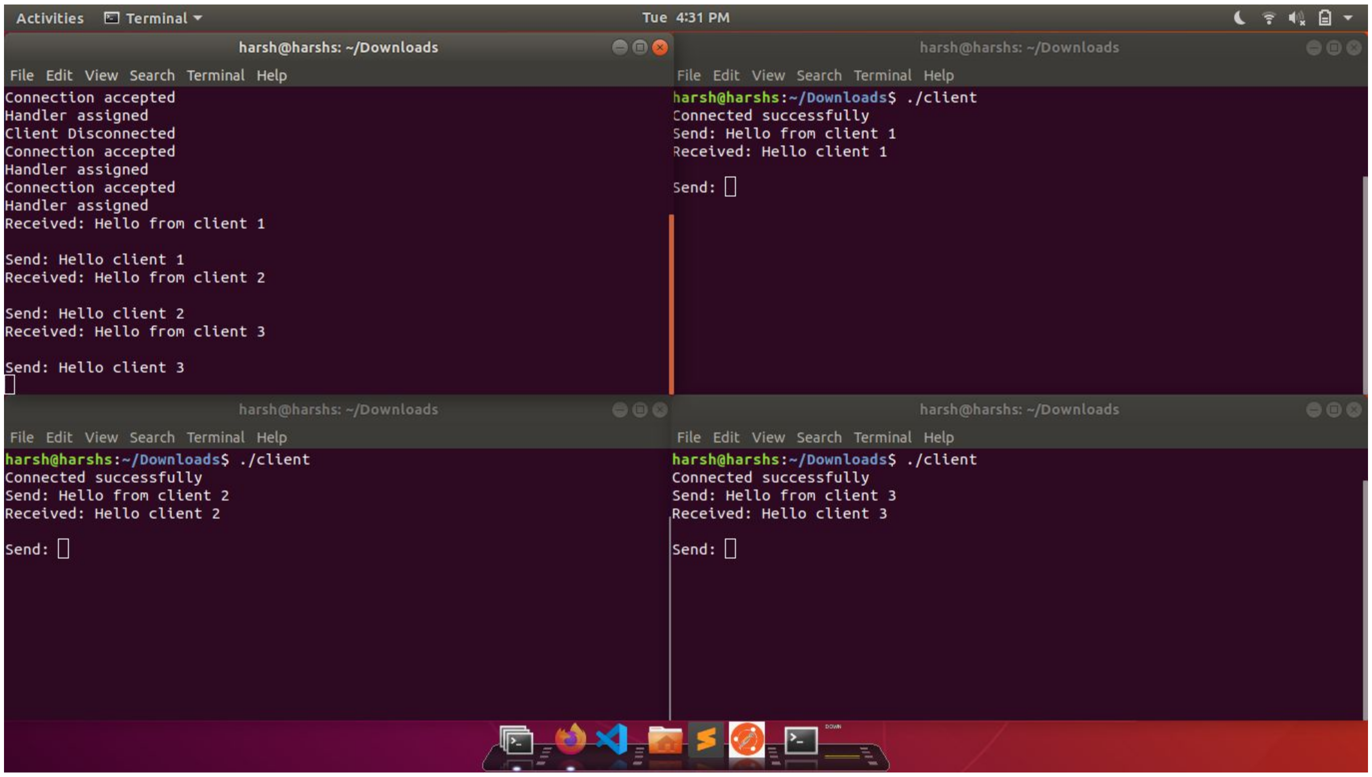
Lab5- Socket programming

TCP Server-Client implementation with multiple clients in C

Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket(node) listens on a particular port at an IP, while another socket reaches out to the other to form a connection. The server forms the listener socket while the client reaches out to the server. TCP/IP provides end-to-end connectivity specifying how data should be formatted, addressed, transmitted, routed, and received at the destination can be used on the internet, and in stand-alone private networks, it is organized into layers.

In this practical, we learned about handling multiple connection requests from clients and also handling the communication requests made by clients and responding to them in accordance with our logic. For the sake of simplicity, in this implementation, we have taken a message as input from the client, displayed it on the server-side, and taken another user input at server-side and sent it back to the client, basically like a chatbot.


Activities
Terminal
Tue 4:31 PM
harsh@harshs: ~/Downloads
File Edit View Search Terminal Help

```
Connection accepted
Handler assigned
Client Disconnected
Connection accepted
Handler assigned
Connection accepted
Handler assigned
Received: Hello from client 1

Send: Hello client 1
Received: Hello from client 2

Send: Hello client 2
Received: Hello from client 3

Send: Hello client 3
```

```
harsh@harshs:~/Downloads$ ./client
Connected successfully
Send: Hello from client 1
Received: Hello client 1

Send: 
```

```
harsh@harshs:~/Downloads$ ./client
Connected successfully
Send: Hello from client 2
Received: Hello client 2

Send: 
```

```
harsh@harshs:~/Downloads$ ./client
Connected successfully
Send: Hello from client 3
Received: Hello client 3

Send: 
```

Conclusion

Through this practical application, we learned about TCP and implemented a basic communication setup between multiple clients and a server like a chatbot.