Faculty of Engineering and Technology

Electrical and Computer Engineering Department

ENCS3320 - Computer Network

Project_1 Report

Prepared by:

Dunia Hamada	1201001	section 1
Tarteel Altaweel	1222670	section 4
Alaa Bader	1200737	section 3

Instructor: Dr. Abdalkarim Awad & Dr. Imad Tarter

Date: 10/11/2023

Table of Contents

List of figures:

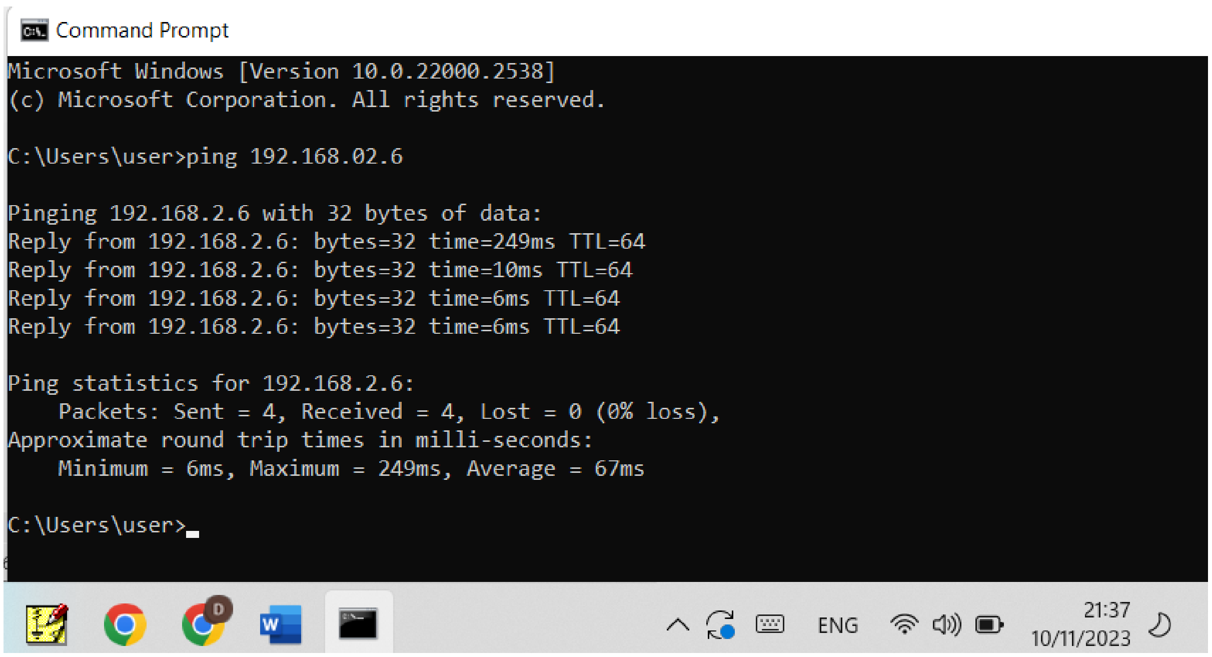
Part 1:

1.1. What Is Ping, Tracert, Nslookup and Telnet?

Ping: is a command-line utility that serves as a test to determine if a networked device is reachable.
Tracert: is a command-line utility that traces the path a protocol packet takes to reach its destination.
Nslookup: is a command used to query domain name and address servers.
Telnet: A network protocol that allows a user to establish a remote connection to a device or server over a TCP/IP network.

1.2. Run Some Commands

1.2.1. Ping a Device in The Same Network e.g. from a laptop to a smartphone



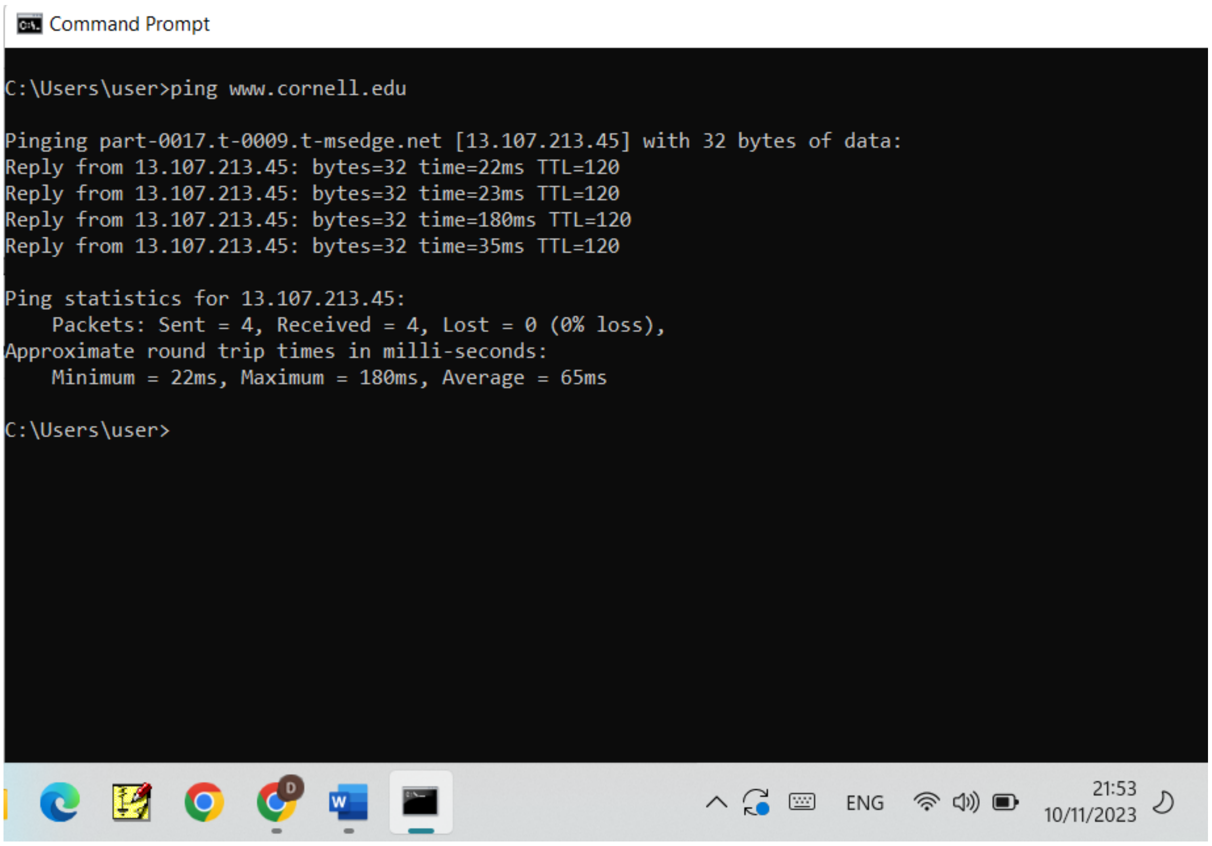
```
Microsoft Windows [Version 10.0.22000.2538]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user>ping 192.168.02.6

Pinging 192.168.2.6 with 32 bytes of data:
Reply from 192.168.2.6: bytes=32 time=249ms TTL=64
Reply from 192.168.2.6: bytes=32 time=10ms TTL=64
Reply from 192.168.2.6: bytes=32 time=6ms TTL=64
Reply from 192.168.2.6: bytes=32 time=6ms TTL=64

Ping statistics for 192.168.2.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 249ms, Average = 67ms

C:\Users\user>_
```

Figure 1:Ping Device in Same Network

Consider Figure 1, When ping a device at IP address 192.168.2.6 , the computer sends four small data packets. If the device is working, it sends these packets back, called "echo reply requests" Each attempt shows as a line in the result. If get a response, it means computer can talk to the device, showing the network is connected. "Request timed out" means the device might be down or blocking pings. "Destination host unreachable" suggests a problem finding the route. The "time" is how long a packet takes to go to the device and back (RTT). "Time to live" (TTL) is when a resource is removed from a cache. The ping command sends 32 bytes of data by default.

In this case, we sent four packets, got all back with no loss, and the RTT values (minimum, maximum, and average) are all 0ms. This means a fast and responsive connection between the computer and the device.

1.2.2. Ping www.cornell.edu

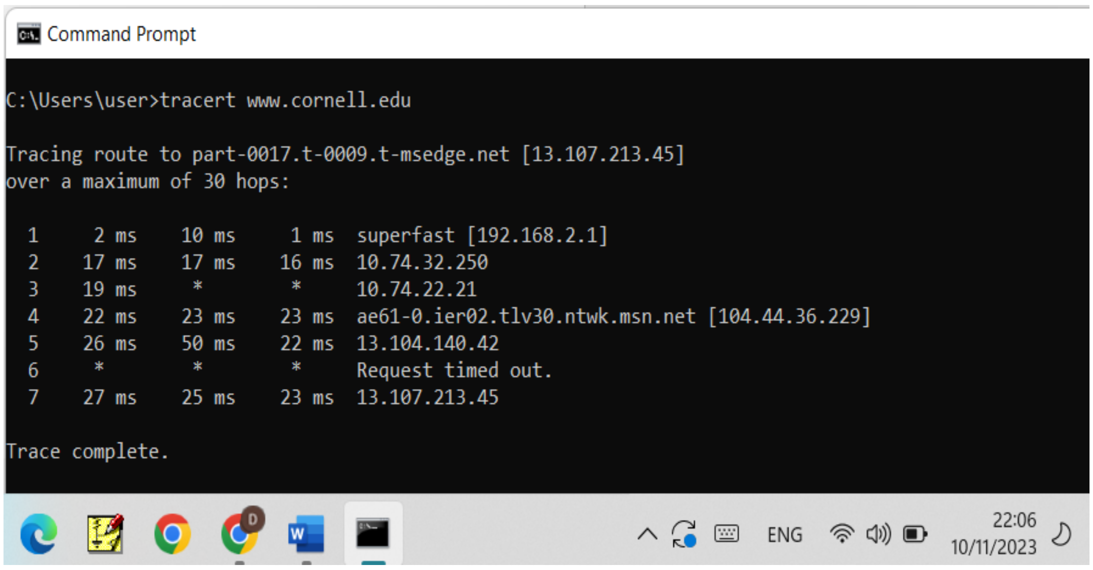

Command Prompt

```
C:\Users\user>ping www.cornell.edu

Pinging part-0017.t-0009.t-msedge.net [13.107.213.45] with 32 bytes of data:
Reply from 13.107.213.45: bytes=32 time=22ms TTL=120
Reply from 13.107.213.45: bytes=32 time=23ms TTL=120
Reply from 13.107.213.45: bytes=32 time=180ms TTL=120
Reply from 13.107.213.45: bytes=32 time=35ms TTL=120

Ping statistics for 13.107.213.45:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 22ms, Maximum = 180ms, Average = 65ms

C:\Users\user>
```

Figure 2:Ping www.cornell.edu

That response shows the URL you are pinging, its IP address, and the packet size. The next four lines show the response details for each packet: the time it took for the response (in milliseconds) and the time-to-live (TTL), which is how long before the packet is discarded. the minimum time for a response (RTT) was 22 ms, the maximum was 180 ms, and the average RTT was 65 ms. The average RTT is calculated by adding the RTT for each packet and dividing by the number of packets (in this case, (22+23+180+35)/4=65), showing the overall responsiveness of the connection.

1.2.3. Tracer www.cornell.edu



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

Tracing route to part-0017.t-0009.t-msedge.net [13.107.213.45]
over a maximum of 30 hops:

  1     2 ms    10 ms     1 ms  superfast [192.168.2.1]
  2    17 ms    17 ms    16 ms  10.74.32.250
  3    19 ms     *        *     10.74.22.21
  4    22 ms    23 ms    23 ms  ae61-0.ier02.tlv30.ntwk.msn.net [104.44.36.229]
  5    26 ms    50 ms    22 ms  13.104.140.42
  6     *        *        *     Request timed out.
  7    27 ms    25 ms    23 ms  13.107.213.45

Trace complete.
```

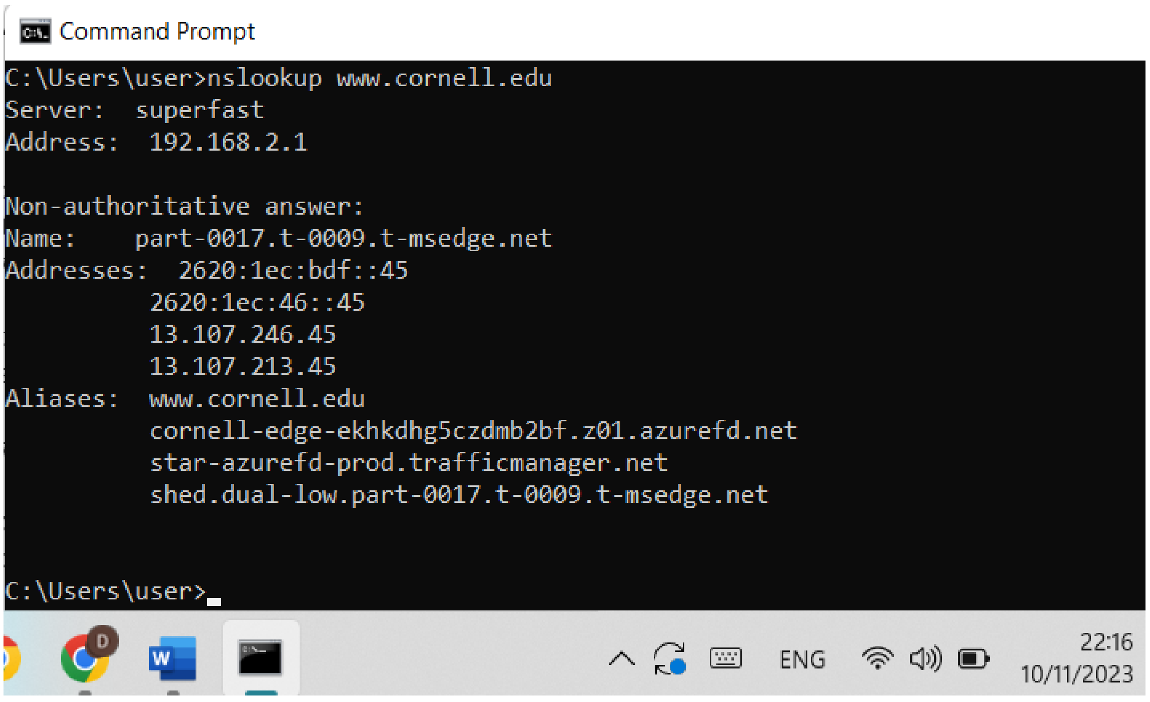
Figure 3:Tracer www.cornell.edu

The displayed output is the result of running the tracert command on a Windows system to trace the network route to "**www.cornell.edu**" It begins by showing the Windows version and then reveals the target destination as "pantheon-systems.map.fastly.net" with the IP address 13.107.213.45. The trace is allowed a maximum of 30 hops.

Each line in the output represents a step, or hop, toward the destination. It includes the hop number, the response time in milliseconds, and the IP address of the network device or router at that step. The journey starts with the first hop having an IP address of 192.168.2.1. As the trace progresses, it shows different IP addresses and corresponding response times for each hop. The final line, "Trace complete," indicates a successful trace to the destination.

This tracert output is valuable for understanding the network path taken to reach "**www.cornell.edu**" providing details like intermediate router IP addresses and response times at each step. It's useful for troubleshooting connectivity issues, identifying latency or performance problems, and gaining insights into the network infrastructure involved in reaching the destination.

1.2.4. Nslookup www.cornell.edu



Command Prompt

```
C:\Users\user>nslookup www.cornell.edu
Server:  superfast
Address:  192.168.2.1

Non-authoritative answer:
Name:    part-0017.t-0009.t-msedge.net
Addresses:  2620:1ec:bdf::45
          2620:1ec:46::45
          13.107.246.45
          13.107.213.45
Aliases:  www.cornell.edu
          cornell-edge-ekhkdhg5czdmb2bf.z01.azurefd.net
          star-azurefd-prod.trafficmanager.net
          shed.dual-low.part-0017.t-0009.t-msedge.net


C:\Users\user>_
```

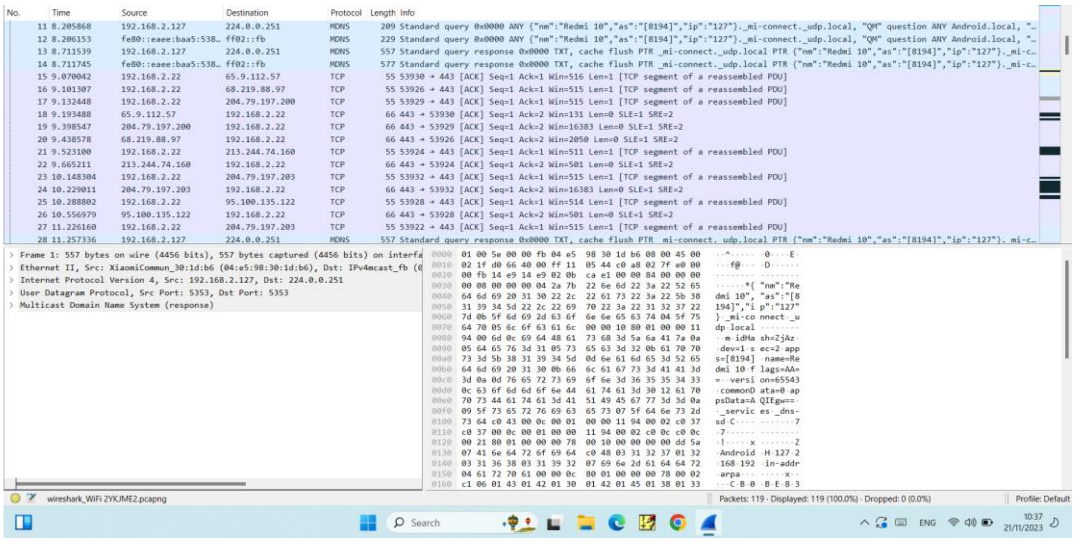
Figure 4:. Nslookup www.cornell.edu

The provided output is the result of running the nslookup command on a Windows system to find the IP address of www.cornell.edu , The command was executed using a DNS server named "superfast" with the IP address 192.168.2.1.

In the non-authoritative answer section, the output reveals that "www.cornell.edu " has an alias named "part-0017.t-0009.t-msedge.net." It provides both IPv6 and IPv4 addresses associated with this alias: 2620:1ec:bdf::45, 2620:1ec:46::45, 13.107.246.45, and 13.107.213.45.Additionally, the output lists aliases for "www.cornell.edu " including "cornell-edge-ekhkdhg5czdmb2bf.z01.azurefd.net" and "star-azurefd-prod.trafficmanager.net." The final line indicates that the alias "www.cornell.edu" is associated with the server "shed.dual-low.part-0017.t-0009.t-msedge.net."

the output provides information about the IP addresses and aliases associated with the domain "www.cornell.edu" and the DNS server used for the lookup.

1.3. wireshark to capture some DNS messages.



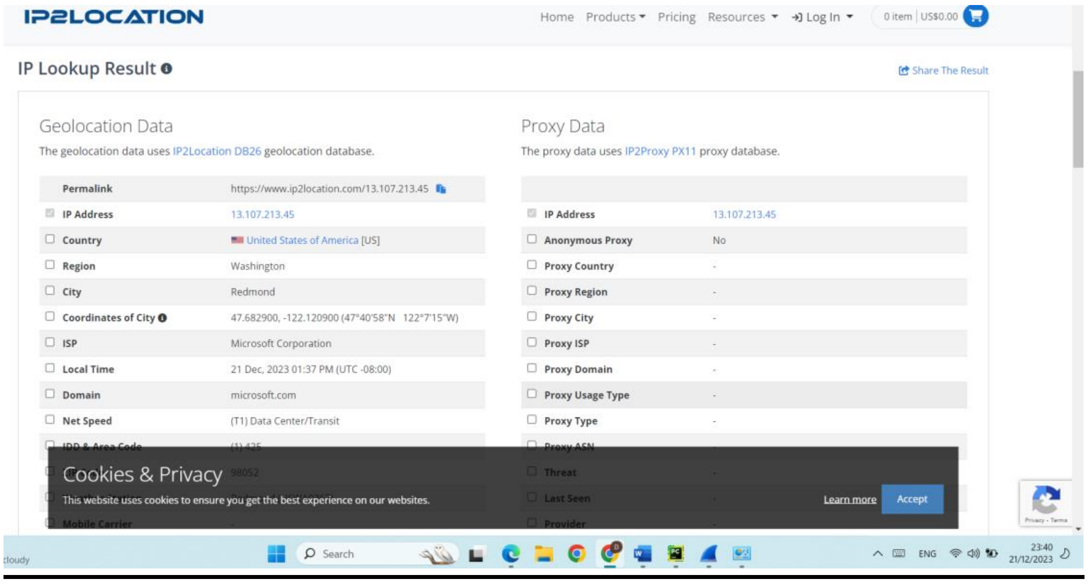
```
No.  Time       Source               Destination      Protocol Length Info
  11 8.205868   192.168.2.127        224.0.0.251      MDNS     209 Standard query 0x0000 ANY {"nm":"Redmi 10","as":"[8194]","ip":"127"}._mi-connect._udp.local, "QM" question ANY Android.local, "…
  12 8.206153   fe80::eaee:baa5:538… ff02::fb         MDNS     229 Standard query 0x0000 ANY {"nm":"Redmi 10","as":"[8194]","ip":"127"}._mi-connect._udp.local, "QM" question ANY Android.local, "…
  13 8.711539   192.168.2.127        224.0.0.251      MDNS     557 Standard query response 0x0000 TXT, cache flush PTR _mi-connect._udp.local PTR {"nm":"Redmi 10","as":"[8194]","ip":"127"}._mi-c…
  14 8.711745   fe80::eaee:baa5:538… ff02::fb         MDNS     577 Standard query response 0x0000 TXT, cache flush PTR _mi-connect._udp.local PTR {"nm":"Redmi 10","as":"[8194]","ip":"127"}._mi-c…
  15 9.070042   192.168.2.22         65.9.112.57      TCP       55 53930 → 443 [ACK] Seq=1 Ack=1 Win=516 Len=1 [TCP segment of a reassembled PDU]
  16 9.101307   192.168.2.22         68.219.88.97     TCP       55 53926 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1 [TCP segment of a reassembled PDU]
  17 9.132448   192.168.2.22         204.79.197.200   TCP       55 53929 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1 [TCP segment of a reassembled PDU]
  18 9.193488   65.9.112.57          192.168.2.22     TCP       66 443 → 53930 [ACK] Seq=1 Ack=2 Win=131 Len=0 SLE=1 SRE=2
  19 9.398547   204.79.197.200       192.168.2.22     TCP       66 443 → 53929 [ACK] Seq=1 Ack=2 Win=16383 Len=0 SLE=1 SRE=2
  20 9.438578   68.219.88.97         192.168.2.22     TCP       66 443 → 53926 [ACK] Seq=1 Ack=2 Win=2050 Len=0 SLE=1 SRE=2
  21 9.523100   192.168.2.22         213.244.74.160   TCP       55 53924 → 443 [ACK] Seq=1 Ack=1 Win=511 Len=1 [TCP segment of a reassembled PDU]
  22 9.665211   213.244.74.160       192.168.2.22     TCP       66 443 → 53924 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
  23 10.148304  192.168.2.22         204.79.197.203   TCP       55 53932 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1 [TCP segment of a reassembled PDU]
  24 10.229011  204.79.197.203       192.168.2.22     TCP       66 443 → 53932 [ACK] Seq=1 Ack=2 Win=16383 Len=0 SLE=1 SRE=2
  25 10.288802  192.168.2.22         95.100.135.122   TCP       55 53928 → 443 [ACK] Seq=1 Ack=1 Win=514 Len=1 [TCP segment of a reassembled PDU]
  26 10.556979  95.100.135.122       192.168.2.22     TCP       66 443 → 53928 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
  27 11.226160  192.168.2.22         204.79.197.203   TCP       55 53922 → 443 [ACK] Seq=1 Ack=1 Win=515 Len=1 [TCP segment of a reassembled PDU]
  28 11.257336  192.168.2.127        224.0.0.251      MDNS     557 Standard query response 0x0000 TXT, cache flush PTR _mi-connect._udp.local PTR {"nm":"Redmi 10","as":"[8194]","ip":"127"}._mi-c…

> Frame 1: 557 bytes on wire (4456 bits), 557 bytes captured (4456 bits) on interfa
> Ethernet II, Src: XiaomiCommun_30:1d:b6 (04:e5:98:30:1d:b6), Dst: IPv4mcast_fb (0
> Internet Protocol Version 4, Src: 192.168.2.127, Dst: 224.0.0.251
> User Datagram Protocol, Src Port: 5353, Dst Port: 5353
> Multicast Domain Name System (response)

0000  01 00 5e 00 00 fb 04 e5  98 30 1d b6 08 00 45 00   ··^·····0·····E·
0010  02 1f d0 66 40 00 ff 11  05 44 c0 a8 02 7f e0 00   ···f@····D······
0020  00 fb 14 e9 14 e9 02 0b  ca e1 00 00 84 00 00 00   ················
0030  00 08 00 00 00 04 2a 7b  22 6e 6d 22 3a 22 52 65   ······*{ "nm":"Re
0040  64 6d 69 20 31 30 22 2c  22 61 73 22 3a 22 5b 38   dmi 10", "as":"[8
0050  31 39 34 5d 22 2c 22 69  70 22 3a 22 31 32 37 22   194]","i p":"127"
0060  7d 0b 5f 6d 69 2d 63 6f  6e 6e 65 63 74 04 5f 75   }·_mi-co nnect·_u
0070  64 70 05 6c 6f 63 61 6c  00 00 10 80 01 00 00 11   dp·local ········
0080  94 00 6d 0c 69 64 48 61  73 68 3d 5a 6a 41 7a 0a   ··m·idHa sh=ZjAz·
0090  05 64 65 76 3d 31 05 73  65 63 3d 32 0b 61 70 70   ·dev=1·s ec=2·app
00a0  73 3d 5b 38 31 39 34 5d  0d 6e 61 6d 65 3d 52 65   s=[8194] ·name=Re
00b0  64 6d 69 20 31 30 0b 66  6c 61 67 73 3d 41 41 3d   dmi 10·f lags=AA=
00c0  3d 0a 0d 76 65 72 73 69  6f 6e 3d 36 35 35 34 33   =··versi on=65543
00d0  0c 63 6f 6d 6d 6f 6e 44  61 74 61 3d 30 12 61 70   ·commonD ata=0·ap
00e0  70 73 44 61 74 61 3d 41  51 49 45 67 77 3d 3d 0a   psData=A QIEgw==·
00f0  09 5f 73 65 72 76 69 63  65 73 07 5f 64 6e 73 2d   ·_servic es·_dns-
0100  73 64 c0 43 00 0c 00 01  00 00 11 94 00 02 c0 37   sd·C···· ·······7
0110  c0 37 00 0c 00 01 00 00  11 94 00 02 c0 0c c0 0c   ·7······ ········
0120  00 21 80 01 00 00 00 78  00 10 00 00 00 00 dd 5a   ·!·····x ·······Z
0130  07 41 6e 64 72 6f 69 64  c0 48 03 31 32 37 01 32   ·Android ·H·127·2
0140  03 31 36 38 03 31 39 32  07 69 6e 2d 61 64 64 72   ·168·192 ·in-addr
0150  04 61 72 70 61 00 00 0c  80 01 00 00 00 78 00 02   ·arpa··· ·····x··
0160  c1 06 01 43 01 42 01 30  01 42 01 45 01 38 01 33   ···C·B·0 ·B·E·8·3

wireshark_WiFi 2YKJME2.pcapng     Packets: 119 · Displayed: 119 (100.0%) · Dropped: 0 (0.0%)     Profile: Default
```

Figure 5:wireshark

This picture shows Wireshark, a tool that looks at the information on a network.

It shows how devices communicate using different methods like TCP , UDP , and DNS .

and Where Devices Are and How They're Connected can find IP addresses assigned to each device on the network. It also shows which entry ports .



IP Lookup Result

Geolocation Data

The geolocation data uses IP2Location DB26 geolocation database.

Permalink	https://www.ip2location.com/13.107.213.45
IP Address	13.107.213.45
Country	United States of America [US]
Region	Washington
City	Redmond
Coordinates of City	47.682900, -122.120900 (47°40'58"N 122°7'15"W)
ISP	Microsoft Corporation
Local Time	21 Dec, 2023 01:37 PM (UTC -08:00)
Domain	microsoft.com
Net Speed	(T1) Data Center/Transit
IDD & Area Code	(1) 425
	98052
Mobile Carrier	-

Proxy Data

The proxy data uses IP2Proxy PX11 proxy database.

IP Address	13.107.213.45
Anonymous Proxy	No
Proxy Country	-
Proxy Region	-
Proxy City	-
Proxy ISP	-
Proxy Domain	-
Proxy Usage Type	-
Proxy Type	-
Proxy ASN	-
Threat	-
Last Seen	-
Provider	-

Cookies & Privacy

This website uses cookies to ensure you get the best experience on our websites.

The information obtained from the ping response typically includes the time it took for the data to travel from source to destination and back. The IP address is specified as belonging to a range assigned to the United States of America, which indicates that the server or device may be located in the United States.

Part 2:

2.1. Required :

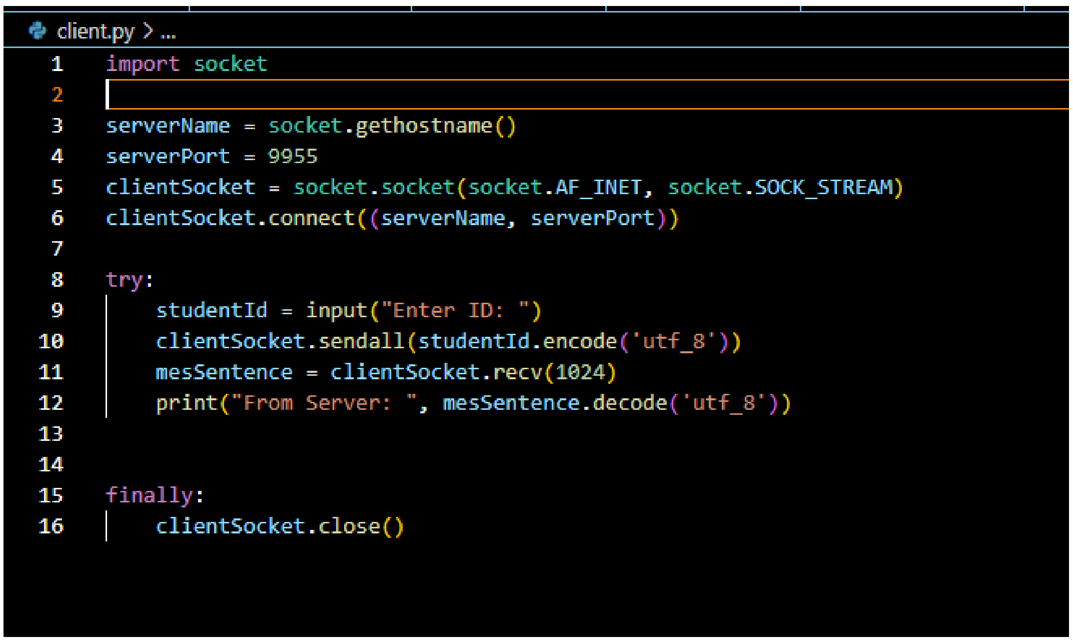
Using socket programming, implement TCP client and server applications in python. The server should listen on port 9955. The client send a message to server . The message should contain one of the students ID. The server should do the following:

- display a message on the server side that the OS will lock screen after 10 seconds
- send a message to the client that the sever will lock screen after 10 seconds
- then wait 10 seconds
- then call a function lock the screen of the operating system

2.2. Run the program using TCP client and server :

Transmission Control Protocol is a basic standard that defines the rules of the internet. provides reliable because there is Three-way handshake , byte stream-oriented , and is a connection between client and server , and is a common protocol used to transport data in digital network communications.

Client code in TCP:

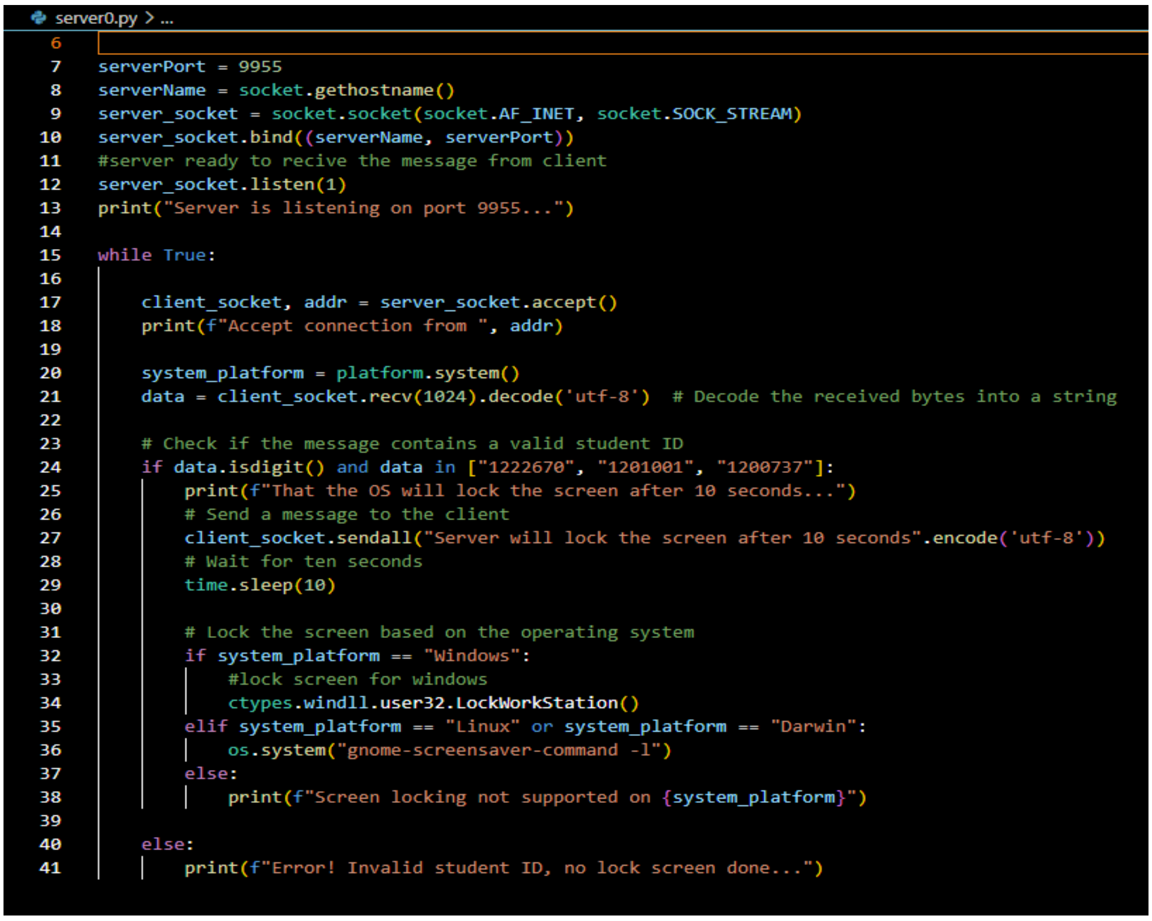


client.py > ...

```python
import socket

serverName = socket.gethostname()
serverPort = 9955
clientSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
clientSocket.connect((serverName, serverPort))

try:
    studentId = input("Enter ID: ")
    clientSocket.sendall(studentId.encode('utf_8'))
    mesSentence = clientSocket.recv(1024)
    print("From Server: ", mesSentence.decode('utf_8'))


finally:
    clientSocket.close()
```

Figure 6:Client Code TCP

Server code in TCP:

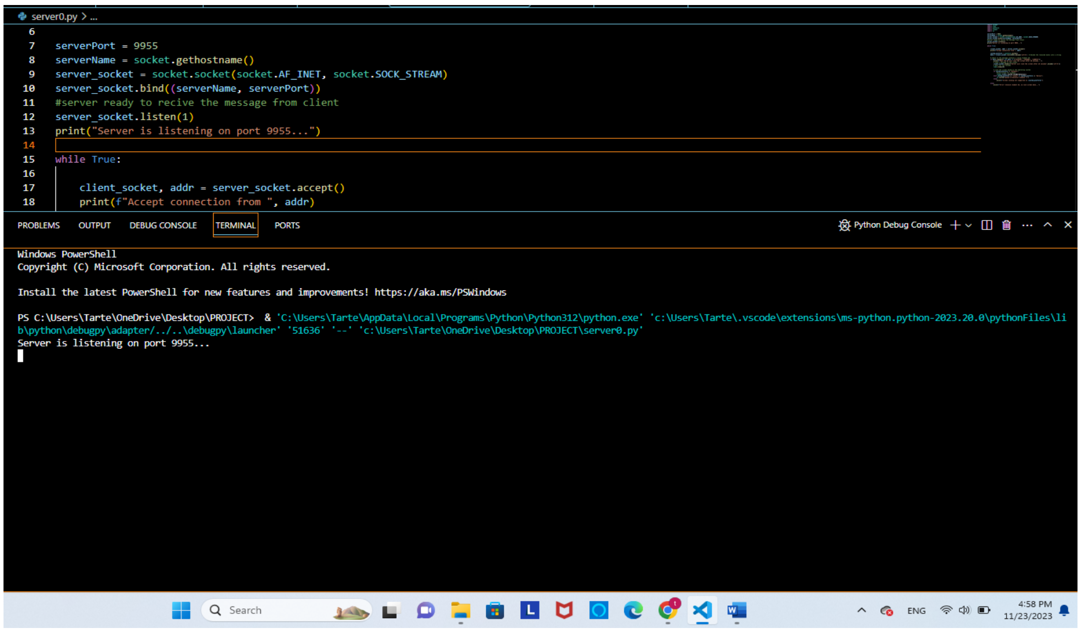


server0.py > ...

```python

serverPort = 9955
serverName = socket.gethostname()
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind((serverName, serverPort))
#server ready to recive the message from client
server_socket.listen(1)
print("Server is listening on port 9955...")

while True:

    client_socket, addr = server_socket.accept()
    print(f"Accept connection from ", addr)

    system_platform = platform.system()
    data = client_socket.recv(1024).decode('utf-8')  # Decode the received bytes into a string

    # Check if the message contains a valid student ID
    if data.isdigit() and data in ["1222670", "1201001", "1200737"]:
        print(f"That the OS will lock the screen after 10 seconds...")
        # Send a message to the client
        client_socket.sendall("Server will lock the screen after 10 seconds".encode('utf-8'))
        # Wait for ten seconds
        time.sleep(10)

        # Lock the screen based on the operating system
        if system_platform == "Windows":
            #lock screen for windows
            ctypes.windll.user32.LockWorkStation()
        elif system_platform == "Linux" or system_platform == "Darwin":
            os.system("gnome-screensaver-command -l")
        else:
            print(f"Screen locking not supported on {system_platform}")

    else:
        print(f"Error! Invalid student ID, no lock screen done...")
```

Figure 7:Server Code TCP

2.2.1 When we run the server, the server should listen on port 9955:

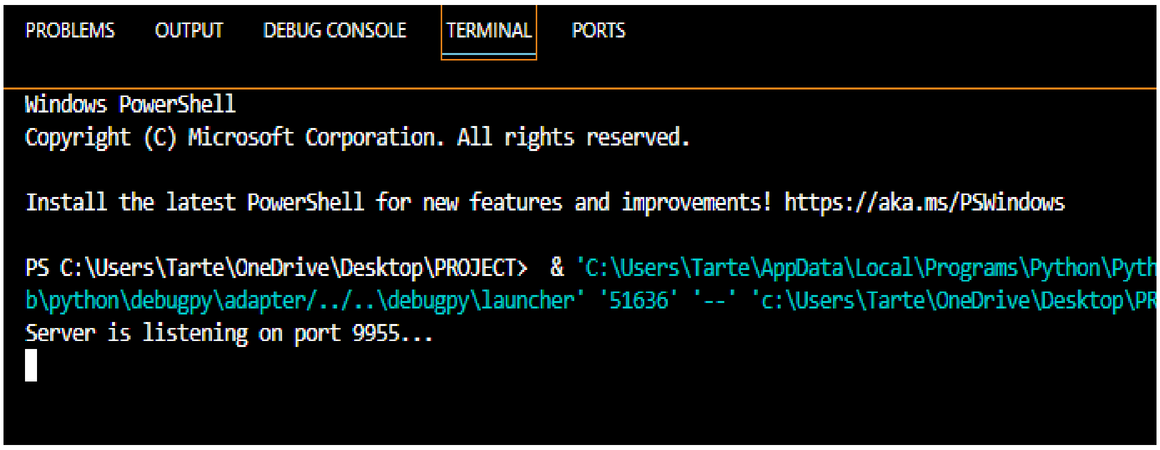


server0.py > ...

```python

serverPort = 9955
serverName = socket.gethostname()
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
server_socket.bind((serverName, serverPort))
#server ready to recive the message from client
server_socket.listen(1)
print("Server is listening on port 9955...")

while True:

    client_socket, addr = server_socket.accept()
    print(f"Accept connection from ", addr)
```

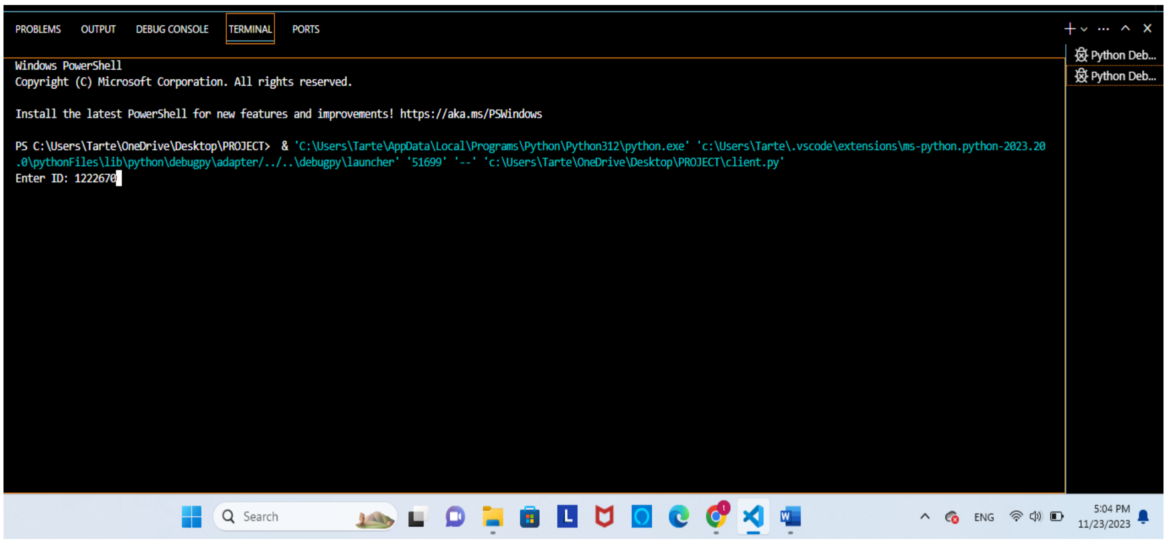
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Tarte\OneDrive\Desktop\PROJECT>  & 'C:\Users\Tarte\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\Tarte\.vscode\extensions\ms-python.python-2023.20.0\pythonFiles\li
b\python\debugpy\adapter/../..\debugpy\launcher' '51636' '--' 'c:\Users\Tarte\OneDrive\Desktop\PROJECT\server0.py'
Server is listening on port 9955...
```

Figure 8:server listen on port 9955



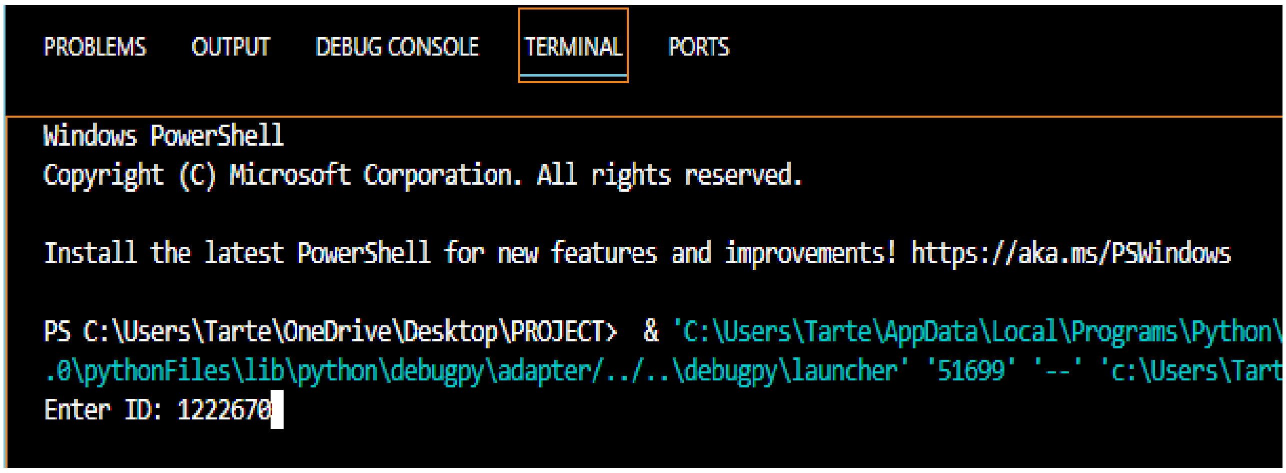
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Tarte\OneDrive\Desktop\PROJECT>  & 'C:\Users\Tarte\AppData\Local\Programs\Python\Pyth
b\python\debugpy\adapter/../..\debugpy\launcher' '51636' '--' 'c:\Users\Tarte\OneDrive\Desktop\PR
Server is listening on port 9955...
```

2.2.2 Then the client send a message to server contain one of the students ID:



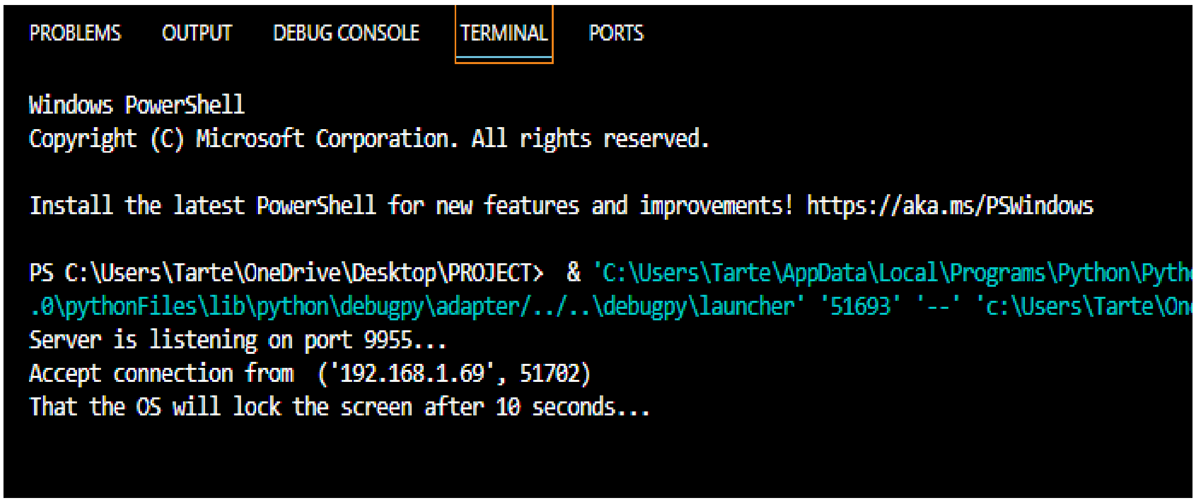
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Tarte\OneDrive\Desktop\PROJECT>  & 'C:\Users\Tarte\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\Tarte\.vscode\extensions\ms-python.python-2023.20
.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '51699' '--' 'c:\Users\Tarte\OneDrive\Desktop\PROJECT\client.py'
Enter ID: 1222670
```

Figure 9:client send a message to server



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Tarte\OneDrive\Desktop\PROJECT>  & 'C:\Users\Tarte\AppData\Local\Programs\Python\
.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '51699' '--' 'c:\Users\Tart
Enter ID: 1222670
```

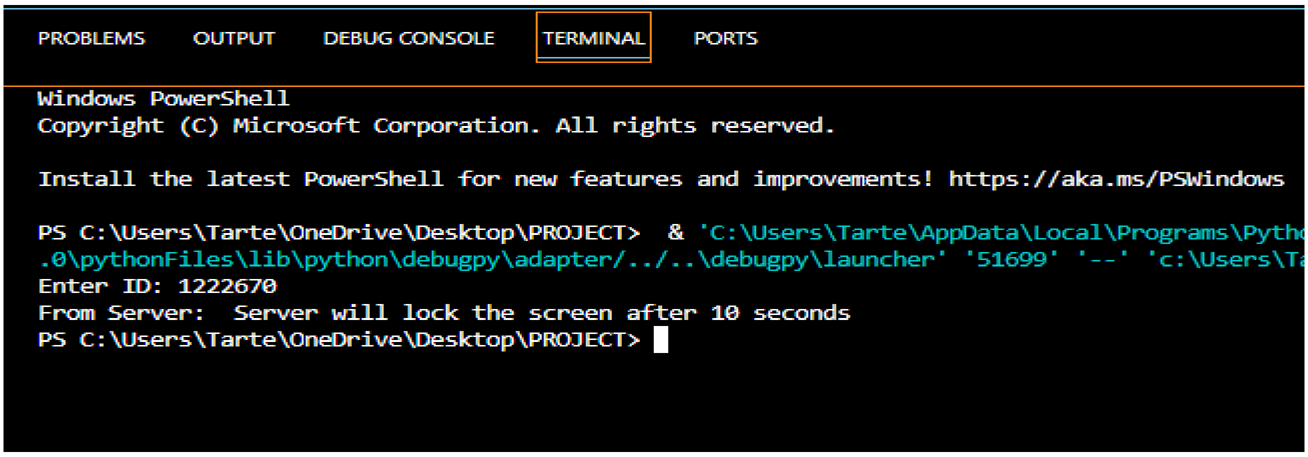
2.2.3 Display a message on the server side that the OS will lock screen after 10 seconds:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Tarte\OneDrive\Desktop\PROJECT>  & 'C:\Users\Tarte\AppData\Local\Programs\Python\Pyth
.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '51693' '--' 'c:\Users\Tarte\On
Server is listening on port 9955...
Accept connection from  ('192.168.1.69', 51702)
That the OS will lock the screen after 10 seconds...
```

Figure 10:display a message on the server

2.2.4 The server send a message to client to lock the screen after 10 seconds:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Tarte\OneDrive\Desktop\PROJECT>  & 'C:\Users\Tarte\AppData\Local\Programs\Pytho
.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '51699' '--' 'c:\Users\Ta
Enter ID: 1222670
From Server:  Server will lock the screen after 10 seconds
PS C:\Users\Tarte\OneDrive\Desktop\PROJECT> 
```

Figure 11:server send a message to client

2.2.5 After waiting 10 seconds the screen will locked:

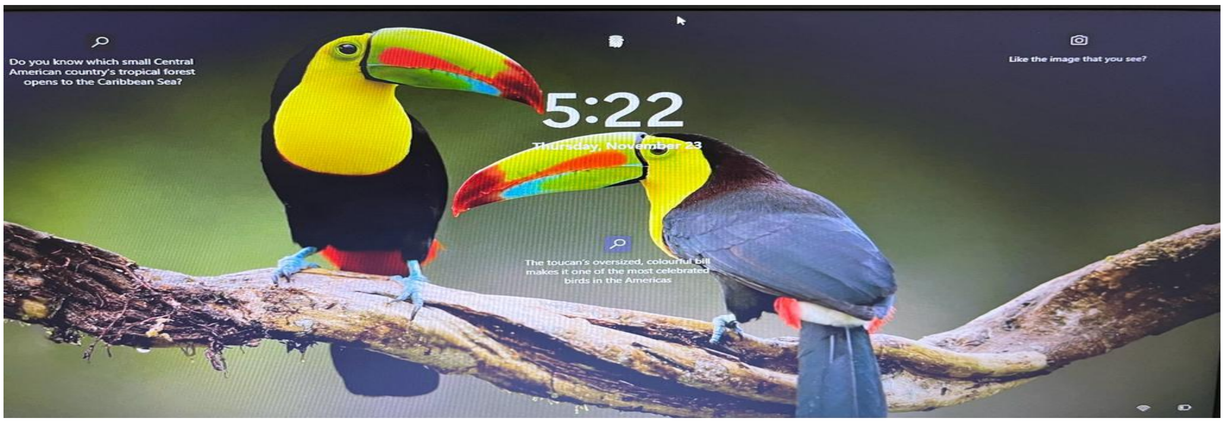

Do you know which small Central American country's tropical forest opens to the Caribbean Sea?
5:22
Thursday, November 23
Like the image that you see?
The toucan's oversized, colourful bill makes it one of the most celebrated birds in the Americas

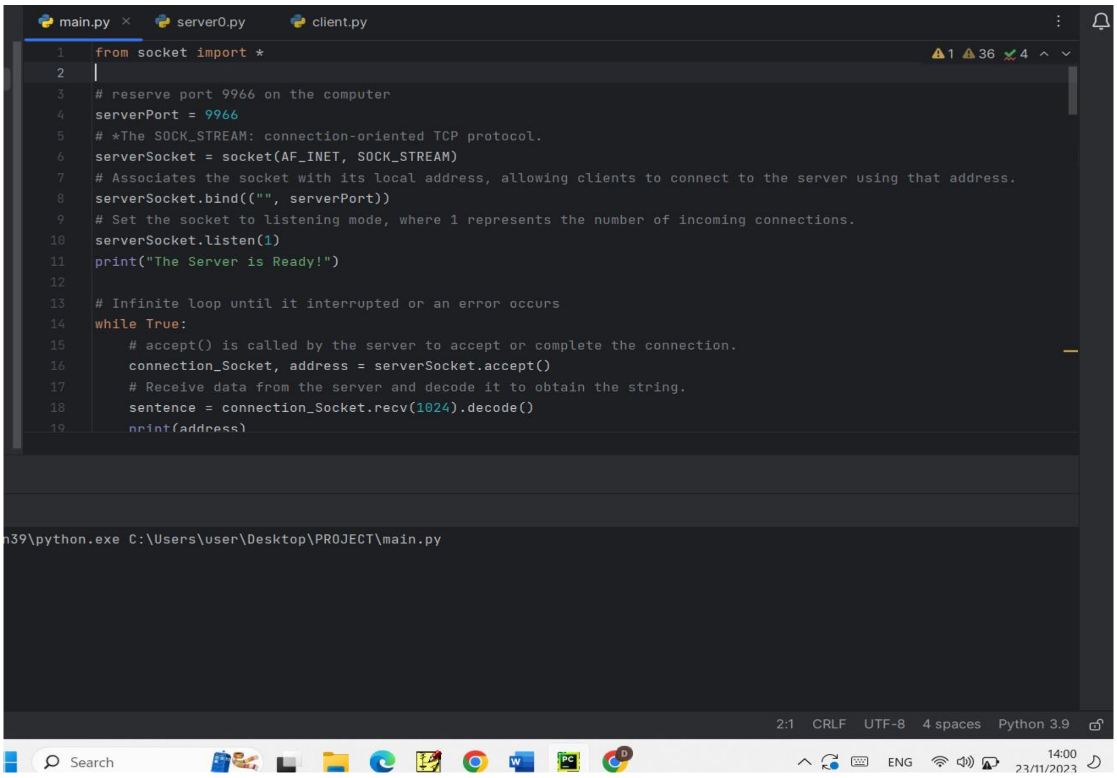
Figure 12:screen locked

Part 3:

In this part, we used socket programming to create a complete web server on port 9966 using the Python programming language, as well as HTML, CSS.

First, we defined a server port which is 9955, then we created a socket instance and pass it two parameters, the first one is AF_INET which means to use IPv4 protocol, and the second one is SOCK_STREAM which means connection-oriented TCP protocol. As shown in figure .

After creating the socket, we associated it with its local address, allowing clients to connect to the server using that address using bind(). Then we put the socket into listening mode to make it ready for the requests.

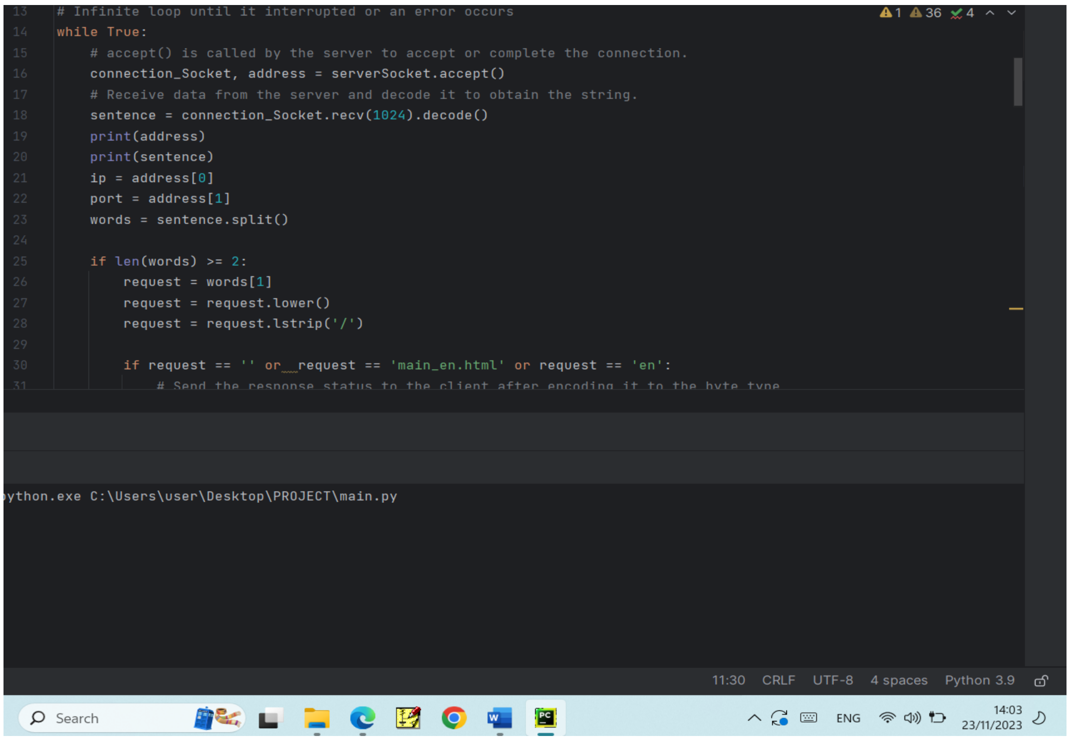


```
from socket import *

# reserve port 9966 on the computer
serverPort = 9966
# *The SOCK_STREAM: connection-oriented TCP protocol.
serverSocket = socket(AF_INET, SOCK_STREAM)
# Associates the socket with its local address, allowing clients to connect to the server using that address.
serverSocket.bind(("", serverPort))
# Set the socket to listening mode, where 1 represents the number of incoming connections.
serverSocket.listen(1)
print("The Server is Ready!")

# Infinite loop until it interrupted or an error occurs
while True:
    # accept() is called by the server to accept or complete the connection.
    connection_Socket, address = serverSocket.accept()
    # Receive data from the server and decode it to obtain the string.
    sentence = connection_Socket.recv(1024).decode()
    print(address)
```

```
n39\python.exe C:\Users\user\Desktop\PROJECT\main.py
```

Figure 13:create the socket

The server will accept and complete the connection by using accept(). Then we will receive the http request from the server and decode it to store it in the "sentence" variable. We will get the request by splitting "sentence" and taking what is after GET /… As shown in figure below.



```
# Infinite loop until it interrupted or an error occurs
while True:
    # accept() is called by the server to accept or complete the connection.
    connection_Socket, address = serverSocket.accept()
    # Receive data from the server and decode it to obtain the string.
    sentence = connection_Socket.recv(1024).decode()
    print(address)
    print(sentence)
    ip = address[0]
    port = address[1]
    words = sentence.split()

    if len(words) >= 2:
        request = words[1]
        request = request.lower()
        request = request.lstrip('/')

        if request == '' or  request == 'main_en.html' or request == 'en':
            # Send the response status to the client after encoding it to the byte type
```

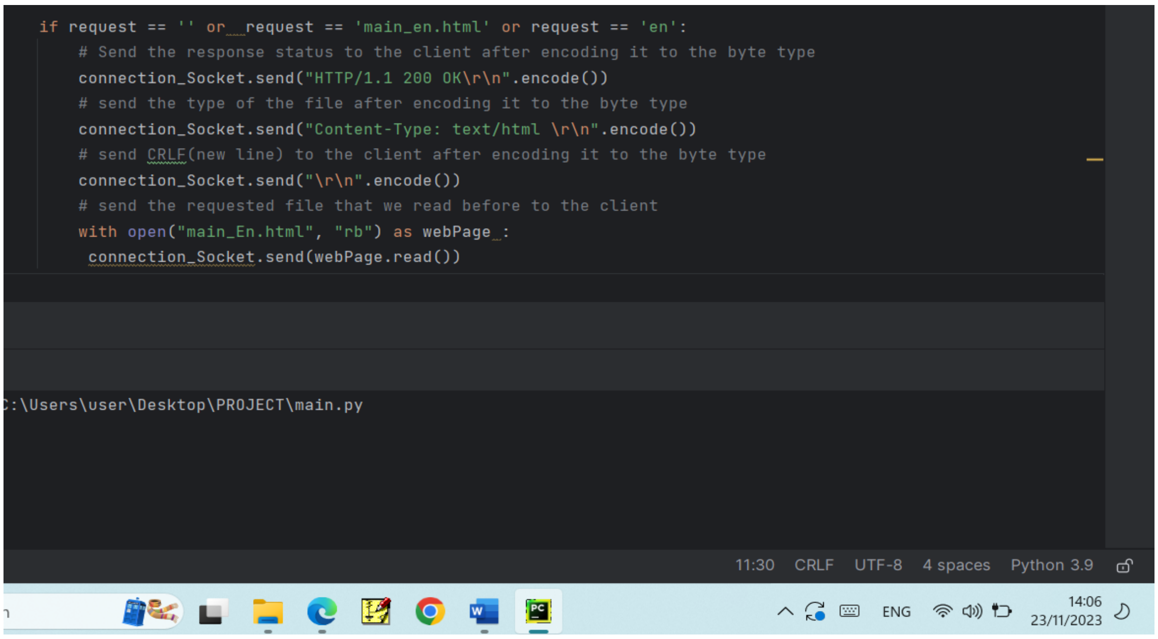
```
ython.exe C:\Users\user\Desktop\PROJECT\main.py
```

```
11:30   CRLF   UTF-8   4 spaces   Python 3.9
```

Figure 14:Accept and complete the connection

3.1. Request the Main Html File

If the request is / or /index.html or /main_en.html or / en then the server should send main_en.html file. If the request is / or main_ar html then the main html file will be sent to the client. To send the html file, the server will first open that file then read it. Then it will send the header which contains of the encoded response status. Finally, the server will send the encoded file that it read previously. As shown in figure below.

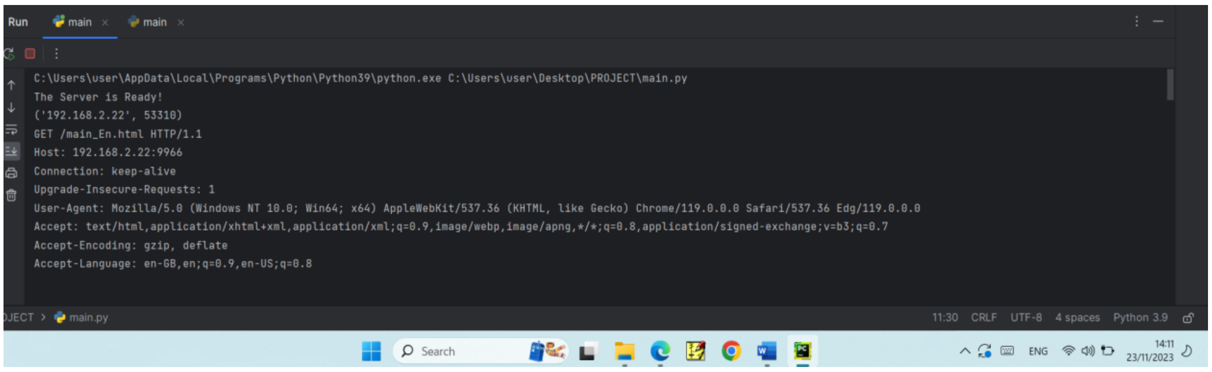


```
if request == '' or  request == 'main_en.html' or request == 'en':
    # Send the response status to the client after encoding it to the byte type
    connection_Socket.send("HTTP/1.1 200 OK\r\n".encode())
    # send the type of the file after encoding it to the byte type
    connection_Socket.send("Content-Type: text/html \r\n".encode())
    # send CRLF(new line) to the client after encoding it to the byte type
    connection_Socket.send("\r\n".encode())
    # send the requested file that we read before to the client
    with open("main_En.html", "rb") as webPage :
     connection_Socket.send(webPage.read())
```

```
:\Users\user\Desktop\PROJECT\main.py
```

```
11:30   CRLF   UTF-8   4 spaces   Python 3.9
```

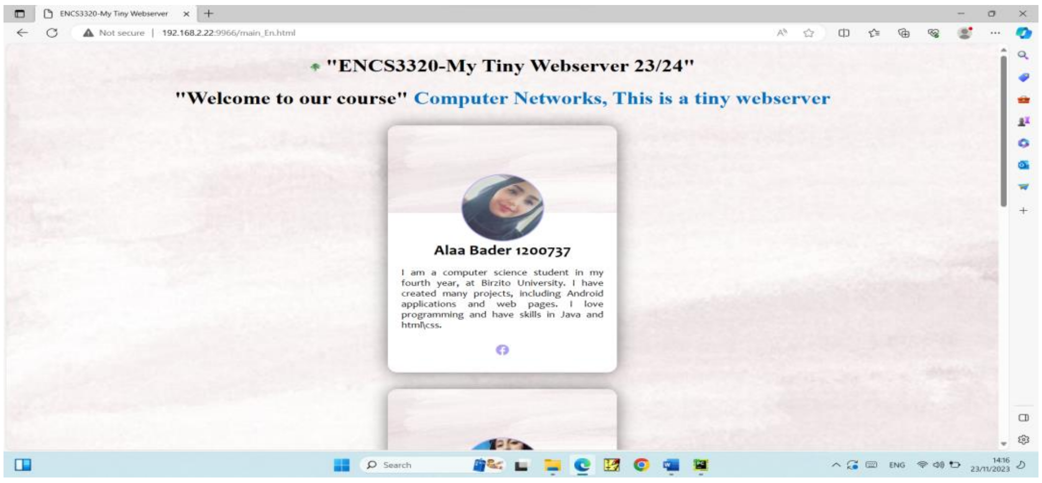
Figure 15: Request the main html file



```
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe C:\Users\user\Desktop\PROJECT\main.py
The Server is Ready!
('192.168.2.22', 53310)
GET /main_En.html HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
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 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 16:Response of local html file

- The local HTML file:



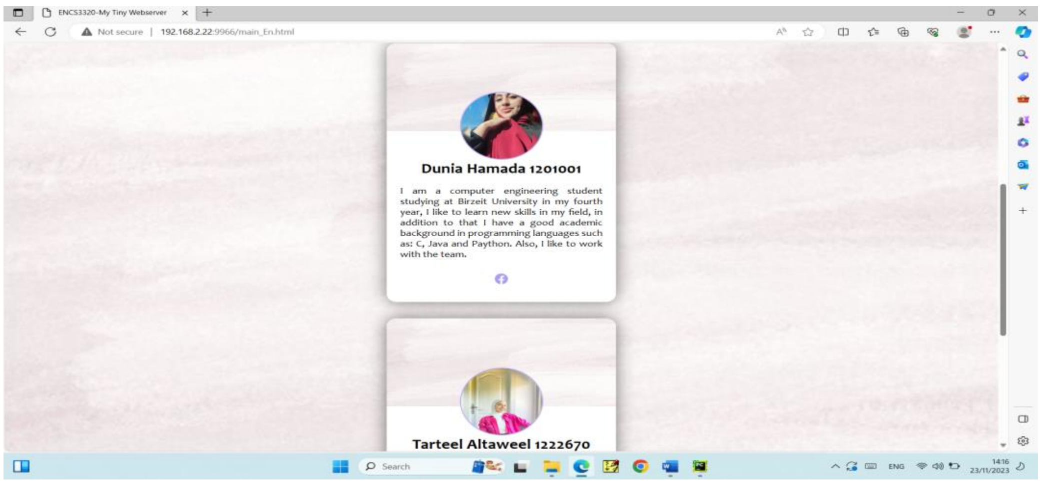
"ENCS3320-My Tiny Webserver 23/24"

"Welcome to our course" Computer Networks, This is a tiny webserver

Alaa Bader 1200737

I am a computer science student in my fourth year, at Birzito University. I have created many projects, including Android applications and web pages. I love programming and have skills in Java and html\css.

ENCS3320-My Tiny Webserver
Not secure | 192.168.2.22:9966/main_En.html

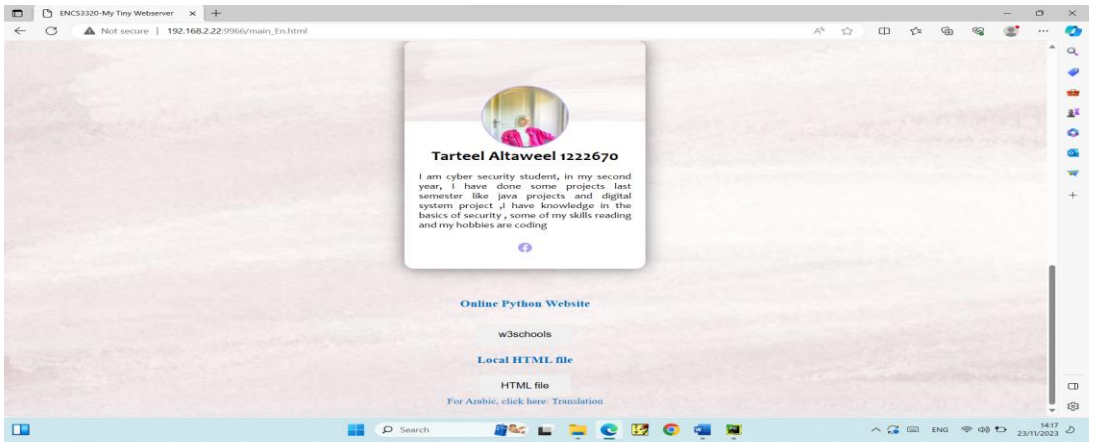


Dunia Hamada 1201001

I am a computer engineering student studying at Birzeit University in my fourth year, I like to learn new skills in my field, in addition to that I have a good academic background in programming languages such as: C, Java and Paython. Also, I like to work with the team.

Tarteel Altaweel 1222670

ENCS3320-My Tiny Webserver
Not secure | 192.168.2.22:9966/main_En.html


Tarteel Altaweel 1222670

I am cyber security student, in my second year, I have done some projects last semester like java projects and digital system project ,I have knowledge in the basics of security , some of my skills reading and my hobbies are coding

Online Python Website

w3schools

Local HTML file

HTML file

For Arabic, click here: Translation

ENCS3320-My Tiny Webserver
Not secure | 192.168.2.22:9966/main_En.html
Search

Figure 17:main_en.html

3.2. Request Arabic version of main_ar.html

If the request is ar or main_ar.html file, then the server will send the **main_ar.html** file by doing the same as the previous steps.

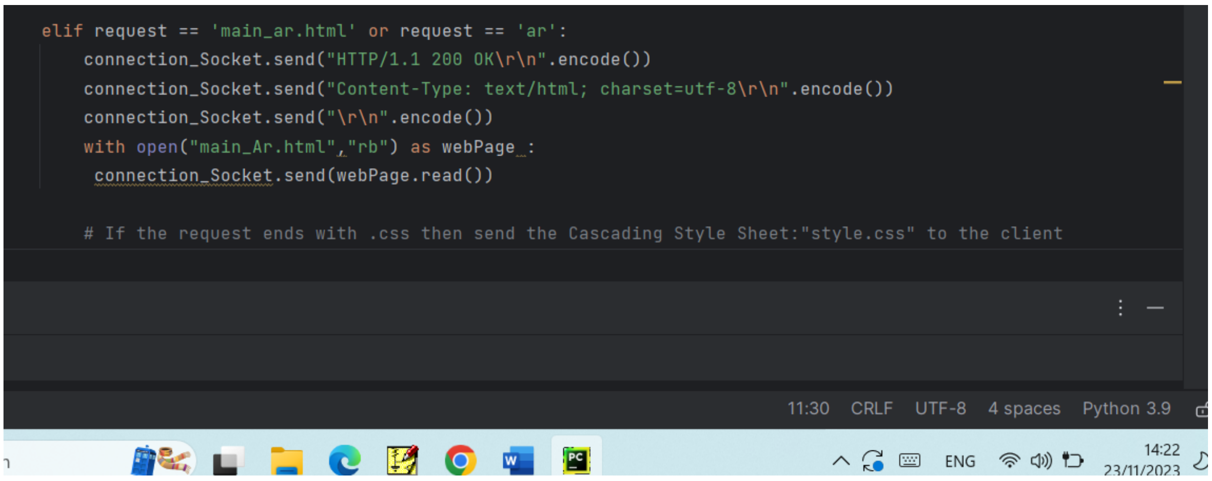
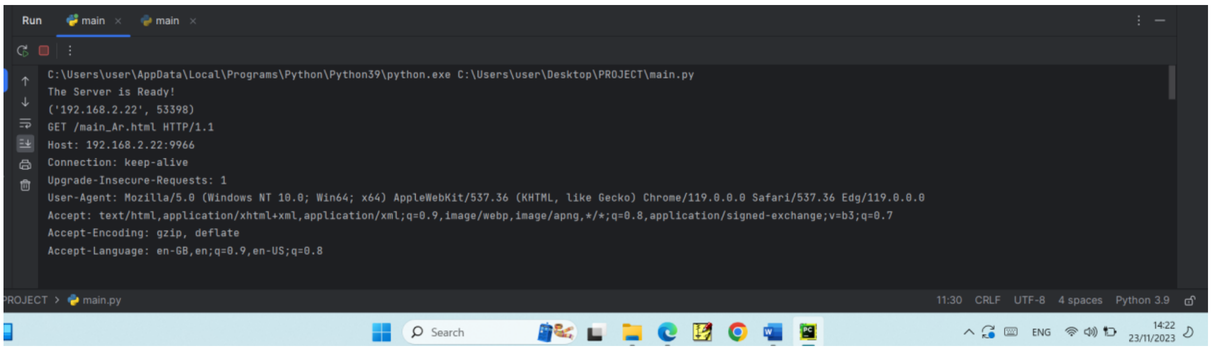

```
elif request == 'main_ar.html' or request == 'ar':
    connection_Socket.send("HTTP/1.1 200 OK\r\n".encode())
    connection_Socket.send("Content-Type: text/html; charset=utf-8\r\n".encode())
    connection_Socket.send("\r\n".encode())
    with open("main_Ar.html","rb") as webPage :
     connection_Socket.send(webPage.read())

    # If the request ends with .css then send the Cascading Style Sheet:"style.css" to the client
```

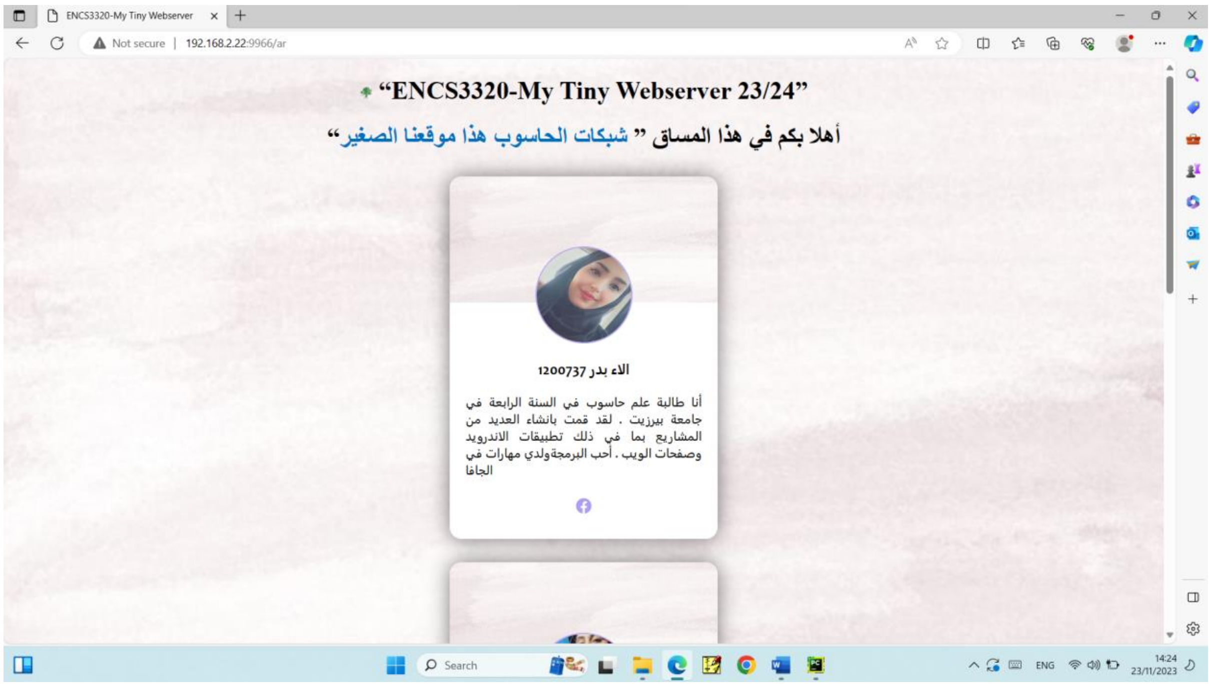
Figure 18:Request Arabic version of main_ar.html



```
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe C:\Users\user\Desktop\PROJECT\main.py
The Server is Ready!
('192.168.2.22', 53398)
GET /main_Ar.html HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
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 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 19:Response of main_ar.html

The main_ar.html:



ENCS3320-My Tiny Webserver

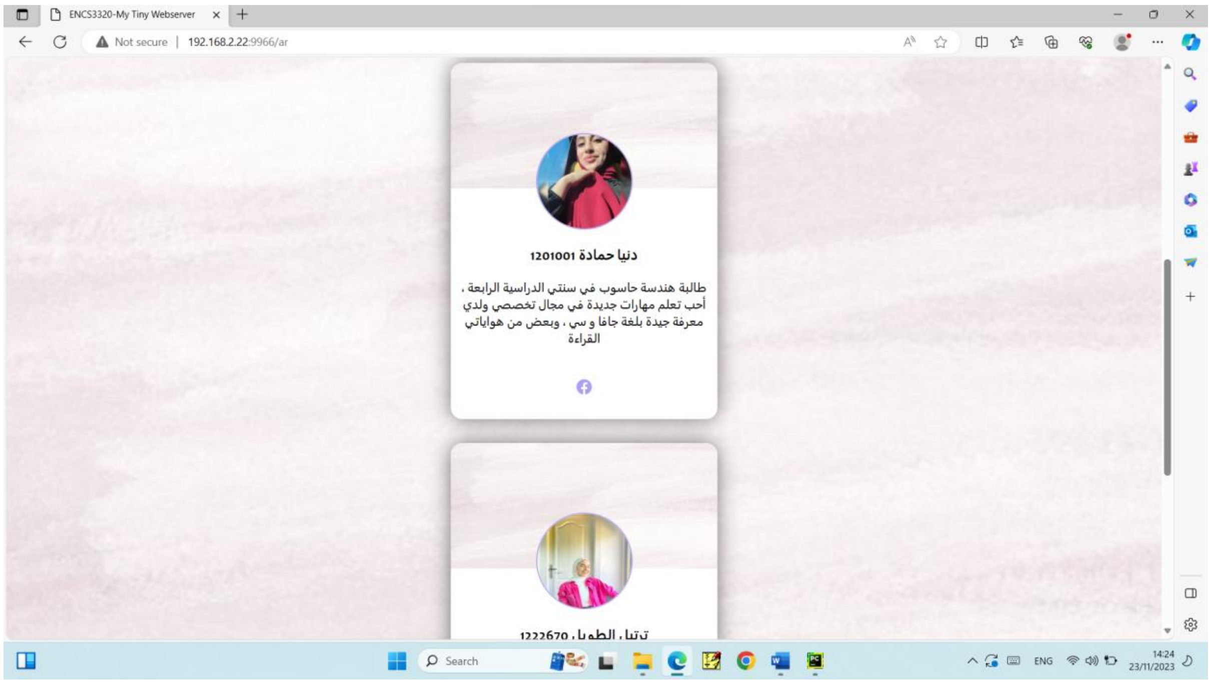
Not secure | 192.168.2.22:9966/ar

“ENCS3320-My Tiny Webserver 23/24”

أهلا بكم في هذا المساق ” شبكات الحاسوب هذا موقعنا الصغير“

الاء بدر 1200737

أنا طالبة علم حاسوب في السنة الرابعة في جامعة بيرزيت . لقد قمت بانشاء العديد من المشاريع بما في ذلك تطبيقات الاندرويد وصفحات الويب . أحب البرمجةولدي مهارات في الجافا



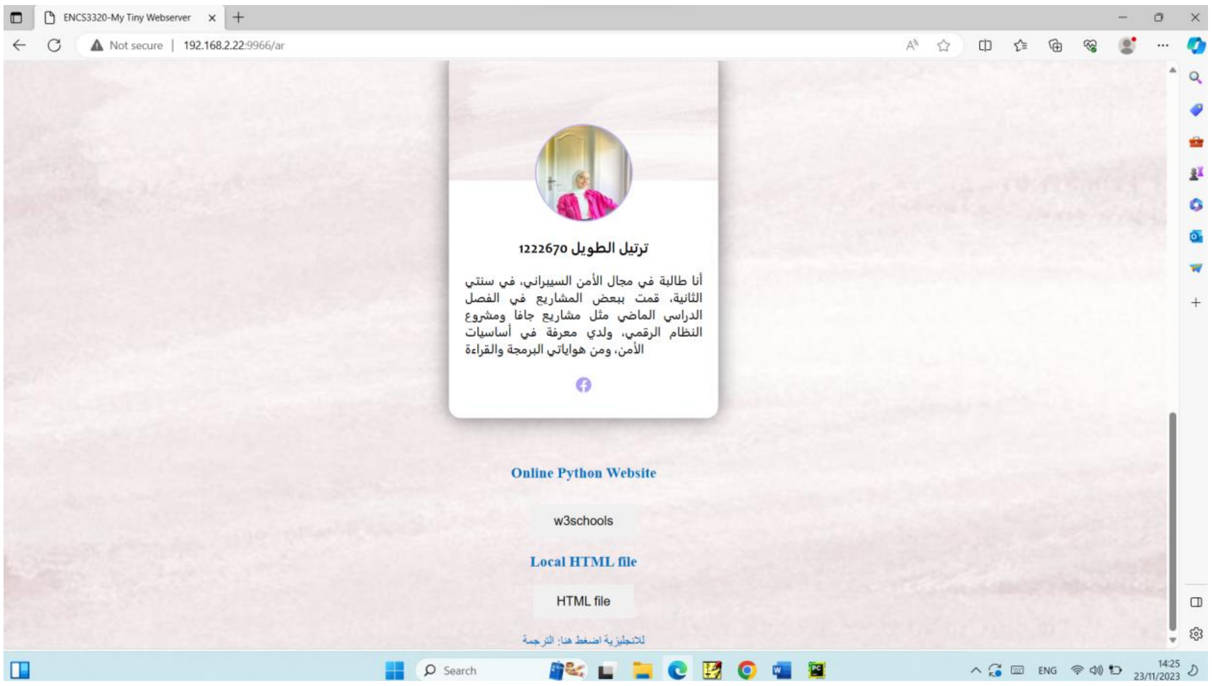
ENCS3320-My Tiny Webserver

Not secure | 192.168.2.22:9966/ar

دنيا حمادة 1201001

طالبة هندسة حاسوب في سنتي الدراسية الرابعة ، أحب تعلم مهارات جديدة في مجال تخصصي ولدي معرفة جيدة بلغة جافا و سي ، وبعض من هواياتي القراءة

[illegible] 1222670



ترتيل الطويل 1222670

أنا طالبة في مجال الأمن السيبراني، في سنتي الثانية، قمت ببعض المشاريع في الفصل الدراسي الماضي مثل مشاريع جافا ومشروع النظام الرقمي، ولدي معرفة في أساسيات الأمن، ومن هواياتي البرمجة والقراءة

Online Python Website

w3schools

Local HTML file

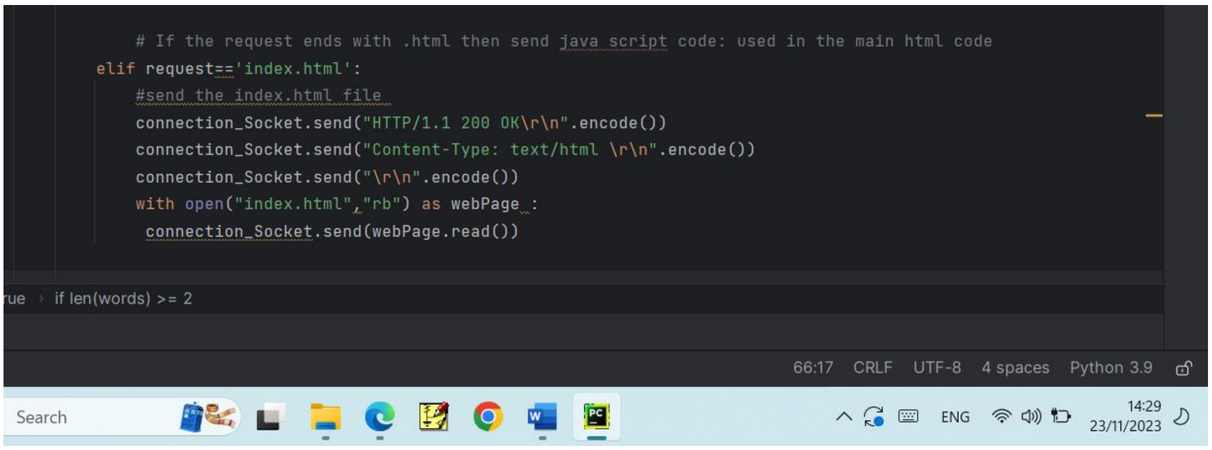
HTML file

للانجليزية اضغط هنا: الترجمة

Figure 20: main_ar.html

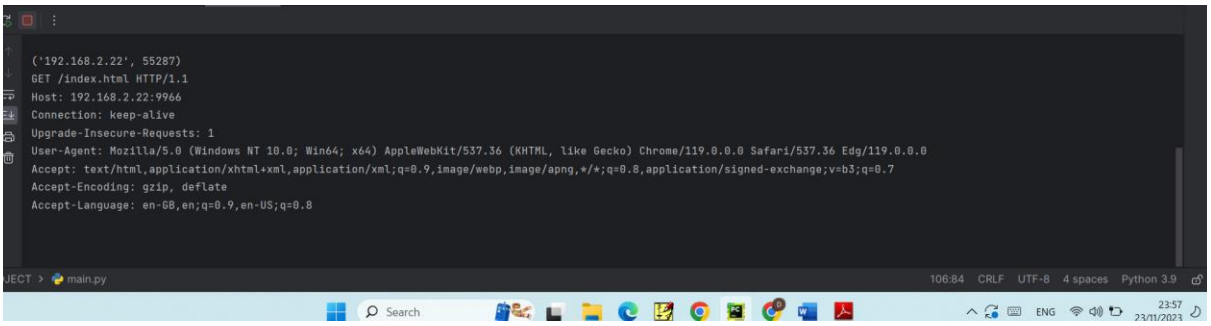
3.3. Request the html file

If the request is /html then the main html file will be sent to the client. To send the html file, theserver will first open that file then read it. Then it will send the header which contains of the encoded response status (which is OK), the encoded content type, and the encoded CRLF.Finally, the server will send the encoded file that it read previously.



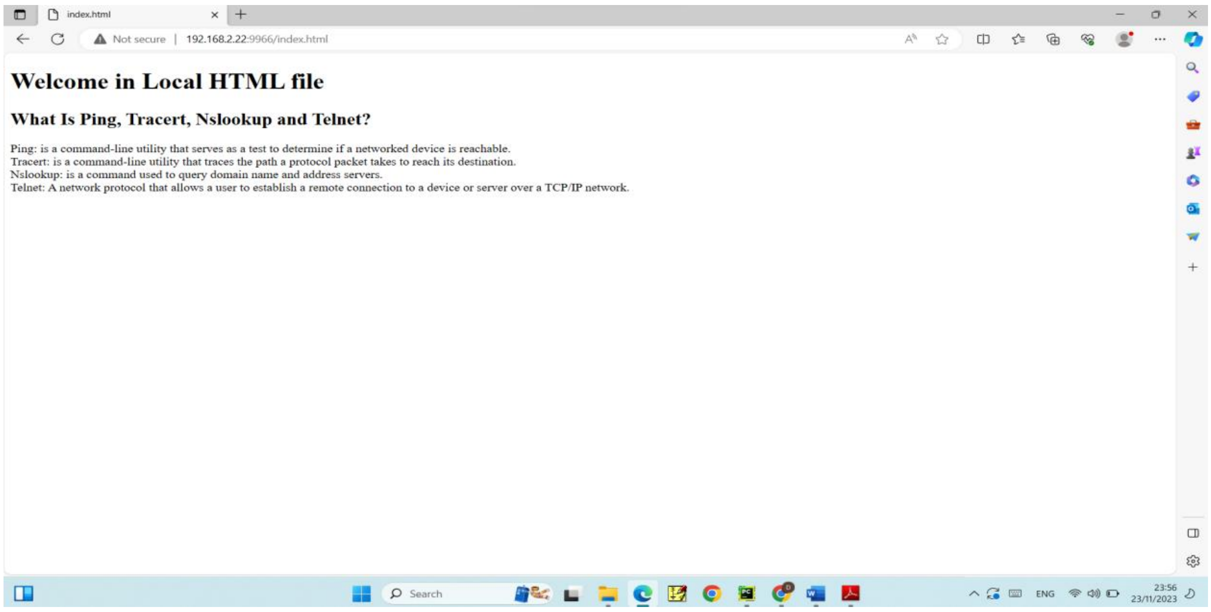
```
    # If the request ends with .html then send java script code: used in the main html code
elif request=='index.html':
    #send the index.html file
    connection_Socket.send("HTTP/1.1 200 OK\r\n".encode())
    connection_Socket.send("Content-Type: text/html \r\n".encode())
    connection_Socket.send("\r\n".encode())
    with open("index.html","rb") as webPage :
     connection_Socket.send(webPage.read())
```

Figure 21:Request .html file



```
('192.168.2.22', 55287)
GET /index.html HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
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 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 22:Response of .html file



192.168.2.22:9966/index.html

Welcome in Local HTML file

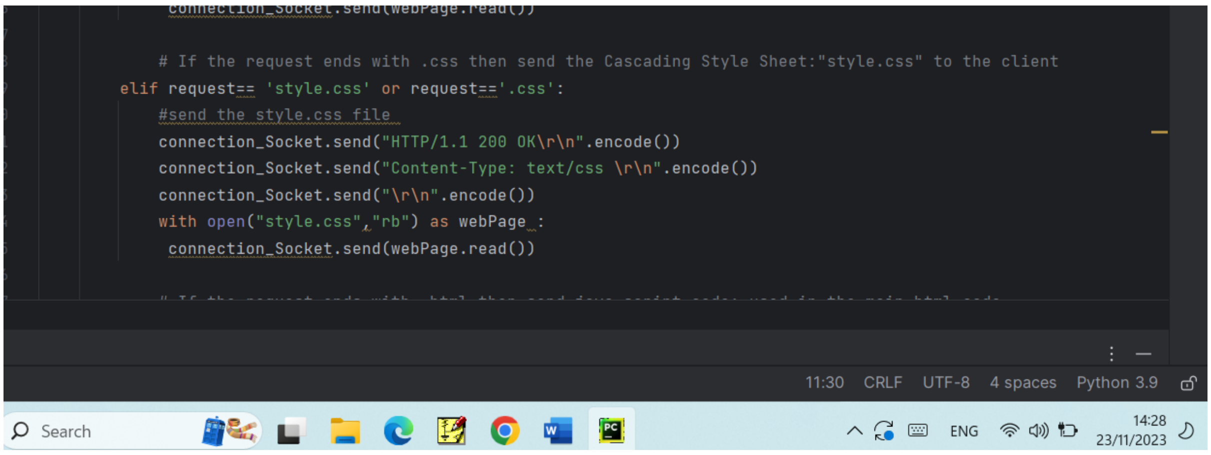
What Is Ping, Tracert, Nslookup and Telnet?

Ping: is a command-line utility that serves as a test to determine if a networked device is reachable.
Tracert: is a command-line utility that traces the path a protocol packet takes to reach its destination.
Nslookup: is a command used to query domain name and address servers.
Telnet: A network protocol that allows a user to establish a remote connection to a device or server over a TCP/IP network.

Figure 23:index.html

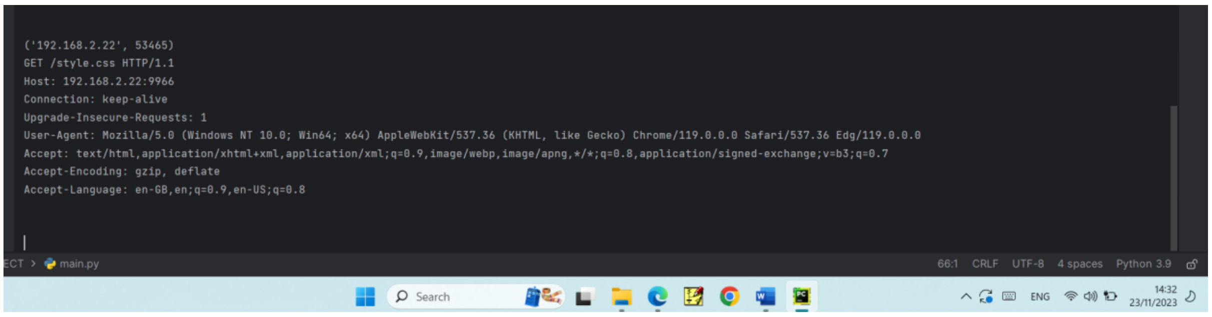
3.4. Request the CSS file

If the request is a CSS file, then the server will send the “style.css” file by doing the same as the previous steps, but changing the type of content to text/CSS.



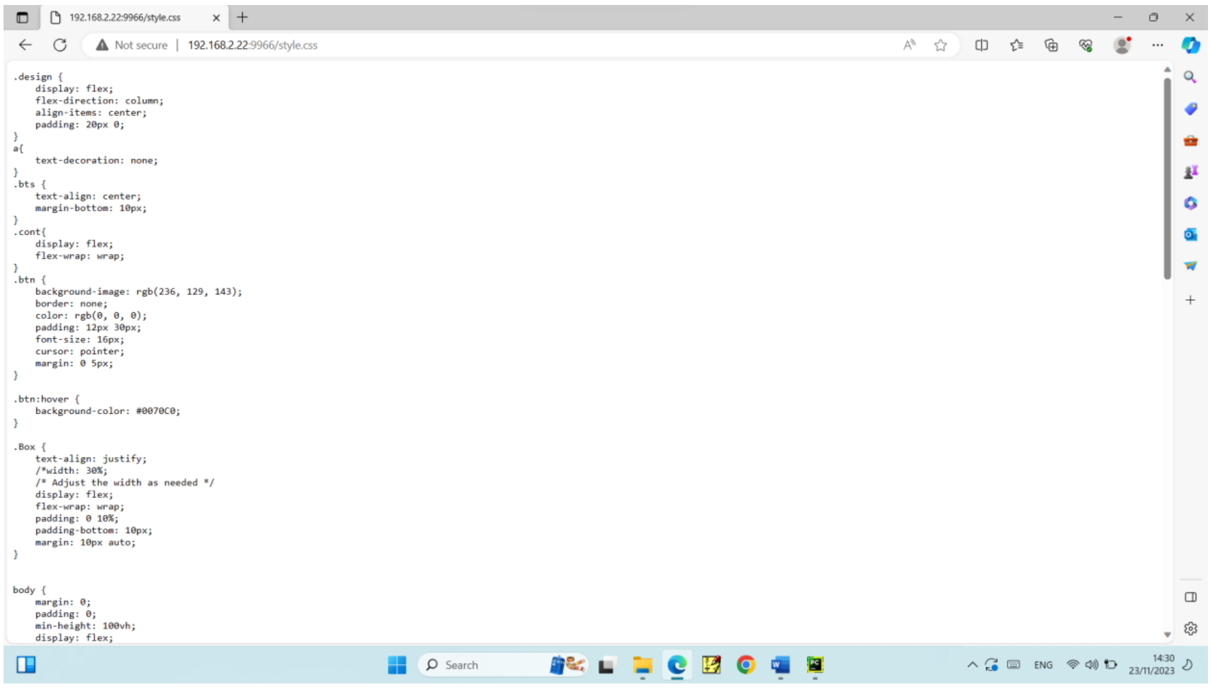
```python
            connection_Socket.send(webPage.read())

        # If the request ends with .css then send the Cascading Style Sheet:"style.css" to the client
    elif request== 'style.css' or request=='.css':
        #send the style.css file
        connection_Socket.send("HTTP/1.1 200 OK\r\n".encode())
        connection_Socket.send("Content-Type: text/css \r\n".encode())
        connection_Socket.send("\r\n".encode())
        with open("style.css","rb") as webPage :
         connection_Socket.send(webPage.read())
```

Figure 24:Request the CSS file



```
('192.168.2.22', 53465)
GET /style.css HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
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 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 25:Response of css file



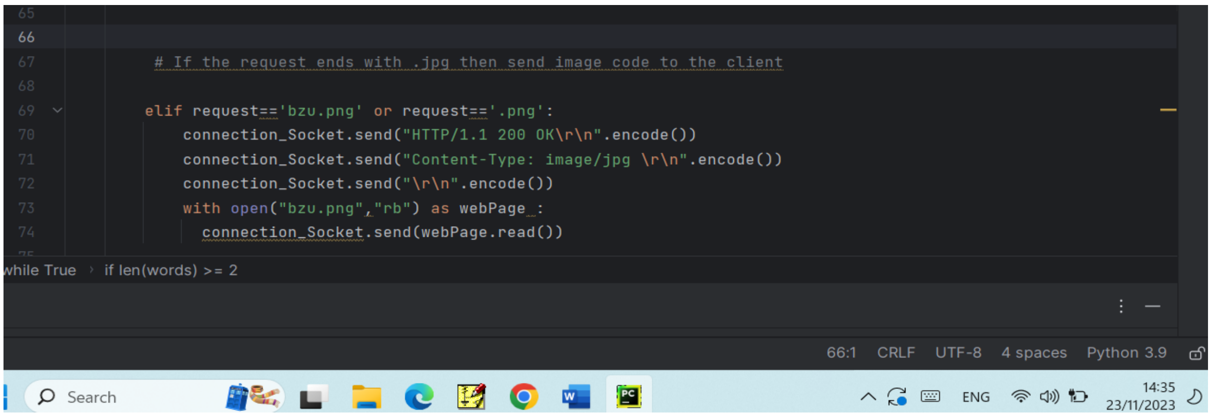
```
.design {
    display: flex;
    flex-direction: column;
    align-items: center;
    padding: 20px 0;
}
a{
    text-decoration: none;
}
.bts {
    text-align: center;
    margin-bottom: 10px;
}
.cont{
    display: flex;
    flex-wrap: wrap;
}
.btn {
    background-image: rgb(236, 129, 143);
    border: none;
    color: rgb(0, 0, 0);
    padding: 12px 30px;
    font-size: 16px;
    cursor: pointer;
    margin: 0 5px;
}

.btn:hover {
    background-color: #0070C0;
}

.Box {
    text-align: justify;
    /*width: 30%;
    /* Adjust the width as needed */
    display: flex;
    flex-wrap: wrap;
    padding: 0 10%;
    padding-bottom: 10px;
    margin: 10px auto;
}


body {
    margin: 0;
    padding: 0;
    min-height: 100vh;
    display: flex;
```

Figure 26:style.css

3.5. Request an image with JPG or PNG format

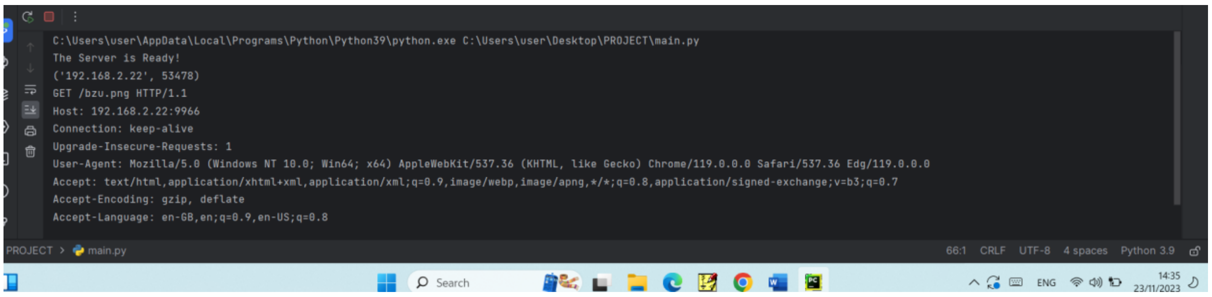
If the request is an image with JPG or PNG format, we will first see if that image is existed to send it. If not, to avoid any errors, we will send a specific image with the requested format.

request the **.png**



```


        # If the request ends with .jpg then send image code to the client

        elif request=='bzu.png' or request=='.png':
            connection_Socket.send("HTTP/1.1 200 OK\r\n".encode())
            connection_Socket.send("Content-Type: image/jpg \r\n".encode())
            connection_Socket.send("\r\n".encode())
            with open("bzu.png","rb") as webPage :
                connection_Socket.send(webPage.read())
```

Figure 27:The request is a .png



```
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe C:\Users\user\Desktop\PROJECT\main.py
The Server is Ready!
('192.168.2.22', 53478)
GET /bzu.png HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
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 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 28:Response .png

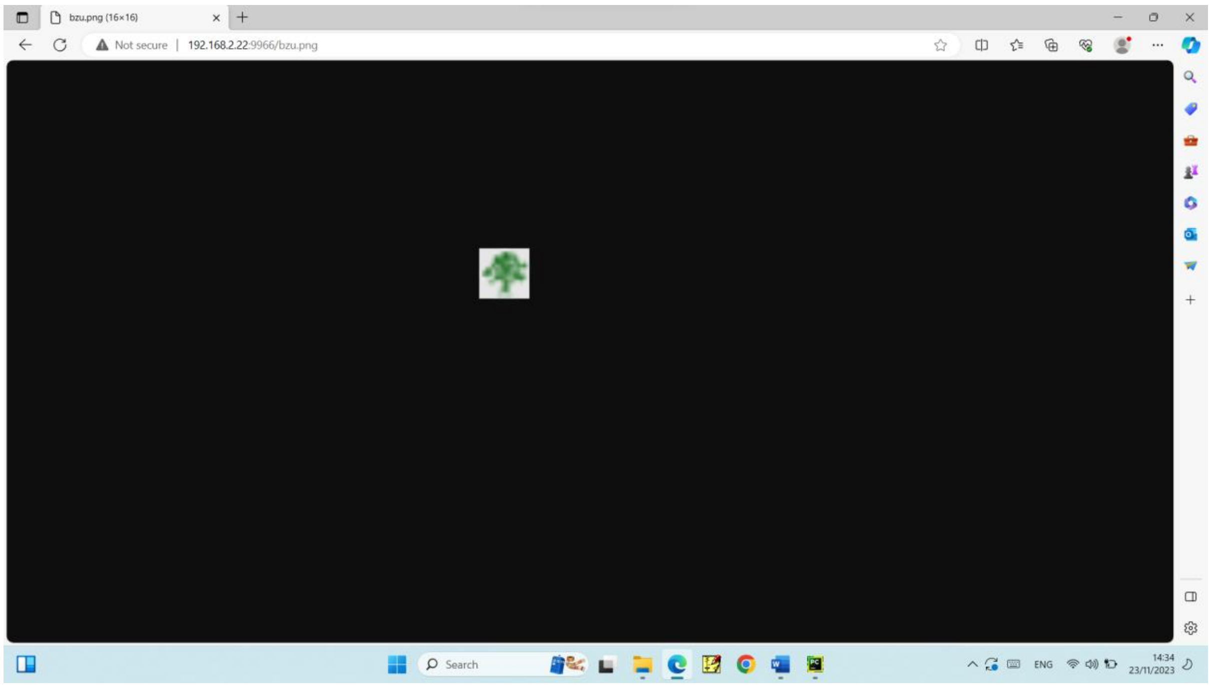
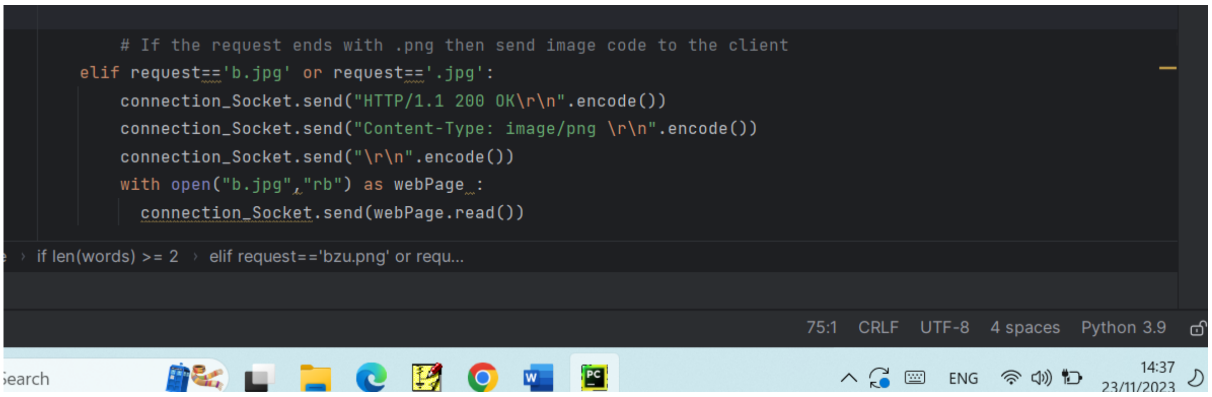


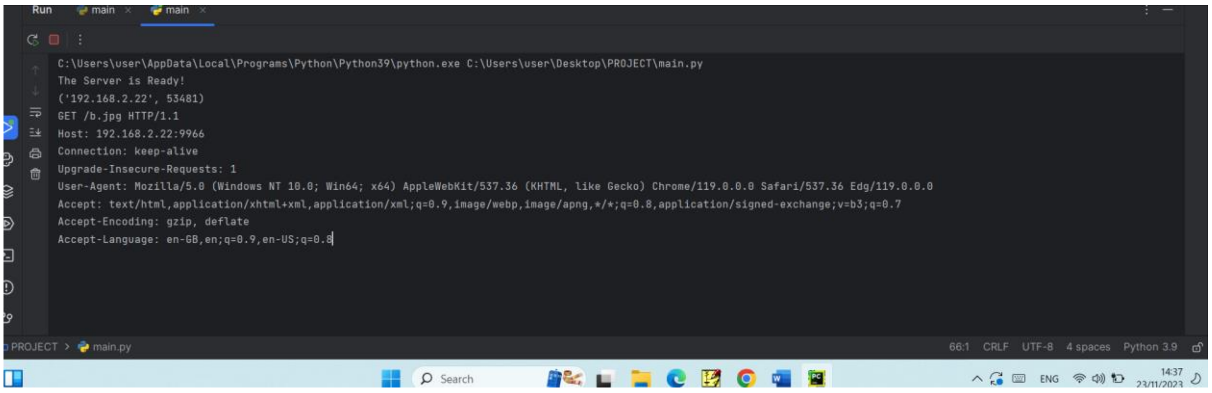
Figure 29:bzu.bng

request the **.jpg**



```
    # If the request ends with .png then send image code to the client
elif request=='b.jpg' or request=='.jpg':
    connection_Socket.send("HTTP/1.1 200 OK\r\n".encode())
    connection_Socket.send("Content-Type: image/png \r\n".encode())
    connection_Socket.send("\r\n".encode())
    with open("b.jpg","rb") as webPage :
      connection_Socket.send(webPage.read())
```

Figure 30:request the .jpg



```
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe C:\Users\user\Desktop\PROJECT\main.py
The Server is Ready!
('192.168.2.22', 53481)
GET /b.jpg HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
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 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 31:response .jpg

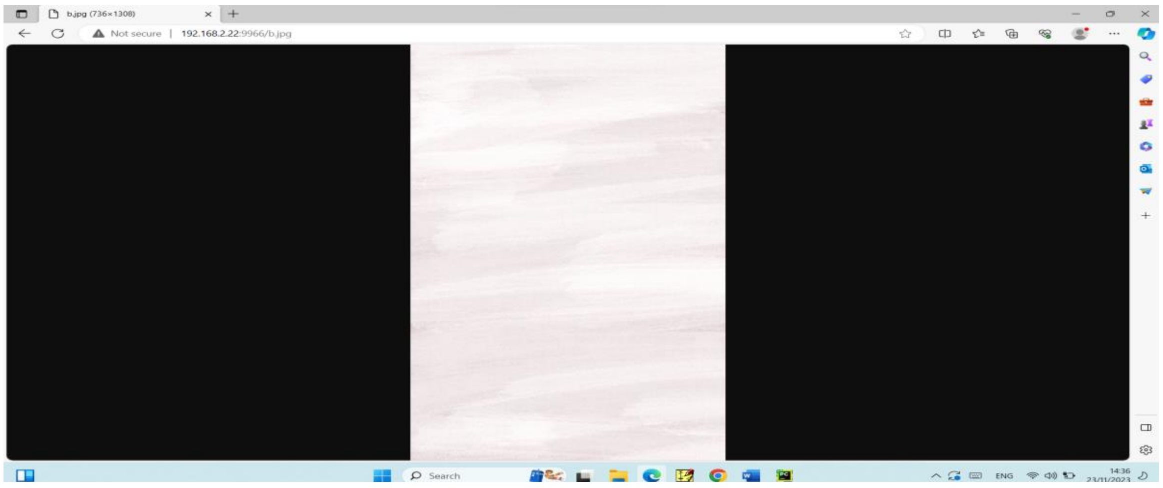
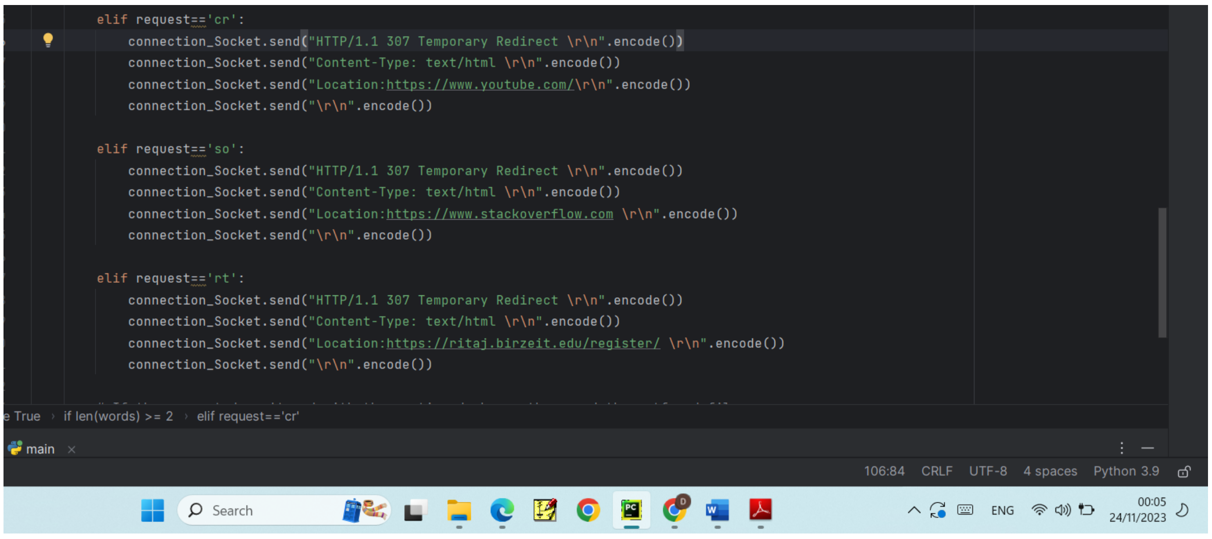


Figure 32:image.jpg

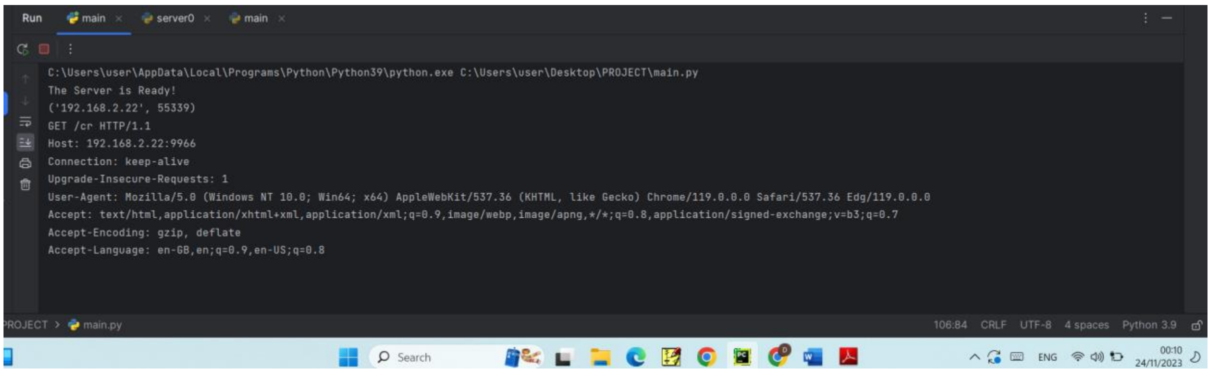
3.7. The status code 307 Temporary Redirect



```
    elif request=='cr':
        connection_Socket.send("HTTP/1.1 307 Temporary Redirect \r\n".encode())
        connection_Socket.send("Content-Type: text/html \r\n".encode())
        connection_Socket.send("Location:https://www.youtube.com/\r\n".encode())
        connection_Socket.send("\r\n".encode())

    elif request=='so':
        connection_Socket.send("HTTP/1.1 307 Temporary Redirect \r\n".encode())
        connection_Socket.send("Content-Type: text/html \r\n".encode())
        connection_Socket.send("Location:https://www.stackoverflow.com \r\n".encode())
        connection_Socket.send("\r\n".encode())

    elif request=='rt':
        connection_Socket.send("HTTP/1.1 307 Temporary Redirect \r\n".encode())
        connection_Socket.send("Content-Type: text/html \r\n".encode())
        connection_Socket.send("Location:https://ritaj.birzeit.edu/register/ \r\n".encode())
        connection_Socket.send("\r\n".encode())
```

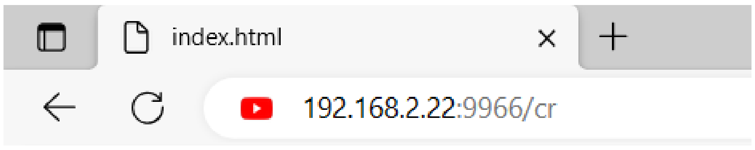
Figure 33:307 Temporary Redirect

A. If the request is /cr then redirect to cornell.edu website

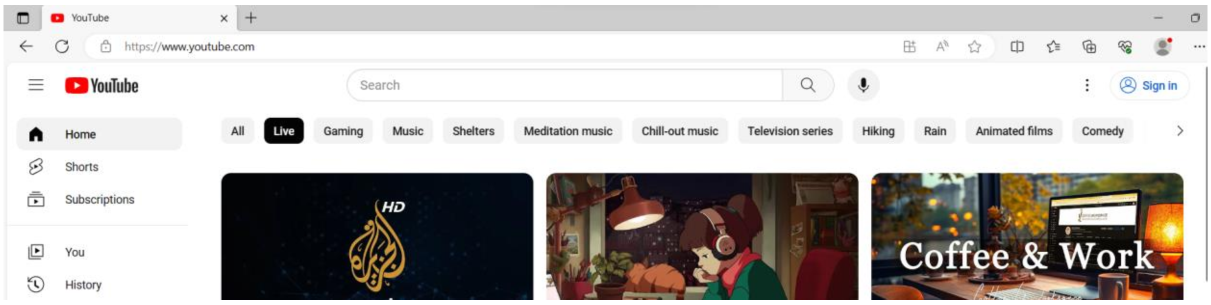


```
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe C:\Users\user\Desktop\PROJECT\main.py
The Server is Ready!
('192.168.2.22', 55339)
GET /cr HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

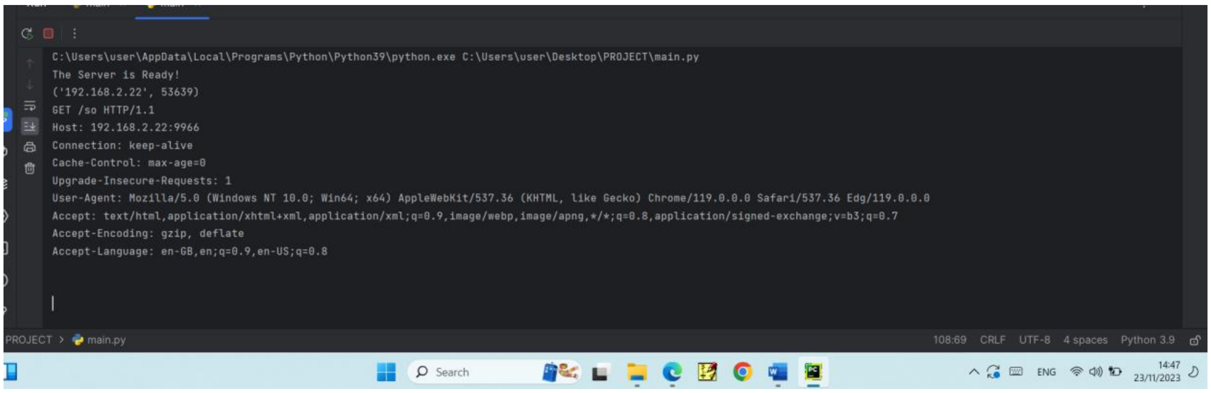
Figure 34:response /cr



Directly transmitted to you tube after sending.

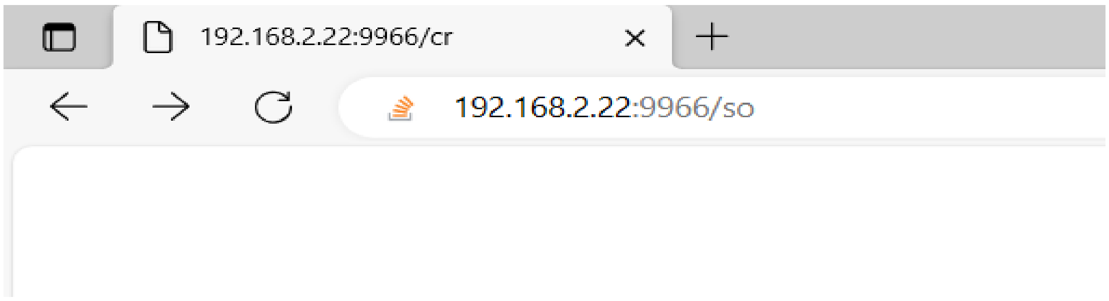


B.If the request is /so then redirect to stackoverflow.com website

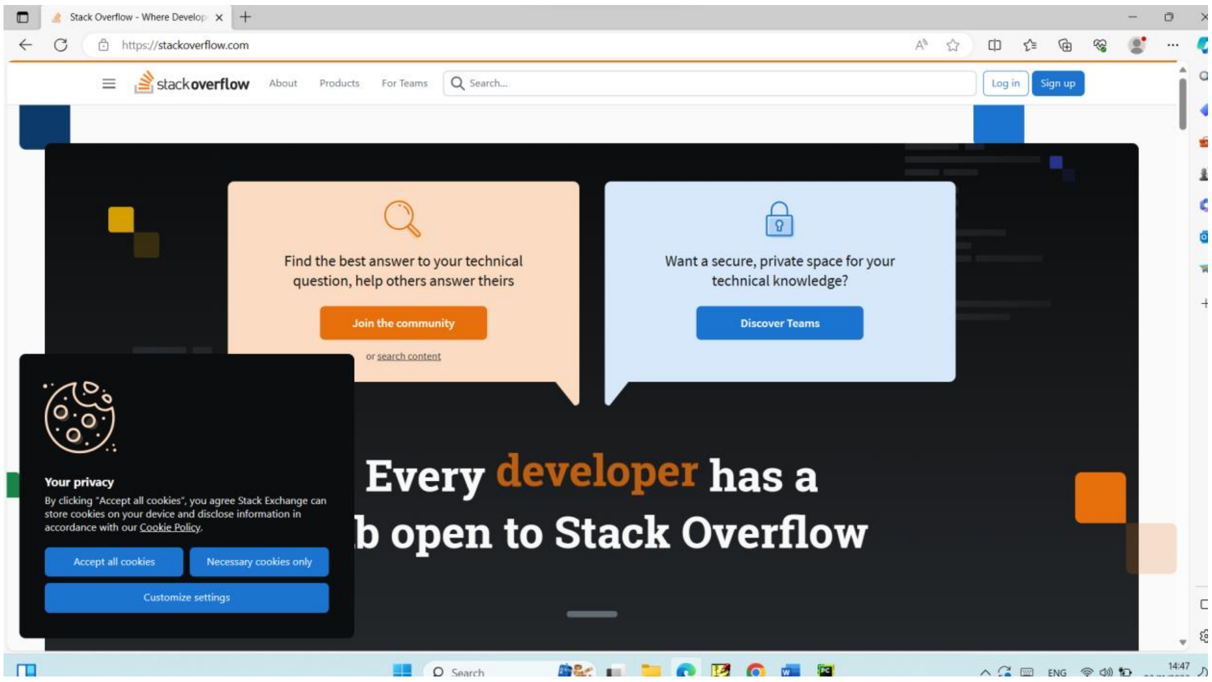


```
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe C:\Users\user\Desktop\PROJECT\main.py
The Server is Ready!
('192.168.2.22', 53639)
GET /so HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 35:response /so



Directly transmitted to stack overflow after sending



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

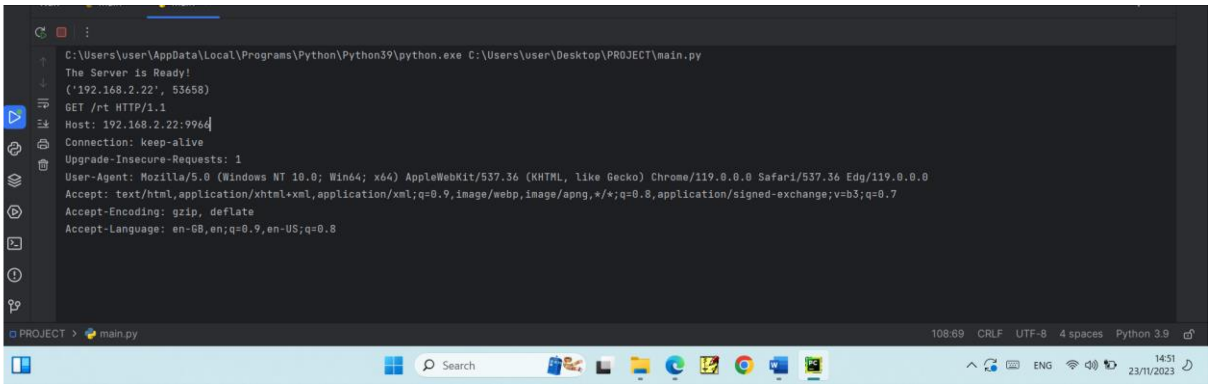
Want a secure, private space for your technical knowledge?

Your privacy

By clicking "Accept all cookies", you agree Stack Exchange can store cookies on your device and disclose information in accordance with our Cookie Policy.

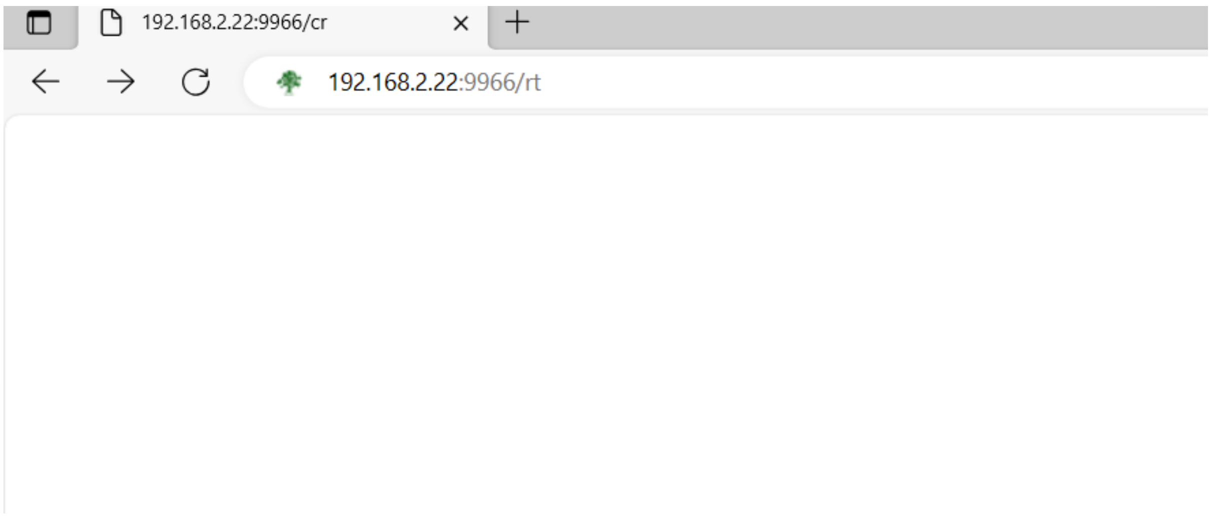
Every developer has a b open to Stack Overflow

C. If the request is /rt then redirect to ritaj website

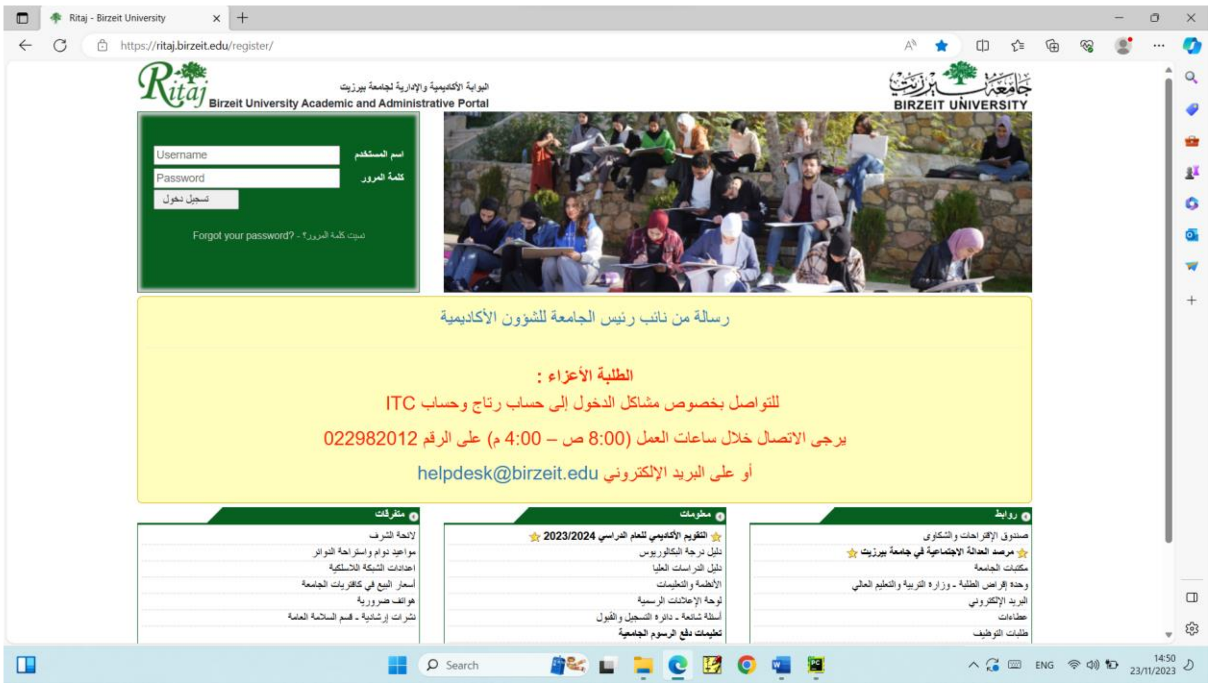


```
C:\Users\user\AppData\Local\Programs\Python\Python39\python.exe C:\Users\user\Desktop\PROJECT\main.py
The Server is Ready!
('192.168.2.22', 53658)
GET /rt HTTP/1.1
Host: 192.168.2.22:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36 Edg/119.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en;q=0.9,en-US;q=0.8
```

Figure 36:response /rt



Directly transmitted to Ritag website after sending


Ritaj - Birzeit University
https://ritaj.birzeit.edu/register/
Search
ENG
14:50
23/11/2023

البوابة الأكاديمية والإدارية لجامعة بيرزيت

Birzeit University Academic and Administrative Portal

BIRZEIT UNIVERSITY

اسم المستخدم	Username
كلمة المرور	Password

تسجيل دخول

Forgot your password? - نسيت كلمة المرور؟

رسالة من نائب رئيس الجامعة للشؤون الأكاديمية

الطلبة الأعزاء :

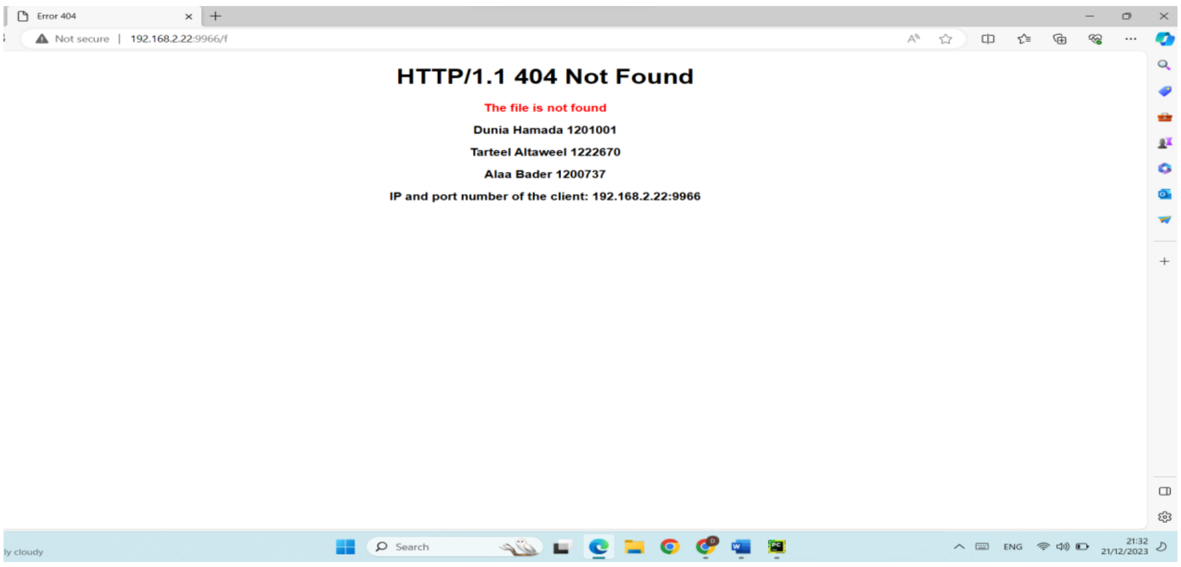
للتواصل بخصوص مشاكل الدخول إلى حساب رتاج وحساب ITC

يرجى الاتصال خلال ساعات العمل (8:00 ص – 4:00 م) على الرقم 022982012

أو على البريد الإلكتروني helpdesk@birzeit.edu

روابط	معلومات	متفرقات
صندوق الإقتراحات والشكاوى	التقويم الأكاديمي للعام الدراسي 2023/2024	لائحة الشرف
مرصد العدالة الاجتماعية في جامعة بيرزيت	دليل درجة البكالوريوس	مواعيد دوام واستراحة الدوائر
مكتبات الجامعة	دليل الدراسات العليا	اعدادات الشبكة اللاسلكية
وحدة إقراض الطلبة - وزارة التربية والتعليم العالي	الأنظمة والتعليمات	أسعار البيع في كافتريات الجامعة
البريد الإلكتروني	لوحة الإعلانات الرسمية	هواتف ضرورية
عطاءات	أسئلة شائعة - دائرة التسجيل والقبول	نشرات إرشادية - قسم السلامة العامة
طلبات التوظيف	تعليمات دفع الرسوم الجامعية	

3.8. Wrong request


Error 404

Not secure | 192.168.2.22:9966/f

HTTP/1.1 404 Not Found

The file is not found

Dunia Hamada 1201001

Tarteel Altaweel 1222670

Alaa Bader 1200737

IP and port number of the client: 192.168.2.22:9966

Figure 37:Wrong request

3.9. Screenshots of working with another device:

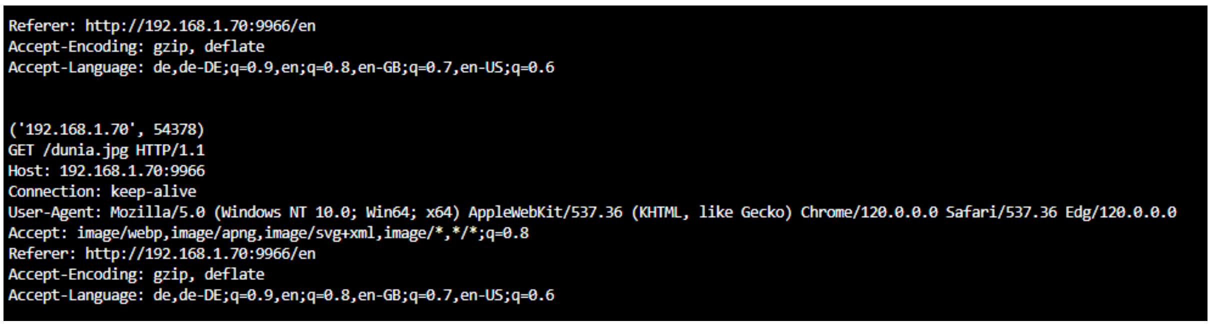
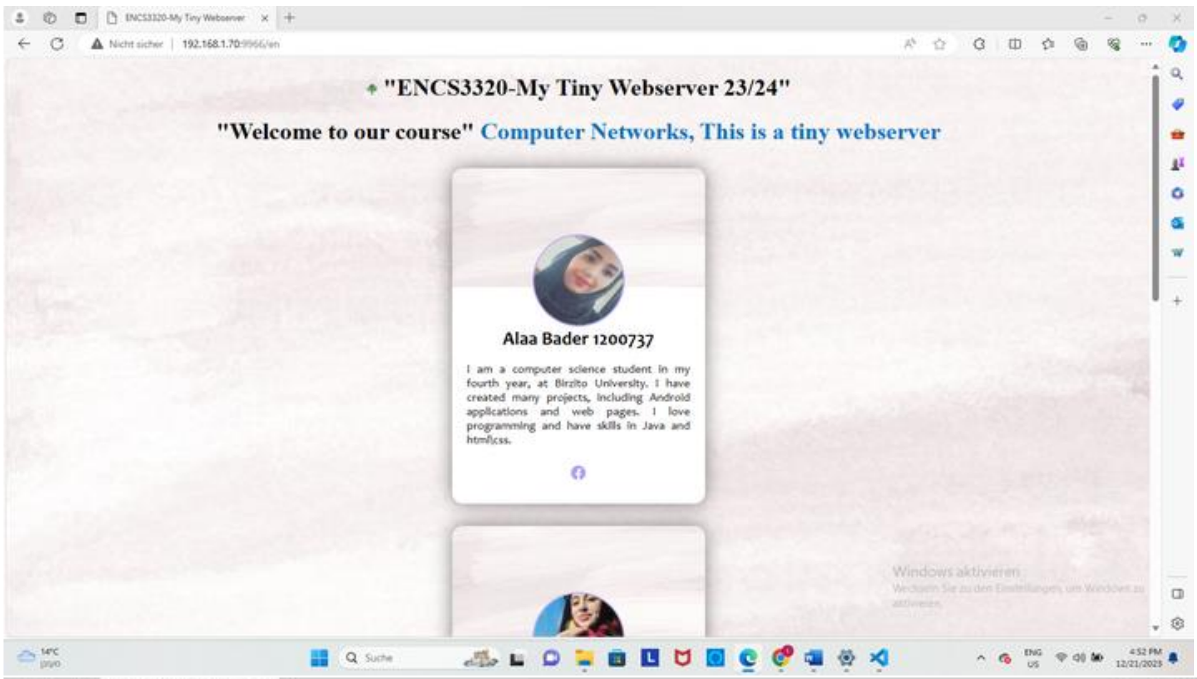

```
Referer: http://192.168.1.70:9966/en
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6


('192.168.1.70', 54378)
GET /dunia.jpg HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Referer: http://192.168.1.70:9966/en
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

Figure 37: Response of local html file



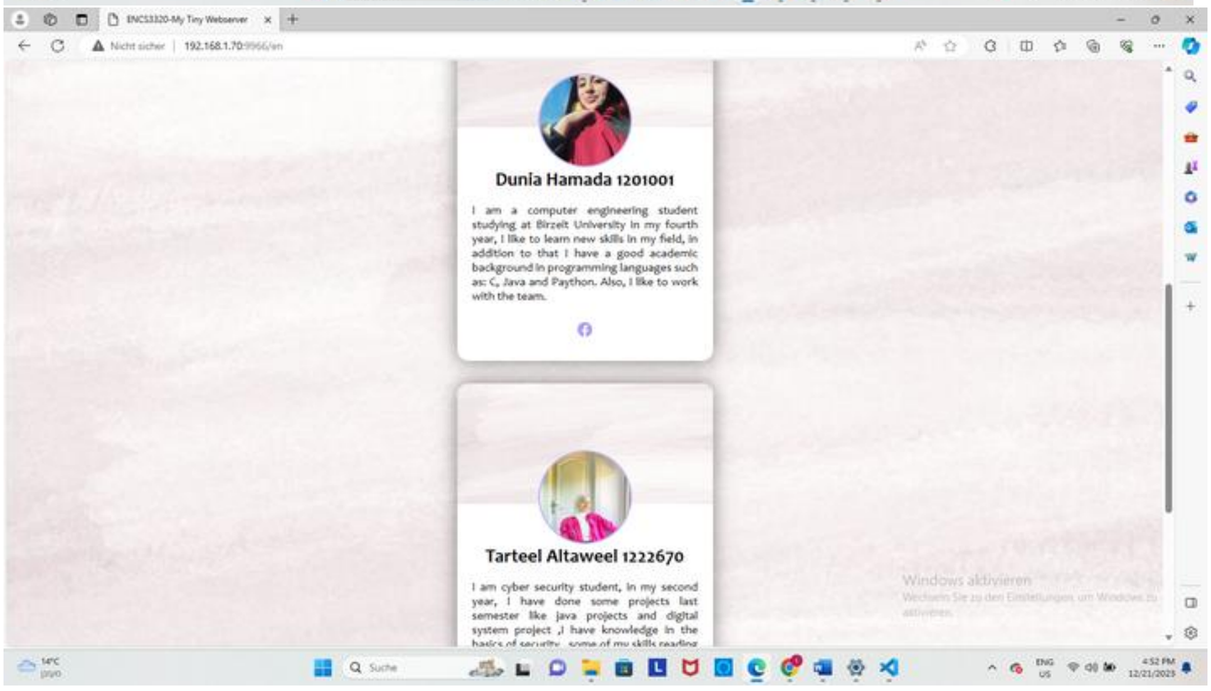
"ENCS3320-My Tiny Webserver 23/24"

"Welcome to our course" Computer Networks, This is a tiny webserver

Alaa Bader 1200737

I am a computer science student in my fourth year, at Birzito University. I have created many projects, including Android applications and web pages. I love programming and have skills in Java and html\css.

ENCS3320-My Tiny Webserver
Nicht sicher | 192.168.1.70:9966/en
Windows aktivieren
Wechseln Sie zu den Einstellungen, um Windows zu aktivieren.
Suche
4:52 PM
12/21/2023



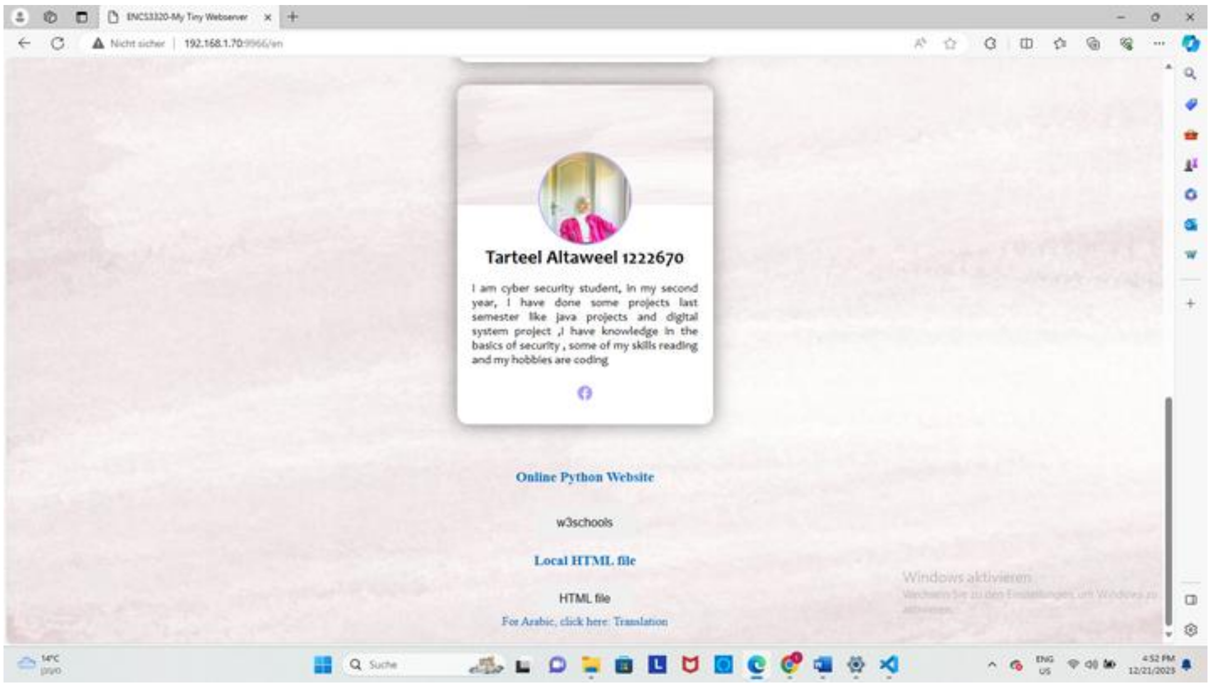
Dunia Hamada 1201001

I am a computer engineering student studying at Birzeit University in my fourth year, I like to learn new skills in my field, in addition to that I have a good academic background in programming languages such as: C, Java and Paython. Also, I like to work with the team.

Tarteel Altaweel 1222670

I am cyber security student, in my second year, I have done some projects last semester like java projects and digital system project ,I have knowledge in the basics of security , some of my skills reading

ENCS3320-My Tiny Webserver
Nicht sicher | 192.168.1.70:9966/en
Windows aktivieren
Wechseln Sie zu den Einstellungen, um Windows zu aktivieren.
Suche
4:52 PM
12/21/2023



Tarteel Altaweel 1222670

I am cyber security student, in my second year, I have done some projects last semester like java projects and digital system project ,I have knowledge in the basics of security , some of my skills reading and my hobbies are coding

Online Python Website

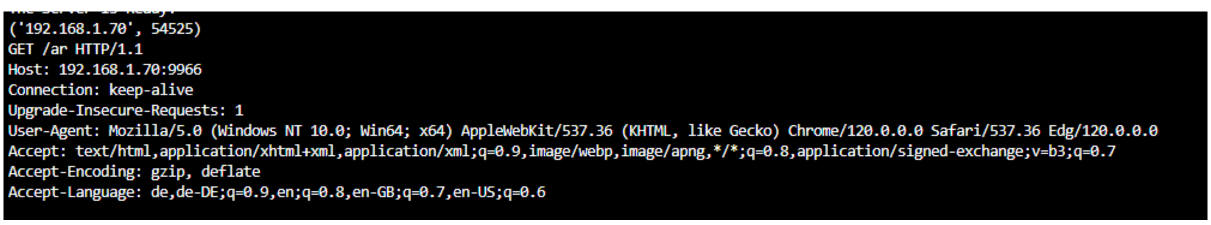
w3schools

Local HTML file

HTML file

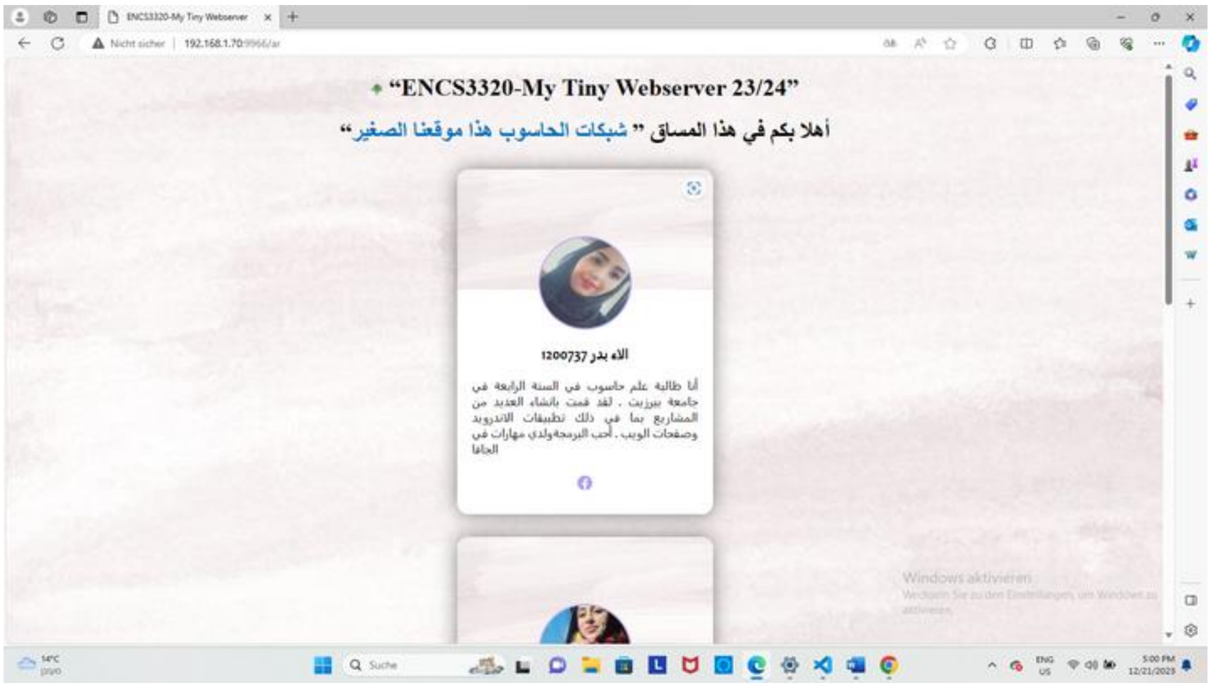
For Arabic, click here: Translation

Figure 38: main_en.html



```
('192.168.1.70', 54525)
GET /ar HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

Figure 38:Response of main_ar.html



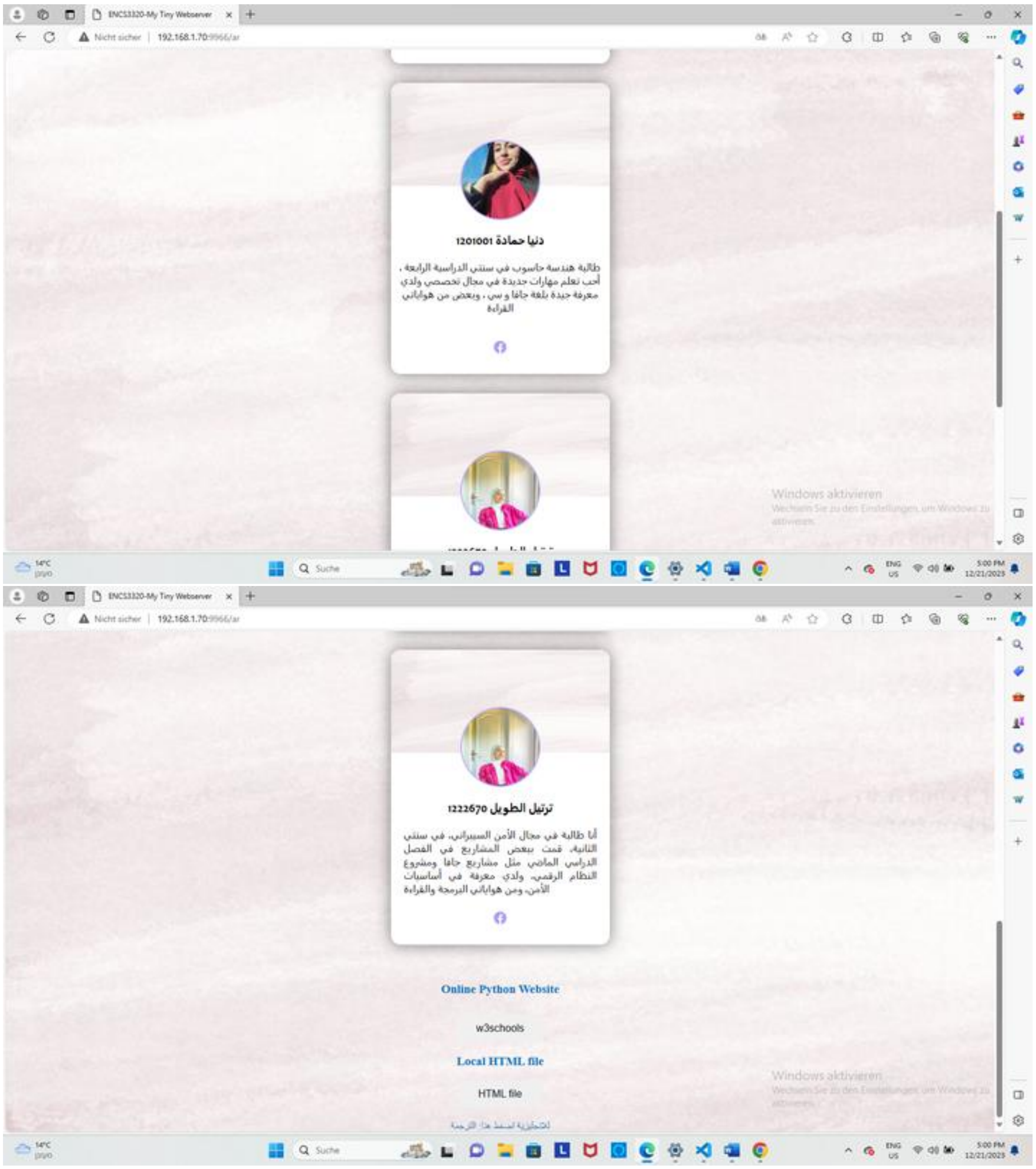
“ENCS3320-My Tiny Webserver 23/24”

أهلا بكم في هذا المساق " شبكات الحاسوب هذا موقعنا الصغير"

الاء بدر 1200737

أنا طالبة علم حاسوب في السنة الرابعة في جامعة بيرزيت . لقد قمت بانشاء العديد من المشاريع بما في ذلك تطبيقات الاندرويد وصفحات الويب . أحب البرمجة ولدي مهارات في الجافا

a



دنيا حمادة 1201001

طالبة هندسة حاسوب في سنتي الدراسية الرابعة ، أحب تعلم مهارات جديدة في مجال تخصصي ولدي معرفة جيدة بلغة جافا و سي ، وبعض من هواياتي القراءة

ترتيل الطويل 1222670

أنا طالبة في مجال الأمن السيبراني، في سنتي الثانية. قمت ببعض المشاريع في الفصل الدراسي الماضي مثل مشاريع جافا ومشروع النظام الرقمي، ولدي معرفة في أساسيات الأمن، ومن هواياتي البرمجة والقراءة

Online Python Website

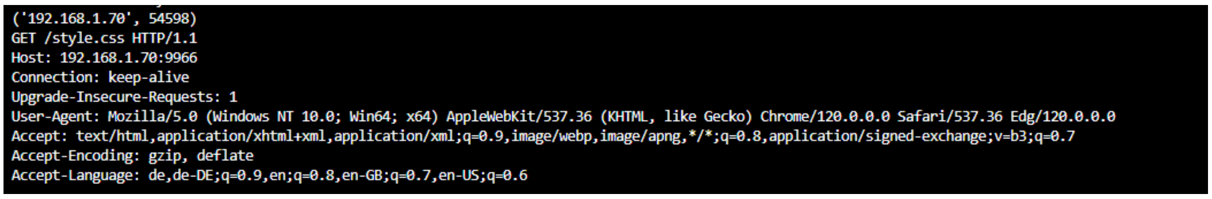
w3schools

Local HTML file

HTML file

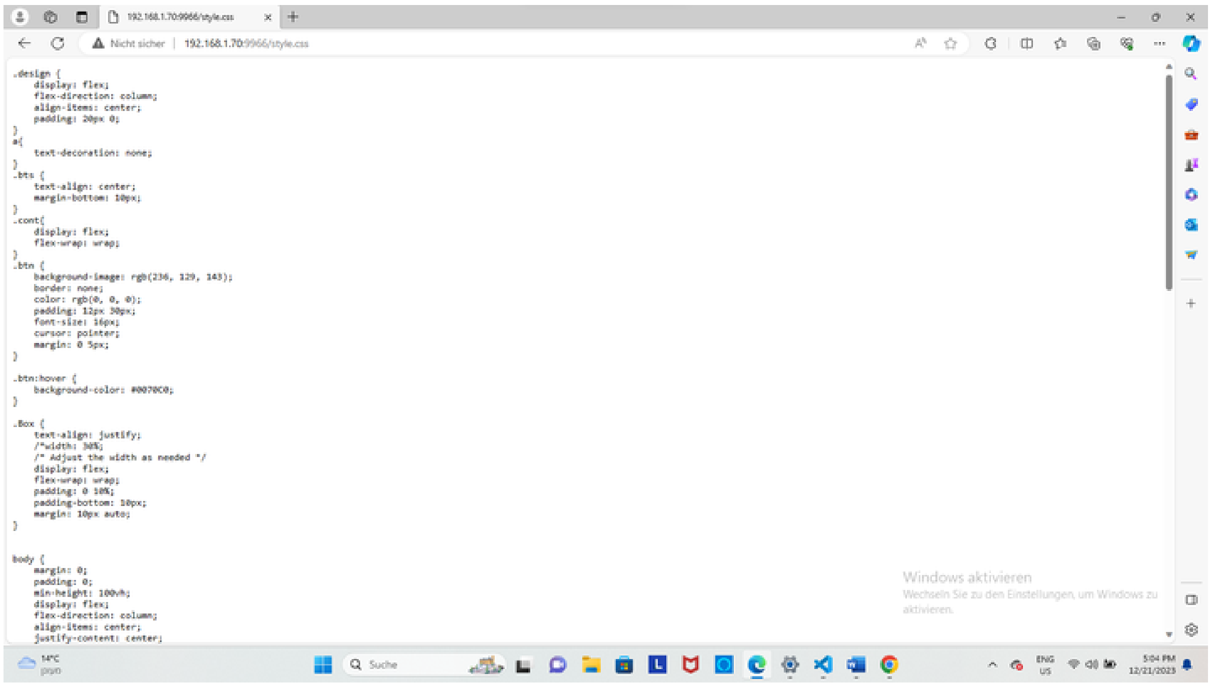
للانجليزية اضغط هنا الترجمة

Figure 39:main_ar.html



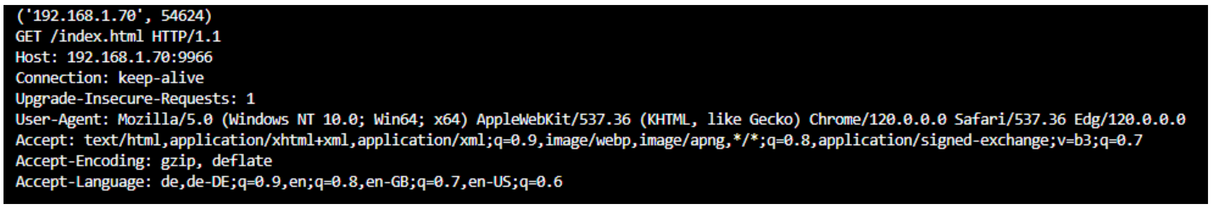
```
('192.168.1.70', 54598)
GET /style.css HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

Figure 40:Response of css file



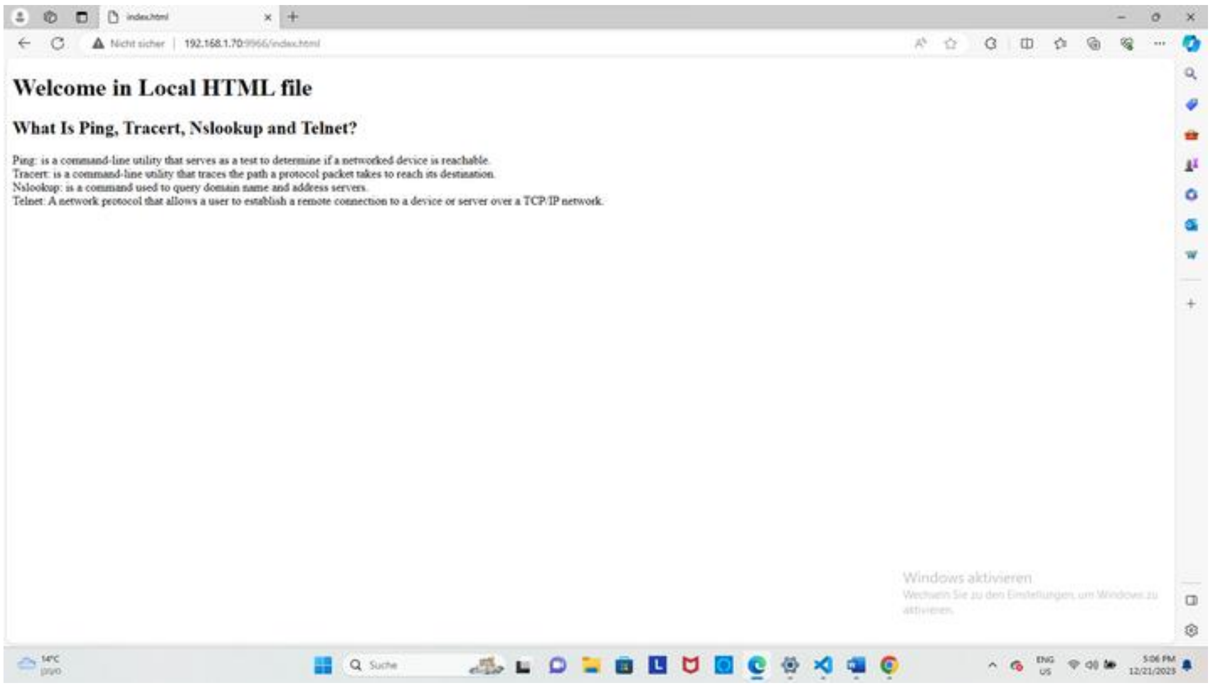
```
.design {
    display: flex;
    flex-direction: column;
    align-items: center;
    padding: 20px 0;
}
a{
    text-decoration: none;
}
.bts {
    text-align: center;
    margin-bottom: 10px;
}
.cont{
    display: flex;
    flex-wrap: wrap;
}
.btn {
    background-image: rgb(236, 129, 143);
    border: none;
    color: rgb(0, 0, 0);
    padding: 12px 30px;
    font-size: 16px;
    cursor: pointer;
    margin: 0 5px;
}

.btn:hover {
    background-color: #0070C0;
}

.Box {
    text-align: justify;
    /*width: 30%;
    /* Adjust the width as needed */
    display: flex;
    flex-wrap: wrap;
    padding: 0 10%;
    padding-bottom: 10px;
    margin: 10px auto;
}


body {
    margin: 0;
    padding: 0;
    min-height: 100vh;
    display: flex;
    flex-direction: column;
    align-items: center;
    justify-content: center;
```

Figure 41:style.css



```
('192.168.1.70', 54624)
GET /index.html HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

Figure 42:Response of .html file

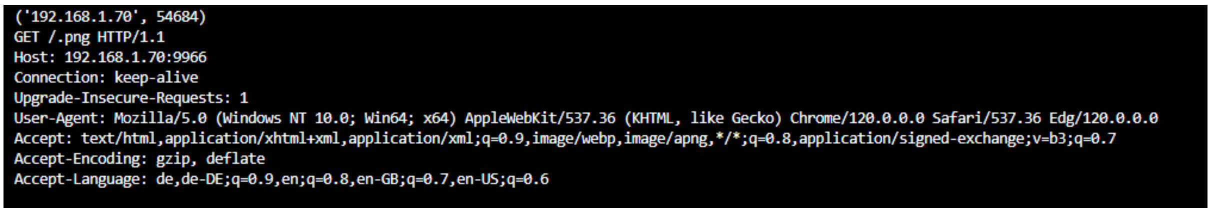


Welcome in Local HTML file

What Is Ping, Tracert, Nslookup and Telnet?

Ping: is a command-line utility that serves as a test to determine if a networked device is reachable.
Tracert: is a command-line utility that traces the path a protocol packet takes to reach its destination.
Nslookup: is a command used to query domain name and address servers.
Telnet: A network protocol that allows a user to establish a remote connection to a device or server over a TCP/IP network.

Figure 43:index.html



```
('192.168.1.70', 54684)
GET /.png HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

Figure 44:Response .png

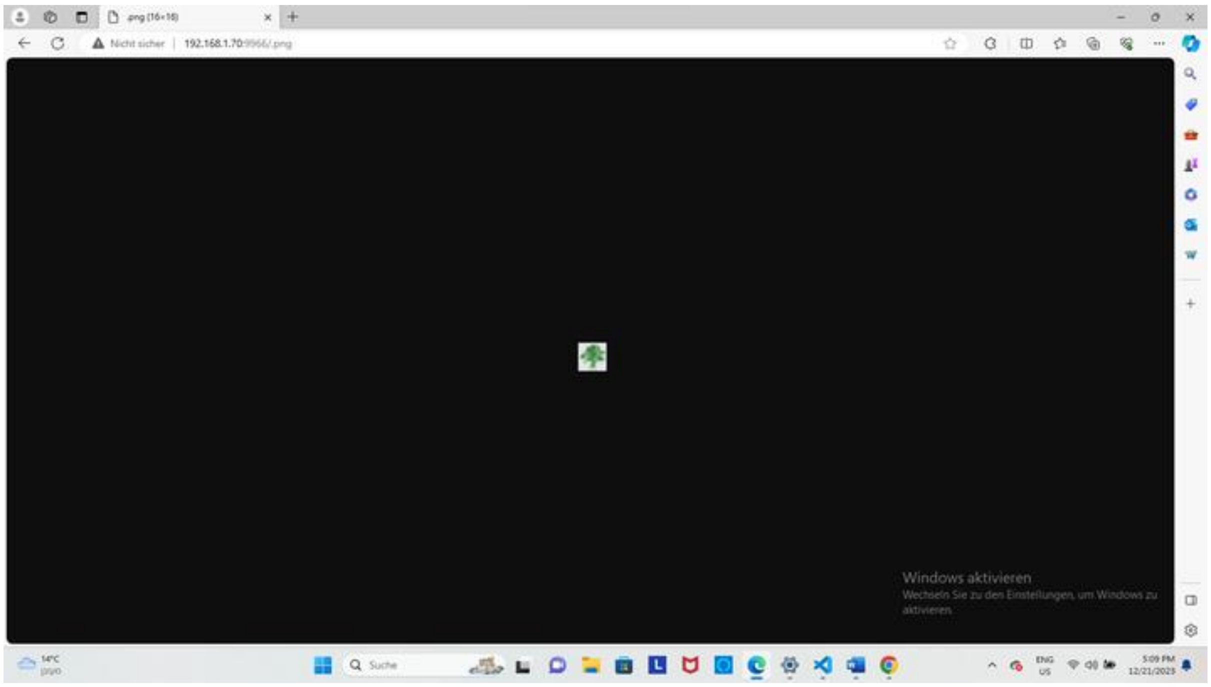
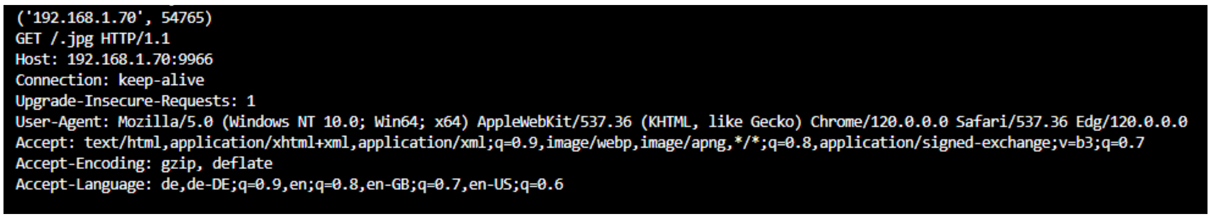


Figure 45:bzu .png



```
('192.168.1.70', 54765)
GET /.jpg HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

Figure 46: response .jpg

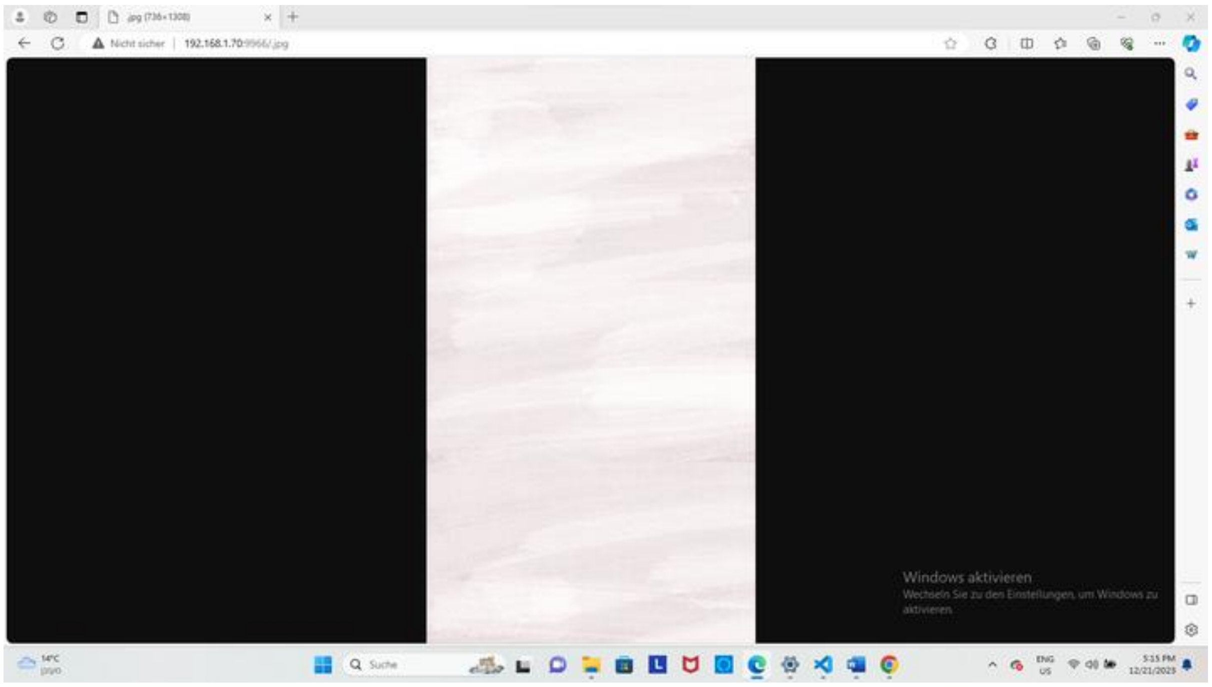
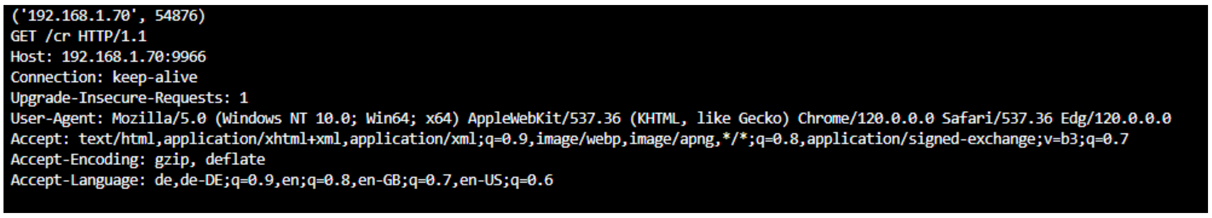
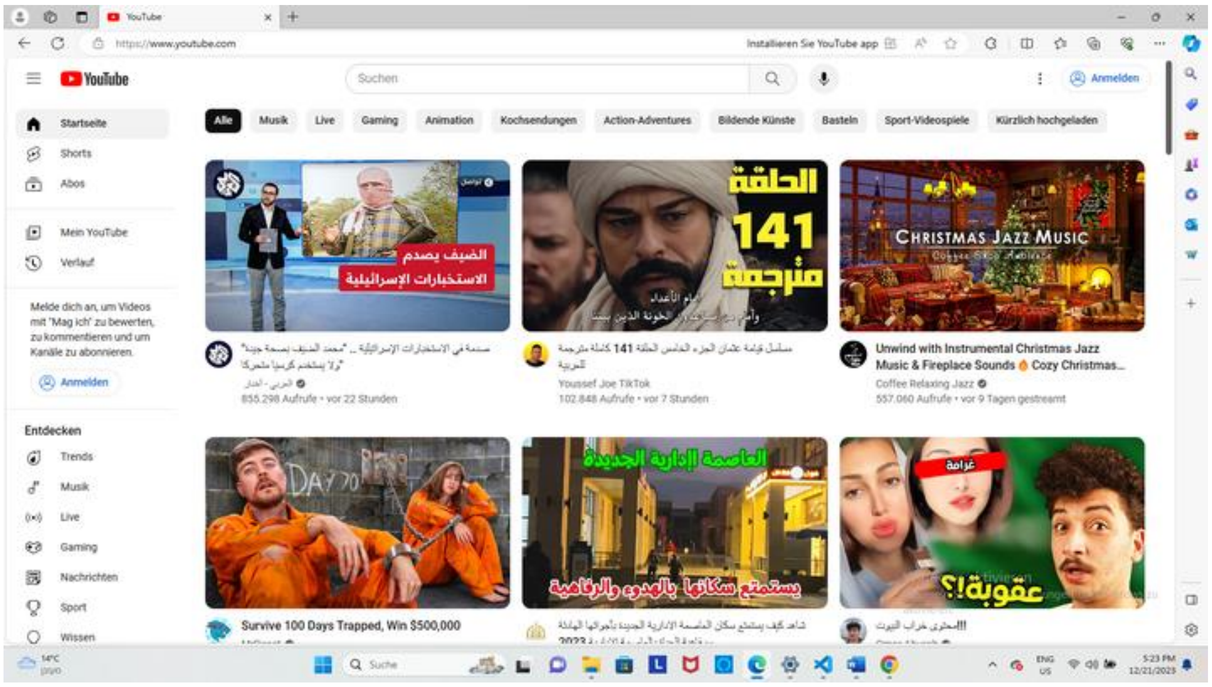

.jpg (736×1308)
Nicht sicher | 192.168.1.70:9966/.jpg
Windows aktivieren
Wechseln Sie zu den Einstellungen, um Windows zu aktivieren.

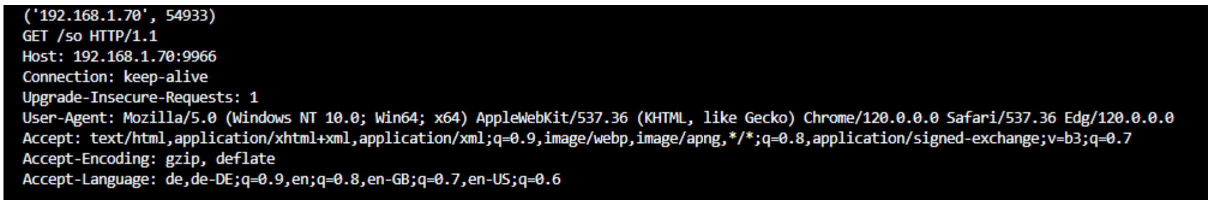
Figure 47: image.jpg



```
('192.168.1.70', 54876)
GET /cr HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

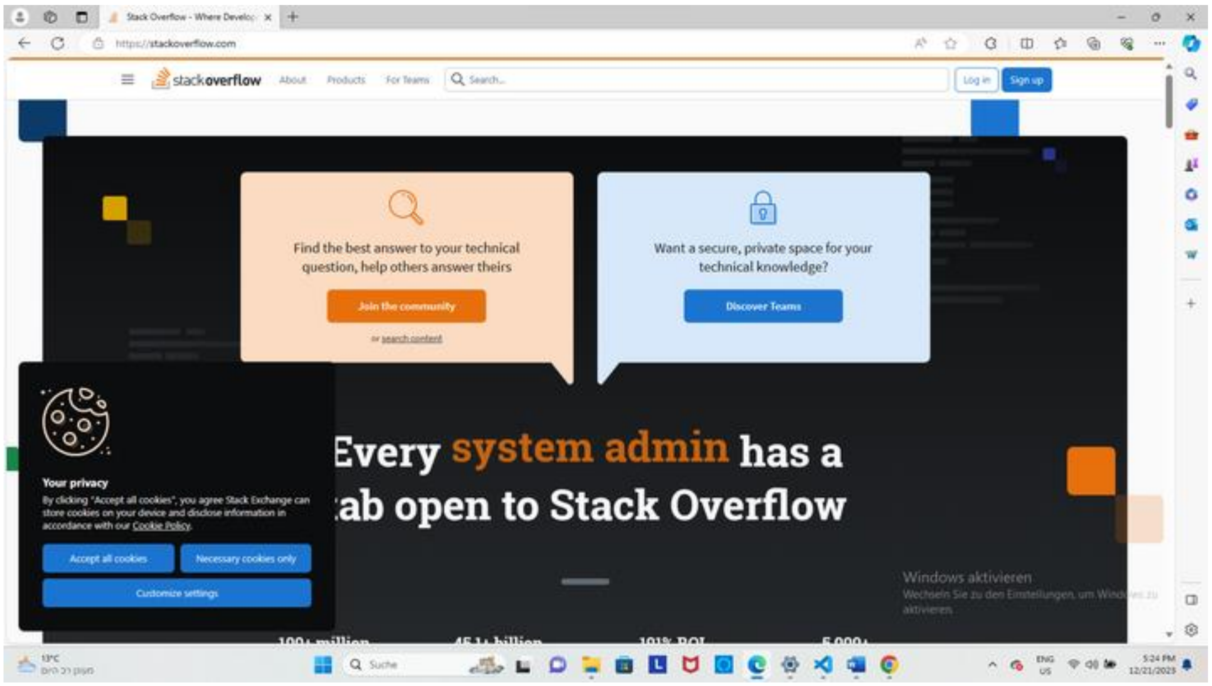
Figure 48:response /cr

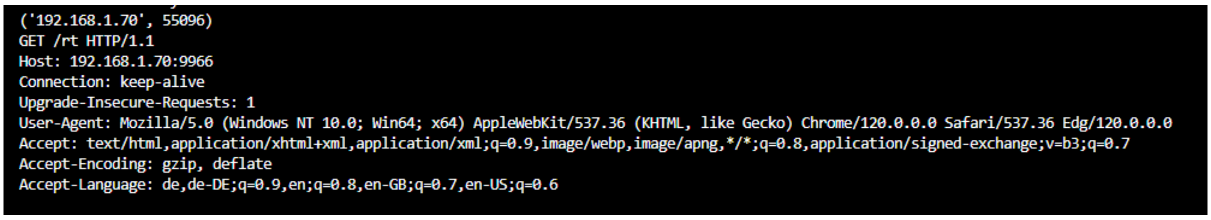




```
('192.168.1.70', 54933)
GET /so HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

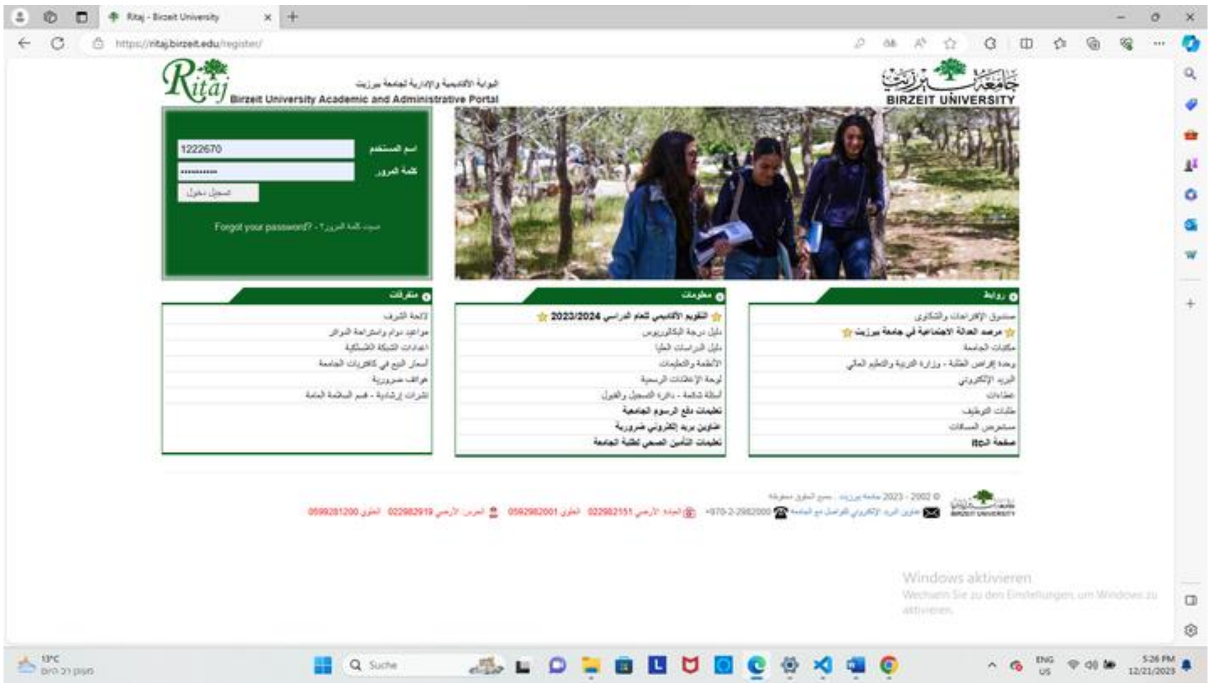
Figure 49:response /so

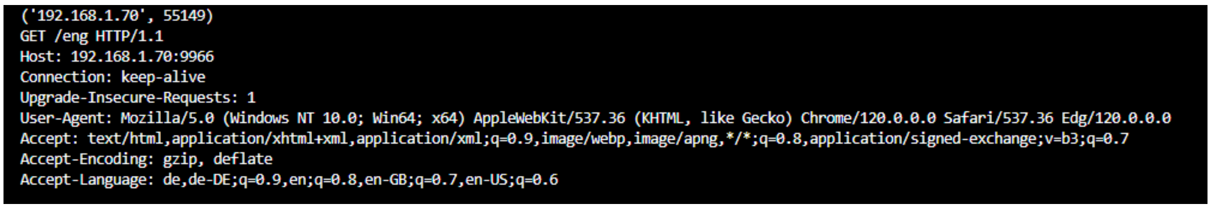

Stack Overflow - Where Develop
https://stackoverflow.com
stackoverflow
About
Products
For Teams
Search...
Log in
Sign up
Find the best answer to your technical question, help others answer theirs
Join the community
or search content
Want a secure, private space for your technical knowledge?
Discover Teams
Every system admin has a tab open to Stack Overflow
Your privacy
By clicking "Accept all cookies", you agree Stack Exchange can store cookies on your device and disclose information in accordance with our Cookie Policy.
Accept all cookies
Necessary cookies only
Customize settings
Windows aktivieren



```
('192.168.1.70', 55096)
GET /rt HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

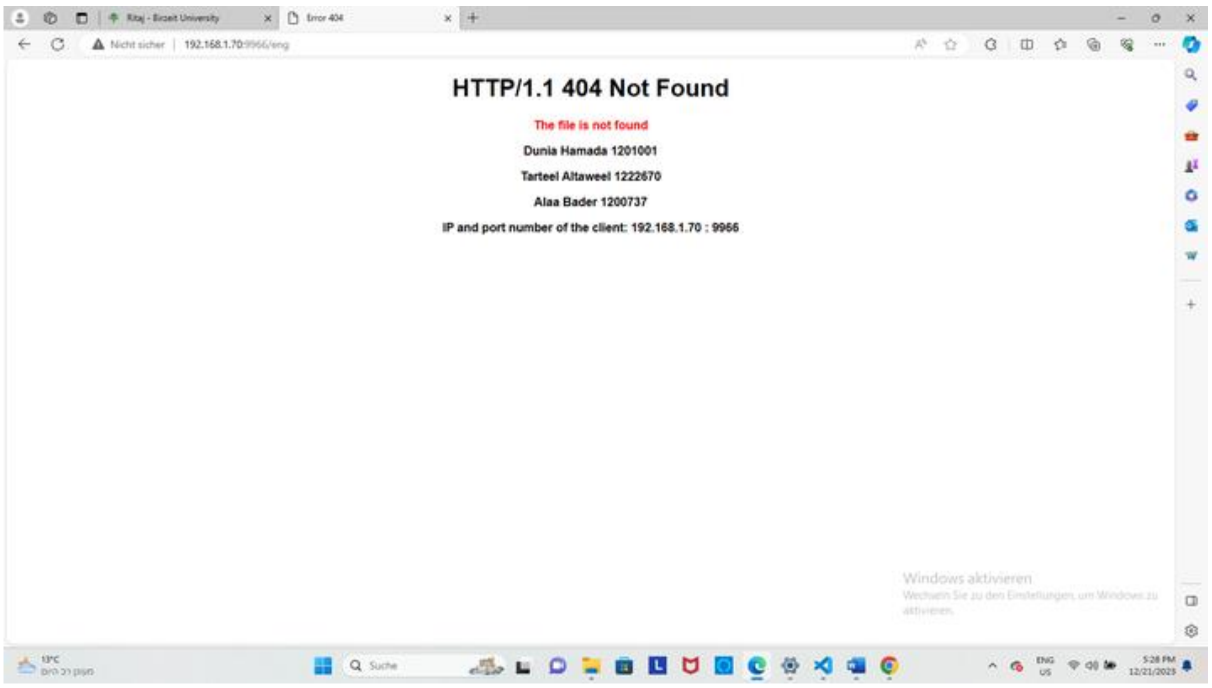
Figure 50:response /rt


Ritaj - Birzeit University
https://ritaj.birzeit.edu/register/
Ritaj
Birzeit University Academic and Administrative Portal
BIRZEIT UNIVERSITY
1222670
Forgot your password?
2023/2024
Windows aktivieren



```
('192.168.1.70', 55149)
GET /eng HTTP/1.1
Host: 192.168.1.70:9966
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36 Edg/120.0.0.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: de,de-DE;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
```

Figure 51: Response of error.html



HTTP/1.1 404 Not Found

The file is not found

Dunia Hamada 1201001

Tarteel Altaweel 1222670

Alaa Bader 1200737

IP and port number of the client: 192.168.1.70 : 9966

Figure 52:error.html