# 文件上传操作手册

## 上传文件

### 第一步

#### **前台选择文件**

调用ElementUI的el-upload组件，获取要上传的文件。实际可根据自身情况进行选择，只要能获得文件即可。如下图1.1：

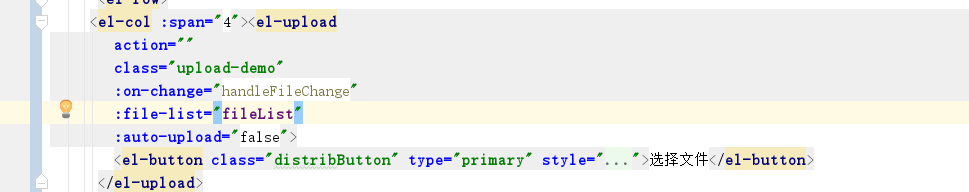


图1.1

上图 通过调用el-upload组件的on-change属性调用handleFileChange方法，如图该方法如下图1.2所示：

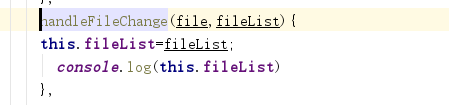


图1.2

从上面方法可以看到第一个形参（file）是当前选择的文件,第二个形参（fileList）是当前选择的所有文件。把所有选择的文件赋值给data声明的fileList变量。现在我们获得了需要上传的所有文件。那么接下来进行提交。

#### 前台提交文件

刚才我们获得了所需要的所有的文件，那么接下来进行上传。因为我们平时是一个表单一个附件，当然也可能一个表单多个附件，可根据实际情况自己选择。无论我们怎样选择，但有的时候并不是只是上传文件，可能还会包含表单的其他的信息，比如某个人的名字和年龄等。

我先把前台的文件上传的接口先展示出来，再来说怎么用。接口如下图1.3：



图1.3

上图是个接口,它有两个形参，第一个形参接受的是表单的非二进制数据，比如某个人的年龄和姓名，他是一个JSON对像。第二个形参是需要上传的文件。把如何调用这个接口和传的内容可参考图1.4：

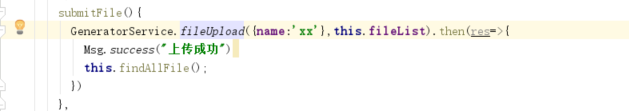


图1.4

上图调用了刚才的接口，第一个形参可以看出是基本的对象，第二个形参是刚才选择的文件列表。

值得注意的是，在图1.4中我们不是直接把文件列表传个server而是把每个文件的raw传给server，至于为什么，自己网上查。现在前台的条件一切准备就绪，接下来看后台如何处理。

### 第二步

#### 后台处理文件

在图1.3中我们将文件上传到（/simple/attachment/Attachment/saveAttachment）controller中,还是首先看这个controller中写了些什么。如下图1.5：

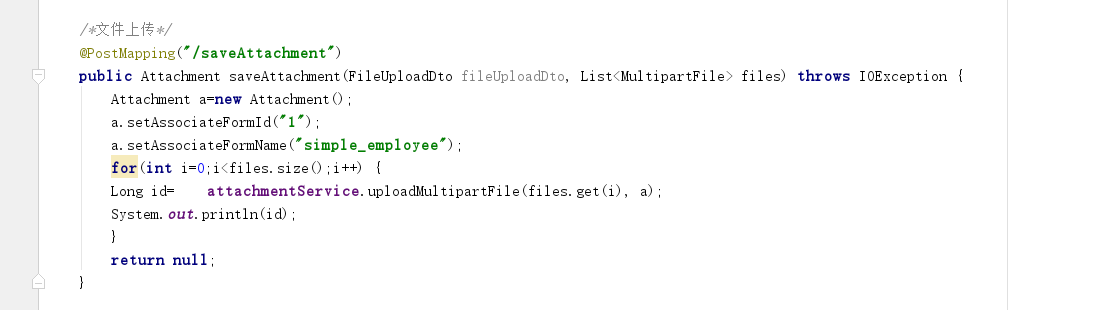


图1.5

在图1.5中saveAttachment接收了两个形参，第一个是一个DTO（数据传输对象），

第二个则是一个List<MultipartFile> files文件，接收了前台传输的数据。值得提醒的是后台的controller的形参名，如果不加@requestParm和@requestPart注解的话（我就没加），必须跟前台的所传输的变量名一一对应,如下图1.6展示了dto跟前台的变量名对应

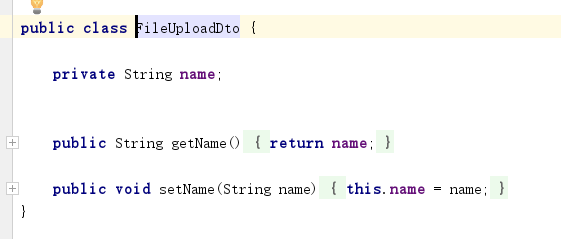


图1.6

如下图1.7展示了文件跟前台的一致：



图1.7

接下来看controller如何处理所上传的文件。

要保存文件，你所做的第一步是在你自己的controller中注入AttachmentService服务，如下图1.8：



图1.8

接下来就能够使用它了。

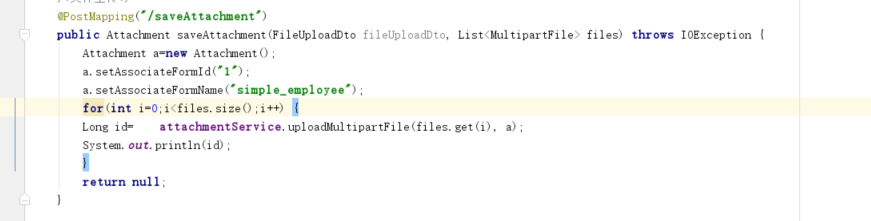


图1.9

在上图1.9中首先实例化了一个Attachment对象，我们所做的是也就是必填的是associateFormId（关联表Id）和associateFormName（关联表名）这个是必须的。根据这两个可以确定数据库的任何一张表的任何一行。假如你新增了一个学生，把你刚新增得到的id和数据库表名（写死）传给这两个属性就行了。具体的Attachment实体各属性解释如下图1.10和1.11



图1.10

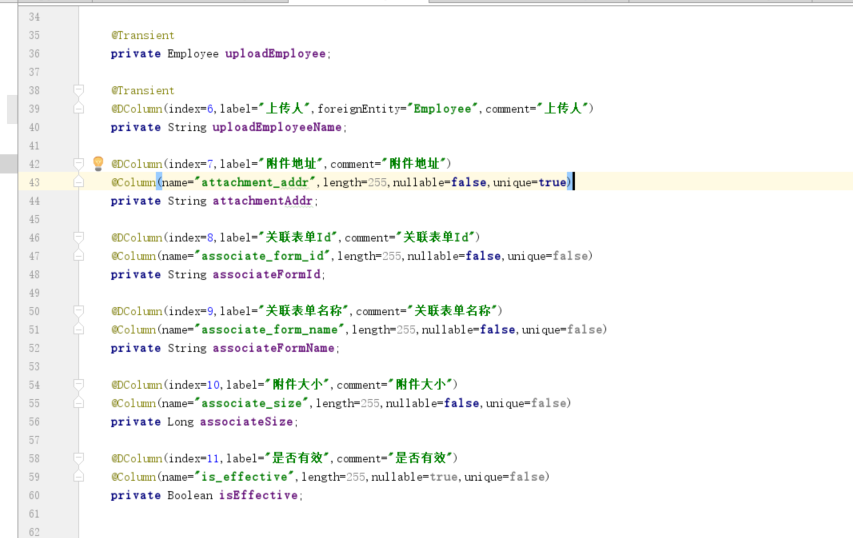


图1.11

那么接下来调用attachmentservice的uploadMultipartFile方法，第一个形参是文件（如果多个进行for循环就行了），第二个是实例化的attachment对象。该对象返回的是新增的一行的id，如下图1.12全步骤。

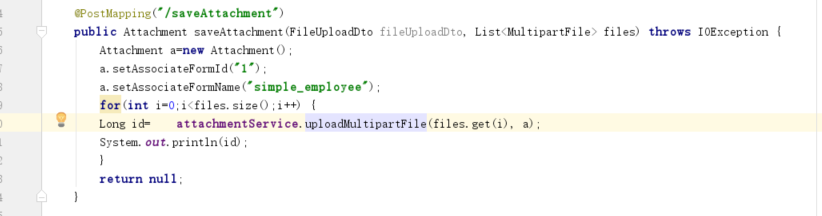
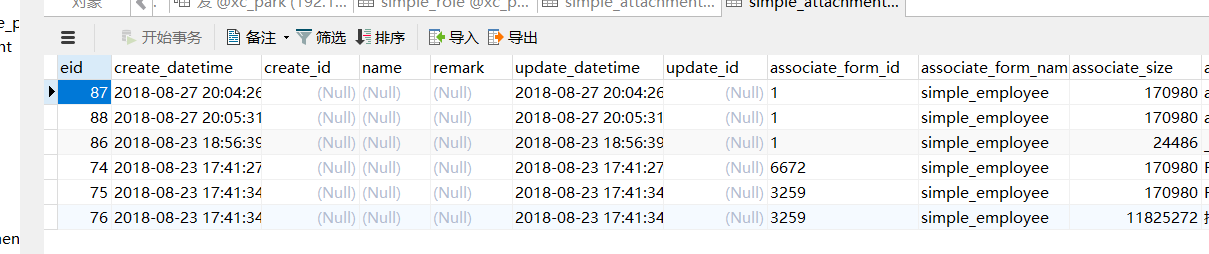


图1.12

可以再数据库查看是否上传成功，如下图1.13



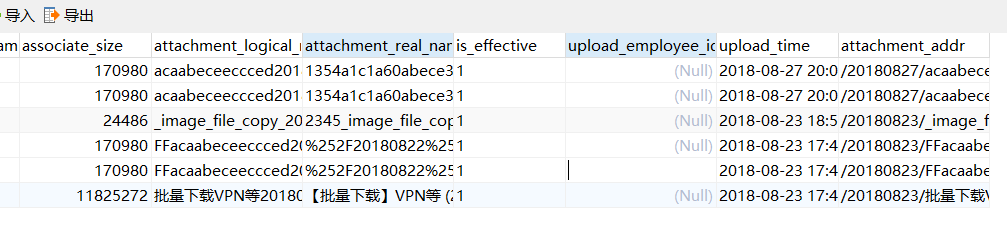


图1.13

## 下载，预览和删除文件

对于下载文件，首先需要把所上传的文件根据条件查询到前台。我写了个根据associate\_form\_id和associate\_form\_name查找符合条件的文件。注意这里返回的是个list，因为一行也许包含了多个文件，方法findAllUploadedFilesByIdANdName（）。如下图1.14所示改方法，传入id和name。

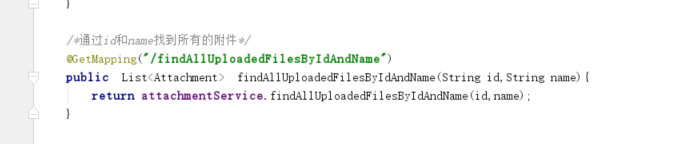


图1.14

所做的效果如图1.15

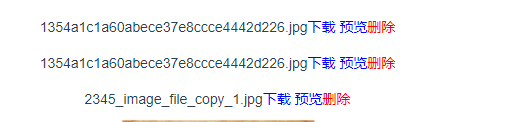


图1.15

你如上图你所展示的是你文件的真实名。提供下载，预览 ，删除功能。接下来看点击下载如何做？这里值得庆幸的是下载和预览只调用一个方法即可。如下图1.16 controller中如何写的

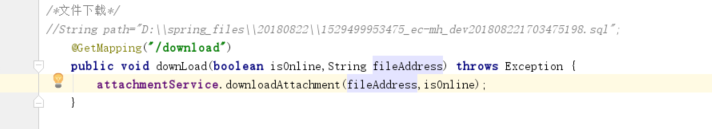


图1.16

调用了attachmentService的downloadAttachment()方法，第一个形参名是文件地址，只需把刚才的上传的文件地址（对于数据库的字段）传进去即可，第二个是一个Boolean的变量，为true表示的是预览功能，为false表示下载。值得遗憾的是只能在线预览图片，文件的话是下载。现在看看删除，如下图1.17

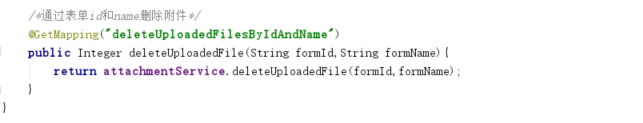


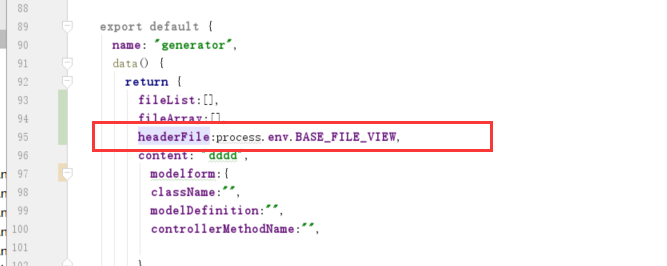
图1.17

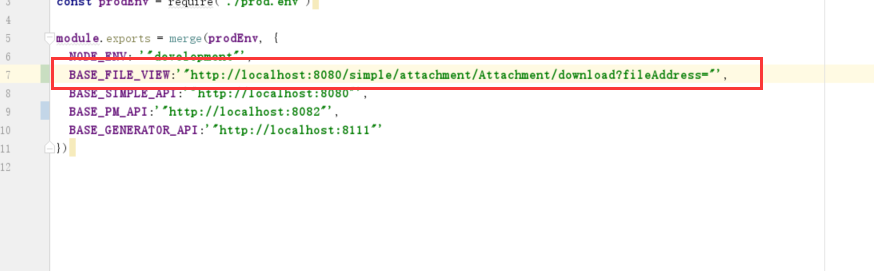
在上图我只写了根据关联表id和关联表名进行删除。如果你调用我写的方法需要的注意的是假如一行有多个附件，会都进行删除。到时你可自己根据条件在attachment表删除哪行即可。

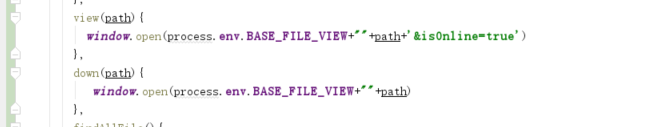
对于图片展示，介绍如下：



只需在img的src传入文件地址。







上面几张图中的process.env.**BASE\_FILE\_VIEW是图片的下载和预览前缀。我这里的是请求的8080端口的attachment/Attachment/download的get接口，你们直接这样写就行。因为simple是必须启动的，启动就能调用这个下载和预览接口，当然你可以自己实现，但那样不好统一。**

# 实际效果图



图2.1

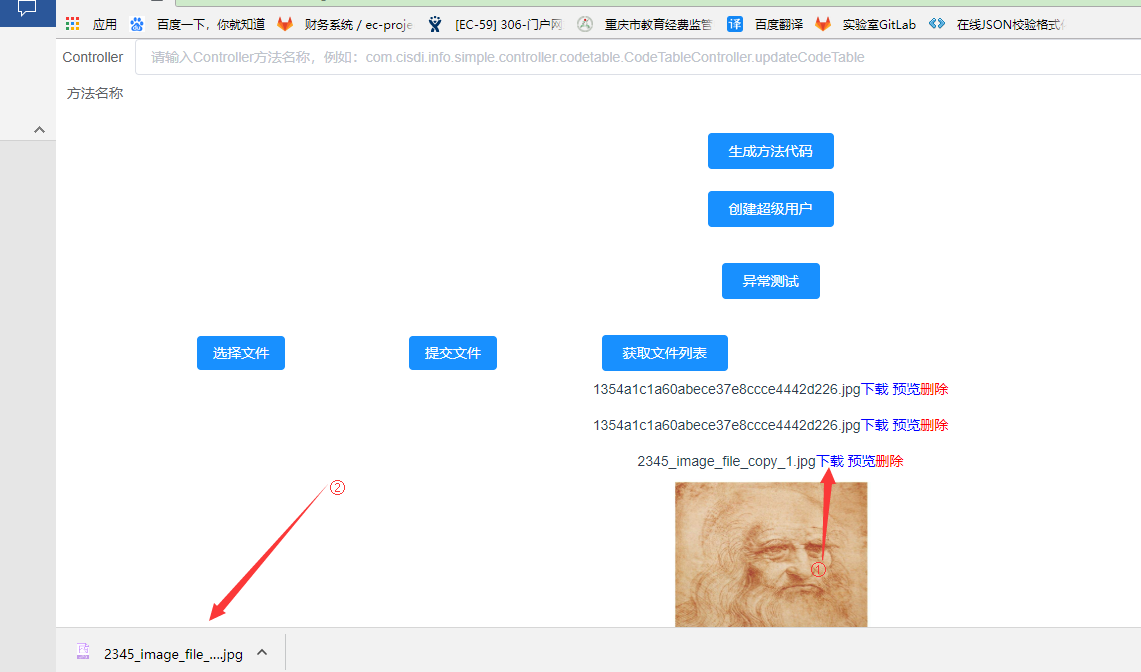


图2.2下载效果图

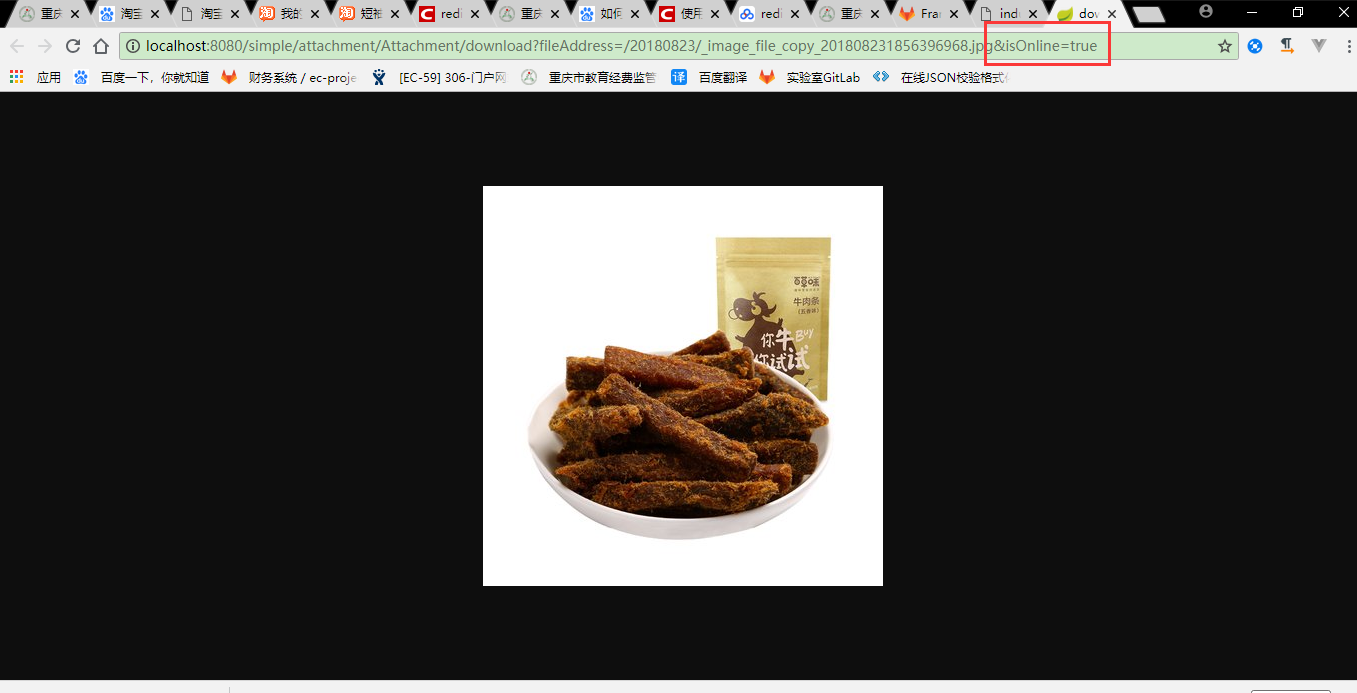


图2.2预览效果图（仅支持图片）