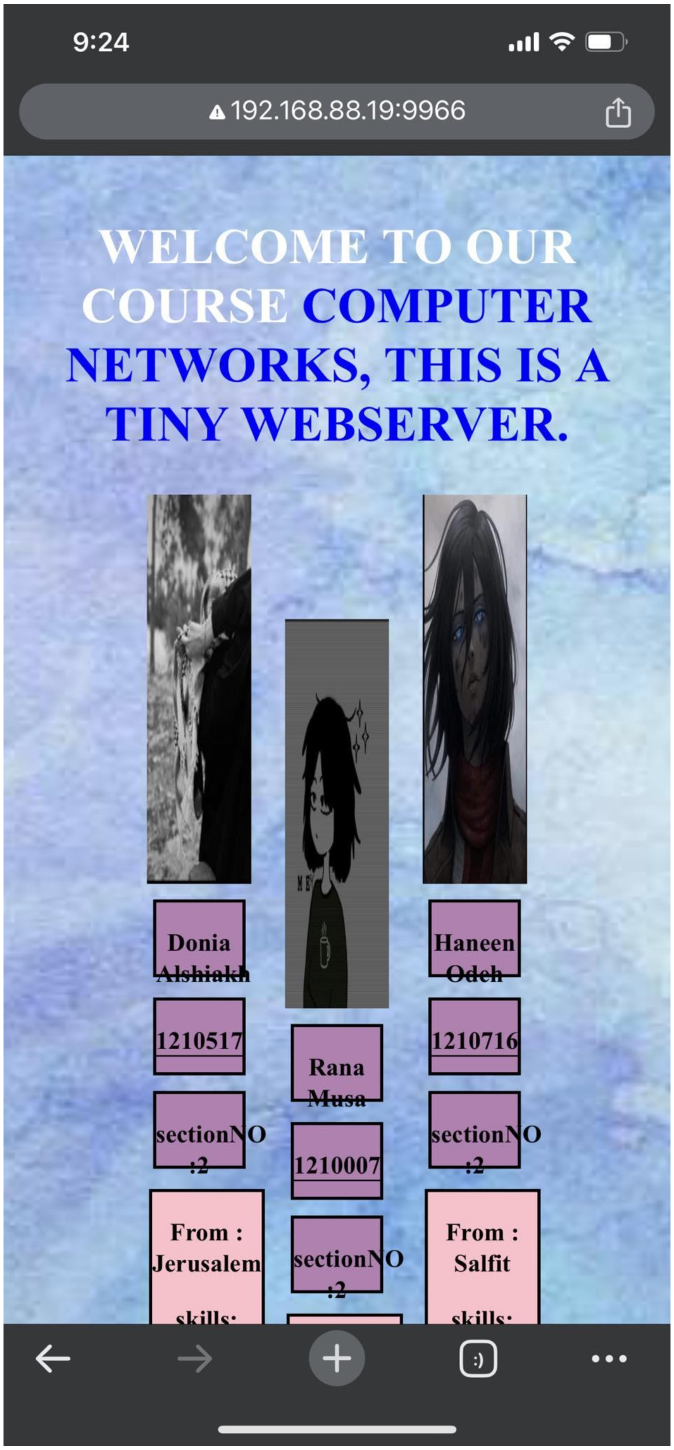
HTTP/1.1 404 Not Found

The file is not found

Rana Musa 1210007
Donia Alshiakh 1210517
Haneen Odah 1210716
Client IP: 127.0.0.1
Client Port: 61891

Figure 40: error webpage

3.3.1. Testing from another device



WELCOME TO OUR COURSE COMPUTER NETWORKS, THIS IS A TINY WEBSERVER.

Donia Alshiakh

1210517

sectionNO :2

From : Jerusalem

skills:

Rana Musa

1210007

sectionNO :2

Haneen Odeh

1210716

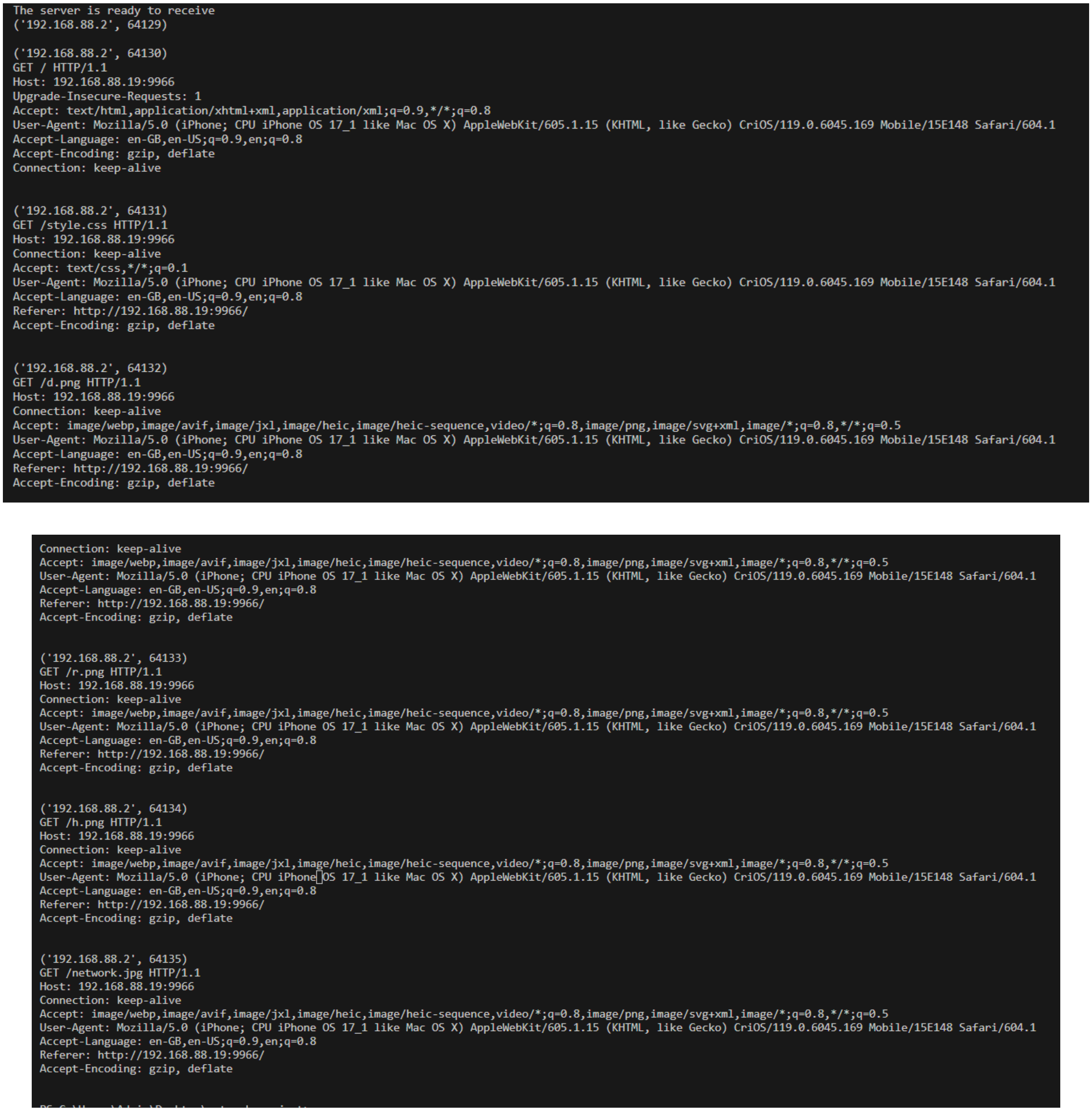
sectionNO :2

From : Salfit

skills:

Figure 41: the test on the phone

3.3.2. On command line Testing from another device



```
The server is ready to receive
('192.168.88.2', 64129)

('192.168.88.2', 64130)
GET / HTTP/1.1
Host: 192.168.88.19:9966
Upgrade-Insecure-Requests: 1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/119.0.6045.169 Mobile/15E148 Safari/604.1
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Accept-Encoding: gzip, deflate
Connection: keep-alive


('192.168.88.2', 64131)
GET /style.css HTTP/1.1
Host: 192.168.88.19:9966
Connection: keep-alive
Accept: text/css,*/*;q=0.1
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/119.0.6045.169 Mobile/15E148 Safari/604.1
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Referer: http://192.168.88.19:9966/
Accept-Encoding: gzip, deflate


('192.168.88.2', 64132)
GET /d.png HTTP/1.1
Host: 192.168.88.19:9966
Connection: keep-alive
Accept: image/webp,image/avif,image/jxl,image/heic,image/heic-sequence,video/*;q=0.8,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/119.0.6045.169 Mobile/15E148 Safari/604.1
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Referer: http://192.168.88.19:9966/
Accept-Encoding: gzip, deflate
```

```
Connection: keep-alive
Accept: image/webp,image/avif,image/jxl,image/heic,image/heic-sequence,video/*;q=0.8,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/119.0.6045.169 Mobile/15E148 Safari/604.1
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Referer: http://192.168.88.19:9966/
Accept-Encoding: gzip, deflate


('192.168.88.2', 64133)
GET /r.png HTTP/1.1
Host: 192.168.88.19:9966
Connection: keep-alive
Accept: image/webp,image/avif,image/jxl,image/heic,image/heic-sequence,video/*;q=0.8,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/119.0.6045.169 Mobile/15E148 Safari/604.1
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Referer: http://192.168.88.19:9966/
Accept-Encoding: gzip, deflate


('192.168.88.2', 64134)
GET /h.png HTTP/1.1
Host: 192.168.88.19:9966
Connection: keep-alive
Accept: image/webp,image/avif,image/jxl,image/heic,image/heic-sequence,video/*;q=0.8,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/119.0.6045.169 Mobile/15E148 Safari/604.1
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Referer: http://192.168.88.19:9966/
Accept-Encoding: gzip, deflate


('192.168.88.2', 64135)
GET /network.jpg HTTP/1.1
Host: 192.168.88.19:9966
Connection: keep-alive
Accept: image/webp,image/avif,image/jxl,image/heic,image/heic-sequence,video/*;q=0.8,image/png,image/svg+xml,image/*;q=0.8,*/*;q=0.5
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 17_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/119.0.6045.169 Mobile/15E148 Safari/604.1
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Referer: http://192.168.88.19:9966/
Accept-Encoding: gzip, deflate
```

Figure 42: the request message

4. Conclusion

To sum up in this project, processes on different hosts communicate through network applications and we tried in this project to write simple applications in python to understand how connection happens and which type of protocols can be more appropriate for different applications

5. Appendices

5.1. For part 2

CLIENT CODE

```
#include <stdio.h> #include <stdlib.h> #include <string.h> #include <unistd.h> #include
<arpa/inet.h> int main() { // Creating a socket int clientSocket = socket(AF_INET,
SOCK_STREAM, 0); if (clientSocket == -1) { perror("Error in creating the socket!");
exit(EXIT_FAILURE); } // Connecting to the server struct sockaddr_in serverAddress;
serverAddress.sin_family = AF_INET; serverAddress.sin_port = htons(9955);
serverAddress.sin_addr.s_addr = inet_addr("127.0.0.1"); if (connect(clientSocket, (struct
sockaddr *)&serverAddress, sizeof(serverAddress)) == -1) { perror("Error connecting to
server"); exit(EXIT_FAILURE); } // Here is where to put the student id needed char
studentID[] = "1210716"; // Send the student ID to the server send(clientSocket,
studentID, strlen(studentID), 0); // Receive the server's response char buffer[1024]; ssize_t
bytesReceived = recv(clientSocket, buffer, sizeof(buffer), 0); if (bytesReceived == -1) {
perror("Error in receiving the data"); exit(EXIT_FAILURE); } buffer[bytesReceived] = '\0';
printf("Server response: %s\n", buffer); // Closing the connection close(clientSocket);
return 0;
```

SERVER CODE

```
#include <stdio.h> #include <stdlib.h> #include <string.h> #include <unistd.h> #include
<arpa/inet.h> void screenLock() { //A function to lock the screen when needed
system("gnome-screensaver-command --lock"); } void handleClient(int clientSocket, char
*data) { char *studentsIDS[] = {"1210716", "1210007", "1210517"}; int ID = 0; for (int i = 0;
i < sizeof(studentsIDS) / sizeof(studentsIDS[0]); ++i) { if (strcmp(data, studentsIDS[i]) ==
0) { ID = 1; break; } } if (ID) { // Displaying message on server side printf("The student Id
has been recieved : %s\n", data); printf("The screen will be locked after 10 seconds\n");
send(clientSocket, "Server will lock screen after 10 seconds\n", 42, 0); // Wait for 10
seconds sleep(10); screenLock(); } else { printf("Invalid student ID or text received. No
```

action taken.\n"); } } int main() { int serverSocket = socket(AF_INET, SOCK_STREAM, 0); if (serverSocket == -1) { perror("Error creating socket"); exit(EXIT_FAILURE); } // Bind the socket to a specific address and port struct sockaddr_in serverAddress; serverAddress.sin_family = AF_INET; serverAddress.sin_addr.s_addr = INADDR_ANY; serverAddress.sin_port = htons(9955); //Specifying the port number if (bind(serverSocket, (struct sockaddr *)&serverAddress, sizeof(serverAddress)) == -1) { perror("Error binding socket"); exit(EXIT_FAILURE); } // Listening for incoming connections if (listen(serverSocket, 1) == -1) { perror("Error listening"); exit(EXIT_FAILURE); } printf("Server currently is listening on port 9955...\n"); while (1) { // Wait for a connection from a client int clientSocket = accept(serverSocket, NULL, NULL); if (clientSocket == -1) { perror("Error accepting connection"); exit(EXIT_FAILURE); } printf("Accepted connection from a client\n"); // Receive data from the client char buffer[1024]; ssize_t bytes = recv(clientSocket, buffer, sizeof(buffer), 0); if (bytes == -1) { perror("Error receiving data"); exit(EXIT_FAILURE); } buffer[bytes] = '\0'; handleClient(clientSocket, buffer); // Close the connection with the client close(clientSocket); } // Close the server socket close(serverSocket); return 0; }

5.2. For part 3

PYTHON CODE

```
#Rana Musa 1210007
#Donia Alshiakh 1210517
#Haneen Odah 1210716

from socket import *
import os

def send_file(connectionSocket, file_path, content_type):
    with open(file_path, "rb") as file:
        file_data = file.read()
        response = f"HTTP/1.1 200 OK\r\nContent-Type:
{content_type}\r\n\r\n".encode() + file_data
        connectionSocket.send(response)
```

```
def send_error_404(connectionSocket, addr):
    names_and_ids = "Rana Musa   1210007"
    names_and_ids += "<br>Donia Alshiakh 1210517"
    names_and_ids += "<br>Haneen Odah 1210716"

    error_message = f"""
        <html>
        <head>
            <title>Error 404</title>
        </head>
        <body style="text-align: center;">
            <h1 style="color: black;">HTTP/1.1 404 Not Found</h1>
            <p style="font-size: 16px; color: red; font-weight: bold;">The file
is not found</p>
            <b>{names_and_ids}</b><br>
            <b>Client IP:</b> {addr[0]}<br>
            <b>Client Port:</b> {addr[1]}
        </body>
        </html>
    """

    response = f"HTTP/1.1 404 Not Found\r\nContent-Type:
text/html\r\n\r\n{error_message}".encode()

    connectionSocket.send(response)






serverPort = 9966
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind(("", serverPort))
serverSocket.listen(1)
print("The server is ready to receive")

while True:
    try:
        connectionSocket, addr = serverSocket.accept()
        sentence = connectionSocket.recv(2048).decode()
        print(addr)
        print(sentence)
        ip = addr[0]
        port = addr[1]

        lines = sentence.split("\r\n")
        request_line = lines[0]
        request_parts = request_line.split()

        if len(request_parts) > 1:
            request_path = request_parts[1]

            if request_path in ['/', '/index.html', '/main_en.html', '/en']:
                send_file(connectionSocket, 'main_en.html', 'text/html')

            elif request_path == '/ar':
                send_file(connectionSocket, 'main_ar.html', 'text/html')

            elif request_path.endswith('.html'):
                send_file(connectionSocket, request_path[1:], 'text/html')

            elif request_path.endswith('.css'):
                send_file(connectionSocket, request_path[1:], 'text/css')

            elif request_path.endswith('.png'):
                send_file(connectionSocket, request_path[1:], 'image/png')

            elif request_path.endswith('.jpg'):
                send_file(connectionSocket, request_path[1:], 'image/jpeg')

            elif request_path == '/cr':
                response = "HTTP/1.1 307 Temporary Redirect\r\nLocation:
http://cornell.edu\r\n\r\n".encode()
                connectionSocket.send(response)

            elif request_path == '/so':
                response = "HTTP/1.1 307 Temporary Redirect\r\nLocation:
http://stackoverflow.com\r\n\r\n".encode()
                connectionSocket.send(response)

            elif request_path == '/rt':
                response = "HTTP/1.1 307 Temporary Redirect\r\nLocation:
https://ritaj.birzeit.edu/register/\r\n\r\n".encode()
                connectionSocket.send(response)

            else:
                send_error_404(connectionSocket, addr)

        connectionSocket.close()

    except Exception as e:
        print("Error:", e)
        connectionSocket.close()
```

CSS CODE

```
body{
    padding: 20px;
    background-image: url(network.jpg);
    background-size: cover;
    background-position: center; /* Center the background image */
    background-repeat: no-repeat;
  }
  h1{
    text-align: center;
    text-transform: uppercase;
    color: #ffffff;
  }
  .group{
    width: 100%;
    text-align: center;
    margin: auto;
  }

  .student{
    width: 20%;
    height: 25px;
    display: inline-block;
    margin: 5px;
    color: black;
    font-weight: bold;


  }
  .info-box {
    width: 100%;
    height: 250px;
    display: inline-block;
    margin: 5px;
    color: black;
    border: 2px solid #000;
    box-sizing: border-box;
    background-color: pink;
}


   .emptiness-box{
    width: 100%;
    height: 300px;
    display: inline-block;
    margin: 5px;
    color: black;
  }
  .section-box {
    width: 80%;
    height: 50px;
    display: inline-block;
    margin: 5px;
    color: black;
    border: 2px solid #000;
    box-sizing: border-box;
    background-color: rgb(182, 127, 178);
}
.names {
  width: 80%;
  height: 50px;
  display: inline-block;
  margin: 5px;
  color: black;
  border: 2px solid #000;
  box-sizing: border-box;
  background-color: rgb(182, 127, 178);
}

.ids {
  width: 80%;
  height: 50px;
  display: inline-block;
  margin: 5px;
  color: black;
  border: 2px solid #000;
  box-sizing: border-box;
  background-color: rgb(182, 127, 178);
}
  .link-box {
    border: 2px solid #000;
    margin-left: 20px;
    padding: 20px;
}

  div img{
    width: 90%;
    height: 1000%;
  }

  .Part0 {
    width: 80%;
    height: 120px;
    display: inline-block;
    margin: 5px;
    color: black;
    border: 2px solid #000;
    box-sizing: border-box;
    background-color: rgb(209, 187, 207);

  }
```