// Server

import java.net.\*;

import java.io.\*;

import java.nio.channels.\*;

public class Server {

protected final String HOST = "";

protected int port;

protected String path = "/home/quanghuy25899/Desktop/226Lab9/";

protected int buf\_size = 1024;

public Server(int port) {

this.port = port;

}

void serve() {

try (

ServerSocket serverSocket = new ServerSocket(port);

Socket clientSocket = serverSocket.accept();

BufferedOutputStream out = new BufferedOutputStream(clientSocket.getOutputStream(), buf\_size);

BufferedReader in = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));

) {

// Read in from Client

String inputLine = in.readLine();

// Concat the file name with file path

String file\_name = path + inputLine;

// Open file then create a input stream

BufferedInputStream fileIn = new BufferedInputStream(new FileInputStream(file\_name));

// Byte type array to store file data

byte[] buf = new byte[buf\_size];

int count;

while ((count = fileIn.read(buf, 0, buf\_size)) != -1) {

// Send data to client

out.write(buf, 0, count);

System.out.println(count);

// System.out.println(buf);

}

// Flush all old data

out.flush();

System.out.println("here");

// Close the file and the input stream

fileIn.close();

} catch (IOException e) {

System.err.println(e);

System.exit(-2);

} catch (SecurityException e) {

System.err.println(e);

System.exit(-3);

} catch (IllegalArgumentException e) {

System.err.println(e);

System.exit(-4);

} catch (IllegalBlockingModeException e) {

System.err.println(e);

System.exit(-6);

}

}

public static void main(String[] args) {

if (args.length != 1) {

System.err.println("Need <port>");

System.exit(-99);

}

Server s = new Server(Integer.valueOf(args[0]));

s.serve();

}

}

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------

// Client

import java.io.\*;

import java.net.\*;

public class Client {

protected String serverName;

protected int serverPort;

protected String message;

protected String path = "/home/quanghuy25899/Desktop/";

protected int buf\_size = 1024;

public Client(String serverName, int serverPort, String message) {

this.serverName = serverName;

this.serverPort = serverPort;

this.message = message;

}

public void connect() throws Exception {

// Concat the file name with file path

String file\_name = path + this.message;

try (

Socket socket = new Socket(serverName, serverPort);

PrintWriter out = new PrintWriter(socket.getOutputStream(), true);

BufferedInputStream in = new BufferedInputStream(socket.getInputStream());

) {

// Send data to server

out.println(this.message);

int count=0;

// Create a byte type array to store file data

byte[] b = new byte[buf\_size];

//Open a file to write data

BufferedOutputStream fileOut = new BufferedOutputStream(new FileOutputStream(file\_name));

// System.out.println(in.read(b));

while ((count = in.read(b, 0, buf\_size)) > 0) {

System.out.println(count);

// Write data from b array to a file

fileOut.write(b, 0, count);

}

// Flush all old data

fileOut.flush();

System.out.println("here");

// Close the file

fileOut.close();

} catch (UnknownHostException e) {

System.err.println(e);

System.exit(-1);

} catch (IOException e) {

System.err.println(e);

System.exit(-2);

} catch (SecurityException e) {

System.err.println(e);

System.exit(-3);

} catch (IllegalArgumentException e) {

System.err.println(e);

System.exit(-4);

}

}

public static void main(String[] args) throws Exception {

if (args.length != 3) {

System.err.println("Need <host> <port> <message>");

System.exit(-2);

}

Client c = new Client(args[0], Integer.valueOf(args[1]), args[2]);

c.connect();

}

}