Донецкий Национальный Технический Университет

Лабораторная работа № 3

Разработка простейших клиент-серверных программ с использованием UDP сокетов

Выполнил:

ст. группы ИПЗ -13

Лысенко А. С.

Проверила:

Скрипник. Т. В.

Красноармейск 2016

Sender.java

**package** Source;  
**import** java.io.File;  
**import** java.io.FileInputStream;  
**import** java.io.FileNotFoundException;  
**import** java.io.IOException;  
**import** java.net.\*;  
**public class** Sender **implements** Runnable  
{  
 **public void** run()  
 {  
 **try** {  
 **byte**[] data = **new byte**[1000];  
 DatagramSocket s = **new** DatagramSocket();  
 InetAddress addr = InetAddress.*getLocalHost*();  
*/\*файл с именем 1.txt должен лежать в корне проекта\*/* FileInputStream fr = **new** FileInputStream(**new** File(**"1.txt"**));  
 DatagramPacket pac;  
 **while** (fr.read(data) != -1) {  
*//создание пакета данных* pac = **new** DatagramPacket(data, data.**length**, addr, 51000);  
 s.send(pac);*//отправление пакета* }  
 fr.close();  
 System.***out***.println(**"Файл отправлен"**);  
 } **catch** (UnknownHostException e)  
 {  
*// неверный адрес получателя* e.printStackTrace();  
 } **catch** (SocketException e)  
 {  
*// возникли ошибки при передаче данных* e.printStackTrace();  
 } **catch** (FileNotFoundException e)  
 {  
*// не найден отправляемый файл* e.printStackTrace();  
 } **catch** (IOException e) {  
  
 e.printStackTrace();  
 }  
 }  
}

Recipient.java

**package** Source;  
  
**import** java.io.File;  
**import** java.io.FileOutputStream;  
**import** java.io.IOException;  
**import** java.net.DatagramPacket;  
**import** java.net.DatagramSocket;  
**import** java.net.SocketTimeoutException;  
  
**public class** Recipient **implements** Runnable  
{  
 **private static void** acceptFile (File file,**int** port, **int** pacSize)**throws** IOException  
 {  
 **byte** data[] = **new byte**[pacSize];  
 DatagramPacket pac = **new** DatagramPacket(data, data.**length**);  
 DatagramSocket s = **new** DatagramSocket(port);  
 FileOutputStream os = **new** FileOutputStream(file);  
 **try** {  
*/\* установка времени ожидания: если в течение 10 секунд  
не принято ни одного пакета, прием данных заканчивается\*/* s.setSoTimeout(10000);  
 **while** (**true**) {  
 s.receive(pac);  
 os.write(data);  
 os.flush();  
 }  
 } **catch** (SocketTimeoutException e) {  
*// если время ожидания вышло* os.close();  
 System.***out***.println(**"Истекло время ожидания, прием данных закончен"**);  
 }  
 }  
 **public void** run() {  
 File file = **new** File(**"1.txt"**);  
 System.***out***.println(**"Прием данных…"**);  
 **try** { *// прием файла  
 acceptFile*(file, 51000, 1000);  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
}

Source.java

**package** Source;  
**public class** Source  
{  
 **public static void** main(String[] args)  
 {  
 **new** Thread(**new** Sender()).start();  
 **new** Thread(**new** Recipient()).start();  
 }  
}