

C++ Week 12 TCP

Exercise 1

1. Within VS create two C++ Console Projects (Server & Client).
2. Within the client add the code to allow the user to type in a message.
3. Add code to the server to capture this transmitted message and display it on the screen.

Exercise 2

Now extend both the client and server so that on receipt of a message the server sends an automatic confirmation message back to the client ("Message Received"), which is then displayed on the client's screen.

Exercise 3

Extend exercise 2 by allowing an unlimited number of messages to be typed into the client and each one should receive a confirmation.

Exercise 4

Extend the programs to terminate both the server and client machine if the user types "SHUTDOWN" into the client. You may find that when testing this the Press Any Key message is displayed after the server and client connections have been closed.

Exercise 5

Modify the server so that each new connection is managed by a thread.

- Within main you will need to implement a loop within which you accept connections and create a new thread. Passing the accepted socket to the thread function.
- Within the thread function have a second loop that receives and sends data.
- There is no need to change the client.

Exercise 6

Rather than transmitting an array of char, we could send a complete object. Add the following class to both the client and server projects.

```
class Data{  
public:  
    int health;  
    char name[50];  
};
```

C++ Week 12 TCP

Now modify the client so that it no longer sends data to the server but simply receives an instance of Data and prints the values of the two member variables.

Within server create a new Thread Function, that declares a static instance of Data and assigns it an initial health and name value. Then within a loop increments the health by one and transmits the data object to the client ever second. Hint Sleep(1000);

Within main comment out the statement that created the previous exercises thread and add a new one that creates your new thread function.