

Program 1 - Words of Wisdom

(Written by Dr. Scott Barlowe)

Description

This program consists of writing a server and a client in both TCP and UDP. The functionality for both protocols should be identical. The objective of the program is to let a client retrieve a random word of wisdom from the server. The overall process (there are differences between the TCP and UDP versions implementations) is that

- The server will load an array of strings from a text file. Each index in the array will hold a word of wisdom to be returned to the client. You have been provided with a **WisdomWord** class which the server should use for loading and retrieving the strings. You have also been provided with a text file with wise sayings. You may add to the wise sayings file, but DO NOT MODIFY THE WISDOMWORD class.
- The client will initiate contact with the server.
- Upon contact, the server will return an integer to the client that represents the size of the string array (i.e. how many snippets of wisdom that the server has loaded).
- The client should send back to the server a random integer between 0 and the size of the array sent by the server. In effect, the client is randomly telling the server which wise saying should be selected based on the size of the server's array.
- After checking that the number is inside the bounds of the array, the server should retrieve the wise saying at that index in the array and send that string to the client. If the number is outside the bounds of the array, the server give an error message and should close the connection to the client.
- The client should print the wise saying and then terminate.

Getting Started

- Download WisdomWord.java and sayings.txt with
wget http://agora.cs.wcu.edu/~sbarlowe/cs465/Project1/WisdomWords.java
wget http://agora.cs.wcu.edu/~sbarlowe/cs465/Project1/sayings.txt

Project Structure

- Create a **project1** directory. Within the **project1** directory
 - Create a directory called **UDP**. Develop your UDP client and UDP server there. Your UDP server should be called **WisdomWordsServUDP.java** and the client should be called **WisdomWordsCLUdp.java**. Place a copy of your **WisdomWords** class and a copy of **sayings.txt** in that directory.
 - Create a directory called **TCP**. Develop your TCP client and TCP server there. Your UDP server should be called **WisdomWordsServTCP.java** and the client should be called **WisdomWordsCLTCP.java**. Place a copy of your **WisdomWords** class and a copy of **sayings.txt** in that directory.
- Use port **5000** for TCP and port **6000** for UDP.
- The server for both TCP and UDP should accept a command line parameter representing the file name.
- The client for both TCP and UDP should accept two command line parameters. The first is the host and the second is the port.
- The TCP server should send the size of the string array to the TCP client immediately after accepting the request from the client and creating the input/output streams.
- Initiating contact is a little more complex with UDP. To initiate contact, the UDP client should create a buffer of size zero and send the **DatagramPacket** using that buffer to the server. The server should expect a buffer size of zero for the first packet.

- In the UDP version, use only two variables for the send and receive buffers in both the client and server. Depending on the situation, I suggest that you either assign each to a new byte array or to the incoming information before reusing so that the information in the array is current.
- Both the TCP and UDP server should accept requests until the server is terminated (with CTL-C).
- An exception should be caught/handled for not specifying the correct address/port.
- Your program must be compiled and executed on agora.

Example TCP Server Output

```
$ java WisdomWordsServTCP sayings.txt
```

Example TCP Client Output

```
$ java WisdomWordsCLTCP agora.cs.wcu.edu 5000
```

```
Connecting to agora on port 5000...
```

```
If the grass is greener on the other side, your neighbors probably have a septic tank.
```

```
$
```

NOTE: The output for UDP would be the same except for using port 6000.

Important Notes

- You may work in teams of one or two. If working in teams of two, each partner's name should be included with the file heading documentation in all files.
- You may talk about the concepts in the assignment, but you may not share code with other teams.
- Your program should be fully documented and follow the class style guidelines. (Use Google's style guidelines except in a few cases. A link to Google's style guidelines and a list of the exceptions to follow instead of those guidelines can be found in Blackboard.)

Submission

While in your **project1** directory (but outside of your TCP and UDP directories), Compress your **UDP** and **TCP** directories together into a single file with the command:

```
$ tar -zcvf Wisdom.tar.gz TCP UDP
```

Submit Wisdom.tar.gz by midnight on Sept 22 by using the command:

```
$ handin.465.1 1 Wisdom.tar.gz
```