Network 1

10636454

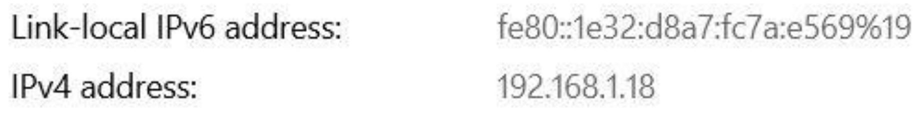
Masa Rateb Abu Aisheh

12027844

First Task :

In this task I will use my computer as server and client in the same time.

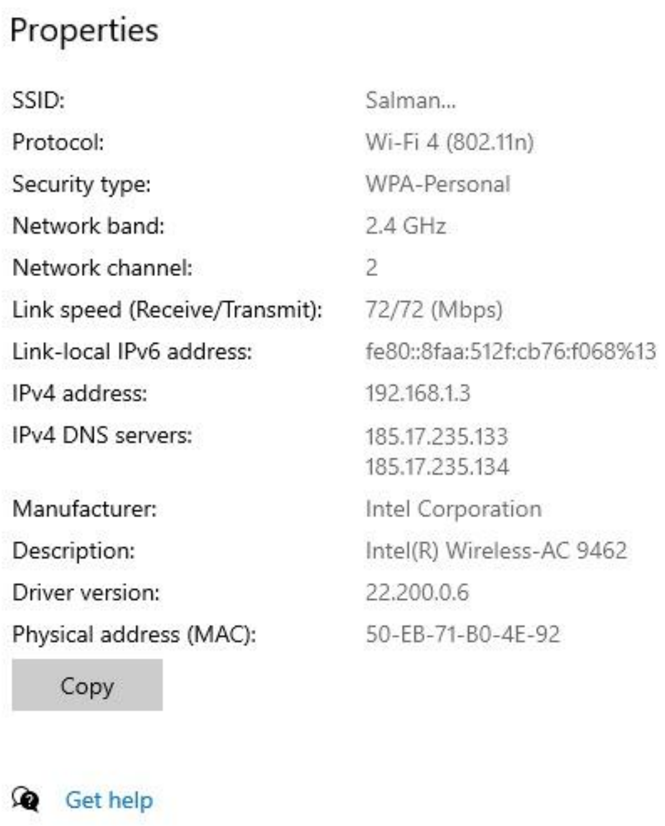
The IP for the Server and client is “192.168.1.18” and The port is “7851”, As we can see in the picture:



Link-local IPv6 address:	fe80::1e32:d8a7:fc7a:e569%19
IPv4 address:	192.168.1.18

And I will use two computers, one for the server (my computer) and one for the client.

The IP for the client is “192.168.1.13” .

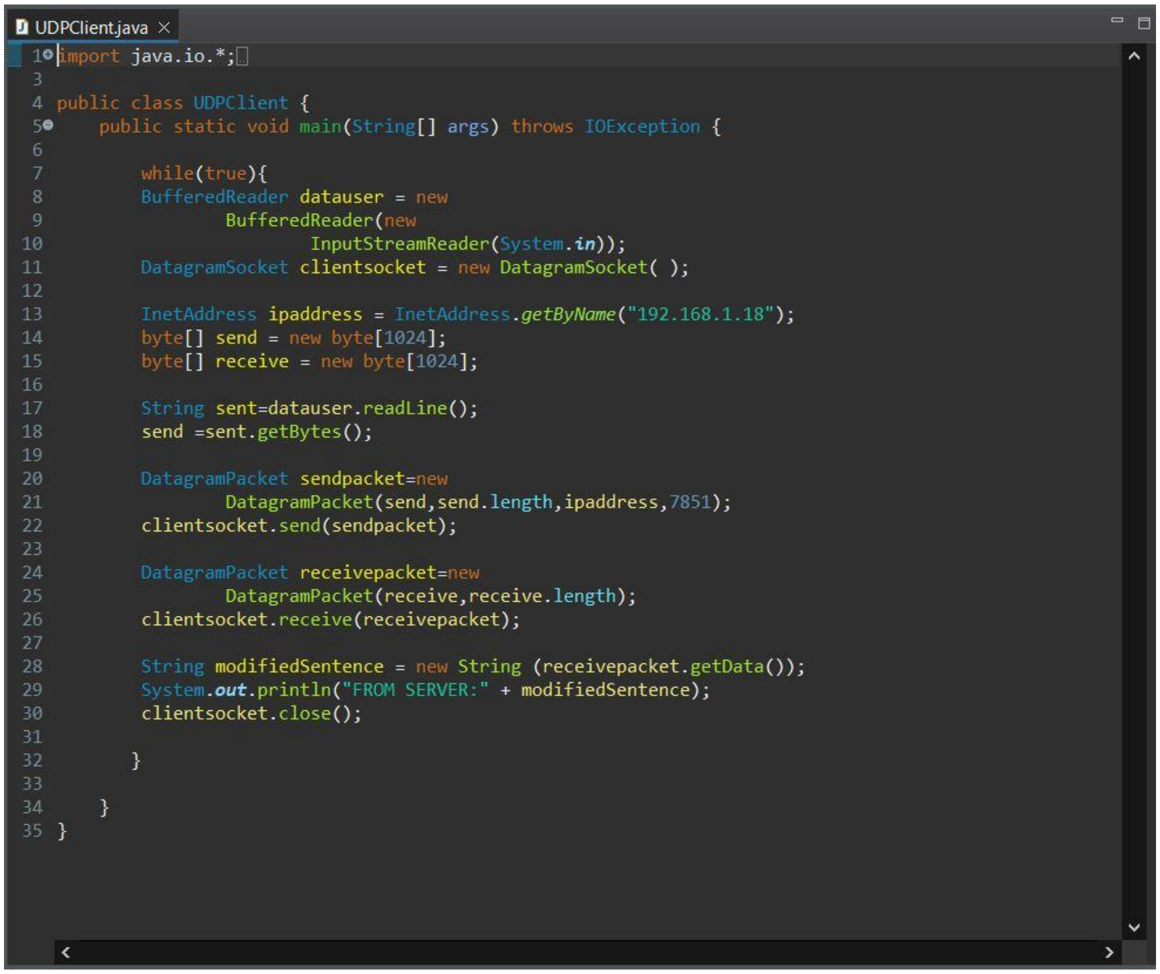


Properties

SSID:	Salman...
Protocol:	Wi-Fi 4 (802.11n)
Security type:	WPA-Personal
Network band:	2.4 GHz
Network channel:	2
Link speed (Receive/Transmit):	72/72 (Mbps)
Link-local IPv6 address:	fe80::8faa:512f:cb76:f068%13
IPv4 address:	192.168.1.3
IPv4 DNS servers:	185.17.235.133 185.17.235.134
Manufacturer:	Intel Corporation
Description:	Intel(R) Wireless-AC 9462
Driver version:	22.200.0.6
Physical address (MAC):	50-EB-71-B0-4E-92

Socket programing by UDB:

UDB Client code:



UDPClient.java

```java
import java.io.*;

public class UDPClient {
    public static void main(String[] args) throws IOException {

        while(true){
        BufferedReader datauser = new
                BufferedReader(new
                        InputStreamReader(System.in));
        DatagramSocket clientsocket = new DatagramSocket( );

        InetAddress ipaddress = InetAddress.getByName("192.168.1.18");
        byte[] send = new byte[1024];
        byte[] receive = new byte[1024];

        String sent=datauser.readLine();
        send =sent.getBytes();

        DatagramPacket sendpacket=new
                DatagramPacket(send,send.length,ipaddress,7851);
        clientsocket.send(sendpacket);

        DatagramPacket receivepacket=new
                DatagramPacket(receive,receive.length);
        clientsocket.receive(receivepacket);

        String modifiedSentence = new String (receivepacket.getData());
        System.out.println("FROM SERVER:" + modifiedSentence);
        clientsocket.close();

       }

    }
}
```

UDB Client code to explain:

1. I will first establish an input stream so that the user can input data (the Vehicle plate ID).

2. DatagramSocket clientSocket = new DatagramSocket(); I shall create the client socket.

3. After that, provide the client with the server's IP address and port number so they can connect.

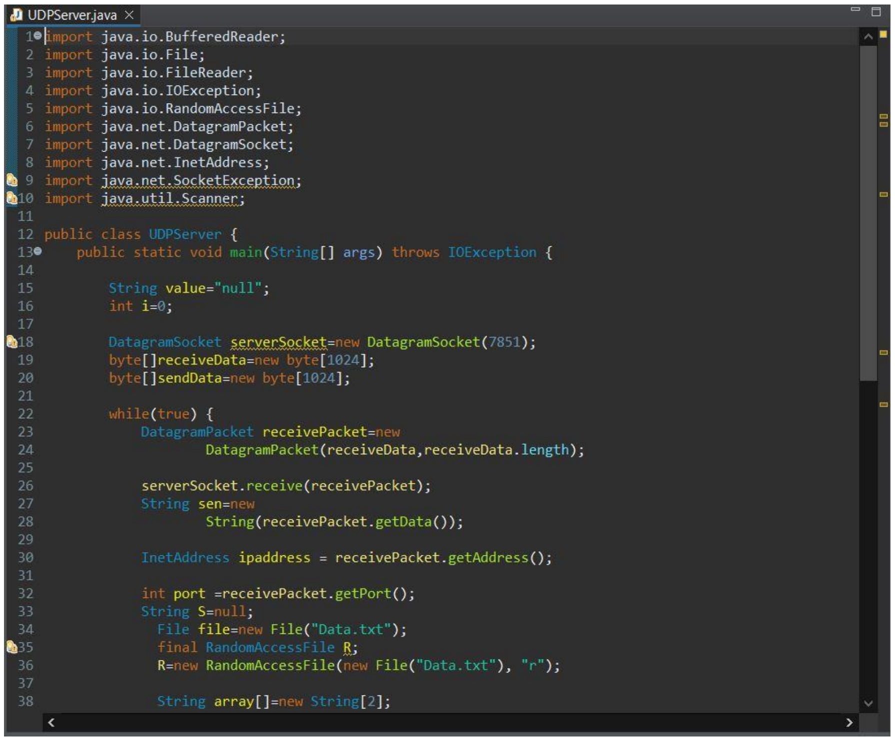
4. create an array of the type "byte" to hold the message and convert it to that format while sending and receiving the message.

5. Store the user's input as a string, convert it to byte format, and then store it in the array (the send array defined above), before preparing the message to be sent by creating the packet (the packet contains the message and the size of message and Ip address and port for server).

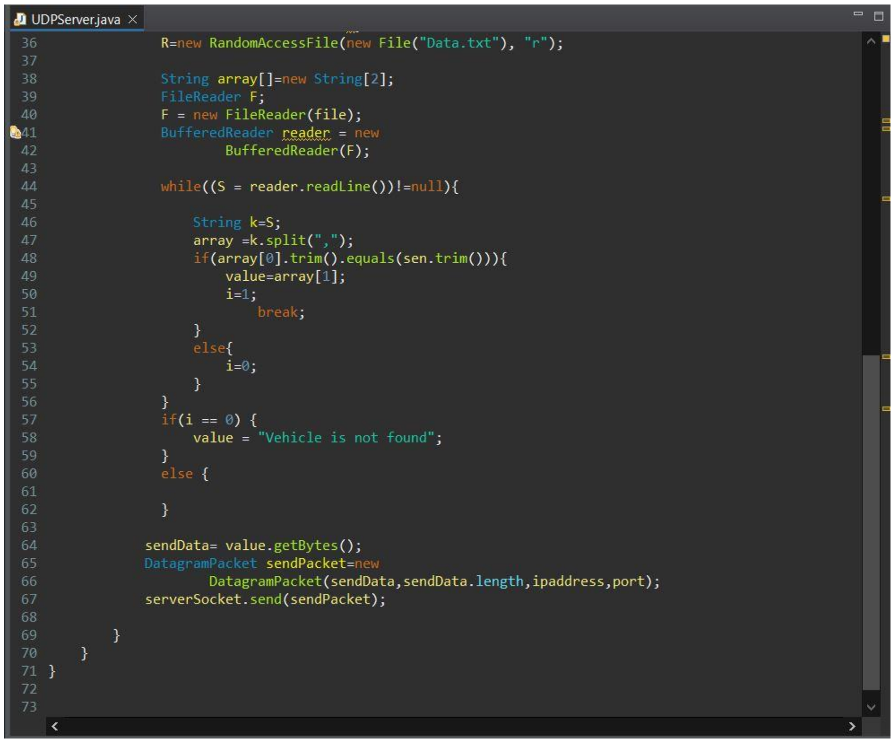
6. Next, create the packet to receive the server's message (contain the message and the size of it).

7. last, print the message after receiving it and cutting off your connection to the server.

UDB Server code:


UDPServer.java

```java
import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.IOException;
import java.io.RandomAccessFile;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;
import java.net.SocketException;
import java.util.Scanner;

public class UDPServer {
    public static void main(String[] args) throws IOException {

        String value="null";
        int i=0;

        DatagramSocket serverSocket=new DatagramSocket(7851);
        byte[]receiveData=new byte[1024];
        byte[]sendData=new byte[1024];

        while(true) {
            DatagramPacket receivePacket=new
                    DatagramPacket(receiveData,receiveData.length);

            serverSocket.receive(receivePacket);
            String sen=new
                    String(receivePacket.getData());

            InetAddress ipaddress = receivePacket.getAddress();

            int port =receivePacket.getPort();
            String S=null;
              File file=new File("Data.txt");
              final RandomAccessFile R;
              R=new RandomAccessFile(new File("Data.txt"), "r");

              String array[]=new String[2];
```



```java
              R=new RandomAccessFile(new File("Data.txt"), "r");

              String array[]=new String[2];
              FileReader F;
              F = new FileReader(file);
              BufferedReader reader = new
                      BufferedReader(F);

              while((S = reader.readLine())!=null){

                  String k=S;
                  array =k.split(",");
                  if(array[0].trim().equals(sen.trim())){
                      value=array[1];
                      i=1;
                          break;
                  }
                  else{
                      i=0;
                  }
              }
              if(i == 0) {
                  value = "Vehicle is not found";
              }
              else {

              }

            sendData= value.getBytes();
            DatagramPacket sendPacket=new
                    DatagramPacket(sendData,sendData.length,ipaddress,port);
            serverSocket.send(sendPacket);

        }
    }
}
```

UDB Server code to explain:

1. I'll define two arrays of type byte and create a socket on port 7851 to store the message in.

2. The server waits for any client to establish a connection.

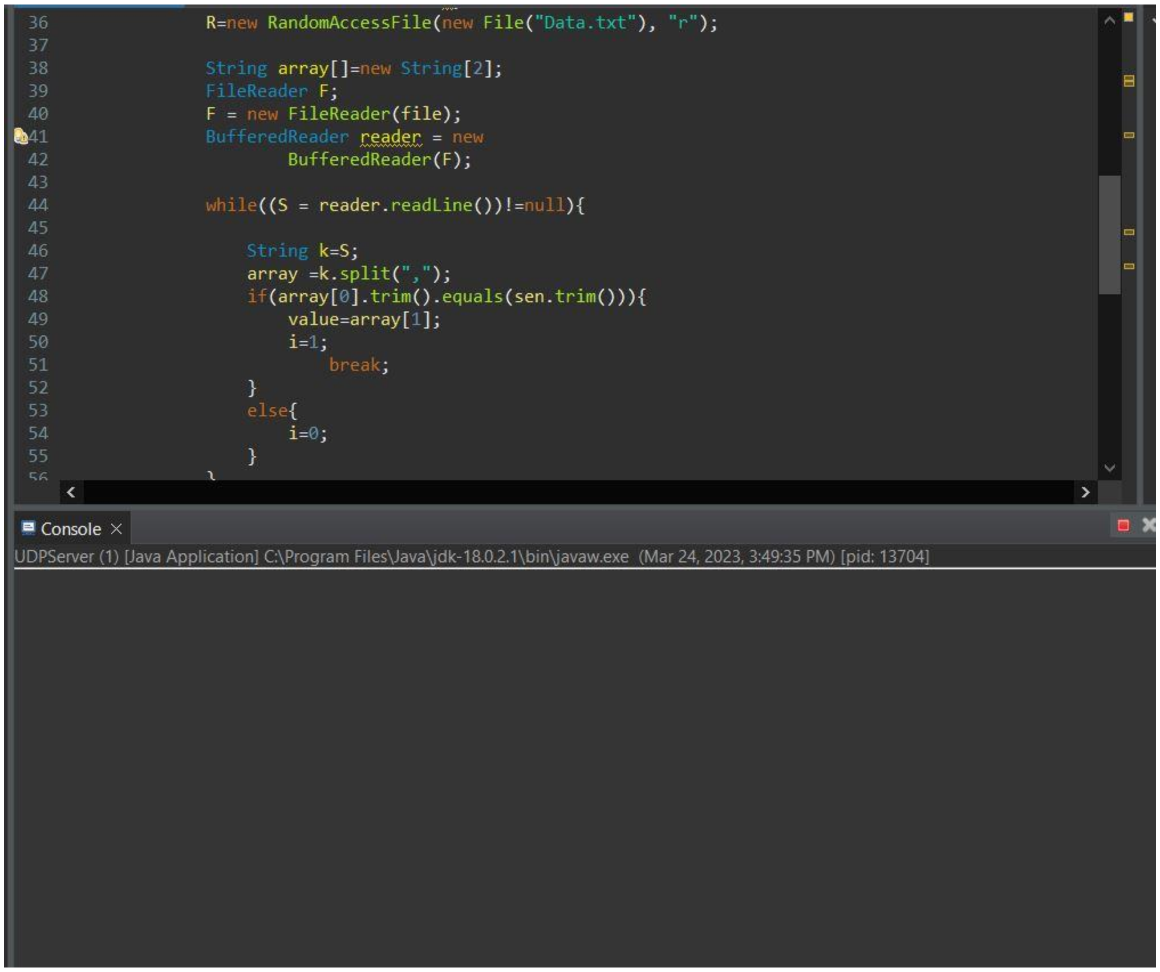
3. After creating the packet to receive the message from the client (which includes the message's size), receive the message, and store it in the string.

4. Obtain the sender's IP address and port number and enter them in the send backet (to make sure that the message arrive to the correct client).

5. After that, we use the file (the server has the file, and the file contains all the information), so we split the line in the file by (,), and then I loop through the file until I reach the id (the message from the client), at which point we store the name, of course, in an array of type byte and send it to the client.

Apply the UDB Code:

1. first I run the code of server in the first computer, and the server wait until the client make connection:

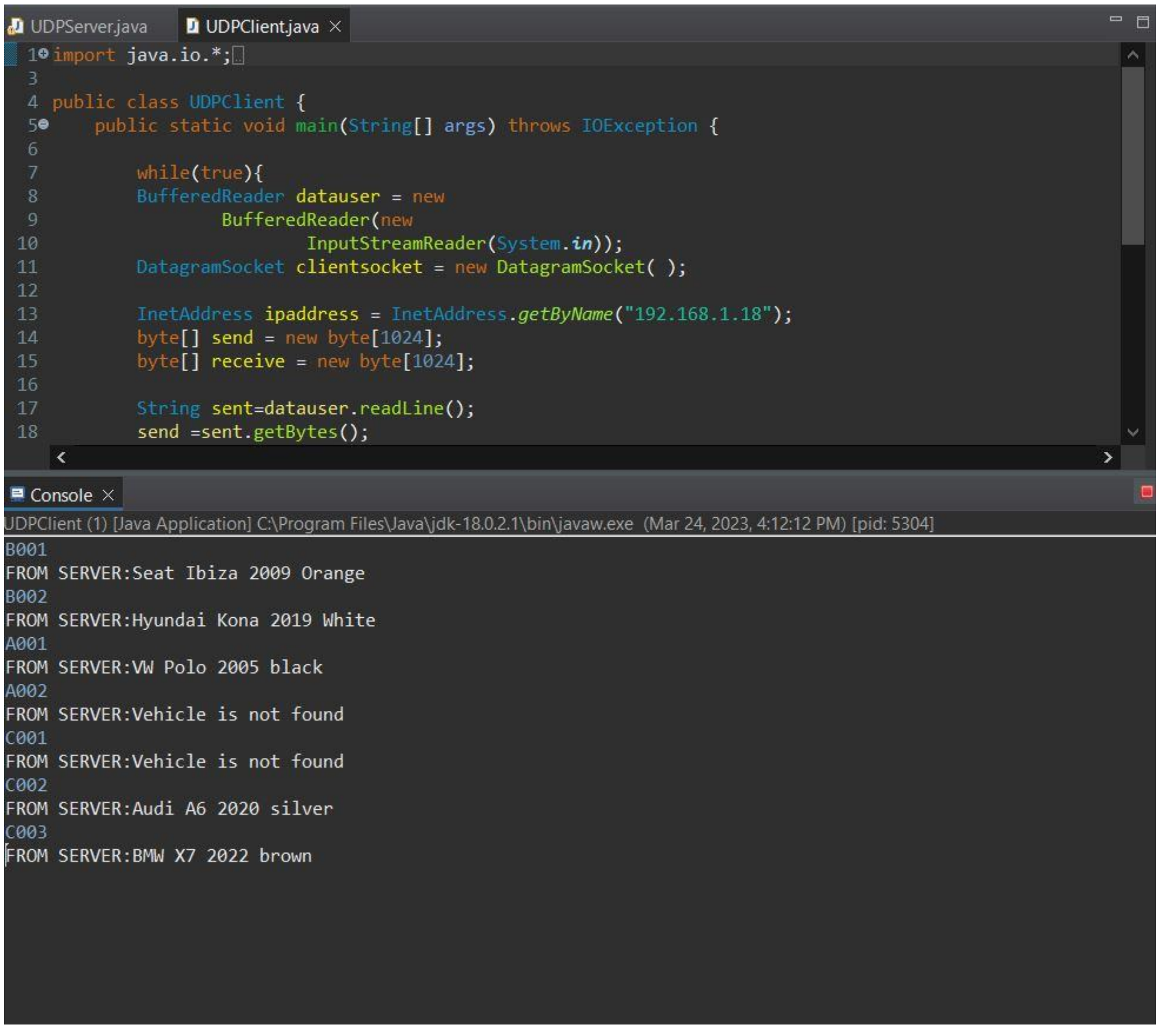


```
R=new RandomAccessFile(new File("Data.txt"), "r");

String array[]=new String[2];
FileReader F;
F = new FileReader(file);
BufferedReader reader = new
        BufferedReader(F);

while((S = reader.readLine())!=null){

    String k=S;
    array =k.split(",");
    if(array[0].trim().equals(sen.trim())){
        value=array[1];
        i=1;
            break;
    }
    else{
        i=0;
    }
}
```

Console

```
UDPServer (1) [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe  (Mar 24, 2023, 3:49:35 PM) [pid: 13704]
```

2. run the client code and put the data then the server gives me the expected response like:

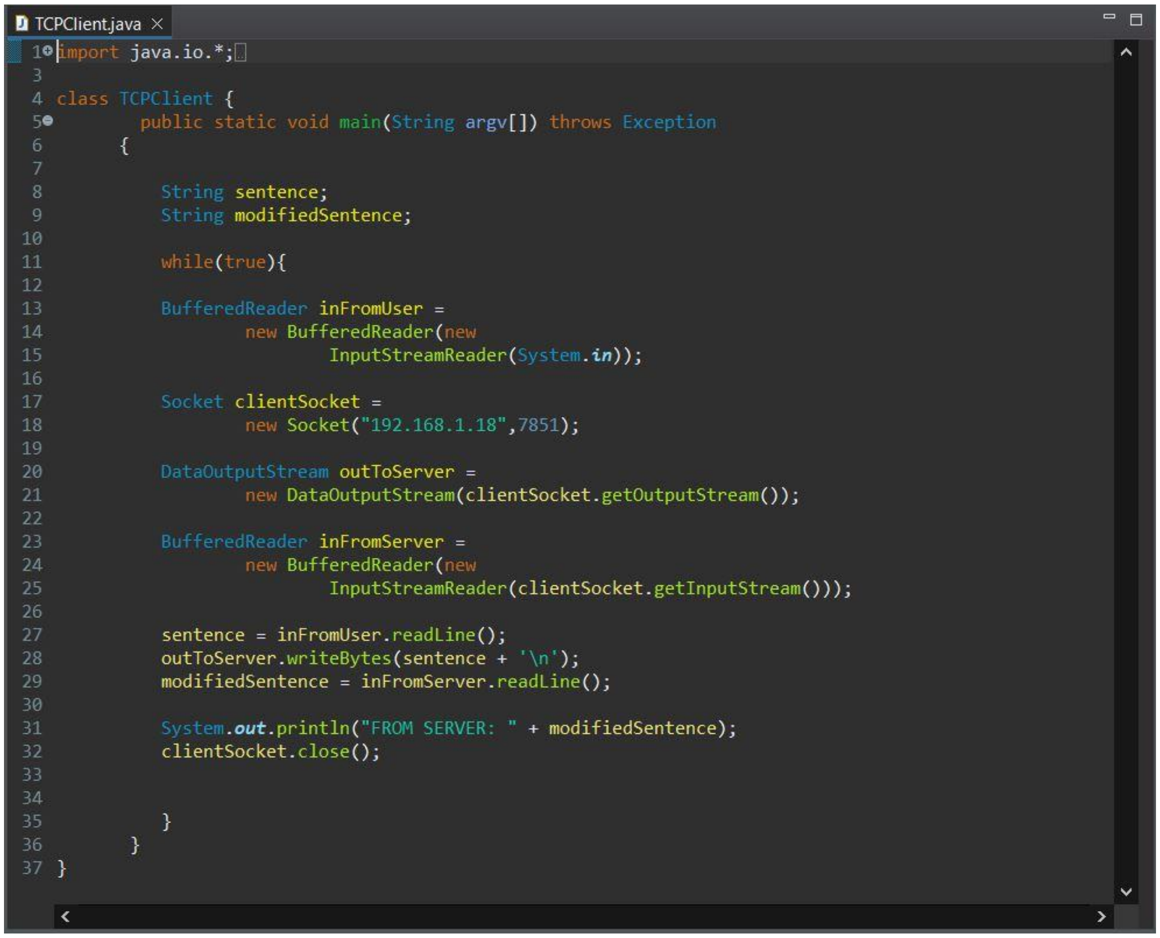


```java
import java.io.*;

public class UDPClient {
    public static void main(String[] args) throws IOException {

        while(true){
        BufferedReader datauser = new
                BufferedReader(new
                        InputStreamReader(System.in));
        DatagramSocket clientsocket = new DatagramSocket( );

        InetAddress ipaddress = InetAddress.getByName("192.168.1.18");
        byte[] send = new byte[1024];
        byte[] receive = new byte[1024];

        String sent=datauser.readLine();
        send =sent.getBytes();
```

```
UDPClient (1) [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe  (Mar 24, 2023, 4:12:12 PM) [pid: 5304]
B001
FROM SERVER:Seat Ibiza 2009 Orange
B002
FROM SERVER:Hyundai Kona 2019 White
A001
FROM SERVER:VW Polo 2005 black
A002
FROM SERVER:Vehicle is not found
C001
FROM SERVER:Vehicle is not found
C002
FROM SERVER:Audi A6 2020 silver
C003
FROM SERVER:BMW X7 2022 brown
```

Socket programing by TCP:

TCP Client code:



TCPClient.java

```java
import java.io.*;

class TCPClient {
        public static void main(String argv[]) throws Exception
        {

            String sentence;
            String modifiedSentence;

            while(true){

            BufferedReader inFromUser =
                    new BufferedReader(new
                            InputStreamReader(System.in));

            Socket clientSocket =
                    new Socket("192.168.1.18",7851);

            DataOutputStream outToServer =
                    new DataOutputStream(clientSocket.getOutputStream());

            BufferedReader inFromServer =
                    new BufferedReader(new
                            InputStreamReader(clientSocket.getInputStream()));

            sentence = inFromUser.readLine();
            outToServer.writeBytes(sentence + '\n');
            modifiedSentence = inFromServer.readLine();

            System.out.println("FROM SERVER: " + modifiedSentence);
            clientSocket.close();


            }
        }
}
```

TCP Client code to explain:

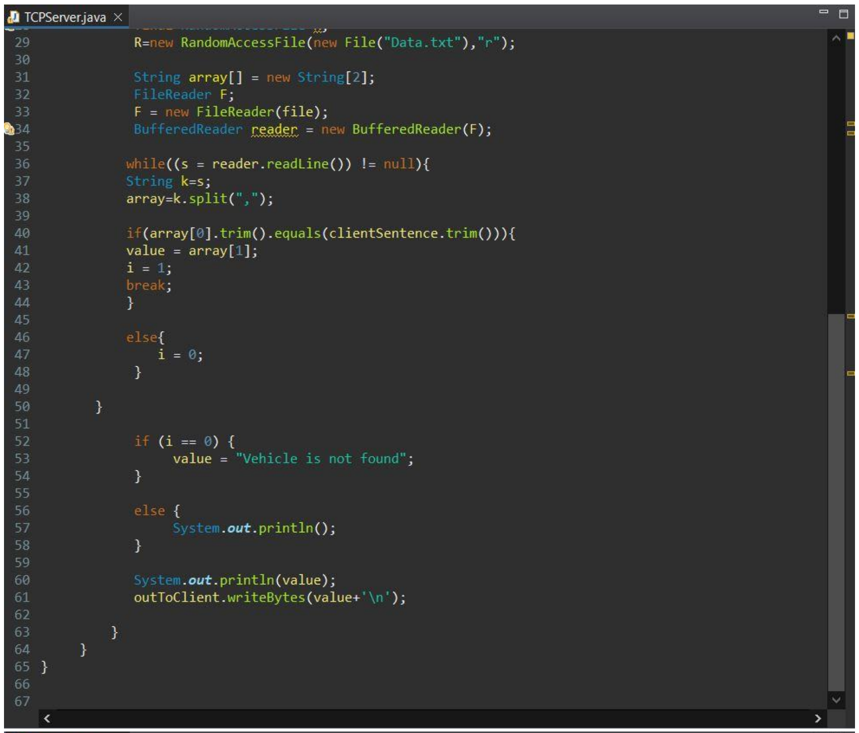
1. we create the Stream that enables the user to enter data, then we create the socket for the client and establish a connection with the server by entering the server's IP address and port number, and last we create the output and input Strems.

2. Save and deliver the user-inputted data to the server.

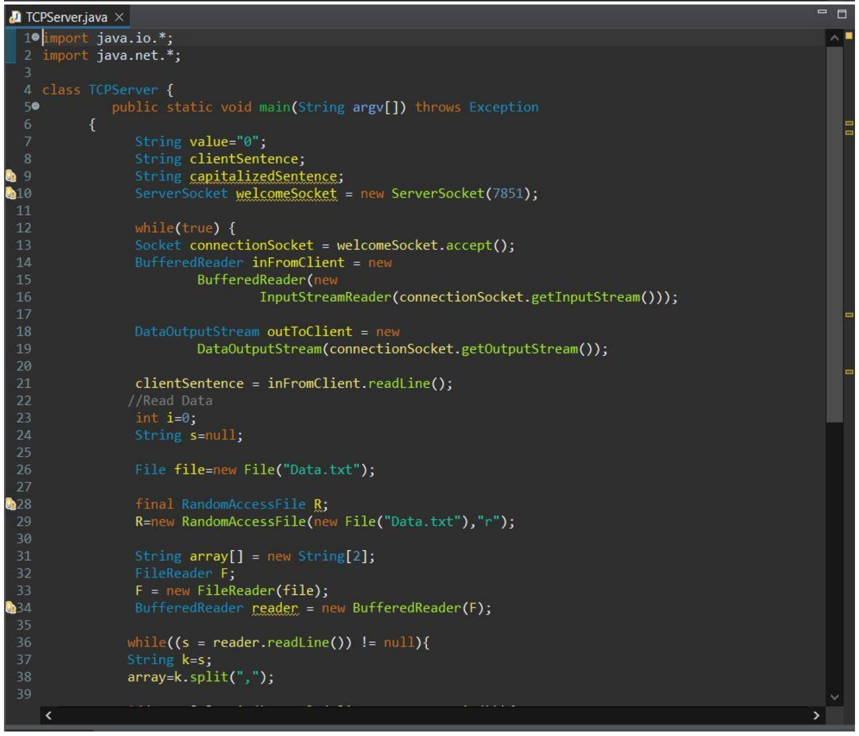
3. print the message you downloaded from the server (receive message).

4. shut down the connection once we have sent and received all the info we require.

TCP Server code:



```
            R=new RandomAccessFile(new File("Data.txt"),"r");

            String array[] = new String[2];
            FileReader F;
            F = new FileReader(file);
            BufferedReader reader = new BufferedReader(F);

           while((s = reader.readLine()) != null){
           String k=s;
           array=k.split(",");

           if(array[0].trim().equals(clientSentence.trim())){
           value = array[1];
           i = 1;
           break;
           }

           else{
               i = 0;
            }

       }

            if (i == 0) {
                 value = "Vehicle is not found";
            }

            else {
                 System.out.println();
            }

            System.out.println(value);
            outToClient.writeBytes(value+'\n');

         }
     }
 }
```



```
import java.io.*;
import java.net.*;

class TCPServer {
         public static void main(String argv[]) throws Exception
      {
            String value="0";
            String clientSentence;
            String capitalizedSentence;
            ServerSocket welcomeSocket = new ServerSocket(7851);

            while(true) {
            Socket connectionSocket = welcomeSocket.accept();
            BufferedReader inFromClient = new
                    BufferedReader(new
                            InputStreamReader(connectionSocket.getInputStream()));

            DataOutputStream outToClient = new
                    DataOutputStream(connectionSocket.getOutputStream());

            clientSentence = inFromClient.readLine();
           //Read Data
            int i=0;
            String s=null;

            File file=new File("Data.txt");

            final RandomAccessFile R;
            R=new RandomAccessFile(new File("Data.txt"),"r");

            String array[] = new String[2];
            FileReader F;
            F = new FileReader(file);
            BufferedReader reader = new BufferedReader(F);

           while((s = reader.readLine()) != null){
           String k=s;
           array=k.split(",");
```

TCP Server code to explain:

1. Before creating the input stream and output stream to accept the message from the client and resend it, we will first construct the socket for the server and wait till the client sends the message.

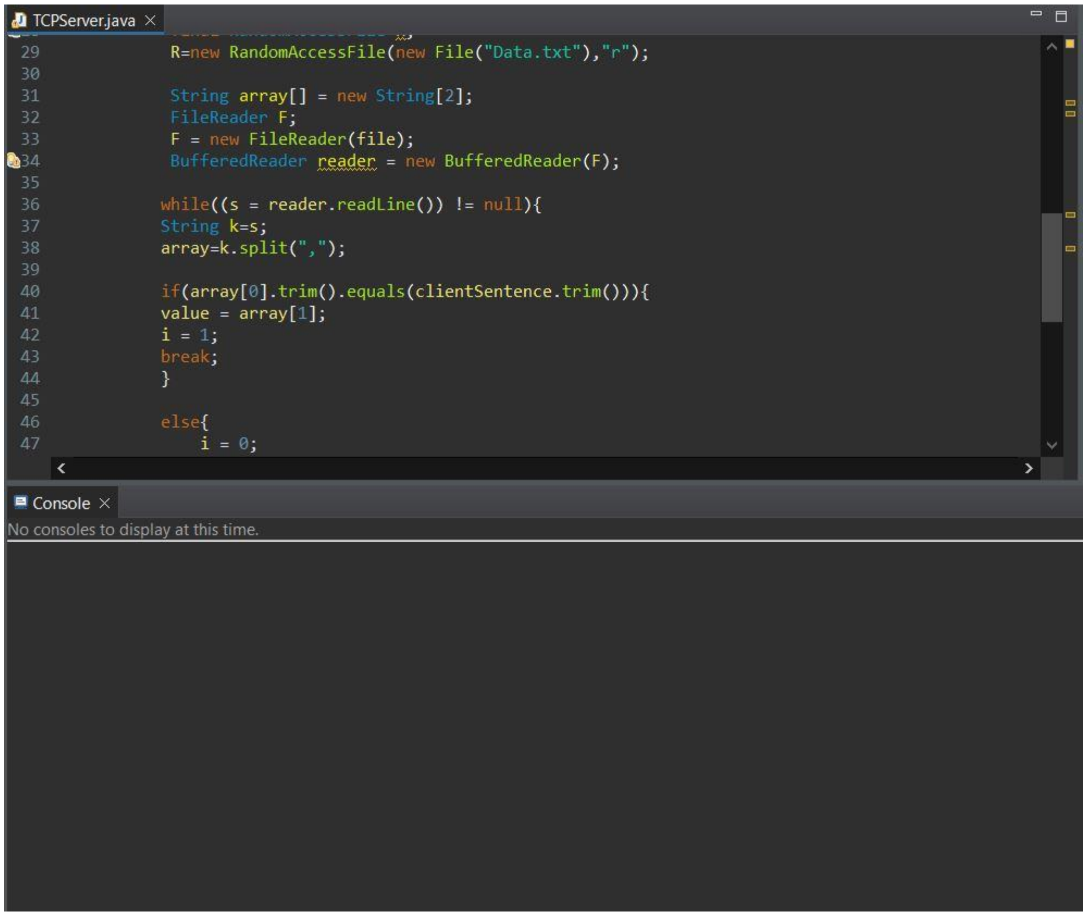
2. after the client establishes a connection with the server, the server reads this message, which we then store in the string.

3. after this, we split the line in the file by (,)and search data by the client message (which we had previously stored in the string).

4. after this, we insert the data in the string and send it to the client.

Apply the TCP Code:

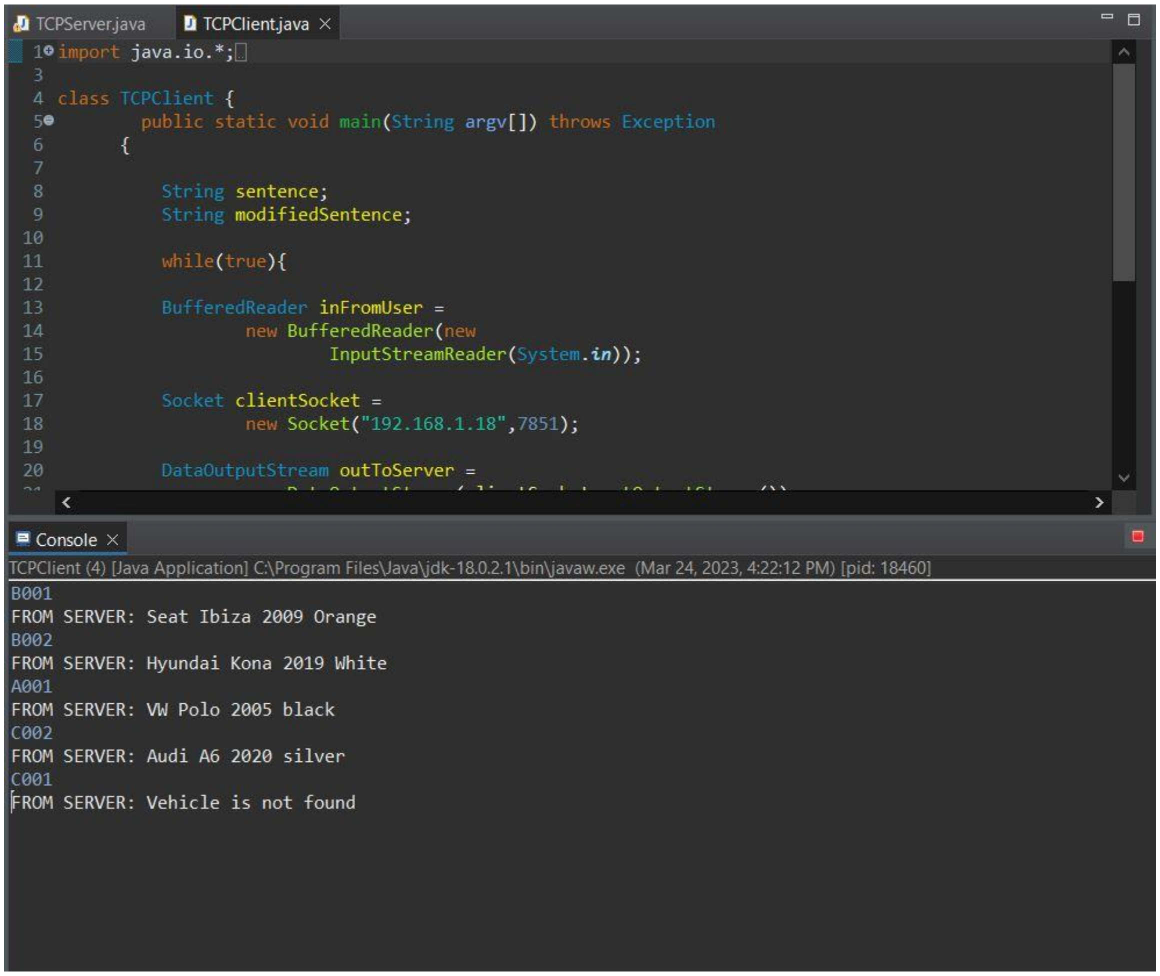
1. first I run the code of server in the first computer, and the server wait until the client make connection:



```java
             R=new RandomAccessFile(new File("Data.txt"),"r");

             String array[] = new String[2];
             FileReader F;
             F = new FileReader(file);
             BufferedReader reader = new BufferedReader(F);

            while((s = reader.readLine()) != null){
            String k=s;
            array=k.split(",");

            if(array[0].trim().equals(clientSentence.trim())){
            value = array[1];
            i = 1;
            break;
            }

            else{
                i = 0;
```

```
No consoles to display at this time.
```

2. run the client code and put the data then the server gives me the expected response like:



```java
import java.io.*;

class TCPClient {
        public static void main(String argv[]) throws Exception
    {

        String sentence;
        String modifiedSentence;

        while(true){

        BufferedReader inFromUser =
                new BufferedReader(new
                        InputStreamReader(System.in));

        Socket clientSocket =
                new Socket("192.168.1.18",7851);

        DataOutputStream outToServer =
```

```
TCPClient (4) [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe  (Mar 24, 2023, 4:22:12 PM) [pid: 18460]
B001
FROM SERVER: Seat Ibiza 2009 Orange
B002
FROM SERVER: Hyundai Kona 2019 White
A001
FROM SERVER: VW Polo 2005 black
C002
FROM SERVER: Audi A6 2020 silver
C001
FROM SERVER: Vehicle is not found
```