

DAY 6——文件传输

问题 6：如何实现文件传输？

6.1 学习内容

文件传输方法，请学习教程的第 6 章 TCP 协议传输文件。

6.2 任务清单

- ① 运行并阅读程序，描绘出主要函数的步骤和程序结构。
- ② 描绘后台线程 SwingWork 的使用方法。
- ③ 将重要功能，记在电子笔记中。
- ④ 在小组内，抽签阐述服务器或客户端的软件结构，并录制视频。
- ⑤ 在自己的程序中实现文件传输。

6.3 关键技术

6.3.1 选择文件

```
private void uploadFileActionPerformed(java.awt.event.ActionEvent evt) {  
    //打开文件选择对话框  
    JFileChooser fileChooser=new JFileChooser();  
    fileChooser.setDialogTitle("选择上传文件");  
    fileChooser.setApproveButtonText("选择");  
    int choice=fileChooser.showOpenDialog(this); //显示对话框  
    if (choice==JFileChooser.APPROVE_OPTION) { //点击选择按钮  
        File file=fileChooser.getSelectedFile();//获取文件对象  
    }  
}
```

6.3.2 启动文件传输的后台线程

```
private void uploadFileActionPerformed(java.awt.event.ActionEvent evt) {  
    //打开文件选择对话框  
    JFileChooser fileChooser=new JFileChooser();  
    fileChooser.setDialogTitle("选择上传文件");  
    fileChooser.setApproveButtonText("选择");  
    int choice=fileChooser.showOpenDialog(this); //显示对话框  
    if (choice==JFileChooser.APPROVE_OPTION) { //点击选择按钮  
        File file=fileChooser.getSelectedFile();//获取文件对象  
        //启动发送文件线程  
        SwingWorker<List<String>,String> sender=new FileSender(file,msg,this);  
  
        sender.addPropertyChangeListener(new PropertyChangeListener() {  
            public void propertyChange(PropertyChangeEvent evt) {  
                if ("progress".equals(evt.getPropertyName())) {  
                    progressBar.setValue((Integer)evt.getNewValue());  
                }  
            }  
        });  
        filePanel.setVisible(true);  
        fileLabel.setText("文件: "+file.getName());  
        sender.execute();  
    }  
}
```

6.3.3 文件发送线程（FileSend.Java）

```
public class FileSender extends SwingWorker<List<String>,String>{//List<String>  
    private File file; //文件  
    private Message msg;//消息类  
    private ClientUI parentUI; //父类  
    private Socket fileSocket; //传送文件的套接字  
    private static final int BUFSIZE=8096; //缓冲区大小  
    private int progress=0; //文件传送进度  
    private String lastResults=null; //传送结果  
    //构造函数  
    public FileSender(File file,Message msg,ClientUI parentUI) {  
        this.file=file;  
        this.msg=msg;  
        this.parentUI=parentUI;  
    }  
}
```

```

@Override
protected List<String> doInBackground() throws Exception {

    fileSocket=new Socket();
    //连接服务器
    SocketAddress remoteAddr=new
InetSocketAddress(msg.getToAddr(),msg.getToPort());
    fileSocket.connect(remoteAddr);
    //构建套接字输出流
    DataOutputStream out=new DataOutputStream(
        new BufferedOutputStream(
            fileSocket.getOutputStream()));
    //构建文件输入流
    DataInputStream in=new DataInputStream(
        new BufferedInputStream(
            new FileInputStream(file)));
    long fileLen=file.length(); //计算文件长度
    //发送文件名称、文件长度
    out.writeUTF(file.getName());
    out.writeLong(fileLen);
    out.flush();
    //传送文件内容
    int numRead=0; //单次读取的字节数
    int numFinished=0; //总完成字节数
    byte[] buffer=new byte[BUFSIZE];
    while (numFinished<fileLen && (numRead=in.read(buffer))!=-1) { //文件可读
        out.write(buffer,0,numRead); //发送
        out.flush();
        numFinished+=numRead; //已完成字节数
        Thread.sleep(2); //演示文件传输进度用
        publish(numFinished+"/"+fileLen+"bytes");
        setProgress(numFinished*100/(int)fileLen);
    } //end while
    in.close();
    //接收服务器反馈信息
    BufferedReader br=new BufferedReader(
        new InputStreamReader(
            fileSocket.getInputStream()));
    String response=br.readLine();//读取返回串
    if (response.equalsIgnoreCase("M_DONE")) { //服务器成功接收
        lastResults= file.getName() +"  传送成功! \n";
    } else if (response.equalsIgnoreCase("M_LOST")) { //服务器接收失败
        lastResults= file.getName() +"  传送失败! \n";
    } //end if
}

```

```

        //关闭流
        br.close();
        out.close();
        fileSocket.close();
        return null;
    } //doInBackground

    protected void process(List<String> middleResults) {
        for (String str:middleResults) {
            parentUI.progressLabel.setText(str);
        }
    }
    @Override
    protected void done() {
        parentUI.progressBar.setValue(parentUI.progressBar.getMaximum());
        parentUI.txtArea.append(lastResults+"\n");
        parentUI.filePanel.setVisible(false);
    }
}

```

6.4 问题讨论

- ① 如何构建 P2P 的 QQ 程序，并可实现双向文件传输？

