Solution 6.4

File TimedServer.java

```
import java.net.*;
import java.io.*;
import java.util.concurrent.*;

class Worker implements Runnable {
   // duration (in seconds) to hold the connection open
   public static final int DURATION = 10;
   private int sleepTime = DURATION;

   private Socket connection;
   private Semaphore sem;

   public Worker(Socket connection, Semaphore sem) {
        this.connection = connection;
        this.sem = sem;
   }

   public void run() {
        try {
             PrintWriter pout = new
PrintWriter(connection.getOutputStream(), true);

             while (sleepTime > 0) {
                  String s = (sleepTime == 1) ? "
second." : " seconds.";
                  pout.println("Sleeping " + sleepTime + "
more " + s);
                  Thread.sleep(1000);
                  sleepTime -= 1;
             }

             // now close the socket connection
             connection.close();

             // increment the semaphore
             sem.release();
        }
        catch (InterruptedException ie) { }
        catch (IOException ioe) { }
   }
}
```

```
public class TimedServer
{
   public static final int PORT = 2500;
   public static final int MAX_CONNECTIONS = 3;

   public static void main(String[] args) {
        Socket connection;
        Semaphore sem = new Semaphore(MAX_CONNECTIONS);

        try {
             ServerSocket server = new ServerSocket(PORT);

             while (true) {
                  sem.acquire();
                  connection = server.accept();

                  Thread worker = new Thread(new
Worker(connection, sem));
                  worker.start();
             }
        }
        catch (InterruptedException ie) { }
        catch (java.io.IOException ioe) { }
   }
}
```

File TimedClient.java

```
/**
 * TimedClient.java
 *
 * @author Gagne, Galvin, Silberschatz
 * Operating System Concepts with Java - Eighth Edition
 * Copyright John Wiley & Sons - 2010.
 */

import java.net.*;
import java.io.*;

public class TimedClient {

   public static void main(String[] args) throws
IOException {
        InputStream in = null;
        BufferedReader bin = null;
        Socket sock = null;

        try {
             sock = new Socket("127.0.0.1",2500);
             in = sock.getInputStream();
             bin = new BufferedReader(new
InputStreamReader(in));

             String line;
             while( (line = bin.readLine()) != null)
                  System.out.println(line);
        }
        catch (IOException ioe) {
                  System.err.println(ioe);
        }
                finally {
                    sock.close();
                }
   }
}
```