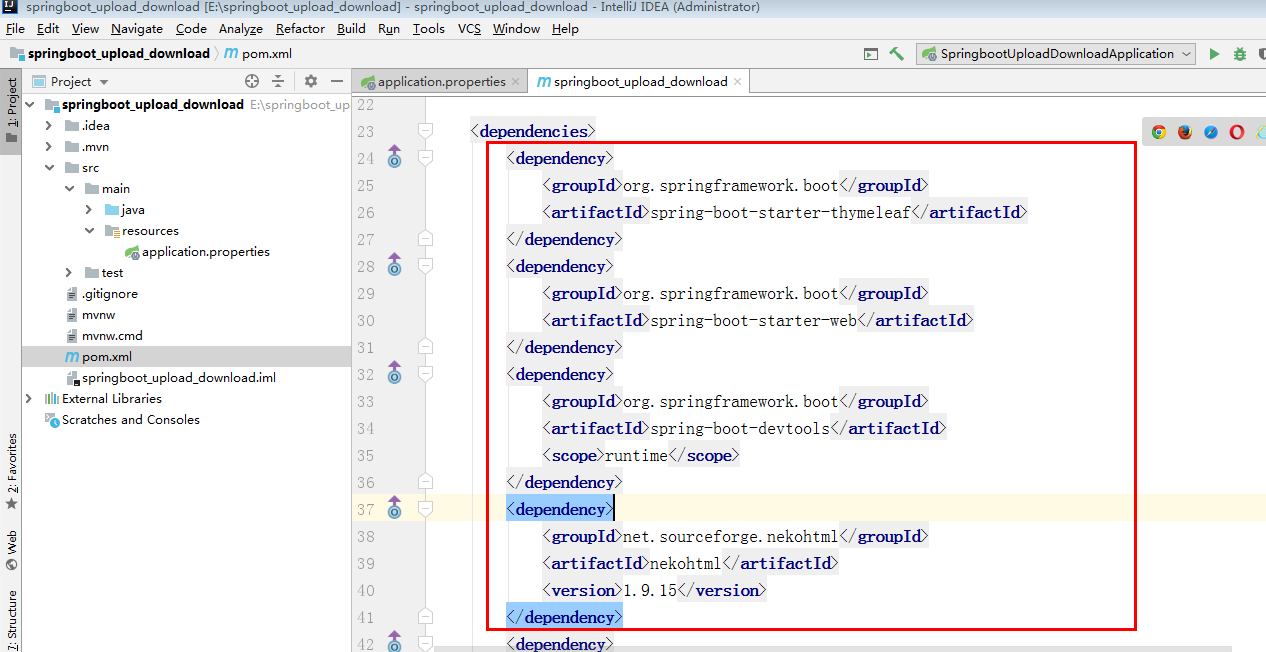
项目中经常会有上传和下载的需求，这篇文章简述一下springboot项目中实现简单的上传和下载。

新建springboot项目，前台页面使用的thymeleaf模板，其余的没有特别的配置，pom代码如下：

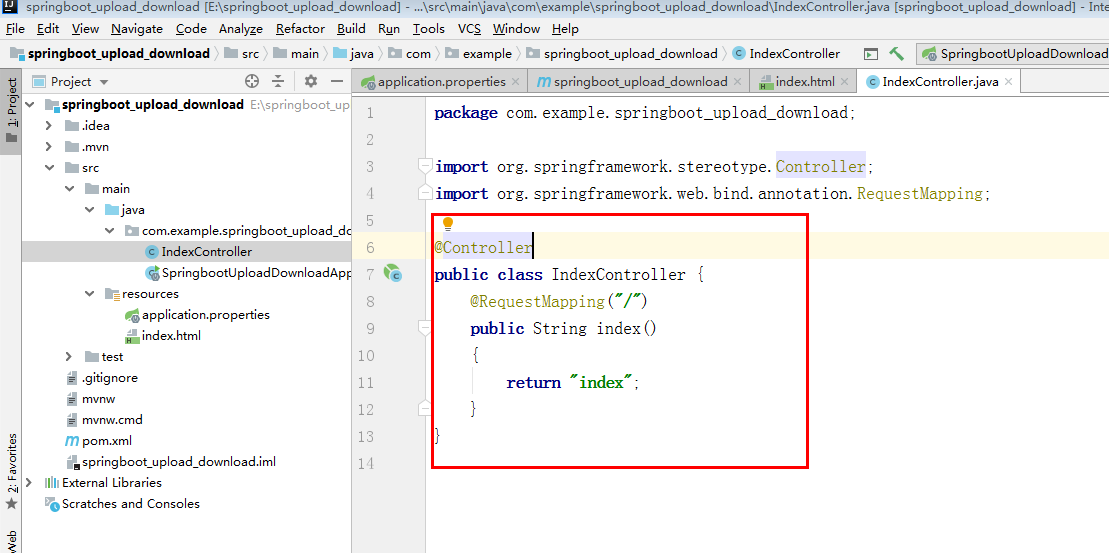


前台页面index.html，其中包含单个上传，下载，批量上传。

<!DOCTYPE **html**>  
<**html lang="en"**>  
<**head**>  
 <**meta charset="UTF-8"**>  
 <**title**>Title</**title**>  
</**head**>  
<**body**>  
<**p**>单文件上传</**p**>  
<**form action="upload" method="POST" enctype="multipart/form-data"**>  
 文件：<**input type="file" name="file"**/>  
 <**input type="submit"**/>  
</**form**>  
<**hr**/>  
<**p**>文件下载</**p**>  
<**a href="download"**>下载文件</**a**>  
<**hr**/>  
<**p**>多文件上传</**p**>  
<**form method="POST" enctype="multipart/form-data" action="batch"**>  
 <**p**>文件1：<**input type="file" name="file"**/></**p**>  
 <**p**>文件2：<**input type="file" name="file"**/></**p**>  
 <**p**><**input type="submit" value="上传"**/></**p**>  
</**form**>

</**body**>  
</**html**>

IndexController只是用来页面的跳转



最后是本文的重点，FileController，其中包含单个上传，单个下载，批量上传对应的方法。需要注意下载功能写的是对应我电脑里面固定位置的文件，仅供大家来参考。以下是代码：

**package** com.example.springboot\_upload\_download;  
  
**import** org.springframework.web.bind.annotation.GetMapping;  
**import** org.springframework.web.bind.annotation.PostMapping;  
**import** org.springframework.web.bind.annotation.RequestMapping;  
**import** org.springframework.web.bind.annotation.RequestParam;  
**import** org.springframework.web.multipart.MultipartFile;  
**import** org.springframework.web.multipart.MultipartHttpServletRequest;  
  
**import** javax.servlet.http.HttpServlet;  
**import** javax.servlet.http.HttpServletMapping;  
**import** javax.servlet.http.HttpServletResponse;  
**import** java.io.\*;  
**import** java.util.List;  
  
@RequestMapping  
**public class** FileController {  
  
 @RequestMapping(value = **"/upload"**)  
 **public** String upload(@RequestParam(**"file"**) MultipartFile file) {  
 **try** {  
 **if** (file.isEmpty()) {  
 **return "文件为空"**;  
 }  
 *// 获取文件名* String fileName = file.getOriginalFilename();  
 System.***out***.println(**"上传的文件名为："** + fileName);  
 *// 获取文件的后缀名* String suffixName = fileName.substring(fileName.lastIndexOf(**"."**));  
 System.***out***.println(**"文件的后缀名为："** + suffixName);  
 *// 设置文件存储路径* String filePath = **"/Users/dalaoyang/Downloads/"**;  
 String path = filePath + fileName;  
 File dest = **new** File(path);  
 *// 检测是否存在目录* **if** (!dest.getParentFile().exists()) {  
 dest.getParentFile().mkdirs();*// 新建文件夹* }  
 file.transferTo(dest);*// 文件写入* **return "上传成功"**;  
 } **catch** (IllegalStateException e) {  
 e.printStackTrace();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 **return "上传失败"**;  
 }  
  
 @PostMapping(**"/batch"**)  
 **public** String handleFileUpload(HttpServletMapping request) {  
 List<MultipartFile> files = ((MultipartHttpServletRequest) request).getFiles(**"file"**);  
 MultipartFile file = **null**;  
 BufferedOutputStream stream = **null**;  
 **for** (**int** i = 0; i < files.size(); ++i) {  
 file = files.get(i);  
 String filePath = **"/Users/dalaoyang/Downloads/"**;  
 **if** (!file.isEmpty()) {  
 **try** {  
 **byte**[] bytes = file.getBytes();  
 stream = **new** BufferedOutputStream(**new** FileOutputStream(  
 **new** File(filePath + file.getOriginalFilename())));*//设置文件路径及名字* stream.write(bytes);*// 写入* stream.close();  
 } **catch** (Exception e) {  
 stream = **null**;  
 **return "第 "** + i + **" 个文件上传失败 ==> "** + e.getMessage();  
 }  
 } **else** {  
 **return "第 "** + i  
 + **" 个文件上传失败因为文件为空"**;  
 }  
 }  
 **return "上传成功"**;  
 }  
  
 @GetMapping(**"/download"**)  
 **public** String downloadFile(HttpServlet request, HttpServletResponse response) {  
 String fileName = **"lhx.jpeg"**;*// 文件名* **if** (fileName != **null**) {  
 *//设置文件路径* File file = **new** File(**"/Users/dalaoyang/Documents/lhx.jpeg"**);  
 *//File file = new File(realPath , fileName);* **if** (file.exists()) {  
 response.setContentType(**"application/force-download"**);*// 设置强制下载不打开* response.addHeader(**"Content-Disposition"**, **"attachment;fileName="** + fileName);*// 设置文件名* **byte**[] buffer = **new byte**[1024];  
 FileInputStream fis = **null**;  
 BufferedInputStream bis = **null**;  
 **try** {  
 fis = **new** FileInputStream(file);  
 bis = **new** BufferedInputStream(fis);  
 OutputStream os = response.getOutputStream();  
 **int** i = bis.read(buffer);  
 **while** (i != -1) {  
 os.write(buffer, 0, i);  
 i = bis.read(buffer);  
 }  
 **return "下载成功"**;  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 } **finally** {  
 **if** (bis != **null**) {  
 **try** {  
 bis.close();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 **if** (fis != **null**) {  
 **try** {  
 fis.close();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 }  
 }  
 **return "下载失败"**;  
 }  
}