

乐字节教育高级架构课程

正所谓“授人以鱼不如授人以渔”，你们想要的 **Java 学习资料** 来啦！

不管你是学生，还是已经步入职场的同行，希望你们都要珍惜眼前的学习机会，奋斗没有终点，知识永不过时。

扫描下方二维码即可领取



乐字节晓啡



乐字节官方交流群



后台代码实现

配置shop-parent项目的pom.xml

```
<!-- 七牛云sdk 依赖-->
<qiniu.version>[7.2.0, 7.2.99]</qiniu.version>
```

```
<!--七牛云SKD-->
<dependency>
  <groupId>com.qiniu</groupId>
  <artifactId>qiniu-java-sdk</artifactId>
  <version>${qiniu.version}</version>
</dependency>
```

配置shop-common项目的pom.xml

```
<!-- 七牛云sdk 依赖 -->
<dependency>
  <groupId>com.qiniu</groupId>
  <artifactId>qiniu-java-sdk</artifactId>
</dependency>
```

shop-common项目编写FileResult.java文件返回对象

```
package com.xxxx.common.result;

import java.io.Serializable;

/**
 * @see(功能介绍) : 公共的文件上传返回对象
 * @version(版本号) : 1.0
 * @author(创建人) : zhoubin
 * @since : JDK 1.8
 */
public class FileResult implements Serializable {

    // success字符串bootstrap file input必须包含该属性
    private String success;
    // error字符串bootstrap file input必须包含该属性
    private String error;
    // 描述信息
    private String message;
    // 文件路径
    private String fileUrl;

    public String getSuccess() {
        return success;
    }
}
```

```
public void setSuccess(String success) {
    this.success = success;
}

public String getError() {
    return error;
}

public void setError(String error) {
    this.error = error;
}

public String getMessage() {
    return message;
}

public void setMessage(String message) {
    this.message = message;
}

public String getFileUrl() {
    return fileUrl;
}

public void setFileUrl(String fileUrl) {
    this.fileUrl = fileUrl;
}
}
```

shop-manager项目application.yml配置文件

```
# Spring
spring:
  servlet:
    multipart:
      # 单个文件上限
      max-file-size: 100MB
      # 单次请求所有文件总上限
      max-request-size: 1000MB
```

Service服务层

UploadService.java

```
package com.xxxx.manager.service;

import com.xxxx.common.result.FileResult;

import java.io.InputStream;
```

/**

```
* 上传服务
*/
public interface UploadService {
    /**
     * 上传服务
     * @param inputStream
     * @param fileName
     * @return
     */
    FileResult upload(InputStream inputStream, String fileName);
}
```

UploadServiceImpl.java

```
package com.xxxx.manager.service.impl;

import com.qiniu.common.QiniuException;
import com.qiniu.http.Response;
import com.qiniu.storage.Configuration;
import com.qiniu.storage.Region;
import com.qiniu.storage.UploadManager;
import com.qiniu.util.Auth;
import com.xxxx.common.result.FileResult;
import com.xxxx.manager.service.UploadService;
import org.springframework.stereotype.Service;

import java.io.InputStream;

/**
 * 上传服务实现类
 *
 * @author zhoubin
 * @since 1.0.0
 */
@Service
public class UploadServiceImpl implements UploadService {

    @Override
    public FileResult upload(InputStream inputStream, String fileName) {
        FileResult fileResult = new FileResult();
        //构造一个带指定 Region 对象的配置类
        Configuration cfg = new Configuration(Region.region0());
        //...其他参数参考类注释
        UploadManager uploadManager = new UploadManager(cfg);
        //...生成上传凭证, 然后准备上传
        String accessKey = "CWAps72aLopYzpOhZ_fm7Qsbx1-tsaXxw5nvuQ0";
        String secretKey = "yywBB12AvRWjYcpcnM-ehMmw5e-zxZD4U_h1V0wx";
        String bucket = "shop-test-demo01";
        //默认不指定key的情况下, 以文件内容的hash值作为文件名
        String key = fileName;
        try {
            Auth auth = Auth.create(accessKey, secretKey);
            String upToken = auth.uploadToken(bucket);
```

```

try {
    Response response = uploadManager.put(inputStream, key, upToken, null, null);
    //解析上传成功的结果
    if (response.getStatusCode()==200){
        fileResult.setSuccess("success");
        fileResult.setMessage("上传成功");
        fileResult.setFileUrl("http://q63yxp74h.bkt.clouddn.com/"+fileName);
        return fileResult;
    }else {
        fileResult.setError("error");
        fileResult.setMessage("上传失败");
        return fileResult;
    }
} catch (QiniuException ex) {
    Response r = ex.response;
    System.err.println(r.toString());
    fileResult.setError("error");
    fileResult.setMessage("上传失败");
    try {
        System.err.println(r.bodyString());
    } catch (QiniuException ex2) {
        //ignore
    }
}
} catch (Exception ex) {
    //ignore
}
return fileResult;
}
}

```

Controller控制层

UploadController.java

```

package com.xxxx.manager.controller;

import com.xxxx.common.result.FileResult;
import com.xxxx.manager.service.UploadService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.multipart.MultipartFile;

import java.io.IOException;
import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;

/**
 * 上传Controller
 *
 * @author zhoubin

```

```
* @since 1.0.0
*/
@Controller
@RequestMapping("fileupload")
public class UploadController {

    @Autowired
    private UploadService uploadService;

    @RequestMapping("save")
    @ResponseBody
    public FileResult upload(MultipartFile file) throws IOException {
        String filename = file.getOriginalFilename();
        String date =
DateTimeFormatter.ofPattern("yyyy/MM/dd/").format(LocalDateDateTime.now());
        filename =
date+System.currentTimeMillis()+filename.substring(filename.lastIndexOf("."));
        return uploadService.upload(file.getInputStream(),filename);
    }
}
```

