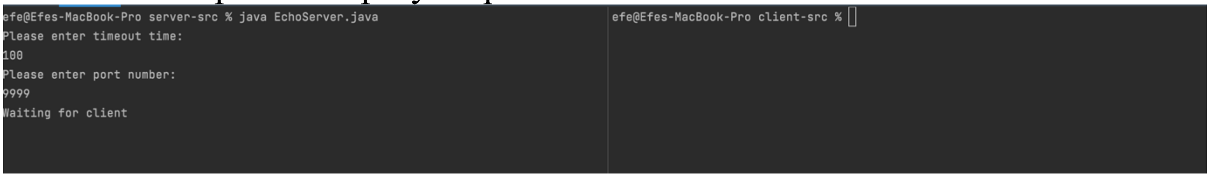
COMP 416: COMPUTER NETWORKS
PROJECT 1 PART 1
REPORT

EFE ERTEM
71739

OVERVIEW

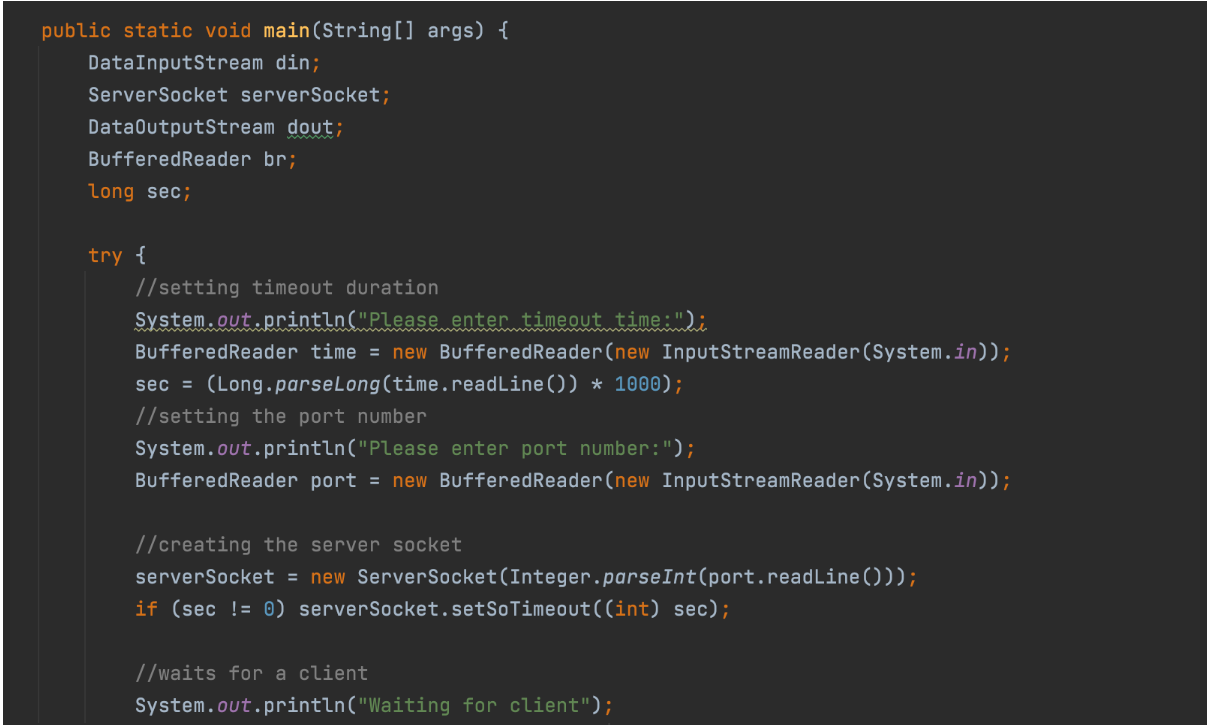
This overview will include snapshot of the app and explaining the processes with these snapshots step-by-step.



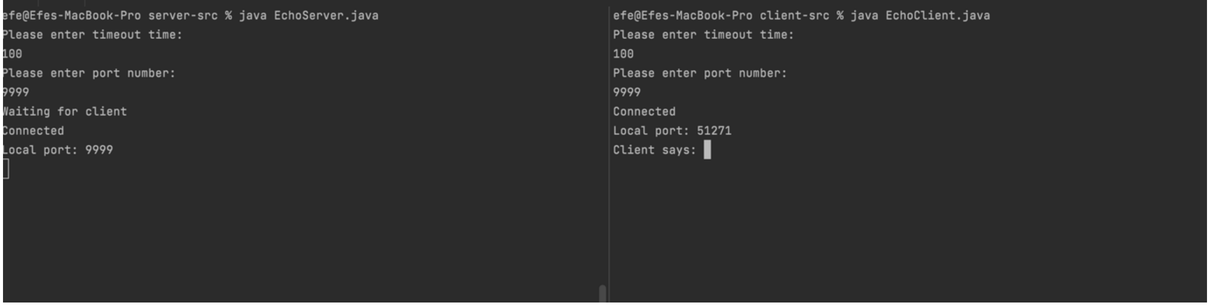
```
efe@Efes-MacBook-Pro server-src % java EchoServer.java
Please enter timeout time:
100
Please enter port number:
9999
Waiting for client
```

```
efe@Efes-MacBook-Pro client-src % 
```

Step 1: Run the program with $java EchoServer.java
Step 2: Enter a value for duration before timeout (in seconds)
Step 3: Enter a value for port number (integer)
Step 4: Waits for the client socket
With these inputs the program takes the values creates a server socket as seen in the snapshot below.



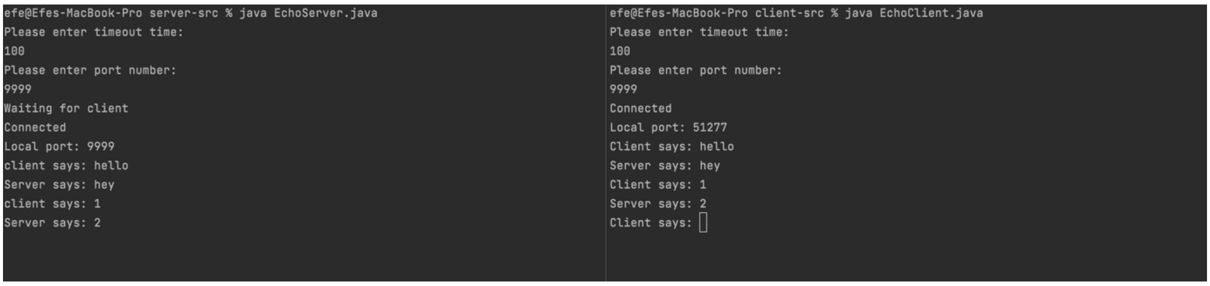
```
public static void main(String[] args) {
    DataInputStream din;
    ServerSocket serverSocket;
    DataOutputStream dout;
    BufferedReader br;
    long sec;

    try {
        //setting timeout duration
        System.out.println("Please enter timeout time:");
        BufferedReader time = new BufferedReader(new InputStreamReader(System.in));
        sec = (Long.parseLong(time.readLine()) * 1000);
        //setting the port number
        System.out.println("Please enter port number:");
        BufferedReader port = new BufferedReader(new InputStreamReader(System.in));

        //creating the server socket
        serverSocket = new ServerSocket(Integer.parseInt(port.readLine()));
        if (sec != 0) serverSocket.setSoTimeout((int) sec);

        //waits for a client
        System.out.println("Waiting for client");
```



```
efe@Efes-MacBook-Pro server-src % java EchoServer.java
Please enter timeout time:
100
Please enter port number:
9999
Waiting for client
Connected
Local port: 9999
```

```
efe@Efes-MacBook-Pro client-src % java EchoClient.java
Please enter timeout time:
100
Please enter port number:
9999
Connected
Local port: 51271
Client says: 
```

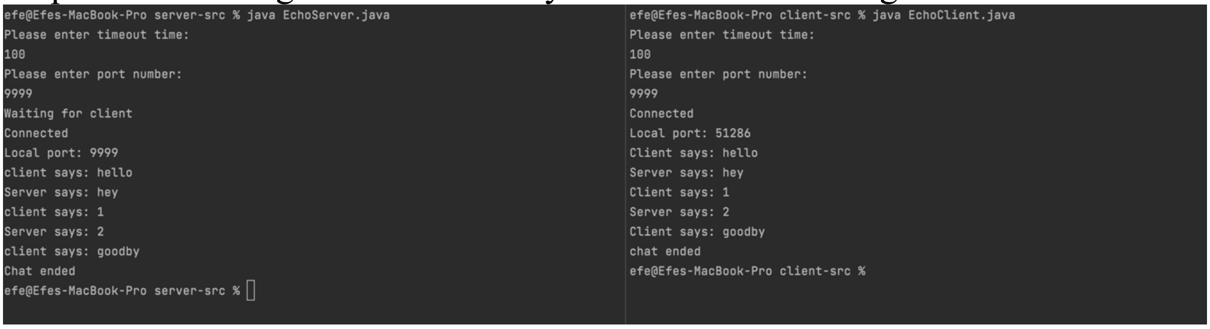
Step 4: Do steps 1-3 for client file (using $java EchoClient.java to run the program)
Step 5: Once the connection is established, local ports are shown on the console.
Step 6: The app waits for the client to start messaging.



```
efe@Efes-MacBook-Pro server-src % java EchoServer.java
Please enter timeout time:
100
Please enter port number:
9999
Waiting for client
Connected
Local port: 9999
client says: hello
Server says: hey
client says: 1
Server says: 2
```

```
efe@Efes-MacBook-Pro client-src % java EchoClient.java
Please enter timeout time:
100
Please enter port number:
9999
Connected
Local port: 51277
Client says: hello
Server says: hey
Client says: 1
Server says: 2
Client says: 
```

Step 7: The messages are sent one-by-one as seen on the figure above.

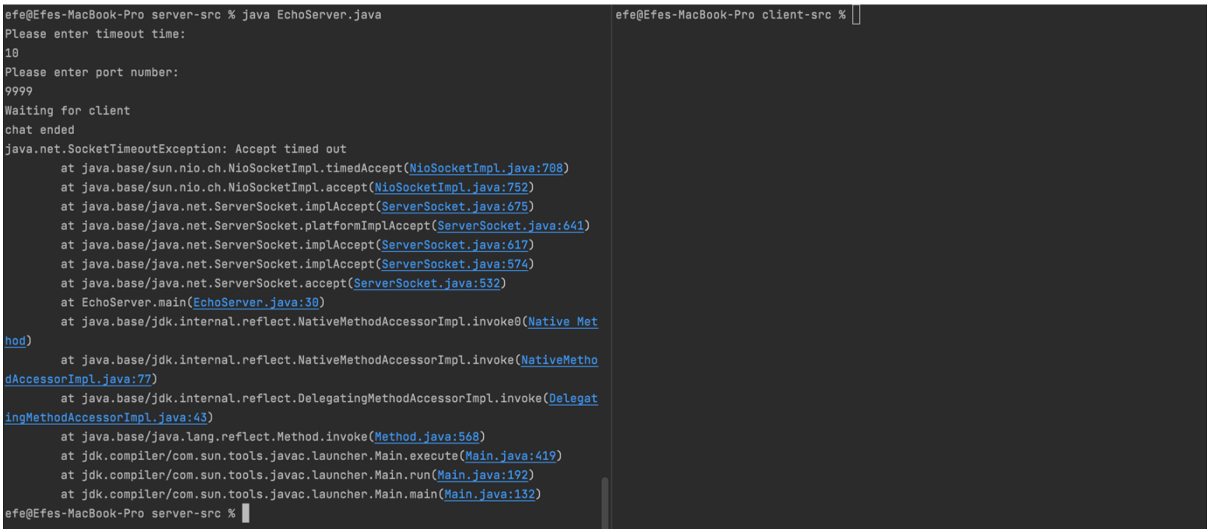


```
efe@Efes-MacBook-Pro server-src % java EchoServer.java
Please enter timeout time:
100
Please enter port number:
9999
Waiting for client
Connected
Local port: 9999
client says: hello
Server says: hey
client says: 1
Server says: 2
client says: goodby
Chat ended
efe@Efes-MacBook-Pro server-src % 
```

```
efe@Efes-MacBook-Pro client-src % java EchoClient.java
Please enter timeout time:
100
Please enter port number:
9999
Connected
Local port: 51286
Client says: hello
Server says: hey
Client says: 1
Server says: 2
Client says: goodby
chat ended
efe@Efes-MacBook-Pro client-src %
```

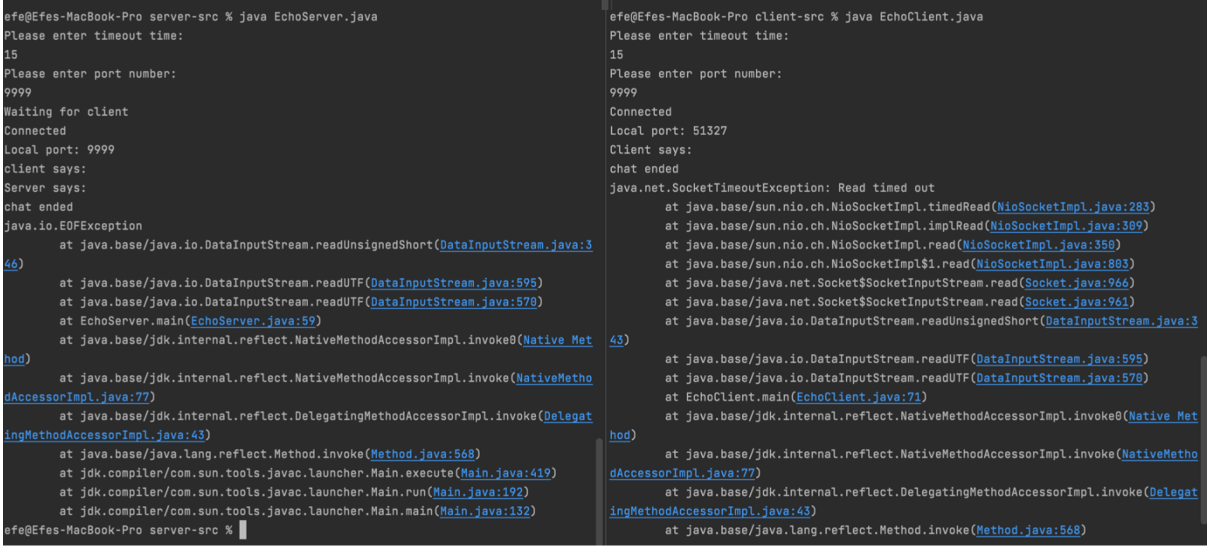
Step 8: Chat ends with the message "goodby"

**Timeout cases are shown below



```
efe@Efes-MacBook-Pro server-src % java EchoServer.java
Please enter timeout time:
10
Please enter port number:
9999
Waiting for client
chat ended
java.net.SocketTimeoutException: Accept timed out
	at java.base/sun.nio.ch.NioSocketImpl.timedAccept(NioSocketImpl.java:708)
	at java.base/sun.nio.ch.NioSocketImpl.accept(NioSocketImpl.java:752)
	at java.base/java.net.ServerSocket.implAccept(ServerSocket.java:675)
	at java.base/java.net.ServerSocket.platformImplAccept(ServerSocket.java:641)
	at java.base/java.net.ServerSocket.implAccept(ServerSocket.java:617)
	at java.base/java.net.ServerSocket.implAccept(ServerSocket.java:574)
	at java.base/java.net.ServerSocket.accept(ServerSocket.java:532)
	at EchoServer.main(EchoServer.java:30)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Met
hod)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMetho
dAccessorImpl.java:77)
	at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegat
ingMethodAccessorImpl.java:43)
	at java.base/java.lang.reflect.Method.invoke(Method.java:568)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.execute(Main.java:419)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.run(Main.java:192)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.main(Main.java:132)
efe@Efes-MacBook-Pro server-src % 
```

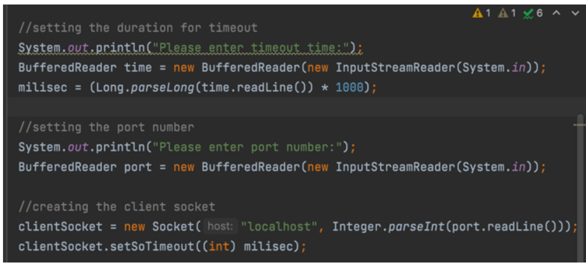
```
efe@Efes-MacBook-Pro client-src % 
```



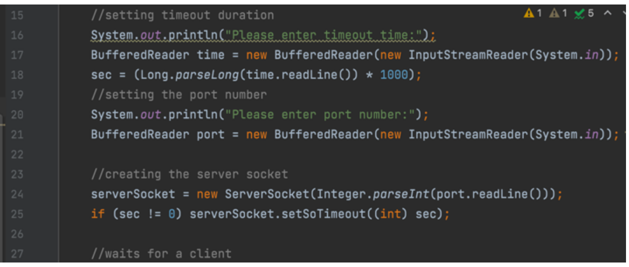
```
efe@Efes-MacBook-Pro server-src % java EchoServer.java
Please enter timeout time:
15
Please enter port number:
9999
Waiting for client
Connected
Local port: 9999
client says:
Server says:
chat ended
java.io.EOFException
	at java.base/java.io.DataInputStream.readUnsignedShort(DataInputStream.java:3
46)
	at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:595)
	at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:570)
	at EchoServer.main(EchoServer.java:59)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Met
hod)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMetho
dAccessorImpl.java:77)
	at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegat
ingMethodAccessorImpl.java:43)
	at java.base/java.lang.reflect.Method.invoke(Method.java:568)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.execute(Main.java:419)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.run(Main.java:192)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.main(Main.java:132)
efe@Efes-MacBook-Pro server-src % 
```

```
efe@Efes-MacBook-Pro client-src % java EchoClient.java
Please enter timeout time:
15
Please enter port number:
9999
Connected
Local port: 51327
Client says:
chat ended
java.net.SocketTimeoutException: Read timed out
	at java.base/sun.nio.ch.NioSocketImpl.timedRead(NioSocketImpl.java:283)
	at java.base/sun.nio.ch.NioSocketImpl.implRead(NioSocketImpl.java:309)
	at java.base/sun.nio.ch.NioSocketImpl.read(NioSocketImpl.java:350)
	at java.base/sun.nio.ch.NioSocketImpl$1.read(NioSocketImpl.java:803)
	at java.base/java.net.Socket$SocketInputStream.read(Socket.java:966)
	at java.base/java.net.Socket$SocketInputStream.read(Socket.java:961)
	at java.base/java.io.DataInputStream.readUnsignedShort(DataInputStream.java:3
43)
	at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:595)
	at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:570)
	at EchoClient.main(EchoClient.java:71)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Met
hod)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMetho
dAccessorImpl.java:77)
	at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegat
ingMethodAccessorImpl.java:43)
	at java.base/java.lang.reflect.Method.invoke(Method.java:568)
```

-standard implementation for socket timeout.
-works well for server socket
-does not work for the client socket

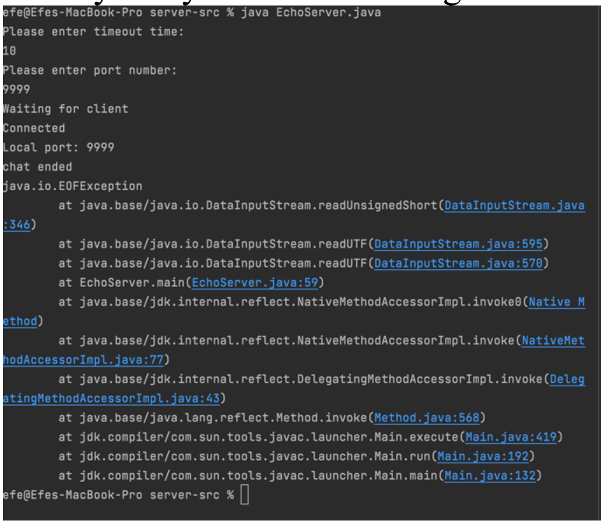


```
//setting the duration for timeout
System.out.println("Please enter timeout time:");
BufferedReader time = new BufferedReader(new InputStreamReader(System.in));
milisec = (Long.parseLong(time.readLine()) * 1000);

//setting the port number
System.out.println("Please enter port number:");
BufferedReader port = new BufferedReader(new InputStreamReader(System.in));

//creating the client socket
clientSocket = new Socket( host: "localhost", Integer.parseInt(port.readLine()));
clientSocket.setSoTimeout((int) milisec);
```

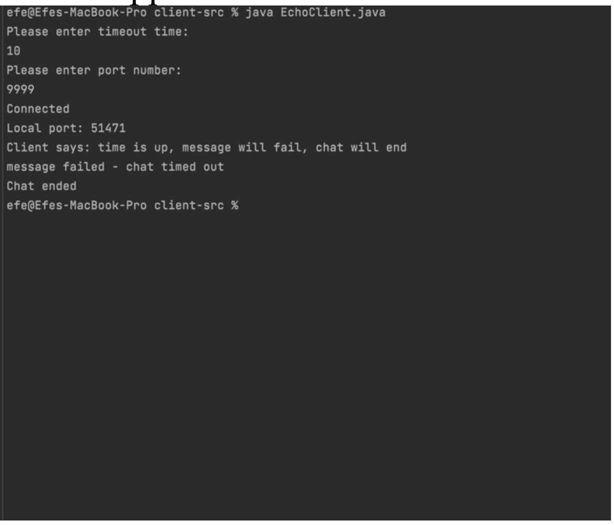


```
//setting timeout duration
System.out.println("Please enter timeout time:");
BufferedReader time = new BufferedReader(new InputStreamReader(System.in));
sec = (Long.parseLong(time.readLine()) * 1000);
//setting the port number
System.out.println("Please enter port number:");
BufferedReader port = new BufferedReader(new InputStreamReader(System.in));

//creating the server socket
serverSocket = new ServerSocket(Integer.parseInt(port.readLine()));
if (sec != 0) serverSocket.setSoTimeout((int) sec);

//waits for a client
```

-In the case below, the client socket had trouble with timeout so I implemented a different kind of solution for this. After the timeout duration is exceeded, when you try to send a message it fails and the app closes.

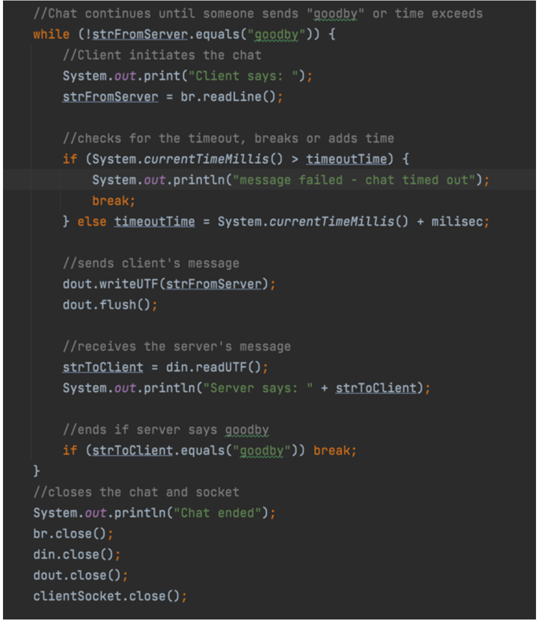


```
efe@Efes-MacBook-Pro server-src % java EchoServer.java
Please enter timeout time:
10
Please enter port number:
9999
Waiting for client
Connected
Local port: 9999
chat ended
java.io.EOFException
	at java.base/java.io.DataInputStream.readUnsignedShort(DataInputStream.java
:346)
	at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:595)
	at java.base/java.io.DataInputStream.readUTF(DataInputStream.java:570)
	at EchoServer.main(EchoServer.java:59)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native M
ethod)
	at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMet
hodAccessorImpl.java:77)
	at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Deleg
atingMethodAccessorImpl.java:43)
	at java.base/java.lang.reflect.Method.invoke(Method.java:568)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.execute(Main.java:419)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.run(Main.java:192)
	at jdk.compiler/com.sun.tools.javac.launcher.Main.main(Main.java:132)
efe@Efes-MacBook-Pro server-src % 
```



```
efe@Efes-MacBook-Pro client-src % java EchoClient.java
Please enter timeout time:
10
Please enter port number:
9999
Connected
Local port: 51471
Client says: time is up, message will fail, chat will end
message failed - chat timed out
Chat ended
efe@Efes-MacBook-Pro client-src %
```

-snapshot of the implementation is below



```
//Chat continues until someone sends "goodby" or time exceeds
while (!strFromServer.equals("goodby")) {
    //Client initiates the chat
    System.out.print("Client says: ");
    strFromServer = br.readLine();

    //checks for the timeout, breaks or adds time
    if (System.currentTimeMillis() > timeoutTime) {
        System.out.println("message failed - chat timed out");
        break;
    } else timeoutTime = System.currentTimeMillis() + milisec;

    //sends client's message
    dout.writeUTF(strFromServer);
    dout.flush();

    //receives the server's message
    strToClient = din.readUTF();
    System.out.println("Server says: " + strToClient);

    //ends if server says goodby
    if (strToClient.equals("goodby")) break;
}
//closes the chat and socket
System.out.println("Chat ended");
br.close();
din.close();
dout.close();
clientSocket.close();
```