springboot(十八)：使用Spring Boot集成FastDFS

 2018/01/16

上篇文章介绍了[如何使用Spring Boot上传文件](http://www.ityouknow.com/springboot/2018/01/12/spring-boot-upload-file.html)，这篇文章我们介绍如何使用Spring Boot将文件上传到分布式文件系统FastDFS中。

这个项目会在上一个项目的基础上进行构建。

**1、pom包配置**

我们使用Spring Boot最新版本1.5.9、jdk使用1.8、tomcat8.0。

<dependency>

<groupId>org.csource</groupId>

<artifactId>fastdfs-client-java</artifactId>

<version>1.27-SNAPSHOT</version>

</dependency>

加入了fastdfs-client-java包，用来调用FastDFS相关的API。

**2、配置文件**

resources目录下添加fdfs\_client.conf文件

connect\_timeout **=** 60

network\_timeout **=** 60

charset **=** UTF**-**8

http**.**tracker\_http\_port **=** 8080

http**.**anti\_steal\_token **=** no

http**.**secret\_key **=** 123456

tracker\_server **=** 192.168**.**53.85**:**22122

tracker\_server **=** 192.168**.**53.86**:**22122

配置文件设置了连接的超时时间，编码格式以及tracker\_server地址等信息

详细内容参考：[fastdfs-client-java](https://github.com/happyfish100/fastdfs-client-java)

**3、封装FastDFS上传工具类**

封装FastDFSFile，文件基础信息包括文件名、内容、文件类型、作者等。

**public** **class** **FastDFSFile** **{**

**private** String name**;**

**private** **byte[]** content**;**

**private** String ext**;**

**private** String md5**;**

**private** String author**;**

*//省略getter、setter*

封装FastDFSClient类，包含常用的上传、下载、删除等方法。

首先在类加载的时候读取相应的配置信息，并进行初始化。

**static** **{**

**try** **{**

String filePath **=** **new** ClassPathResource**(**"fdfs\_client.conf"**).**getFile**().**getAbsolutePath**();;**

ClientGlobal**.**init**(**filePath**);**

trackerClient **=** **new** TrackerClient**();**

trackerServer **=** trackerClient**.**getConnection**();**

storageServer **=** trackerClient**.**getStoreStorage**(**trackerServer**);**

**}** **catch** **(**Exception e**)** **{**

logger**.**error**(**"FastDFS Client Init Fail!"**,**e**);**

**}**

**}**

**文件上传**

**public** **static** String**[]** **upload(**FastDFSFile file**)** **{**

logger**.**info**(**"File Name: " **+** file**.**getName**()** **+** "File Length:" **+** file**.**getContent**().**length**);**

NameValuePair**[]** meta\_list **=** **new** NameValuePair**[**1**];**

meta\_list**[**0**]** **=** **new** NameValuePair**(**"author"**,** file**.**getAuthor**());**

**long** startTime **=** System**.**currentTimeMillis**();**

String**[]** uploadResults **=** **null;**

**try** **{**

storageClient **=** **new** StorageClient**(**trackerServer**,** storageServer**);**

uploadResults **=** storageClient**.**upload\_file**(**file**.**getContent**(),** file**.**getExt**(),** meta\_list**);**

**}** **catch** **(**IOException e**)** **{**

logger**.**error**(**"IO Exception when uploadind the file:" **+** file**.**getName**(),** e**);**

**}** **catch** **(**Exception e**)** **{**

logger**.**error**(**"Non IO Exception when uploadind the file:" **+** file**.**getName**(),** e**);**

**}**

logger**.**info**(**"upload\_file time used:" **+** **(**System**.**currentTimeMillis**()** **-** startTime**)** **+** " ms"**);**

**if** **(**uploadResults **==** **null)** **{**

logger**.**error**(**"upload file fail, error code:" **+** storageClient**.**getErrorCode**());**

**}**

String groupName **=** uploadResults**[**0**];**

String remoteFileName **=** uploadResults**[**1**];**

logger**.**info**(**"upload file successfully!!!" **+** "group\_name:" **+** groupName **+** ", remoteFileName:" **+** " " **+** remoteFileName**);**

**return** uploadResults**;**

**}**

使用FastDFS提供的客户端storageClient来进行文件上传，最后将上传结果返回。

根据groupName和文件名获取文件信息。

**public** **static** FileInfo **getFile(**String groupName**,** String remoteFileName**)** **{**

**try** **{**

storageClient **=** **new** StorageClient**(**trackerServer**,** storageServer**);**

**return** storageClient**.**get\_file\_info**(**groupName**,** remoteFileName**);**

**}** **catch** **(**IOException e**)** **{**

logger**.**error**(**"IO Exception: Get File from Fast DFS failed"**,** e**);**

**}** **catch** **(**Exception e**)** **{**

logger**.**error**(**"Non IO Exception: Get File from Fast DFS failed"**,** e**);**

**}**

**return** **null;**

**}**

**下载文件**

**public** **static** InputStream **downFile(**String groupName**,** String remoteFileName**)** **{**

**try** **{**

storageClient **=** **new** StorageClient**(**trackerServer**,** storageServer**);**

**byte[]** fileByte **=** storageClient**.**download\_file**(**groupName**,** remoteFileName**);**

InputStream ins **=** **new** ByteArrayInputStream**(**fileByte**);**

**return** ins**;**

**}** **catch** **(**IOException e**)** **{**

logger**.**error**(**"IO Exception: Get File from Fast DFS failed"**,** e**);**

**}** **catch** **(**Exception e**)** **{**

logger**.**error**(**"Non IO Exception: Get File from Fast DFS failed"**,** e**);**

**}**

**return** **null;**

**}**

**删除文件**

**public** **static** **void** **deleteFile(**String groupName**,** String remoteFileName**)**

**throws** Exception **{**

storageClient **=** **new** StorageClient**(**trackerServer**,** storageServer**);**

**int** i **=** storageClient**.**delete\_file**(**groupName**,** remoteFileName**);**

logger**.**info**(**"delete file successfully!!!" **+** i**);**

**}**

使用FastDFS时，直接调用FastDFSClient对应的方法即可。

**4、编写上传控制类**

从MultipartFile中读取文件信息，然后使用FastDFSClient将文件上传到FastDFS集群中。

**public** String **saveFile(**MultipartFile multipartFile**)** **throws** IOException **{**

String**[]** fileAbsolutePath**={};**

String fileName**=**multipartFile**.**getOriginalFilename**();**

String ext **=** fileName**.**substring**(**fileName**.**lastIndexOf**(**"."**)** **+** 1**);**

**byte[]** file\_buff **=** **null;**

InputStream inputStream**=**multipartFile**.**getInputStream**();**

**if(**inputStream**!=null){**

**int** len1 **=** inputStream**.**available**();**

file\_buff **=** **new** **byte[**len1**];**

inputStream**.**read**(**file\_buff**);**

**}**

inputStream**.**close**();**

FastDFSFile file **=** **new** FastDFSFile**(**fileName**,** file\_buff**,** ext**);**

**try** **{**

fileAbsolutePath **=** FastDFSClient**.**upload**(**file**);** *//upload to fastdfs*

**}** **catch** **(**Exception e**)** **{**

logger**.**error**(**"upload file Exception!"**,**e**);**

**}**

**if** **(**fileAbsolutePath**==null)** **{**

logger**.**error**(**"upload file failed,please upload again!"**);**

**}**

String path**=**FastDFSClient**.**getTrackerUrl**()+**fileAbsolutePath**[**0**]+** "/"**+**fileAbsolutePath**[**1**];**

**return** path**;**

**}**

请求控制，调用上面方法saveFile()。

**@PostMapping(**"/upload"**)** *//new annotation since 4.3*

**public** String **singleFileUpload(@RequestParam(**"file"**)** MultipartFile file**,**

RedirectAttributes redirectAttributes**)** **{**

**if** **(**file**.**isEmpty**())** **{**

redirectAttributes**.**addFlashAttribute**(**"message"**,** "Please select a file to upload"**);**

**return** "redirect:uploadStatus"**;**

**}**

**try** **{**

*// Get the file and save it somewhere*

String path**=**saveFile**(**file**);**

redirectAttributes**.**addFlashAttribute**(**"message"**,**

"You successfully uploaded '" **+** file**.**getOriginalFilename**()** **+** "'"**);**

redirectAttributes**.**addFlashAttribute**(**"path"**,**

"file path url '" **+** path **+** "'"**);**

**}** **catch** **(**Exception e**)** **{**

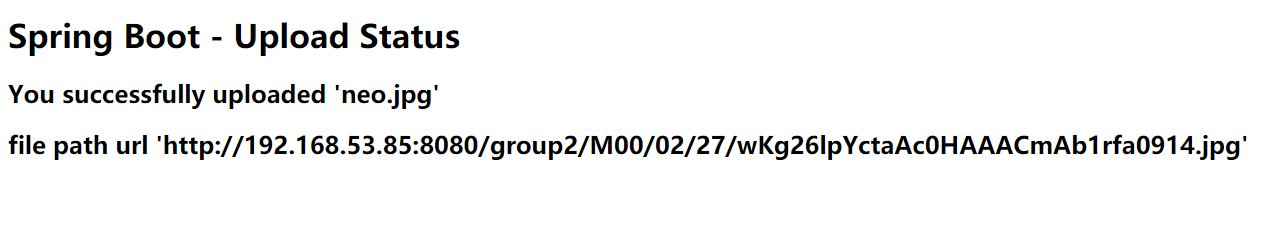
logger**.**error**(**"upload file failed"**,**e**);**

**}**

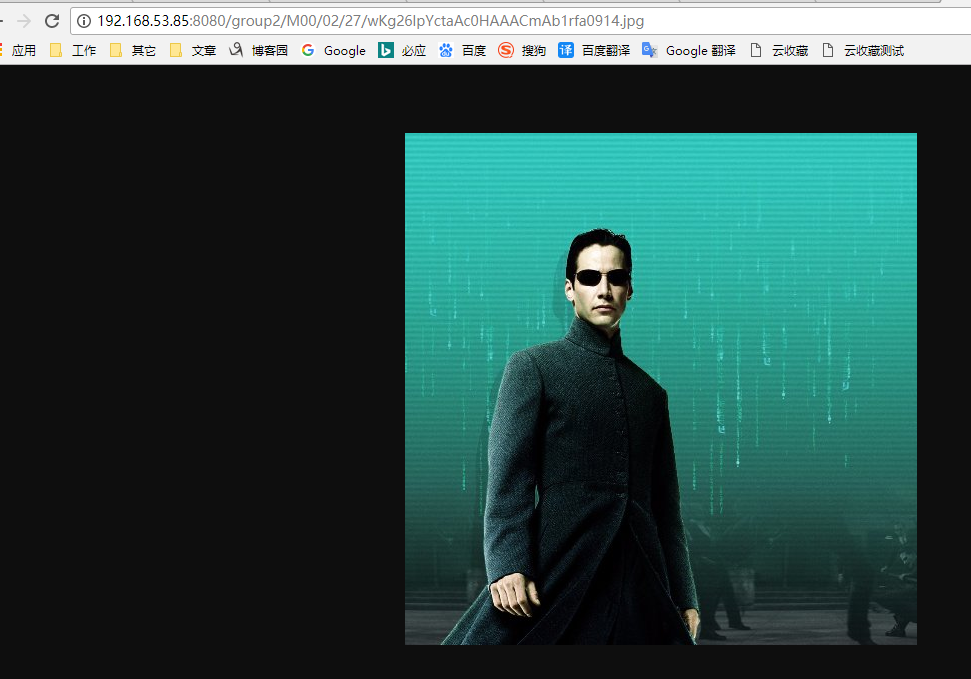
**return** "redirect:/uploadStatus"**;**

**}**

上传成功之后，将文件的路径展示到页面，效果图如下：



在浏览器中访问此Url，可以看到成功通过FastDFS展示：



这样使用Spring Boot 集成FastDFS的案例就完成了。