Socket Programming

Ninth Laboratory Report for CENG 3331

Submitted by

Michael Lanford

1816337

Computer Engineering

University of Houston-Clear Lake

Houston, Texas 77058

11/17/22

Abstract

The goal of this lab was to introduce us to socket programming by using java. During this lab, we used the command prompt to accomplish 3 different goals given to us in the lab manual while also using some code given to us.

Write-Up

Introduction

A socket is an endpoint of a communication link between two different programs and is bound to a port number. The best example of socket programming in the real world is internet browsers. The browsers use it to fetch web pages from a server and display them.

Task 1:

We began our experiment by going into the command prompt and compiling the DateServer.java and DateClient.java programs using the javac command. Following this, we ran the DateServer program using the java command and then opened a new command prompt. We ran the DateClient program using the java command in this new command prompt, which opened a text box. After using the ipconfig command to find the IP address on the machine, we put the IP address into the text box, and it changed to show the current time on the system. The screenshot of these commands can be found in the appendix under Figure 1 and the output of the text box can be found under Figure 2.

Task 2:

For task 2, we continued this experiment by opening up a new command prompt and compiling the CapitalizeServer.java and CapitalizeClient.java programs using the javac command. Following this, we ran the CapitalizeServer program using the java command and then opened a new command prompt. We ran the CapitalizeClient program using the java command in this new command prompt, which prompted us to enter our IP address. After entering the IP, we were allowed to type in sentences that would then be

automatically capitalized after pressing the enter key. The screenshot of these commands and the output can be found in the appendix under Figure 3.

Task 3:

For this final task, we concluded this experiment by opening a new command prompt and compiling the Chat Server.java and ChatClient.java programs using the javac command. Following this, we ran the ChatServer program using the java command and then opened up 3 new command prompts. In each of these, we ran the ChatClient program using the java command, which opened up a text box. In each text box, we entered the IP address after finding it using the ipconfig command, and then chose a screen name for each. This enabled us to be able to use each chat box to talk to the other two. The screenshot of these commands can be seen in Figure 4 of the appendix while the output of this can be seen in Figure 5. Finally, we answered a question given to us in the lab manual. Socket programming works by having two endpoints, each with an address, be connected after one end initiates a connection to the other, which begins communication. After this, data can be exchanged until done and the endpoints can be closed.

Appendix

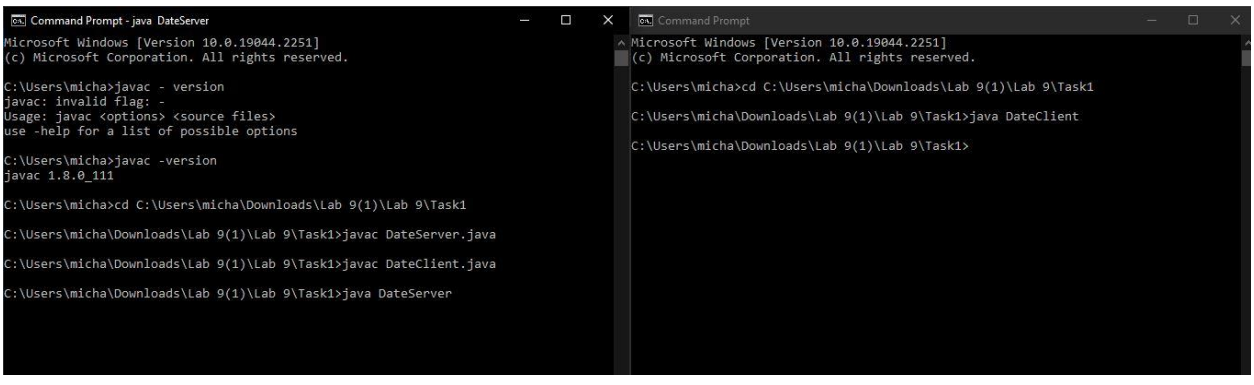

Fig. 1. Task 1 Command Windows

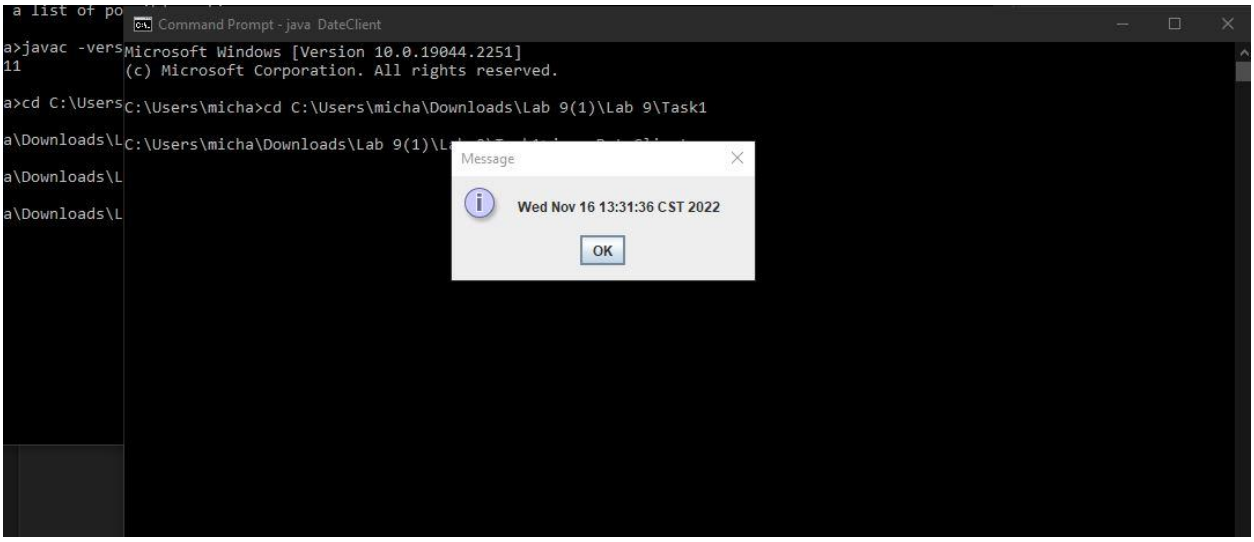

Fig. 2. Task 1 Output

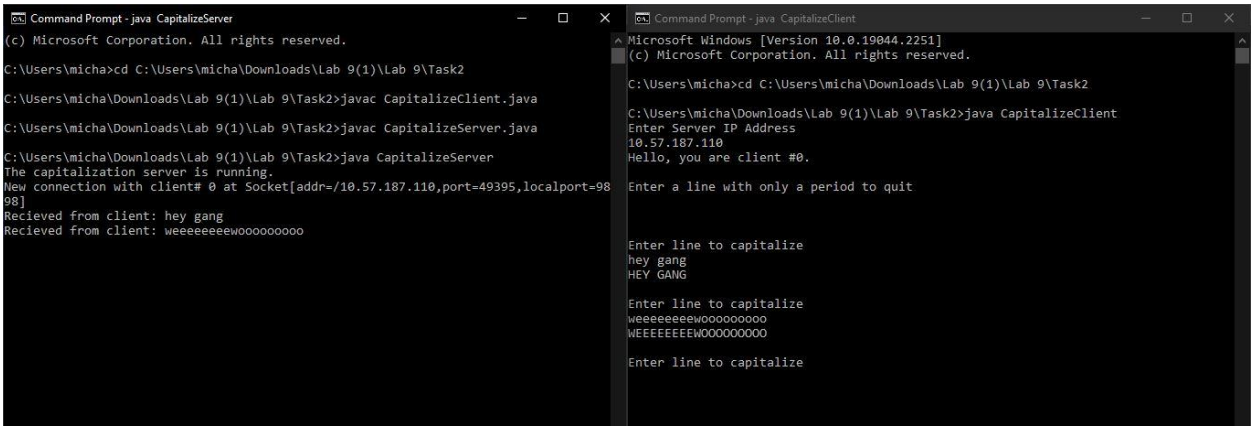

Fig. 3. Task 2 Command Windows and Output

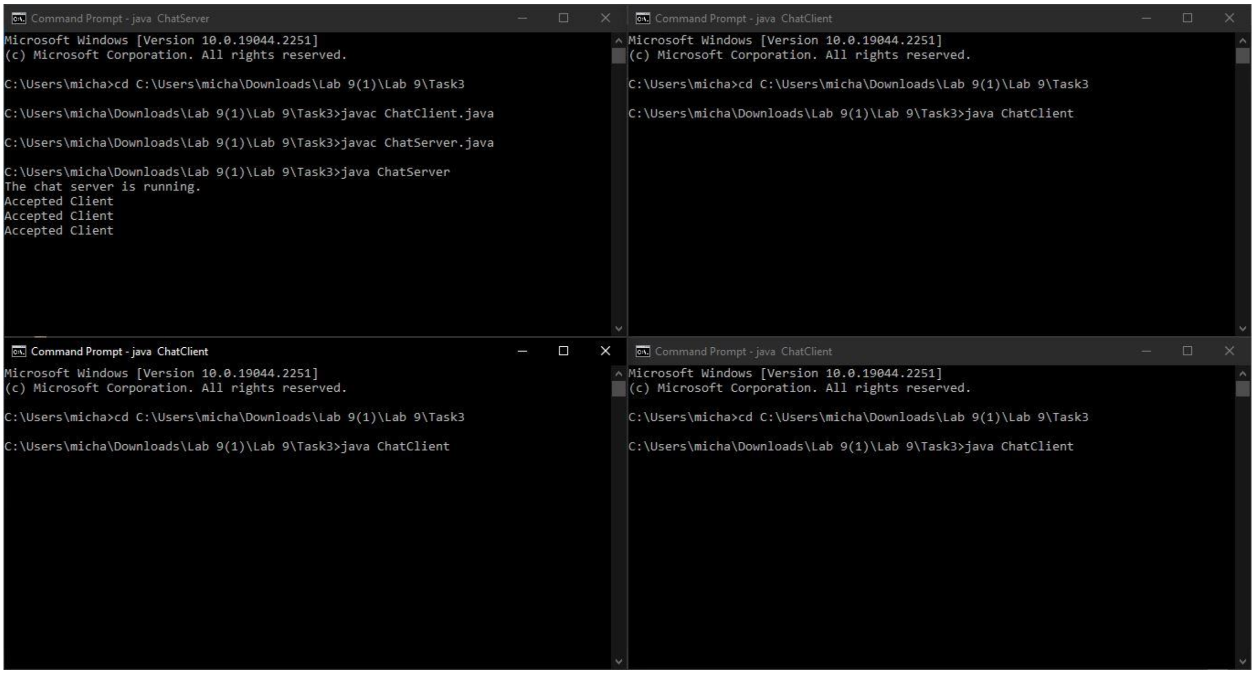

Fig. 4. Task 3 Command Windows

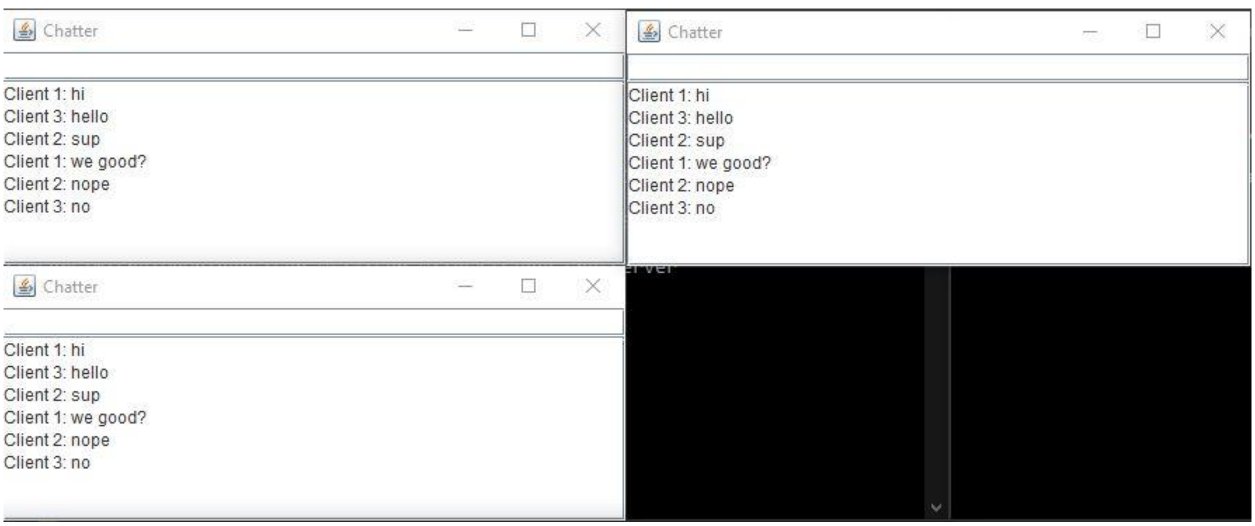

Fig. 5. Task 3 Output

Conclusion

In conclusion, this experiment was a good introduction to socket programming using java. It allowed us to get a deeper understanding through the command prompt and answering some questions.