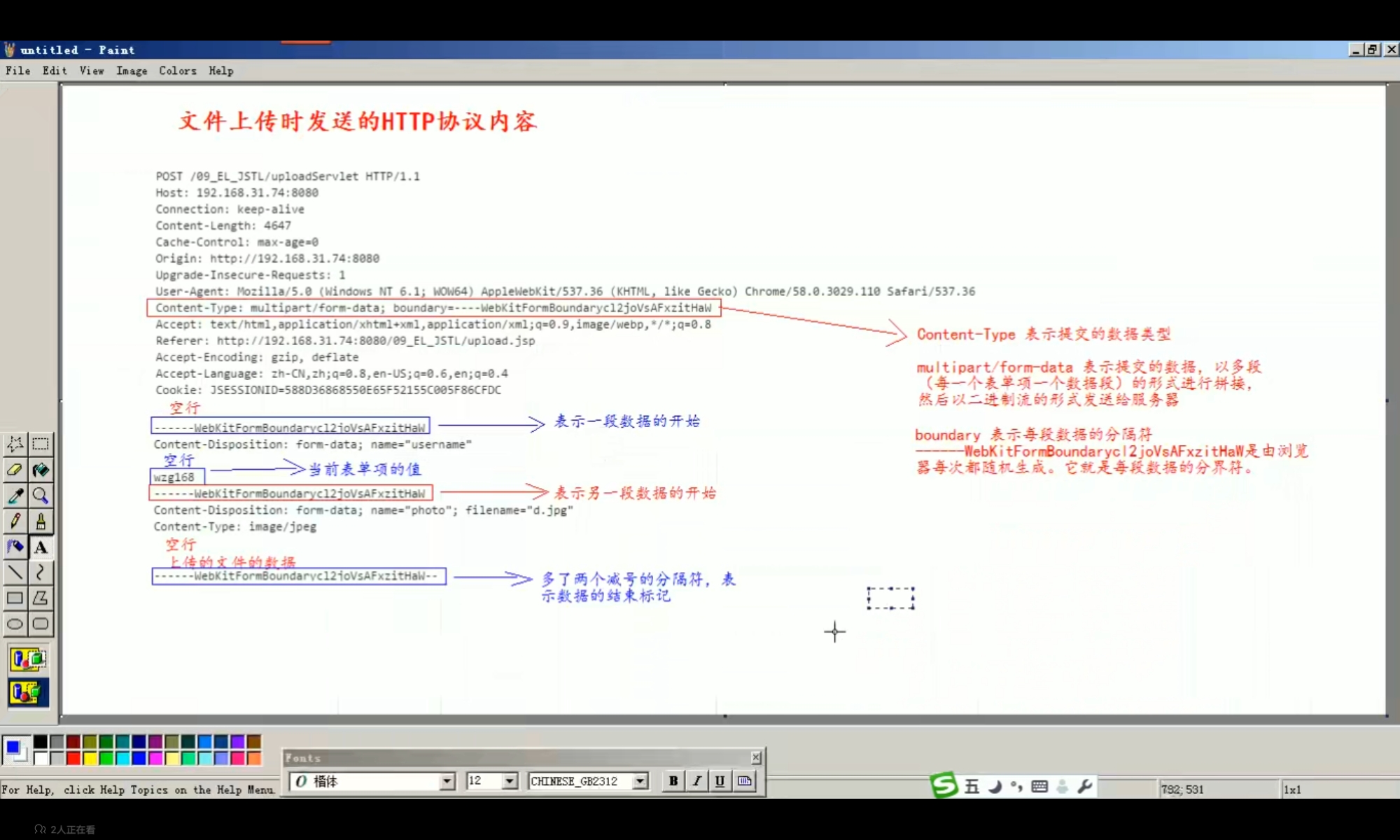
文件的上传和下载

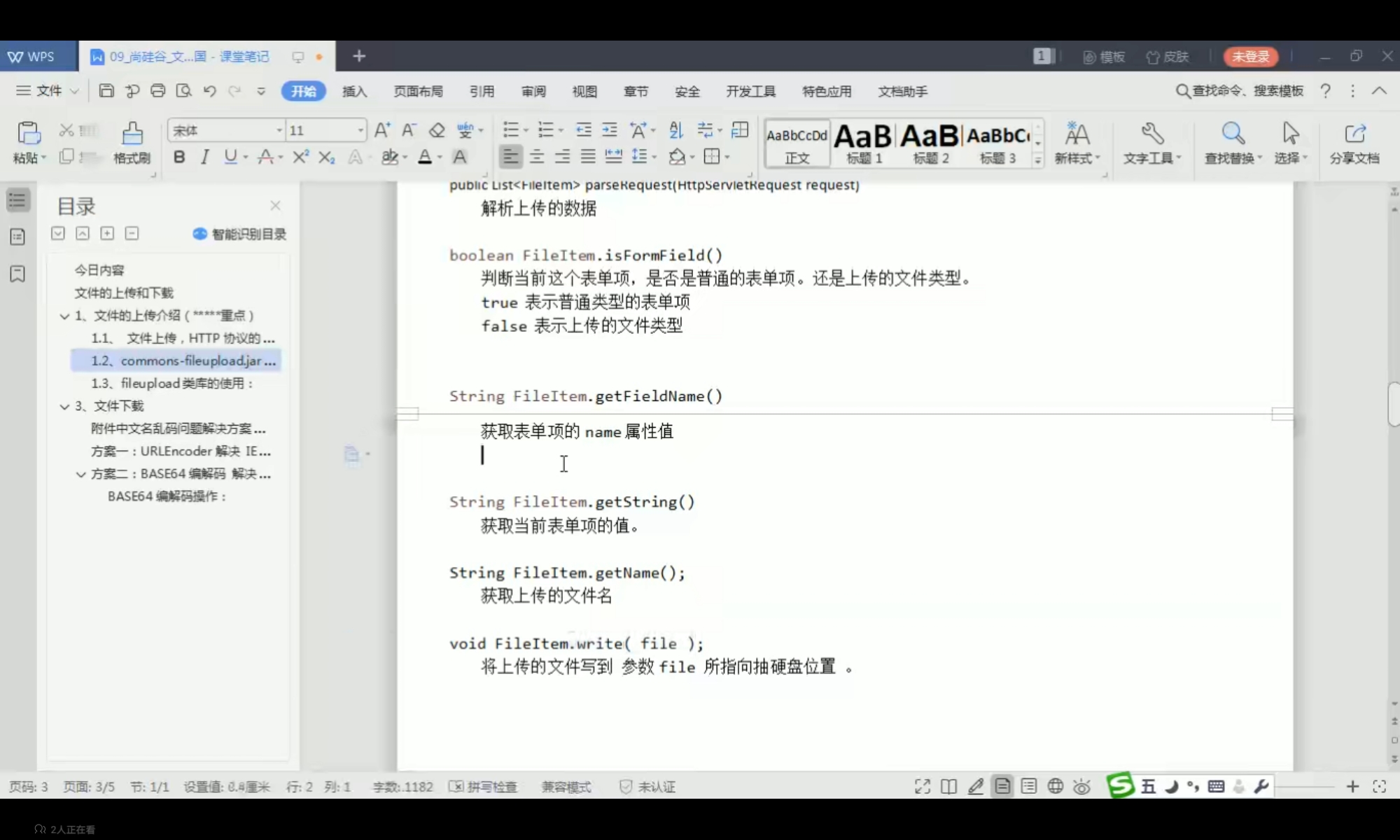
1. 文件的上传介绍
2. 要有一个form标签，method=post请求
3. Form标签的encTyp[e属性值必须为multipart/form-data值
4. 在form标签中使用input type=file添加上传的文件
5. 编写服务器代码接受，处理上传的数据

encType=multipart/form-data表示提交的数据，以多段（每一个表单向一个数据段）的形式进行拼接，然后以二进制流的形式发送给服务器

 1.1文件上的http协议内容

1.2commons-fileupload.jar常用API介绍说明

Commons-fileupload.jar需要依赖commons-io.jar这个包，所以两个包我们都要导入

 Commons-fileupload.jar和commons-io.jar常用的类

1. 文件的下载
2. *//1.获取要下载的文件名*String downloadFileName=**"2.jpg"**;  
   *//2.读取要下载的文件内容（通过ServletContext对象可以读取）*ServletContext servletContext = getServletContext();  
   *//获取要下载的文件类型*String mimeType = servletContext.getMimeType(**"/file/"** + downloadFileName);  
   System.***out***.println(**"下载的文件类型:"**+mimeType);  
   *//4.在回传前，通过响应头告诉客户端返回的数据类型*resp.setContentType(mimeType);  
   *//5.还要告诉客户端收到的数据是用于下载使用（还是使用响应头）  
   //Content-Disposition响应头表示收到的数据怎么处理  
   //attachment表示附件，表示下载使用  
   //filename= 表示指定下载的文件名  
   //url编码是把汉字转换成%xx%xx的格式*resp.setHeader(**"Content-Disposition"**,**"attachment;filename="**+ URLEncoder.*encode*(**"狗.jpg"**,**"UTF-8"**));  
   *// 获取响应的输出流  
   // 读取输入流中全部数据，复制给输出流，输出给客户端*InputStream resourceAsStream = servletContext.getResourceAsStream(**"/file/"** + downloadFileName);  
   OutputStream outputStream=resp.getOutputStream();  
   *//3.把下载的文件内容回传给客户端*IOUtils.*copy*(resourceAsStream,outputStream);