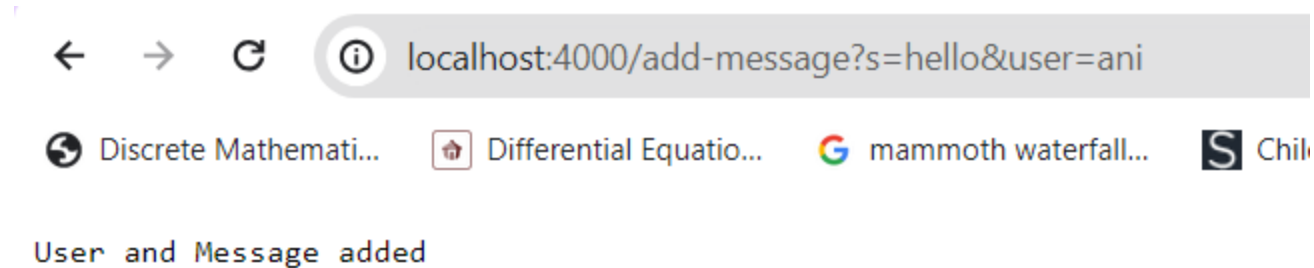
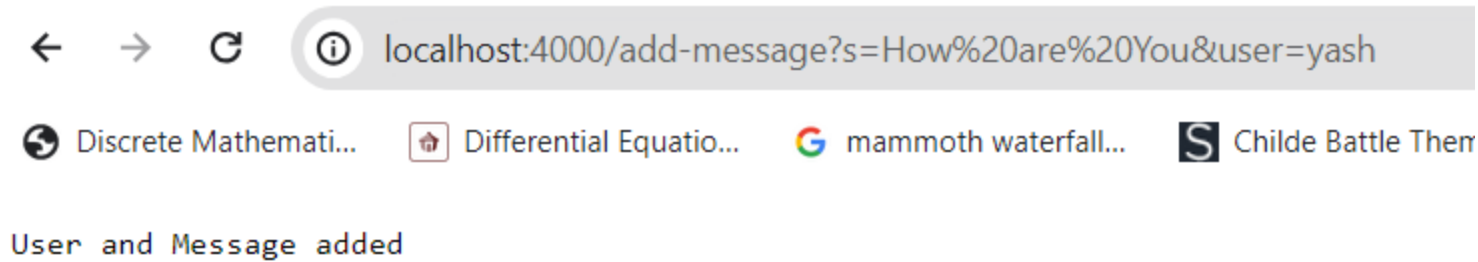


cse-15L-report

Part 1



I used the NumberServer.java template from Week 2 lab and modify it to accept strings. I don't necessarily use a method but the handler method which is in the main method of the ChatServer. I think the relevant arguments are the queries because we're outputting the words inputted after the "=". As for values when I put in an integer after the =, it converts it into a string and still outputs it.



When I added another message and user, it was added to the same ArrayList that contains the first pair of messages and users. It shares the same relevant arguments as the first pair of messages and users. It can also takes in integer as a string and still outputs it.

```
import java.io.IOException;
import java.net.URI;
import java.util.*;
class Handler implements URLHandler{
    ArrayList<String> message = new ArrayList<String>();
    ArrayList<String> user = new ArrayList<String>();
    public String handleRequest(URI url){
        if(url.getPath().equals("/")){ //detects path
            if(user.size() == 0 && message.size() == 0){
                return "Nothing";
            }else{

                StringBuilder result = new StringBuilder();
                //iterating through the messages
                for(int i = 0; i< message.size(); i++){
                    result.append(user.get(i)).append(" : ").append(message.get(i)).append("\n");
                }
                return result.toString();
            }
        }else{
            if(url.getPath().contains("/add-message")){
```

```
String query = url.getQuery(); //string contains the query after the ?
String[] parameters = query.split("&"); //splits the message and user
for (String words : parameters) { //iterate through the two elements (message and user) in parameters[]
    String[] userInput = words.split("=");
    if (userInput[0].equals("s")) { //words inputed after s
        message.add(userInput[1]);
    } else {
        if (userInput[0].equals("user")){ //words inputed after user
            user.add(userInput[1]);
        }
    }
}
return "User and Message added";
}
return "404 Not Found:";
}
}

}

class ChatServer {
    public static void main(String[] args) throws IOException {
        if(args.length == 0){
            System.out.println("Missing port number! Try any number between 1024 to 49151");
            return;
        }

        int port = Integer.parseInt(args[0]);

        Server.start(port, new Handler());
    }
}
```

Part 2

```
15106@LAPTOP-ENK5V0L1 MINGW64 ~/Documents/GitHub/cse-15L-report/lab-report3 (main)
$ ls /c/Users/15106/.ssh/id_ed25519.pub
/c/Users/15106/.ssh/id_ed25519.pub
```

```
15106@LAPTOP-ENK5V0L1 MINGW64 ~/Documents/GitHub/cse-15L-report/lab-report3 (main)
$ ls /c/Users/15106/.ssh/id_ed25519.pub
/c/Users/15106/.ssh/id_ed25519.pub
```

```
15106@LAPTOP-ENK5V0L1 MINGW64 ~/Documents/GitHub/cse-15L-report/lab-report3 (main)
$ ssh anw031@ieng6.ucsd.edu
Last login: Sat Jan 27 13:12:11 2024 from 128.54.131.220
Hello anw031, you are currently logged into ieng6-202.ucsd.edu
```

You are using 0% CPU on this system

Cluster Status

Hostname	Time	#Users	Load	Averages
ieng6-201	13:10:01	14	0.95,	0.54, 0.68
ieng6-202	13:10:01	9	0.48,	0.20, 0.24
ieng6-203	13:10:01	19	0.39,	0.60, 0.56

To begin work for one of your courses [cs15lwi24], type its name at the command prompt. (For example, "cs15lwi24", without the quotes).

```
[anw031@ieng6-202]:~:89$ cs15wi24
```

```
bash: cs15wi24: command not found...
```

```
[anw031@ieng6-202]:~:90$ cs15lwi24
```

```
Sat Jan 27, 2024 1:13pm - Prepping cs15lwi24
```

```
[anw031@ieng6-202]:anw031:91$
```

Part 3

I didn't know that I could see all the files I downloaded in my download folder. I also didn't know that Java is capable of making a website server because I thought only JavaScript or any front-end/back-end languages could do that. I think it was cool that we could connect to the CSE computers and have access to our data. I wonder if it only works if we're in UCSD or somewhere else.