CS 360 In Class Lab Java New IO
April 27, 2012

A Ping/Pong server will accept either the ASCII string PING or PONG and return the opposite string.

You need to write a single threaded server that will accept multiple PING/PONG sessions. You must use telnet as the client application to send and receive PING/PONG data. Note that telnet does include the newline character when sending data across the wire.

Telnet on Windows sends each character as you type, telnet on Linux sends data after a carriage return. On Linux, the command:

telnet localhost 12348

will connect to port 12348 on the local machine.

You must produce a server that will use Java's New IO to implement this protocol. Use the slides from Monday as a guide.

Download the CS360_PingPong_Lab_Student Eclipse project. You will find a completely written PingPongConnection class, a partially written PingPongServer class. You will need to fill in the run() method in the server class. These are the only two files you need to update for this portion of the assignment. PingPongHandler.java is used in the **Challenge** portion of this assignment.

Challenge:

Use a ThreadPoolExecutor to manage a thread pool. When a connection is readable or writable, dispatch a handler (a class that implements Runnable) to the thread pool to complete the task. Hint: once you launch a thread to read or write, remove that particular flag from the S*electionKey's interestOps().* You can use the PingPongHandler.java as your handler. Much of the code in the server run() method should migrate to the hanlder's run() method.