



Electrical & Computer Engineering

COMPUTER NETWORKS

ENCS3320

Project #1 Report

Instructor: Dr. Abdalkarim Awad

Students: **IDs:**

Lama Batta 1210922

Yuna Nawahda 1211524

Hanadi Asfour 1210209

Date: 8/5/2024

Part1:

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

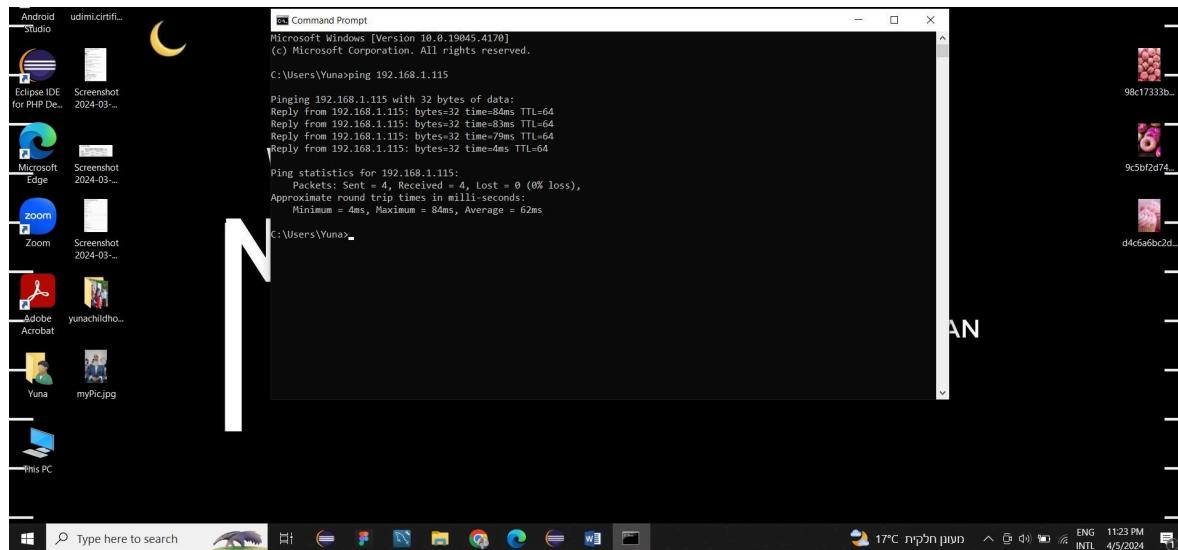
Answer:

- **ping:** It's a command that's used to check connection details between the user's device and another device or server, by sending packages to test it. (Useful to check if a device is reachable).
- **tracert:** a command showing details about a packet's path from one device to another, including the routers in between.
- **nslookup:** a command used to get the addresses of a certain domain name.
- **telnet:** a command used to test if a computer is willing to connect to another remote computer in the network, to manage it.

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

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

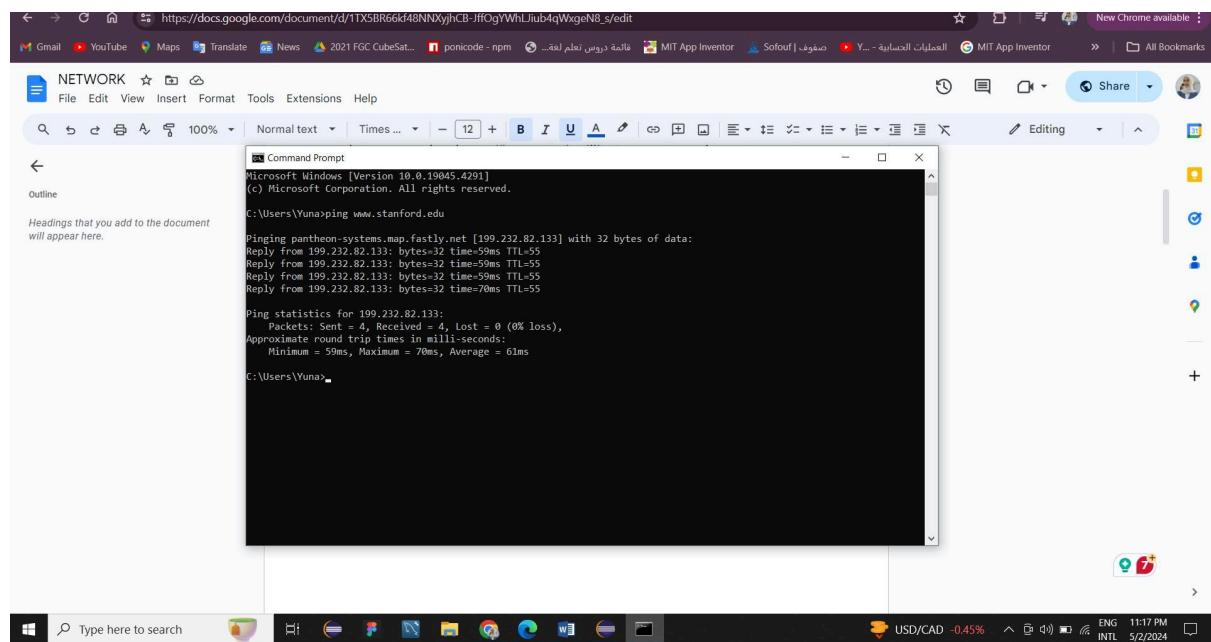
Answer:



The first four lines show the response to each request, including the time the device took to respond in milliseconds. The "TTL" stands for "Time to Live" and refers to the number of hops a packet can pass through before it is discarded. The bottom section shows that all packets were sent and received successfully and the round-trip time of the packets.

b) ping www.stanford.edu

Answer:



```
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

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

Pinging pantheon-systems.map.fastly.net [199.232.82.133] with 32 bytes of data:
Reply from 199.232.82.133: bytes=32 time=59ms TTL=55
Reply from 199.232.82.133: bytes=32 time=59ms TTL=55
Reply from 199.232.82.133: bytes=32 time=59ms TTL=55
Reply from 199.232.82.133: bytes=32 time=70ms TTL=55

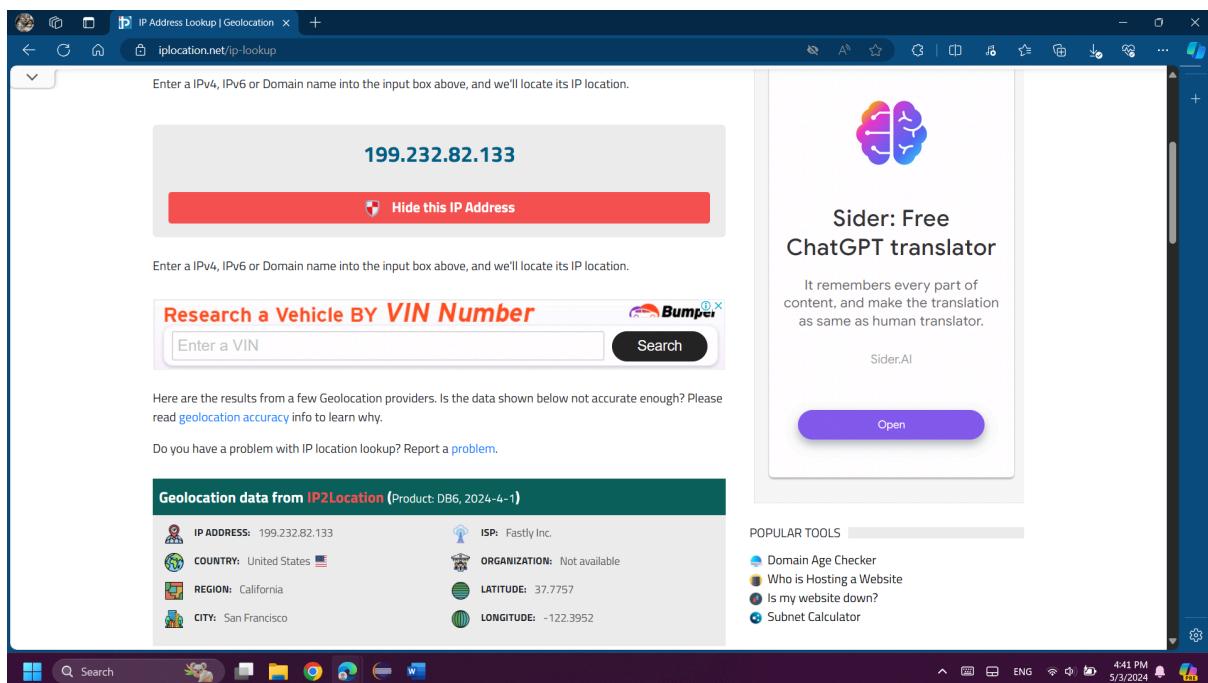
Ping statistics for 199.232.82.133:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 59ms, Maximum = 70ms, Average = 61ms

C:\Users\Yuna>
```

When pinging www.stanford.edu, four 32-byte packets of Internet Control Message Protocol (ICMP) Echo Request were sent to the server with a domain name of “pantheon-systems.map.fastly.net”. The server then responded with ICMP Echo Reply packets. All 4 of 4 packets sent were received (0% loss). The average time it took to receive replies for all packets (RTT round-trip time) was 61ms long. The IP address of www.stanford.edu was shown to be [199.232.82.133]. The TTL (Time to Live) was 55, this tells the network to allow the packet to go through 55 hops before it is discarded. This ping command is used for checking connectivity and response time.

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

Answer:



Yes, by entering the IP address 199.232.82.133 into iplocation.net, the location shows to be in the USA. This method was used since the geographical location cannot be determined by looking at an IP address only. It is important to note that the location of the website in a Content Delivery Network (CDN) and the use of VPNs can affect the accuracy of the geolocation. Therefore, the IP address might be shown to be located in (USA, California, San Francisco), but it might not be 100% accurate.

d) tracert www.stanford.edu

Answer:

```
Pinging pantheon-systems.map.fastly.net [199.232.82.133] with 32 bytes of data:
Reply from 199.232.82.133: bytes=32 time=59ms TTL=55
Reply from 199.232.82.133: bytes=32 time=59ms TTL=55
Reply from 199.232.82.133: bytes=32 time=59ms TTL=55
Reply from 199.232.82.133: bytes=32 time=70ms TTL=55

Ping statistics for 199.232.82.133:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip time in milli-seconds:
        Minimum = 59ms, Maximum = 70ms, Average = 61ms

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

Tracing route to pantheon-systems.map.fastly.net [146.75.122.133]
over a maximum of 30 hops:
1  11 ms   6 ms  328 ms www.webgui.Noklawifi.com [192.168.1.254]
2  16 ms   10 ms  12 ms ADSL-185.17.235.204.mada.ps [185.17.235.204]
3  13 ms   2 ms   5 ms   172.16.250.93
4  99 ms   51 ms  42 ms  10.160.160.253
5  145 ms   65 ms  82 ms me1-b5-link.ip.twelve99.net [62.115.34.30]
6  134 ms   77 ms  75 ms ffe-bb2-link.ip.twelve99.net [62.115.124.60]
7  66 ms   67 ms  66 ms ffe-b5-link.ip.twelve99.net [62.115.136.219]
8  68 ms   67 ms  65 ms fastly-ic-37940.ip.twelve99-cust.net [62.115.167.159]
9  64 ms   65 ms  69 ms  146.75.122.133

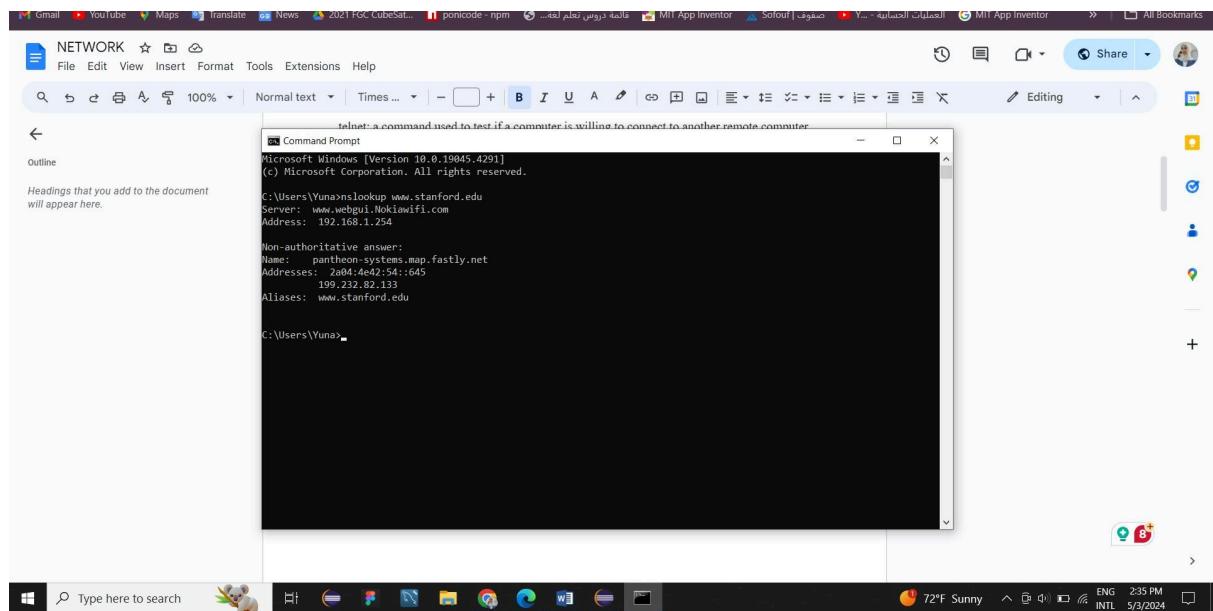
Trace complete.

C:\Users\Yuna>
```

The first line represents the IP address of the device. It connects the network from which this command was made to the internet (network gateway). The address of the router from which the ISP of the device is communicating during the request send. The rows between 3 to 9 represent the all hops of other networks, some global networks, and other third-party internet business organisations. Some of them belong to our ISP. And for the times represented on them are the packets sent.

e) nslookup www.stanford.edu

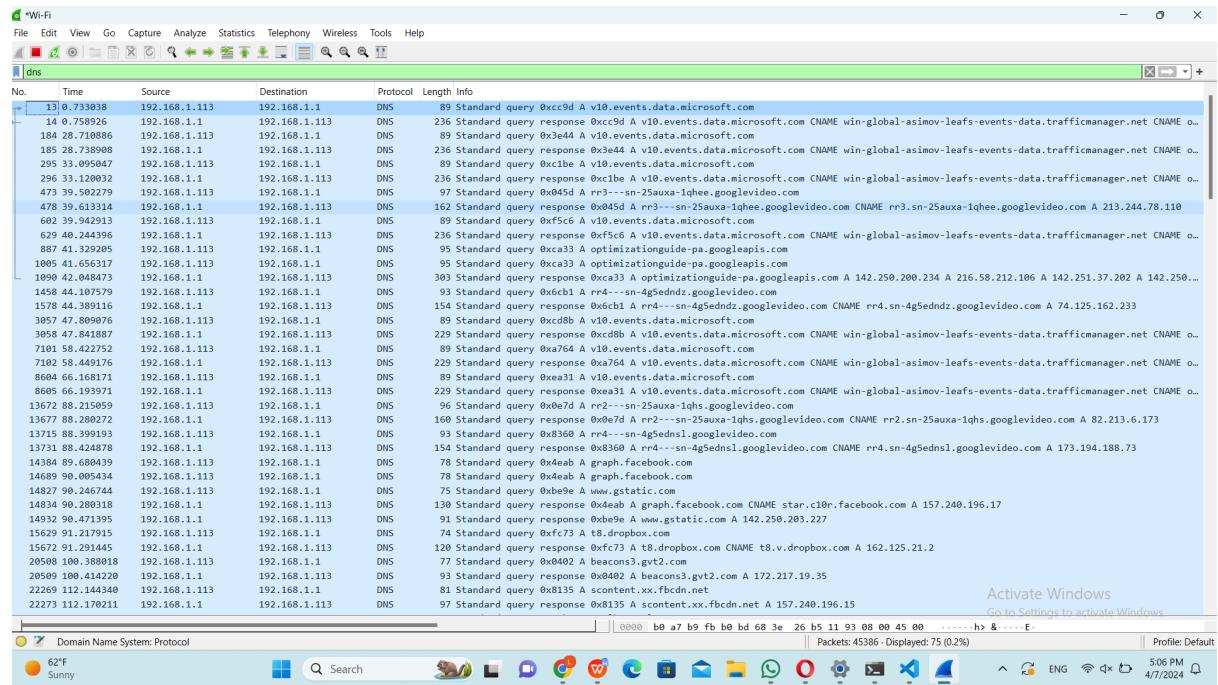
Answer:



NSLookup is a tool used to obtain domain names, IP addresses, and more. It stands for “Name Server Look Up”. The server is www.webgui.Nokiawifi.com and the address for it is 192.168.1.254

3- Use wireshark to capture some DNS messages.

Answer:



The client device sends a DNS query to the DNS server and the Wireshark will display detailed information about each DNS message, including questions and responses. It represents the source and IP address distribution, the domain, and type of query, the response will be containing the resolved IP address, the response type, and other details.

Part2:

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

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

A peer should display something like:

- Peer **First name Last name**
- 1- received a message from **First name Last name** at **Time**
 - 2- received a message from **First name Last name** at **Time**
 - 3- received a message from **First name Last name** at **Time**

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

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

You can use the IP address of the sender to distinguish between different senders.

The broadcast address of a network with subnet mask 255.255.255.0 is similar to the IP address of a host in that network with last octet **255**. For instance, if the IP address is 192.168.1.12 and the subnet mask is 255.255.255.0, then the broadcast address is 192.168.1.**255**.

If the subnet mask 255.255.0.0 then the last 2 octets of broadcast IP are **255.255**

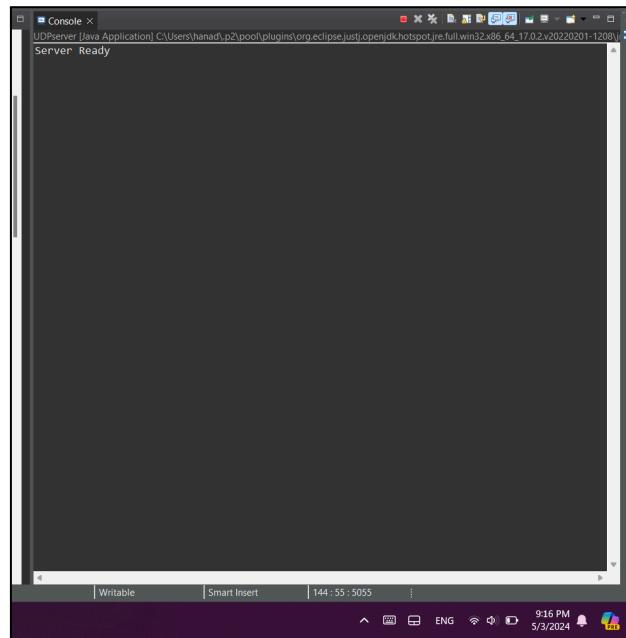
Each student should run a client or server (group of 3 students should contain at least 2 clients and one server)

For each run, provide screenshots of the run and provide a brief description.

Make sure that the computers are within the same subnet. For example, the IP of the first computer is 192.168.1.10, subnet mask 255.255.255.0 and the IP of the second computer is 192.168.1.11 and the subnet mask 255.255.255.0 the IP of the third computer is 192.168.1.12 and the subnet mask 255.255.255.0.

Solution:

1. Running Server:



A screenshot of the Eclipse IDE's Console window. The title bar reads "Console X UDPserver [Java Application] C:\Users\hanad\p2\pool\plugins\org.eclipse.jst\openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208". The main area of the console shows the message "Server Ready". The bottom status bar indicates the time as 144:55 : 5055 and the date as 5/3/2024.

When running the server, a confirmation message is printed indicating the server was successfully opened on port 5051 and is ready to receive connections to it by clients.

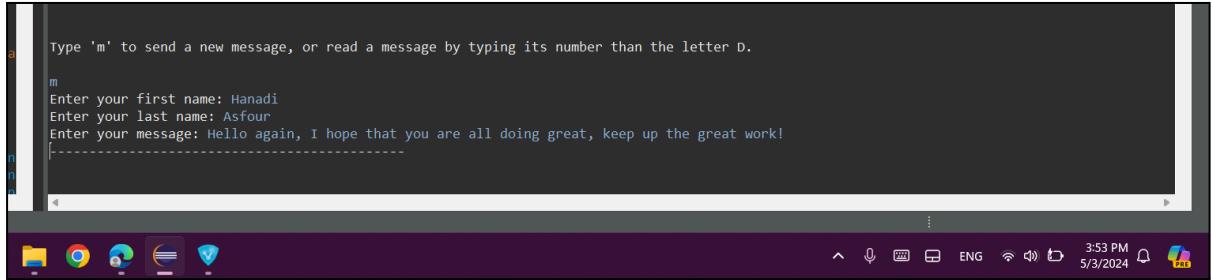
2. Connecting Clients:



A screenshot of the Eclipse IDE's Console window. The title bar reads "Console X UDPserver [Java Application] C:\Users\hanad\p2\pool\plugins\org.eclipse.jst\openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208". The main area of the console shows the messages "Server Ready", "connection from YUNA NAWAHDAH", "connection from lama batta", and "connection from Hanadi Asfour". The bottom status bar indicates the time as 3:49 PM and the date as 5/3/2024.

When a client successfully connects to the server, a confirmation message prints this peer's first and last name. The client connected to the server by specifying the server's IP address and port number (5051).

3. Sending a new message to the server.



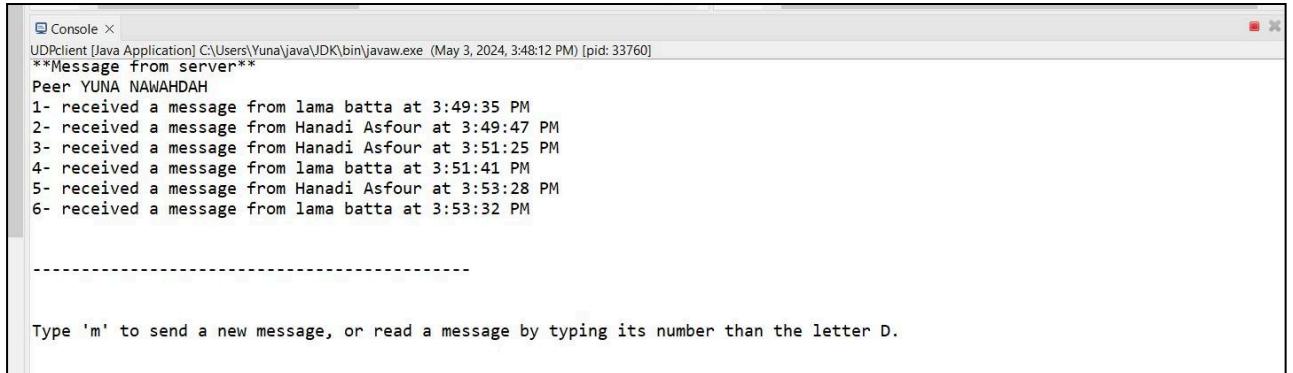
A screenshot of a terminal window titled "Console". The window shows the following text:

```
Type 'm' to send a new message, or read a message by typing its number than the letter D.  
m  
Enter your first name: Hanadi  
Enter your last name: Asfour  
Enter your message: Hello again, I hope that you are all doing great, keep up the great work!
```

The terminal window has a dark background. At the bottom, there is a taskbar with icons for File, Copy, Paste, and others. On the right side of the taskbar, it shows the date and time as "3:53 PM 5/3/2024".

When a peer types the letter "m" on an empty line in their terminal, they trigger the "send new message" feature, they are then asked to enter their first name, last name, and message content. A packet filled with this info as a byte array is then sent to the server. The server instantly distributes this new message to all peers except the sender himself.

4. Receiving messages as a peer.

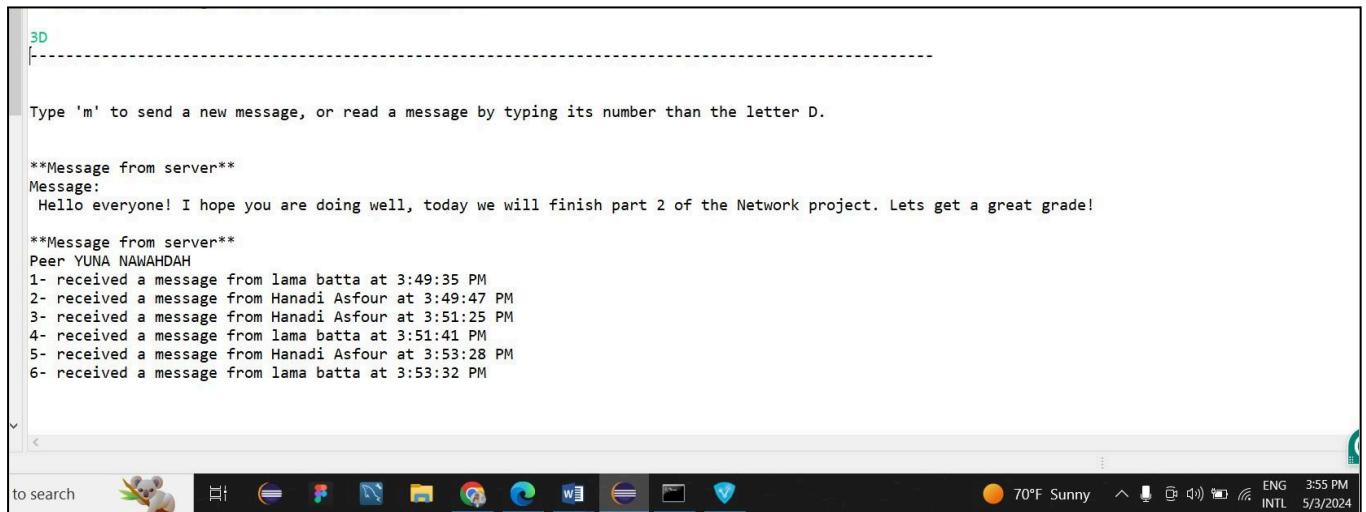


A screenshot of a terminal window titled "Console". The window shows the following text:

```
Console ×  
UDPClient [Java Application] C:\Users\Yuna\java\JDK\bin\javaw.exe (May 3, 2024, 3:48:12 PM) [pid: 33760]  
**Message from server**  
Peer YUNA NAWAHDAH  
1- received a message from lama batta at 3:49:35 PM  
2- received a message from Hanadi Asfour at 3:49:47 PM  
3- received a message from Hanadi Asfour at 3:51:25 PM  
4- received a message from lama batta at 3:51:41 PM  
5- received a message from Hanadi Asfour at 3:53:28 PM  
6- received a message from lama batta at 3:53:32 PM  
-----  
Type 'm' to send a new message, or read a message by typing its number than the letter D.
```

As long as the client is connected to the server, they will receive messages from other peers. The messages received are shown as a numbered list of the oldest to the latest. Each message is associated with a sender and a timestamp of when received.

5. Reading a received message



The screenshot shows a terminal window with the following text:

```
3D
-----
Type 'm' to send a new message, or read a message by typing its number than the letter D.

**Message from server**
Message:
Hello everyone! I hope you are doing well, today we will finish part 2 of the Network project. Lets get a great grade!

**Message from server**
Peer YUNA NAWAHDAH
1- received a message from lama batta at 3:49:35 PM
2- received a message from Hanadi Asfour at 3:49:47 PM
3- received a message from Hanadi Asfour at 3:51:25 PM
4- received a message from lama batta at 3:51:41 PM
5- received a message from Hanadi Asfour at 3:53:28 PM
6- received a message from lama batta at 3:53:32 PM
```

The terminal window has a dark theme with a light-colored text area. The system tray at the bottom shows various icons and the date/time: 70°F Sunny, ENG, 3:55 PM, 5/3/2024.

When a peer enters a desired number from the listed message printed to the terminal followed by the letter “D” on an empty line, the client will send a packet to the server holding this information. The server will parse this input and extract the referenced message from the list of the received messages for that peer. Then by encapsulating the obtained message content to a datagram packet, it is sent back to the peer that requested this message. The message content is received and printed to the peers’ terminal. The server also resends the list of messages to that peer only to remind them of their inbox content.

In the picture above, the peer chose to read the content of message 3 so they typed “3D” referencing the message sent by “Hanadi Asfour” at 3:51:25 PM. The content of the message sent by the server was printed under the title Message from the server.

Note:

The code can be found on GitHub through this link:

[hanadiasfour/NetworkProject_part2 \(github.com\)](https://github.com/hanadiasfour/NetworkProject_part2)

Part3:

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

Answer:

An ETag is a response header field value that is a cache validator. Moreover, it helps servers determine if a client's cached version of a resource is up to date. The server compares the ETag in the client request header with the ETag of the current version of the resource, if the dates match, the server sends a 304 Not Modified response, else it sends the updated resource. This way, we reduce unnecessary data transfers making responses faster and more efficient.

English Page:

File | C:/AppServ/www/Net/main_en.html

https://ritaj.birzeit.e... Textbook_economic... Second - Google Dr... HTML background... aboutus.html Gmail YouTube Maps Translate News 2021 FGC CubeSat... Other favorites

Welcome to our course Computer Networks, ENCS3320-tiny webserver

Local HTML File myform w3schools Python

example of .jpg image example of .png image

Team Member 1

Name: Yuna Nawahda
ID: 1211524



I am Yuna Nawahda a 3rd year Computer science student at [Birzeit University](#), Al-bireh Palestine.

My student ID is 1211524

- Education:**
 - [Sep-2021 - Jul 2025] Computer Science bachelor degree from [Birzeit University](#), Berzeit Palestine.
 - [Sep-2018 - Jul 2021] High School "TAWJIH" Scientific Field at [Ulmoo Al-had School](#), Al-Quds Palestine.
- Projects:** The projects that I worked:
 - JAWWAL-fleet management system database for fleet management system using Java and MySQL.
 - Developed an unbeatable Tic-Tac-Toe game utilizing the Minimax algorithm.
 - Created a Gaza Strip Map application implementing Dijkstra's shortest path algorithm for GIS applications.
 - Designed a Martyr Data Management System with advanced features using JavaFX, incorporating circular linked lists and AVL trees.
- Hobbies:**
 - art,drawing,baking,playing beskitball,singing,researching,playing piano,first aid.
- Skills:**
 - coding skills,time management skills ,soft skills ,project management ,research methodology,Knowledge of [Software]: JAVA, JS, HTML, CSS, Linux, Arduino,android studio,PHP SQL

Team Member 2

Name: Hanadi Asfour
ID: 1210209



I am Hanadi Asfour a 3rd year Computer science student at [Birzeit University](#), Ramallah Palestine.
My student ID is 1210209

Average of 98%

- Education:**
 - [Sep-2021 - Jul 2025] Computer Science bachelor degree from [Birzeit University](#), Berzeit Palestine. Deans Honor's List in all 5 semesters until now.
 - [Jul 2021] High School "TAWJIH" Scientific Field .
- Projects:** The projects that I worked:
 - [car agency] A dealership management system that provides inventory management, appointment scheduling, sales, payments, and analytics.
 - [Tic-Tac-Toe] This Tic-Tac-Toe game has different gameplay modes including an unbeatable AI using the Minimax algorithm.
 - [Gaza-map] Using Dijkstra's algorithm to find the best route between cities in the Gaza strip.
 - [Huffman-compression] Assigning variable-length codes to characters based on their frequencies for data compression and decompression.
 - [Max-LED-Lighting] Dynamic Programming for optimizing LED connections without wire crossing
- Hobbies:**
 - drawing, gymnastics, playing instruments including the recorder and piano, playing ping-pong, video games, and taking care of cats.
- Skills:**
 - Programming Languages: Java/JavaFX , C , Python , MySQL , XML , 3D Modeling , Mobile Development , Time Management , Teamwork , problem-solving , communication skills.

Team Member 3

Name: Lama Battah
ID: 1210922



I am Lama Battah a 3rd year Computer science student at [Birzeit University](#), Palestine.
My student ID is 1210922

- Education:**
 - [Sep-2021 - Jul 2025] Computer Science bachelor degree from [Birzeit University](#), Berzeit Palestine.
 - [Jul 2021] High School "TAWJIH" Scientific Field at [Bunat al bira al jadeda Secondary School](#).
- Projects:** The projects that I worked:
 - the Veterinary Clinic Management System project, I focused on backend development, coding functions for managing patient records, appointments, and medical history using Java and SQL.
 - In the Car Agency System project, I programmed backend functionalities to manage inventory and process customer requests using Java.
- Hobbies:**
 - art,drawing,programming,crafts,communication skills.
- Skills:**
 - Java/javaFX , C , DataBase , MySQL , Linux , Python , Html/Css , Teamwork , Communication skills

Arabic Page:

مرحباً بكم في دورتنا شبكات الحاسوب، ENCS3320 - الخادم الويب الصغير

مثل على صورة.jpg

مثل على صورة.png

عضو الفريق 1

الاسم: يوتا نواهضة
الرقم الجامعي: 12111524

انا يوتا نواهضة، طالبة في السنة الثالثة من كلية علوم الحاسوب في جامعة بير زيت، بيرزيت، فلسطين.
رقمي الجامعي هو 12111524

المشاريع:
• مدرستي: تطبيق خطي يربط بين مدارس مختلفة في فلسطين، يتيح للمدارس الاتصال والتعاون.
• Tic-Tac-Toe: تطبيق بسيط لמשחק تايك تاك تو.
• AVL: تطبيق برمجي ي Handle AVL Trees.
• الهرابيات: تطبيق يساعد المعلمين في إنشاء وتعديل دروسهم.
• المهام: تطبيق يساعد المعلمين في إنشاء وتعديل المهام.
• مهارات البرمجة، مهارات إدارة الوقت، مهارات الاتصال، إدارة المشاريع، معرفة بـ APIs، Java، سكريبت، HTML، CSS، Linux، Arduino، Android، PHP، SQL.

عضو الفريق 2

الاسم: هنادي عصافور
الرقم الجامعي: 1210209

انا هنادي عصافور، طالبة في السنة الثالثة من كلية علوم الحاسوب في جامعة بير زيت، رام الله، فلسطين.
رقمي الجامعي هو 1210209

المشاريع:
• [وكالة السيارات]: تطبيق لإدارة المطارات، جدولة المواعيد، السعيمات، الدفعات، والتحليلات.
• [أقنية لغوية وأدراك]: تطبيق يحل مشكلة الأقنية اللغوية.
• [خريطة غرب]: تطبيق خوارزمية Dijkstra على أفضل مسار بين المدن في قطاع غرب فلسطين.
• [مضطط فاهان]: تطبيق يتيح للمعلمين إنشاء وتعديل دروسهم.
• [ضوء LED]: برمجة ديناميكية لتحسين توصيات LED دون تفاصيل الأستان.

عضو الفريق 3

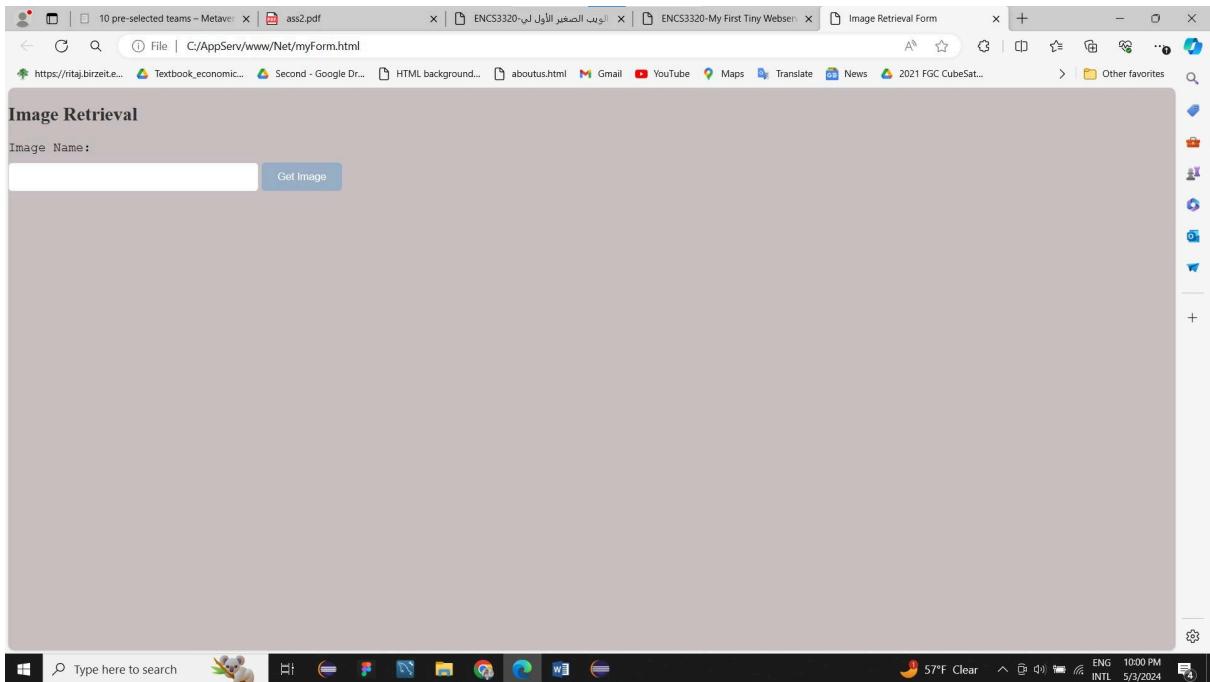
الاسم: لمى بطيه
الرقم الجامعي: 12110922

انا لمى بطيه، طالبة في السنة الثالثة من كلية علوم الحاسوب في جامعة بير زيت، فلسطين.
رقمي الجامعي هو 12110922

المشاريع:
• [بنية بيته]: تطبيق لإدارة عيادة الطبيب، يتيح للمعلمين إنشاء وتعديل دروسهم.
• [مشروع نظام إدارة عيادة الطبيب]: تطبيق يتيح للمعلمين إنشاء وتعديل دروسهم.
• [الهرابيات]: تطبيق يساعد المعلمين في إنشاء وتعديل دروسهم.
• المهام: تطبيق يساعد المعلمين في إنشاء وتعديل المهام.

Pressing Buttons on the Page:

- Local HTML myform.html page:



- Link to W3school:

Code:

```

118     fileInputStream = new FileInputStream("src/part3" + path);
119 } else {
120     fileInputStream = new FileInputStream("src/part3/download.jpg");
121 }
122 byte[] data = new byte[fileInputStream.available()];
123 fileInputStream.read(data);
124 contentType = "image/jpg";
125 out.write("HTTP/1.1 200 OK\r\n");
126 out.write(("Content-Length: " + data.length + "\r\n").getBytes());
127 out.write(("Content-Type: " + contentType + "\r\n").getBytes());
128 out.write("\r\n".getBytes());
129 out.write(data);
130 fileInputStream.close();
131
132 } else if (path.equals("/so")) {
133     out.write(("HTTP/1.1 307 Temporary Redirect\r\n" + "Content-Type: text/html\r\n" + "Location: "
134             + "https://stackoverflow.com" + "\r\n" + "\r\n").getBytes());
135     out.flush();
136 } else if (path.equals("/itc")) {
137     out.write(("HTTP/1.1 307 Temporary Redirect\r\n" + "Content-Type: text/html\r\n" + "Location: "
138             + "https://itc.birzeit.edu" + "\r\n" + "\r\n").getBytes());
139 } else {
140     //if the request is wrong or the file doesn't exist the server should send a HTML error page
141     sendNotFound(out, clientSocket);
142 }
143
144
145 out.flush();
146 }
147 //send a simple HTML error page
148 private static void sendNotFound(OutputStream out, Socket clientSocket) throws IOException {
149     String htmlResponse =
150         "<!DOCTYPE html>\r\n"
151         + "<html lang=\"en\"\r\n"
152         + "<head>\r\n"
153         + "    <meta charset=\"UTF-8\">\r\n"
154         + "    <meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0\">\r\n"
155         + "    <title>Error 404</title>\r\n"
156         + "    <style>\r\n"
157         + "        body {\r\n"
158         + "            font-family: Arial, sans-serif;\r\n"
159         + "            background-color: #888890;\r\n"
160         + "        }\r\n"
161         + "        .container {\r\n"
162         + "            max-width: 600px;\r\n"
163         + "            margin: 50px auto;\r\n"
164         + "            padding: 20px;\r\n"
165         + "            background-color: #fff;\r\n"
166         + "            border-radius: 5px;\r\n"
167         + "            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);\r\n"
168         + "        }\r\n"
169         + "        h1{\r\n"
170         + "            color: #000000;\r\n"
171         + "        }\r\n"
172         + "        h2{\r\n"
173         + "            color: red;\r\n"
174         + "        }\r\n"
175         + "        strong {\r\n"
176         + "            font-weight: bold;\r\n"
177         + "        }\r\n"
178         + "</style>\r\n"
179         + "</head>\r\n"
180         + "<body>\r\n"
181         + "    <div class=\"container\">\r\n"
182         + "        <h1>Error 404</h1>\r\n"
183         + "        <h2>The file is not found.</h2>\r\n"
184         + "        <p><strong>Lama Batta-121092</strong></p>\r\n"
185         + "        <p><strong>Yuna Nawahda-1211524</strong></p>\r\n"
186         + "        <p><strong>Hanadi Asfour-1210209</strong></p>\r\n"
187         + "        <p><strong>Client IP:</strong> " + clientSocket.getInetAddress().getHostAddress() + "</p>\r\n"
188         + "        <p><strong>Client Port:</strong> " + clientSocket.getPort() + "</p>\r\n"
189         + "    </div>\r\n"
190         + "</body>\r\n"
191         + "</html>";
192
193     out.write(("HTTP/1.1 404 Not Found\r\n" +
194             "Content-Type: text/html\r\n" +
195             "Content-Length: " + htmlResponse.length() + "\r\n" +
196             "\r\n" + htmlResponse).getBytes());
197     out.flush();
198     clientSocket.close();
199     out.close();
200 }
201 }
202
203

```



Code Explanation:

Socket programming was used to implement a web server in java. The server is constantly listening on port 6060 to accept incoming connections from all users. After the server accepts a client, the server reads the input stream and extracts the request line to check if it is a GET method. The path URL is then parsed and actions are taken accordingly, then the client socket is closed and the server is ready to accept a new request.

If the path represents “/”, “/index.html”, “/main_en.html”, or “/en”, the server opens the main_en.html page. Similarly, if the path is “/ar” or “/main_ar.html”, the server sends the main_ar.html page. A path with “.html” will open an HTML file with content-type: text/html, and “.css” will open a .css file with content-type: text/css. Having a path with “.jpg” or “.png” will display images with types JPG and PNG respectively. If the client enters the path as “/so” or “/itc”, the server will redirect them to stackoverflow.com and itc.birzeit.edu websites respectively. Any unrecognised URL path will cause a 404 error, which displays an error page to the user, and sends an appropriate HTML error response to the output stream towards the client.

There are three types of response messages the server can send out. First, the 200 OK response, which indicates the request is valid and actions were taken accordingly. Second, the 307 Redirect response, which informs the client that the requested resource has been temporarily moved and they will be directed to a new location specified in the location header. Last, the 404 Error, which indicates that the media format of the requested data is not supported by the server.

Opening English page:



Terminal output:

```
Console X
<terminated> Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 4, 2024, 2:35:41 AM - 2:41:02 AM) [pid: 18336]
Client connected from 0:0:0:0:0:0:1,Port:59138
Received request: GET /index.html HTTP/1.1
Connection closed

Client connected from 0:0:0:0:0:0:1,Port:59161
Received request: GET / HTTP/1.1
Connection closed

Client connected from 0:0:0:0:0:0:1,Port:59151
Received request: GET /main_en.html HTTP/1.1
Connection closed

Server listening on port 6060
Client connected from 0:0:0:0:0:0:1,Port:55272
Received request: GET /en HTTP/1.1
Connection closed
```

Opening Arabic page:

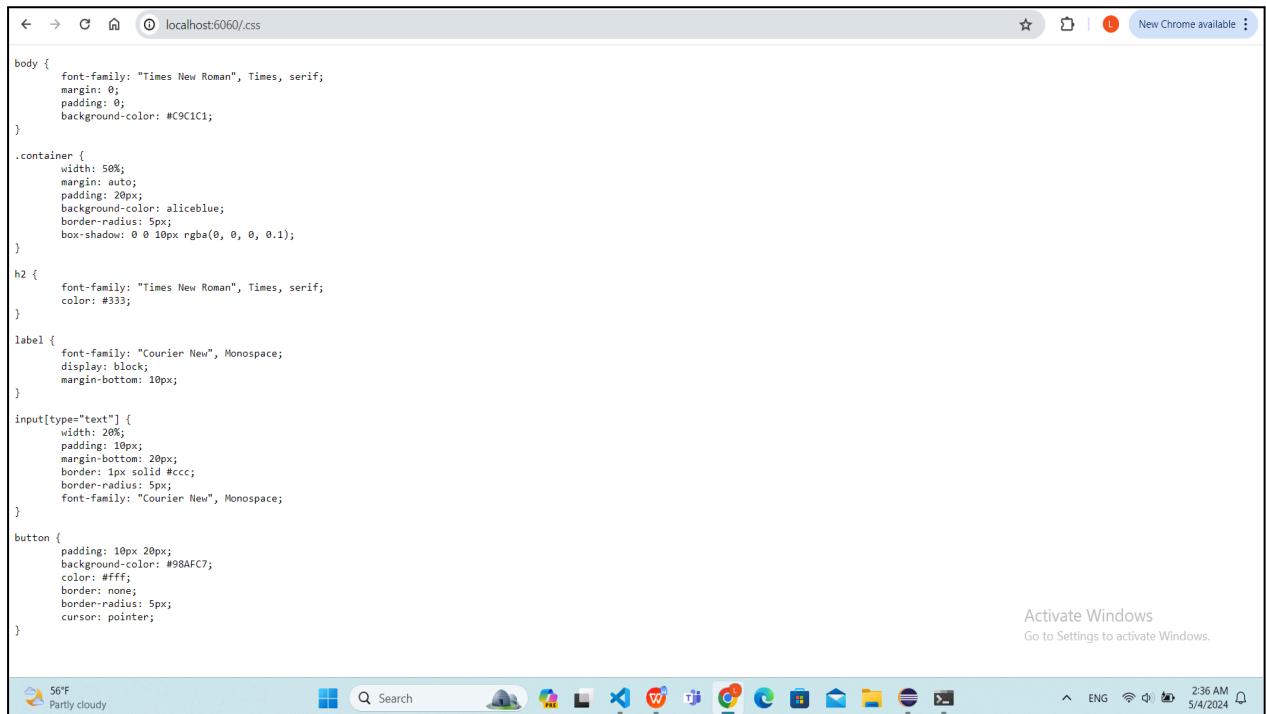


Terminal Output:

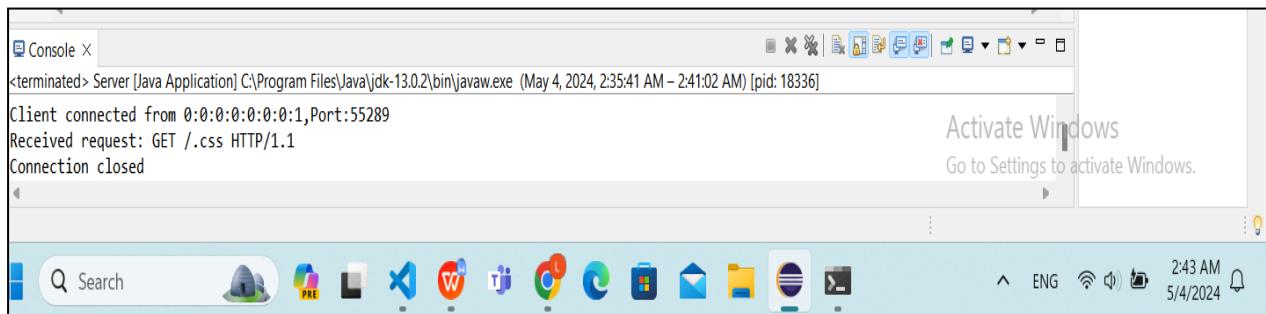
```
Console X
<terminated> Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 4, 2024, 2:35:41 AM – 2:41:02 AM) [pid: 18336]
Client connected from 0:0:0:0:0:1,Port:55279
Received request: GET /ar HTTP/1.1
Connection closed

Client connected from 0:0:0:0:0:1,Port:59173
Received request: GET /main_ar.html HTTP/1.1
Connection closed
```

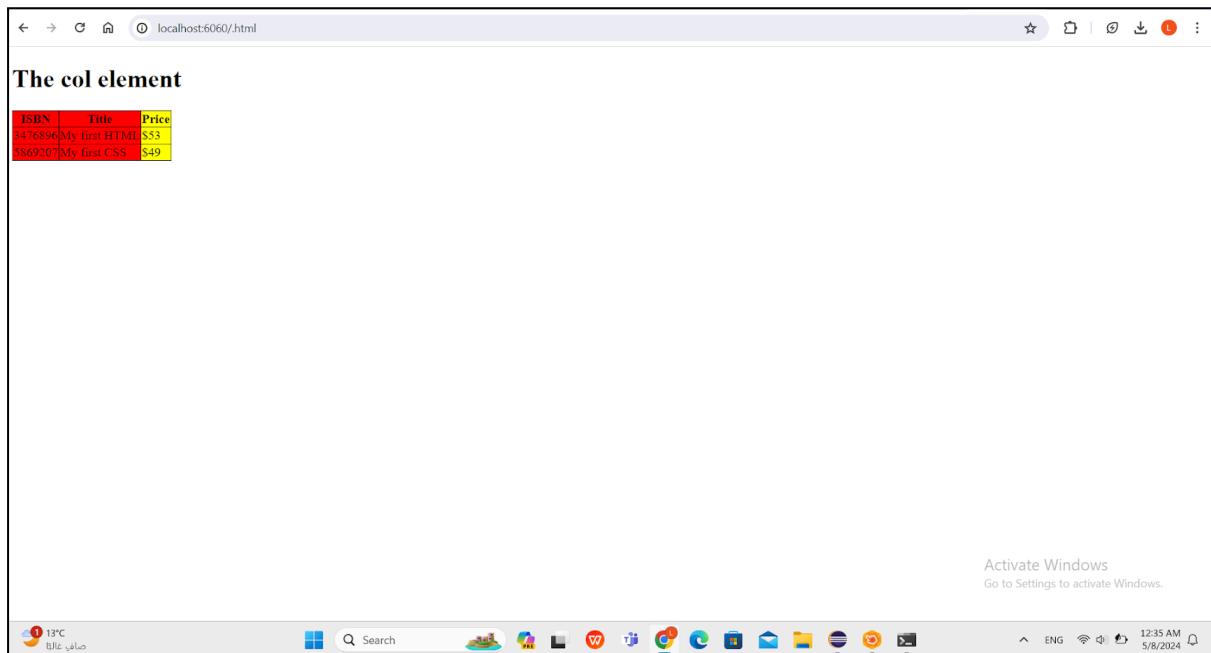
Opening a .css file:



Terminal Output:



Opening html file:

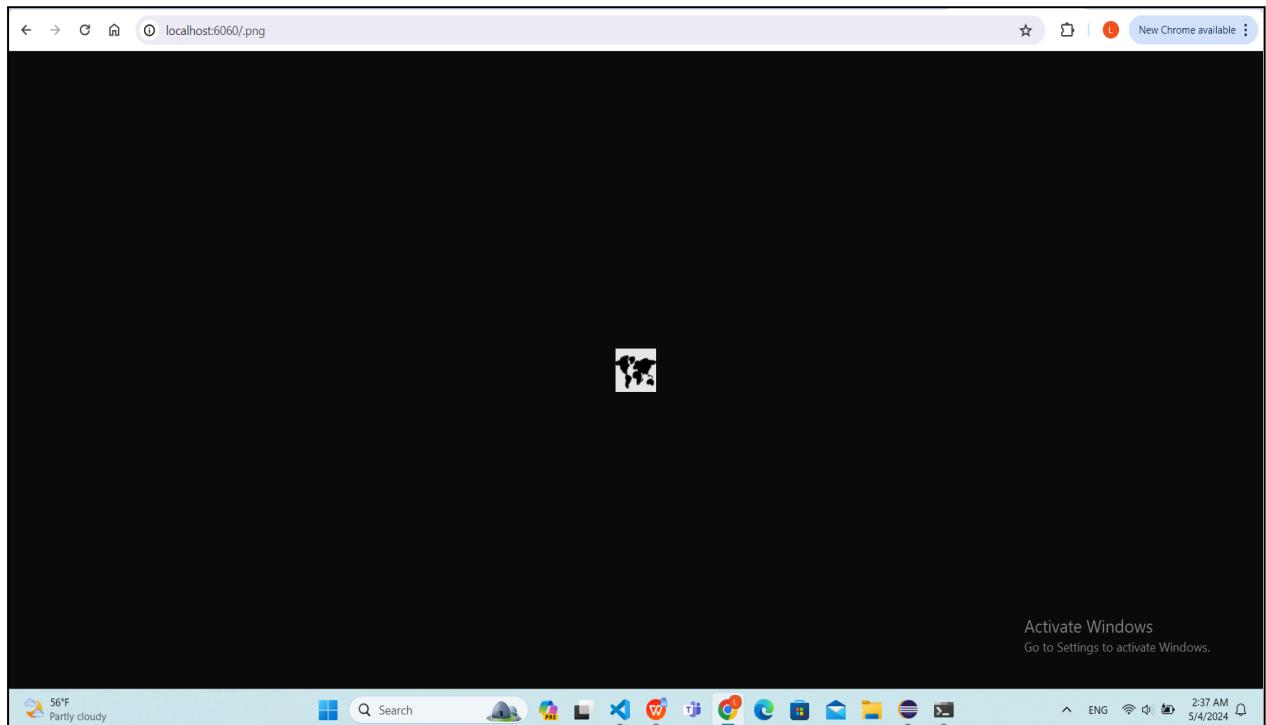


Terminal Output:

```
Client connected from 0:0:0:0:0:1,Port:59133
Received request: GET /.html HTTP/1.1
Connection closed
Client connected from 0:0:0:0:0:1,Port:59138
```

At the top of the terminal window, there is a watermark that says "Activate Windows Go to Settings to activate Windows." The taskbar at the bottom shows various pinned icons and the system tray with the date and time.

Opening .png photo:

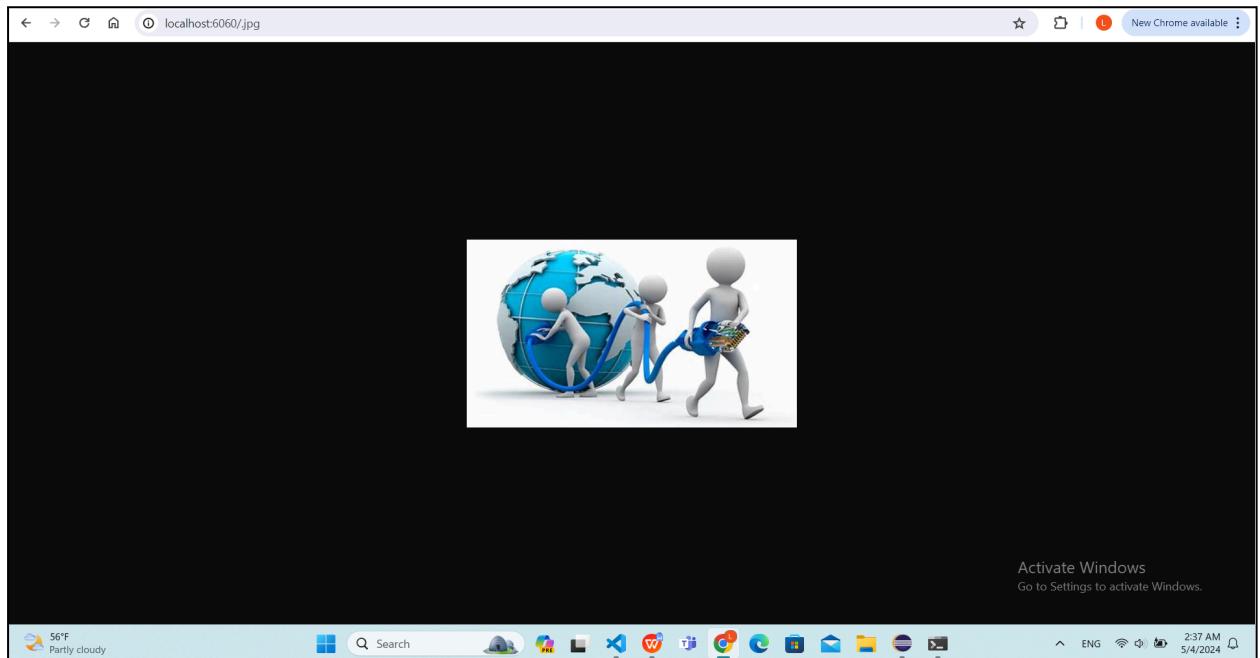


Terminal Output:

```
Client connected from 0:0:0:0:0:1,Port:55290
Received request: GET /.png HTTP/1.1
Connection closed
```

A screenshot of a terminal window displaying server logs. The logs show a client connection from 0:0:0:0:0:1 on port 55290, a received GET request for /.png, and the connection being closed. The terminal interface includes a search bar and a taskbar at the bottom with various application icons.

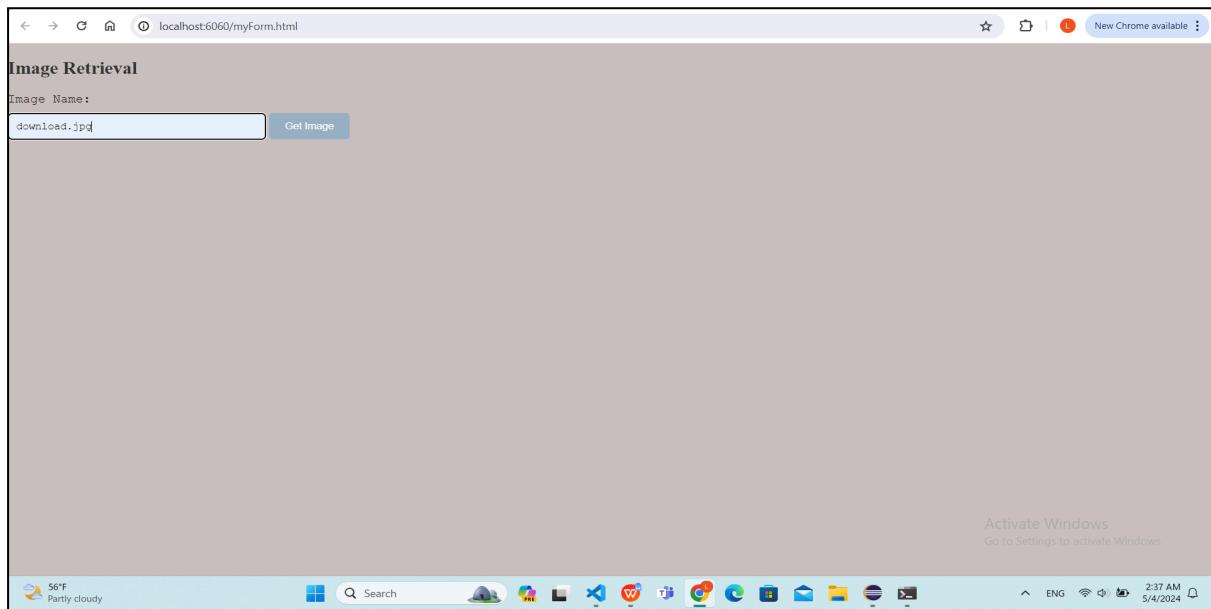
Opening .jpg photo:



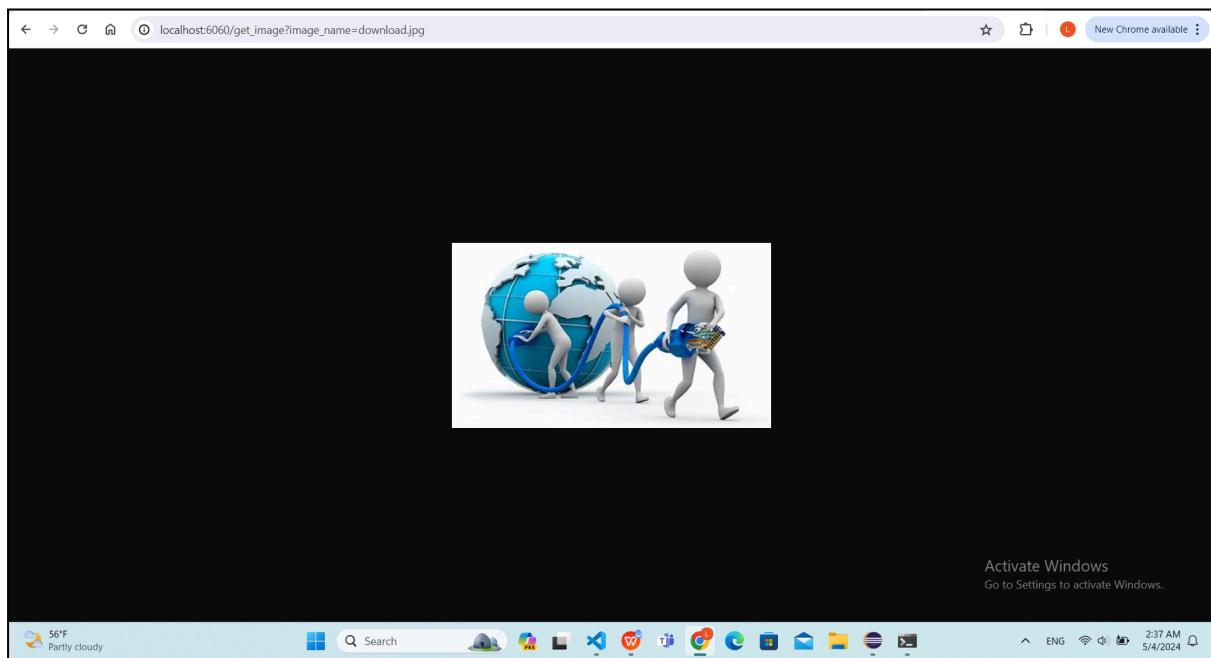
Terminal Output:

```
Console x
<terminated> Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 4, 2024, 2:35:41 AM – 2:41:02 AM) [pid: 18336]
Client connected from 0:0:0:0:0:1,Port:55293
Received request: GET /.jpg HTTP/1.1
Connection closed
```

Using Myform.html request:



Result:



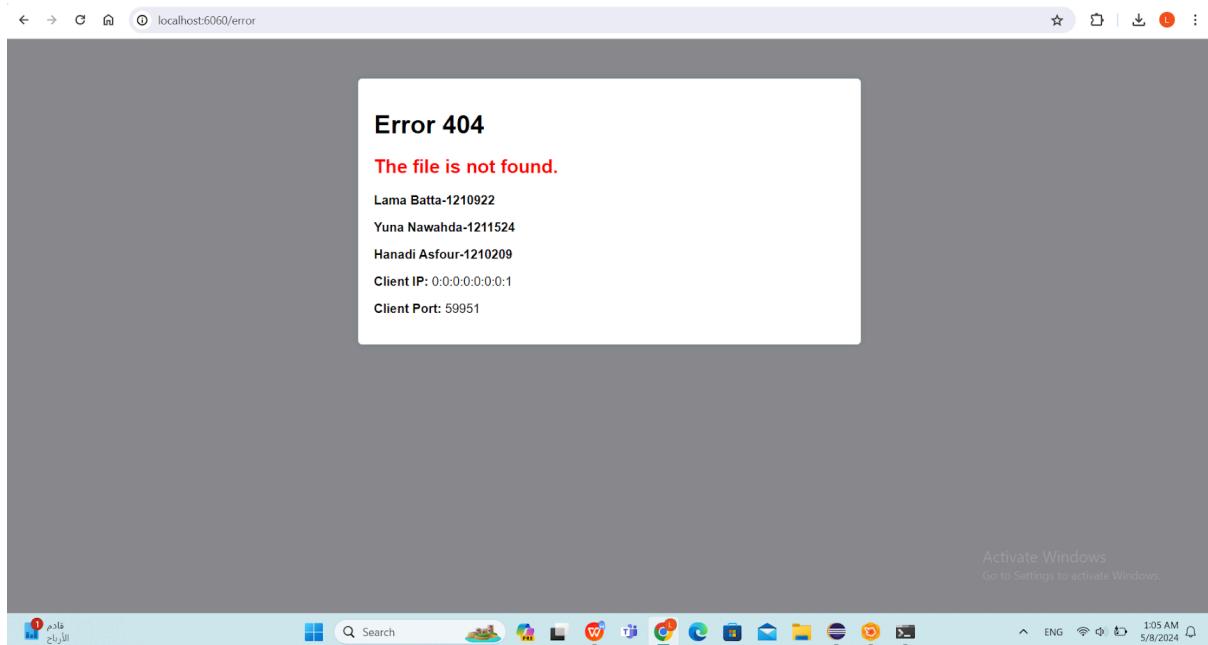
Terminal Output:

```
Console ×
<terminated> Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 4, 2024, 2:35:41 AM – 2:41:02 AM) [pid: 18336]
Connection closed
Client connected from 0:0:0:0:0:1,Port:55301 <terminated> Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 4, 2024, 2:35:41 AM – 2:41:02 AM) [pid: 18336]
Received request: GET /myForm.html HTTP/1.1
Connection closed
Client connected from 0:0:0:0:0:1,Port:55303
Received request: GET /get_image?image_name=download.jpg HTTP/1.1
Connection closed
```

Activate Windows
Go to Settings to activate Windows.

2:44 AM 5/4/2024

Error page:



Terminal Output:

A screenshot of a Java application running in a console window titled 'Console'. The window shows the following text:

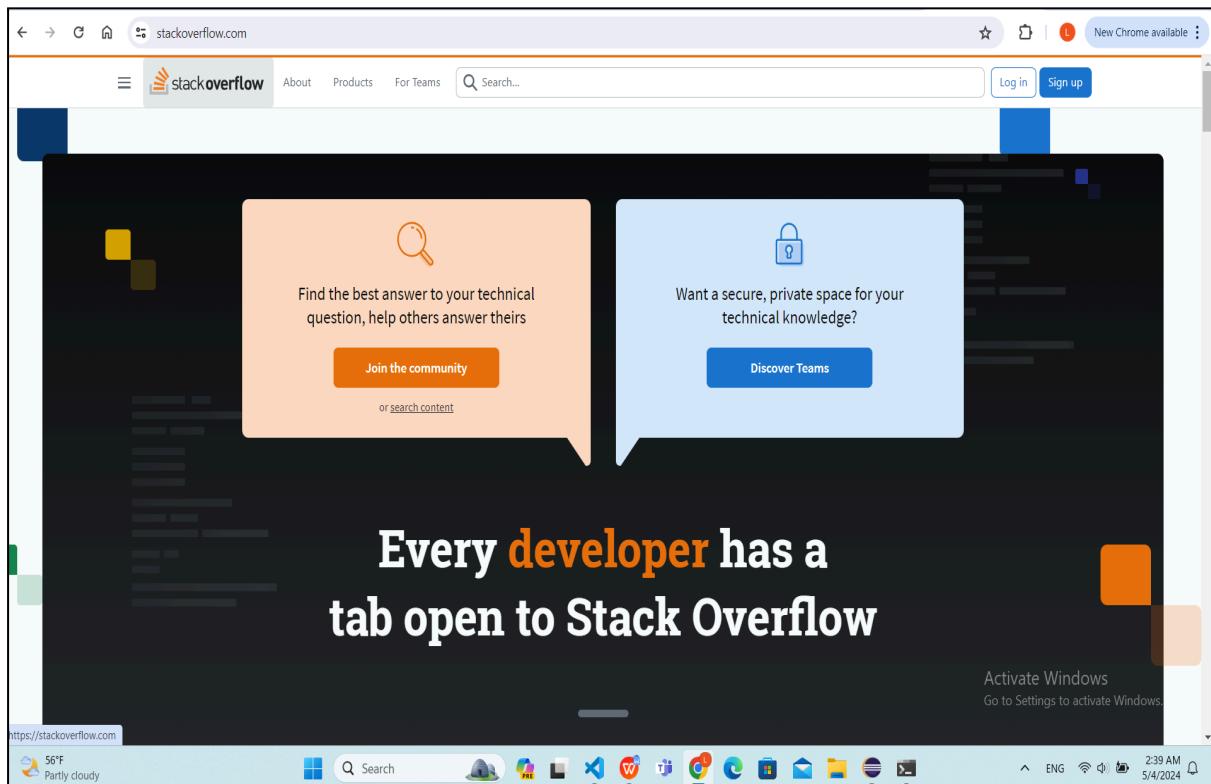
```
Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 8, 2024, 1:05:04 AM) [pid: 58700]
Server listening on port 6060
Client connected from 0:0:0:0:0:1,Port:59951
Received request: GET /error HTTP/1.1
Connection closed.
```

At the bottom right of the console window, there is an 'Activate Windows' watermark with the text 'Go to Settings to activate Windows.'. The taskbar at the bottom of the screen shows various pinned icons and the system tray with the date and time '5/8/2024 106 AM'.

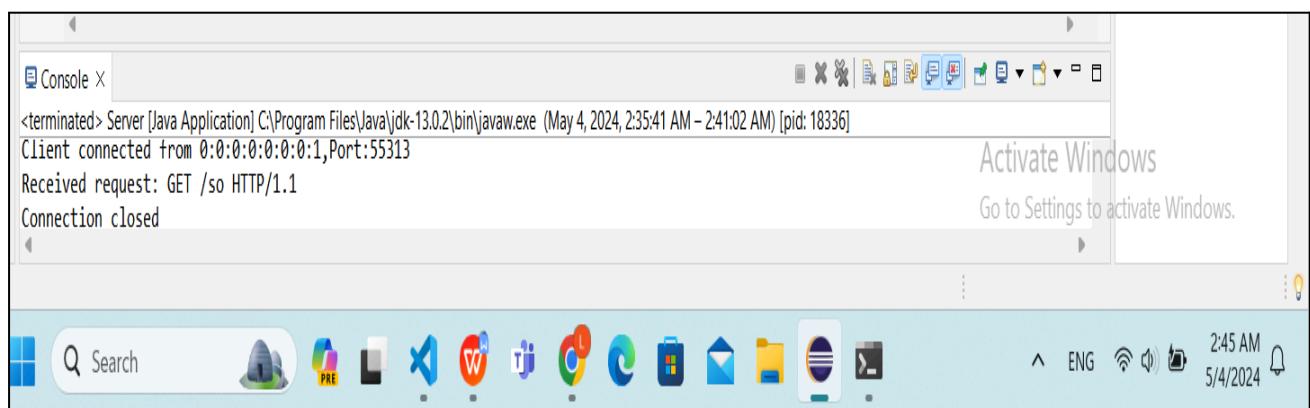
Redirecting to Stack overflow using /so:



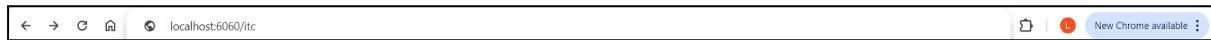
Stack overflow page:



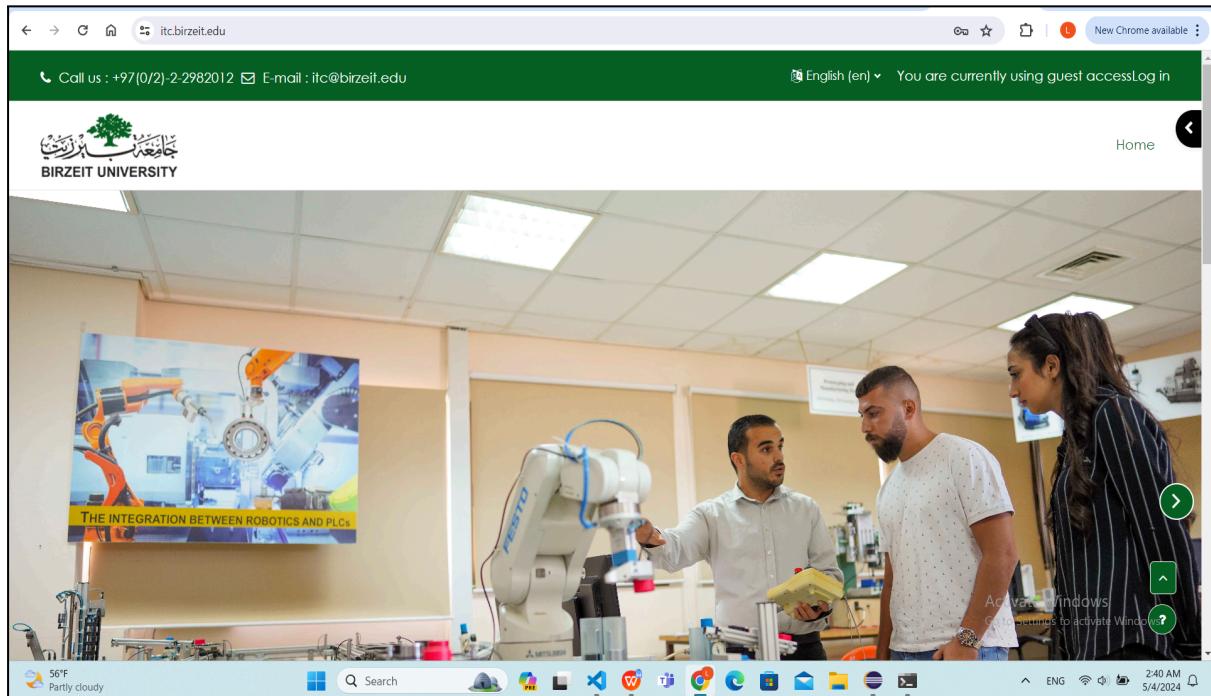
Terminal Output:



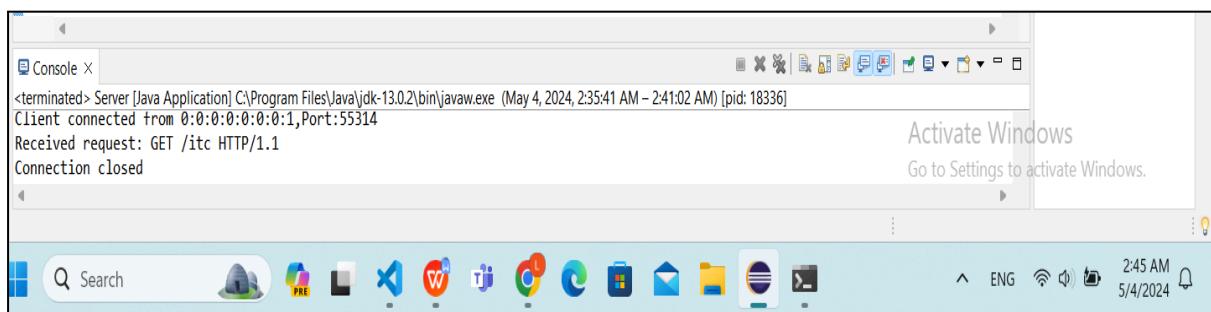
Redirecting to ITC using /itc:



ITC Page opens:



Terminal Output:



Actions Using phone:

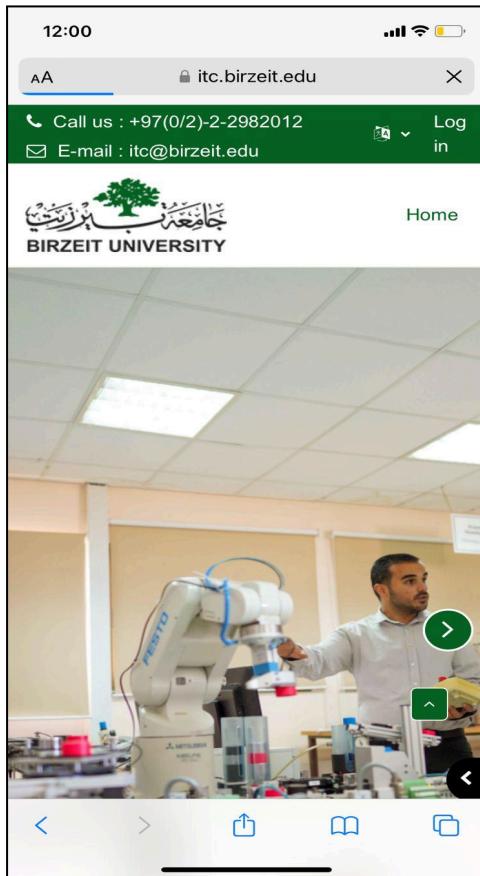
/itc:



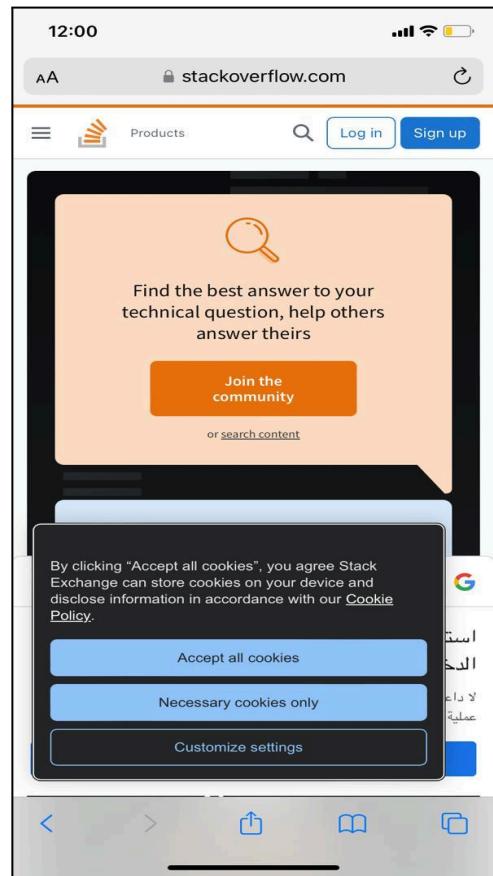
/so:



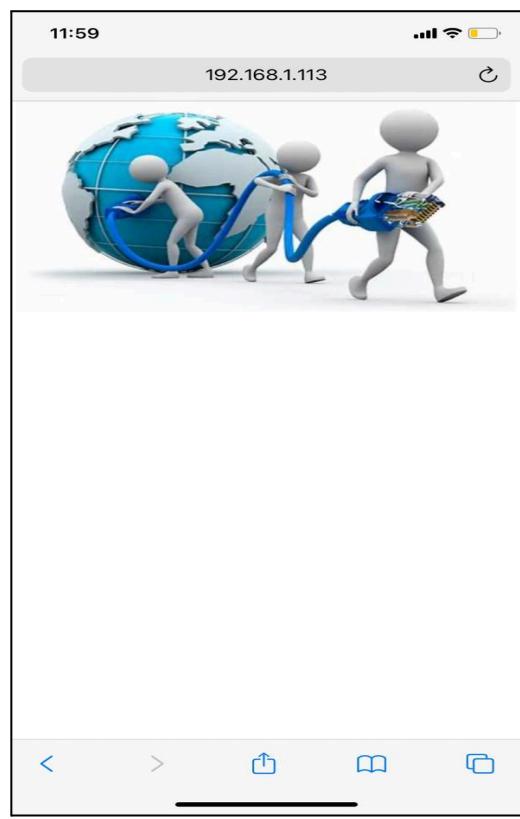
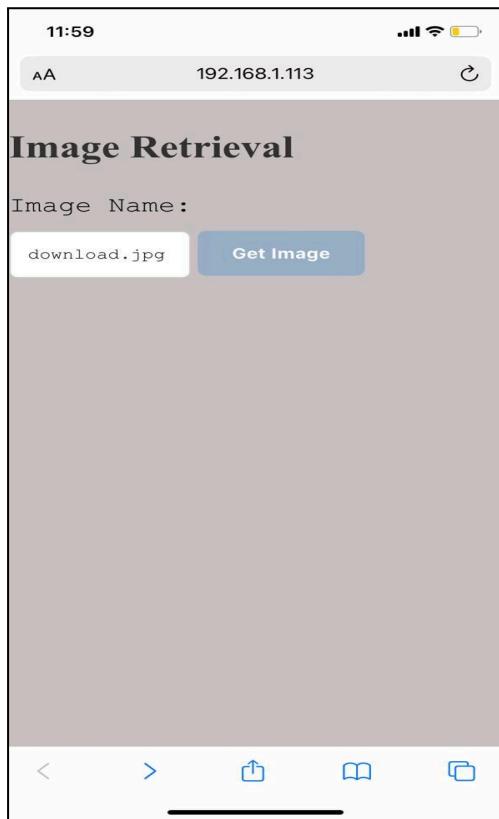
ITC Page opens:



Stack overflow Page opens:



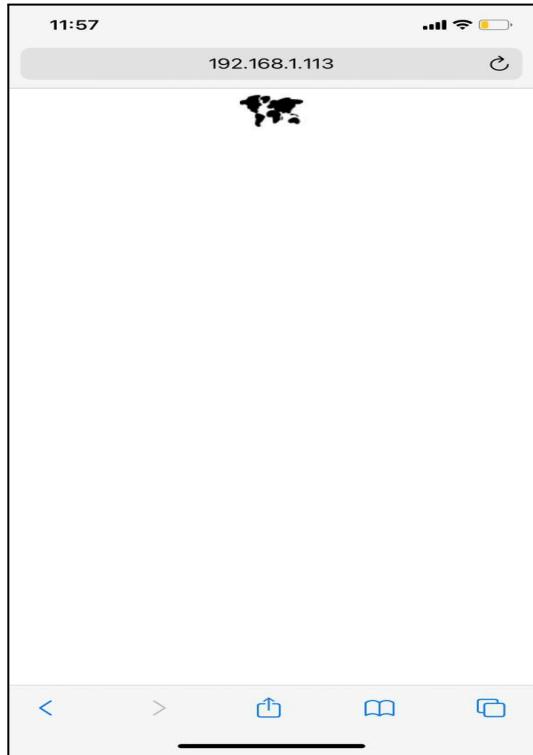
Using Myform.html:



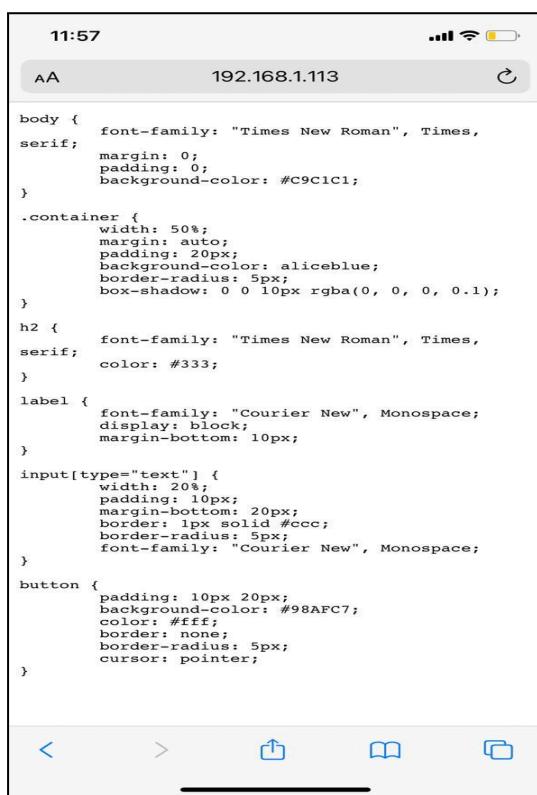
Opening jpg photo:



Opening png photo:



Css file:



```
body { font-family: "Times New Roman", Times, serif; margin: 0; padding: 0; background-color: #c9c1c1; }

.container {
    width: 50%; margin: auto; padding: 20px; background-color: aliceblue; border-radius: 5px; box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

h2 { font-family: "Times New Roman", Times, serif; color: #333; }

label { font-family: "Courier New", Monospace; display: block; margin-bottom: 10px; }

input[type="text"] { width: 20%; padding: 10px; margin-bottom: 20px; border: 1px solid #ccc; border-radius: 5px; font-family: "Courier New", Monospace; }

button { padding: 10px 20px; background-color: #98AFC7; color: #fff; border: none; border-radius: 5px; cursor: pointer; }
```

Error page:



English page:



Welcome to our course Computer Networks, ENCS3320-tiny webserver

Local AT&T File manager w3schools Python

 example of gif image

 example of jpg image

Team Member 1

Name: Yuna Nawahda
ID: 1211524

I am Yuna Nawahda a 3rd year Computer science student at Berzeit University, Berzeit, Palestine. My student ID is 1211524.

Education:

1. I.Sep 2021 - Jul 2023/ Computer Science bachelor degree from Berzeit University, Berzeit, Palestine.
2. I.Sep 2018 - Jul 2021/ High School "TAWFIK" Scientific Field at Al-Azhar School, Al-Zank Palestine.

Projects: The projects that i worked:

- 1. CRM: A fleet management system database for fleet management system using Java and MySQL.

Arabic page:



1:09
AA 192.168.1.113

-ENCS3320- مرحباً بكم في دورتنا شبكات الحاسوب، الخادم الويب الصغير

w3schools ملف محلي HTML

 مثال على صورة jpg

 مثال على صورة png

عضو الفريق 1

الاسم: يونا نواهدة
الرقم الجامعي: 1211524

أنا يونا نواهدة، طالبة في السنة الثالثة من كلية علم الحاسوب في جامعة بيروت، البير، فلسطين. رقمي الجامعي هو 1211524.

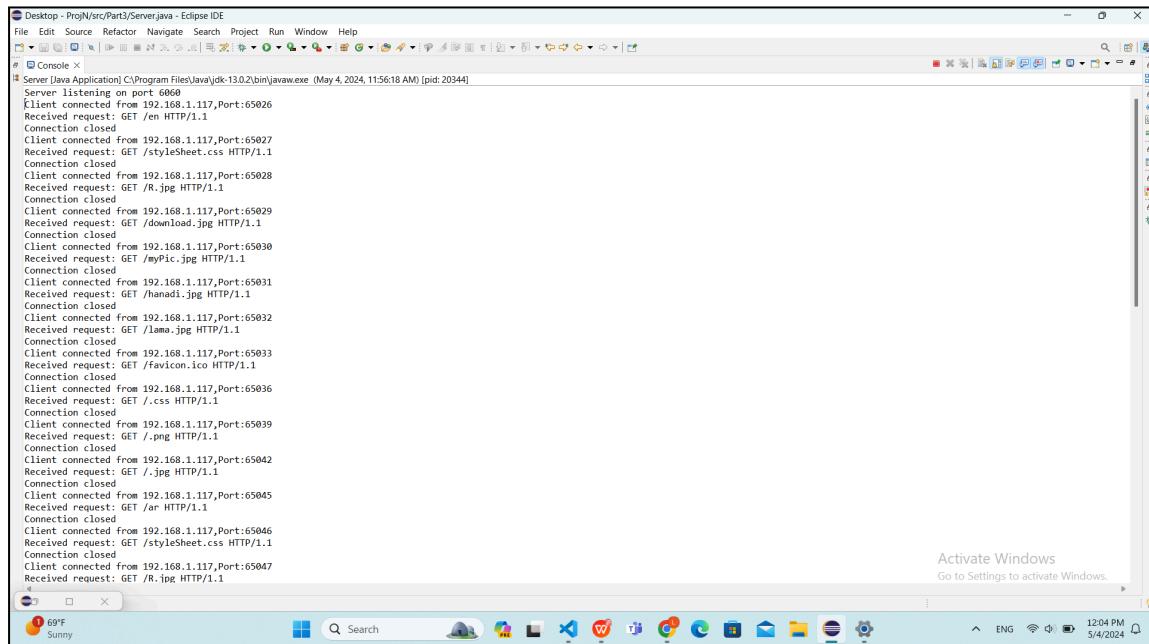
التعليم:

- 1. [سبتمبر- 2021 - يوليو 2025] درجة الميكالوريوس في علوم الحاسوب من جامعة بيروت، بيروت، فلسطين.
- 2. [سبتمبر- 2018 - يوليو 2021] الثانوية العامة "التجديف" في مجال المعلومات في مدرسة علام الفارسية، القدس، فلسطين.

• التخصص: الشاشة، البرمجة، علم البيانات.

Terminal Output from Request using phone :

The phone connected from the IP address (192.168.1.117) to the server of address (192.168.1.113)



Desktop - ProjN/src/Part3/Server.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

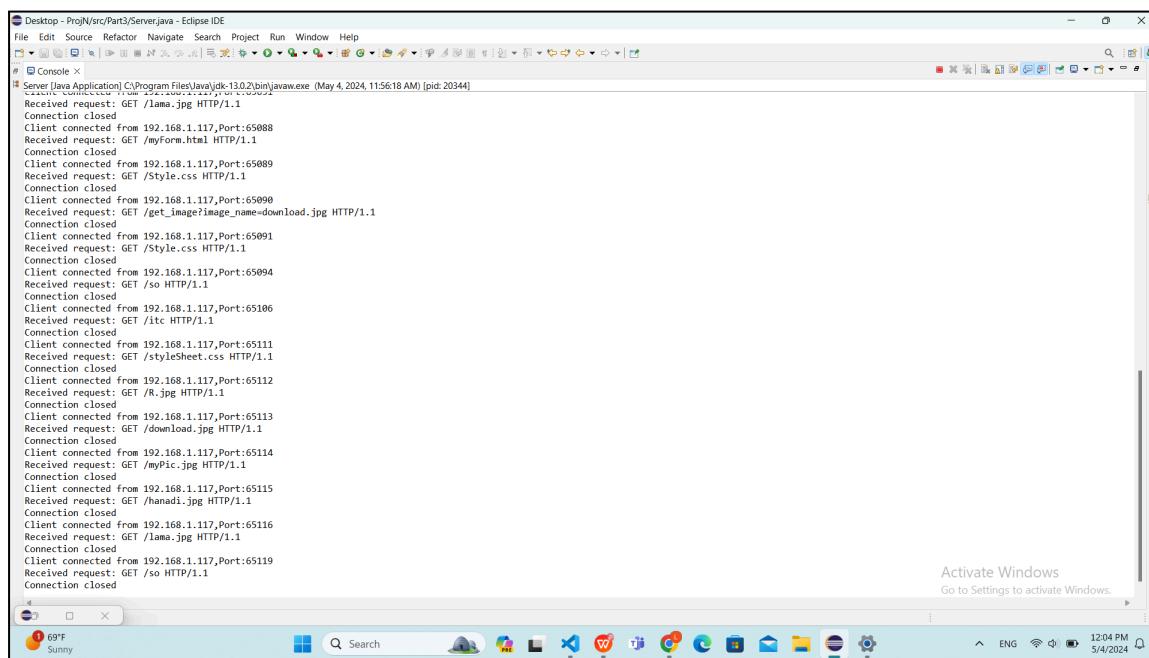
Console

Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 4, 2024, 11:56:18 AM) [pid: 20344]

```
Server listening on port 6090
Client connected from 192.168.1.117,Port:65926
Received request: GET /en HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65927
Received request: GET /styleSheet.css HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65928
Received request: GET /R.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65929
Received request: GET /download.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65930
Received request: GET /myPic.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65931
Received request: GET /handai.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65932
Received request: GET /flame.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65933
Received request: GET /favicon.ico HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65936
Received request: GET /.css HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65939
Received request: GET /.png HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65942
Received request: GET /.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65945
Received request: GET /ar HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65946
Received request: GET /styleSheet.css HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65947
Received request: GET /R.jpg HTTP/1.1
```

Activate Windows
Go to Settings to activate Windows.

69°F Sunny 12:04 PM 5/4/2024



Desktop - ProjN/src/Part3/Server.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Console

Server [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (May 4, 2024, 11:56:18 AM) [pid: 20344]

```
Server listening on port 6090
Client connected from 192.168.1.117,Port:65988
Received request: GET /lama.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65989
Received request: GET /myForm.html HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65989
Received request: GET /Style.css HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65990
Received request: GET /get_image?image_name=download.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65991
Received request: GET /Style.css HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65994
Received request: GET /so HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65106
Received request: GET /tc HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65111
Received request: GET /styleSheet.css HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65112
Received request: GET /R.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65113
Received request: GET /download.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65114
Received request: GET /myPic.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65115
Received request: GET /handai.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65116
Received request: GET /lama.jpg HTTP/1.1
Connection closed
Client connected from 192.168.1.117,Port:65119
Received request: GET /so HTTP/1.1
Connection closed
```

Activate Windows
Go to Settings to activate Windows.

69°F Sunny 12:04 PM 5/4/2024