Донецкий Национальный Технический Университет

Лабораторная работа № 1

Изучение протоколов FTP и TFTP

Выполнил:

ст. группы ИПЗ -13

Лысенко А. С.

Проверила:

Скрипник. Т. В.

Красноармейск 2016

Server.java

**package** FileWork;  
  
**import** java.io.IOException;  
**import** java.io.ObjectInputStream;  
**import** java.io.ObjectOutputStream;  
**import** java.net.ServerSocket;  
**import** java.net.Socket;  
**import** java.net.UnknownHostException;  
  
**public class** Server **implements** Runnable  
{  
 **static private** ServerSocket *server*;  
 **static private** Socket *connection*;  
 **static private** ObjectOutputStream *output*;  
 **static private** ObjectInputStream *input*;  
 **public void** run() {  
  
 **try** {  
 *server* = **new** ServerSocket(1337,50);  
 **while**(**true**)  
 {  
 *connection* = *server*.accept();  
 *output* = **new** ObjectOutputStream(*connection*.getOutputStream());  
 *input* = **new** ObjectInputStream(*connection*.getInputStream());  
 *output*.writeObject(**"Вы прислали:"** + (String) *input*.readObject());  
 }  
 } **catch** (UnknownHostException e) {  
 e.printStackTrace();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 } **catch** (ClassNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

Client.java

**package** FileWork;  
  
**import** javax.swing.\*;  
**import** java.io.IOException;  
**import** java.io.ObjectInputStream;  
**import** java.io.ObjectOutputStream;  
**import** java.net.InetAddress;  
**import** java.net.Socket;  
**import** java.net.UnknownHostException;  
  
**public class** Client **implements** Runnable  
 {  
  
 **static private** Socket *connection*;  
 **static private** ObjectOutputStream *output*;  
 **static private** ObjectInputStream *input*;  
 **public void** run()  
 {  
 **try** {  
 **while**(**true**)  
 {  
 *connection* = **new** Socket(InetAddress.*getByName*(**"127.0.0.1"**),1337);  
 *output* = **new** ObjectOutputStream(*connection*.getOutputStream());  
 *input* = **new** ObjectInputStream(*connection*.getInputStream());  
 JOptionPane.*showMessageDialog*(**null**, *input*.readObject());  
 }  
 } **catch** (UnknownHostException e) {  
 e.printStackTrace();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 } **catch** (ClassNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
 **public static void** sendData(Object obj) **throws** IOException {  
 *output*.flush();  
 *output*.writeObject(obj);  
 }  
 }

GUI.java

**package** FileWork;  
**import** javax.swing.\*;  
**import** java.io.FileNotFoundException;  
  
**public class** GUI **extends** JFrame  
{  
 **public** GUI() **throws** FileNotFoundException  
 {  
 setContentPane(**new** FileWork());  
 setSize(300, 300);  
 setTitle(**"Отправка файла"**);  
 setResizable(**false**);  
 setLocationRelativeTo(**null**);  
 setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);  
 show();  
 }  
  
}

FileWork.java

**package** FileWork;  
**import** javax.swing.\*;  
**import** java.awt.\*;  
**import** java.awt.event.ActionEvent;  
**import** java.io.\*;  
  
**import static** FileWork.Client.*sendData*;  
  
  
**public class** FileWork **extends** JPanel  
{**private static** String *fileName* = **"C://blog/a.txt"**;  
 **private static** String *text* = **""**;**static** String *textFromFile*;  
 **public** FileWork() **throws** FileNotFoundException  
 {  
 JLabel l = **new** JLabel(**"Путь к файлу: "**);  
 JLabel l1 = **new** JLabel(**"Текст файла: "**);  
 JTextField t = **new** JTextField(15);  
 JTextArea t1 = **new** JTextArea(10,15);  
 t1.setFont(**new** Font(**"Times New Roman"**, Font.***ITALIC***, 14));  
 t1.setLineWrap(**true**);  
 t1.setWrapStyleWord(**true**);  
 JScrollPane scroll = **new** JScrollPane();  
 JButton b = **new** JButton(**" Записать "**);  
 JButton b1 = **new** JButton(**" Читать "**);  
 JButton b2 = **new** JButton(**" Удалить "**);  
 JButton b3 = **new** JButton(**"Копировать"**);  
 *fileName* = t.getText();  
 *text* = t1.getText();  
 setLayout(**new** FlowLayout());  
 add(l);  
 add(t);  
 add(l1);  
 add(**new** JScrollPane(t1));  
 add(scroll);  
 add(b);  
 add(b1);  
 add(b3);  
 add(b2);  
 b.addActionListener(**new** AbstractAction() {  
 @Override  
 **public void** actionPerformed(ActionEvent e) {  
 *fileName* = t.getText();  
 *text* = t1.getText();  
 FileWork.*write*(*fileName*,*text*);  
 **try** {  
 *sendData*(*fileName*);  
 *sendData*(*text*);  
 } **catch** (IOException e1) {  
 e1.printStackTrace();  
 }  
 JOptionPane.*showMessageDialog*(**null**,**"Записано!"**);  
 }  
 });  
 b1.addActionListener(**new** AbstractAction() {  
 @Override  
 **public void** actionPerformed(ActionEvent e) {  
 **try** {  
 *textFromFile* = FileWork.*read*(*fileName*);  
 } **catch** (FileNotFoundException e1) {  
 e1.printStackTrace();  
 }  
 t1.setText(*textFromFile*);  
 t.setText(**""**);  
 **try** {  
 *sendData*(*textFromFile*);  
 } **catch** (IOException e1) {  
 e1.printStackTrace();  
 }  
 JOptionPane.*showMessageDialog*(**null**, **"Прочитано!"**);  
 }  
 });  
 b3.addActionListener(**new** AbstractAction() {  
 @Override  
 **public void** actionPerformed(ActionEvent e) {  
 **try** {  
 *copy*(*fileName*);  
 t.setText(**""**);  
 t1.setText(*text*);  
 } **catch** (FileNotFoundException e1) {  
 e1.printStackTrace();  
 }  
 **try** {  
 *sendData*(*text*);  
 } **catch** (IOException e1) {  
 e1.printStackTrace();  
 }  
 JOptionPane.*showMessageDialog*(**null**, **"Копировано!"**);  
 }  
 });  
 b2.addActionListener(**new** AbstractAction() {  
 @Override  
 **public void** actionPerformed(ActionEvent e) {  
 **try** {  
 FileWork.*delete*(*fileName*);  
 } **catch** (FileNotFoundException e1) {  
 e1.printStackTrace();  
 }  
 t1.setText(**"Файл удален"**);  
 t.setText(**""**);  
 **try** {  
 *sendData*(*fileName*);  
 } **catch** (IOException e1) {  
 e1.printStackTrace();  
 }  
 JOptionPane.*showMessageDialog*(**null**, **"Удалено!"**);  
 }  
  
 });}  
 **static** File *file* = **new** File(*fileName*);  
 **public static void** write(String fileName, String text) {  
 File file = **new** File(fileName);  
 **try** {  
 **if**(!file.exists()){  
 file.createNewFile();  
 }  
 PrintWriter out = **new** PrintWriter(file.getAbsoluteFile());  
 **try** {  
 out.print(text);  
 } **finally** {  
 out.close();  
 }  
 } **catch**(IOException e) {  
 **throw new** RuntimeException(e);  
 }  
 }  
  
 **public static** String read(String fileName) **throws** FileNotFoundException {  
 StringBuilder sb = **new** StringBuilder();  
  
 *exists*(fileName);  
  
 **try** {  
  
 BufferedReader in = **new** BufferedReader(**new** FileReader(*file*.getAbsoluteFile()));  
 **try** {  
  
 String s;  
 **while** ((s = in.readLine()) != **null**) {  
 sb.append(s);  
 sb.append(**"\n"**);  
 }  
 } **finally** {  
 in.close();  
 }  
 } **catch**(IOException e) {  
 **throw new** RuntimeException(e);  
 }  
  
 **return** sb.toString();  
 }  
 **private static void** exists(String fileName) **throws** FileNotFoundException {  
 File file = **new** File(fileName);  
 **if** (!file.exists()){  
 **throw new** FileNotFoundException(file.getName());  
 }  
 }  
 **public static void** copy(String fileName) **throws** FileNotFoundException {  
 *exists*(FileWork.*fileName*);  
 StringBuilder sb = **new** StringBuilder();  
 String oldFile = *read*(fileName);  
 sb.append(oldFile);  
 sb.append(*text*);  
 *write*(fileName, sb.toString());  
 }  
  
 **public static void** delete(String nameFile) **throws** FileNotFoundException {  
 *exists*(nameFile);  
 **new** File(nameFile).delete();  
 }  
}

Source.java

**package** Source;  
**import** FileWork.\*;  
**import** java.io.FileNotFoundException;  
**public class** Source  
{  
 **public static void** main(String[] args)**throws** FileNotFoundException  
 {  
 **new** GUI();  
 **new** Thread(**new** Client()).start();  
 **new** Thread(**new** Server()).start();  
 }  
}