

Please visit website: <http://cxyroad.com>

Spring Boot整合阿里云OSS文件上传

=====

第一部分：准备工作

1. 注册阿里云账号： 如果尚未拥有阿里云账号，需要先注册一个账号并创建一个 OSS 存储空间。

2. 获取Access Key和Secret Key： 在阿里云控制台中创建一个 Access Key 和 Secret Key，用于访问 OSS。

第二部分：添加依赖

在Spring Boot项目的`pom.xml`文件中添加阿里云OSS的依赖：

```
...
<!--oss对象存储-->
<dependency>
  <groupId>com.alibaba.cloud</groupId>
  <artifactId>spring-cloud-starter-alicloud-oss</artifactId>
  <version>2.2.0.RELEASE</version>
</dependency>
...
```


第三部分：配置阿里云OSS

```
...
spring:
  cloud:
    alicloud:
```

```
secret-key:
access-key:
oss:
  endpoint:
  bucket:
util:
  enabled: false
```

...

第四部分：创建Controller

创建一个Controller类，用于处理文件上传请求：

...

```
package com.atguigu.gulimall.thirdparty.controller;

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.common.utils.BinaryUtil;
import com.aliyun.oss.model.MatchMode;
import com.aliyun.oss.model.PolicyConditions;
import com.atguigu.common.utils.R;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;
import org.springframework.web.multipart.MultipartFile;

import java.io.IOException;
import java.io.InputStream;
import java.nio.charset.StandardCharsets;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.LinkedHashMap;
import java.util.Map;

@RequestMapping("/api/files")
public class OssController {
```

```

@Autowired
private OSS ossClient;

@Value("${spring.cloud.alicloud.oss.endpoint}")
private String endpoint;

@Value("${spring.cloud.alicloud.secret-key}")
private String secretKey;

@Value("${spring.cloud.alicloud.access-key}")
private String accessId;

@Value("${spring.cloud.alicloud.oss.bucket}")
private String bucket;

/**
 * 文件流上传
 * @param file
 * @return
 */
@PostMapping("/upload")
public String uploadFile(@RequestParam("file") MultipartFile file) {
    String objectName = "gulimall-basc/";
    try {
        InputStream inputStream = file.getInputStream();
        String originalFilename = file.getOriginalFilename();
        objectName=objectName+originalFilename;
        // 上传文件到 OSS
        ossClient.putObject(bucket, objectName, inputStream);
        System.out.println("文件上传成功");
    } catch (OSSEException oe) {
        // 处理 OSS 异常
        return R.error("OSS上传失败");
    } catch (ClientException ce) {
        // 处理 Client 异常
        System.out.println("Client Error Message:" + ce.getMessage());
        return R.error("OSS上传失败");
    } catch (IOException e) {
        e.printStackTrace();
    }
    String url="https://" + bucket + "." + endpoint + "/" + objectName;
    return R.ok().put("data", url);
}

/**
 * Oss 获取服务端签名, 直传
 * @return
 */
@RequestMapping("/oss/policy")

```

```

public R policy() {
    // https://gulimall-hello.oss-cn-beijing.aliyuncs.com/hahaha.jpg
    host的格式为 bucketname.endpoint
    String host = "https://" + bucket + "." + endpoint;
    // callbackUrl为 上传回调服务器的URL, 请将下面的IP和Port配置为您
    自己的真实信息。
    // String callbackUrl = "http://88.88.88.88:8888";
    String format = new SimpleDateFormat("yyyy-MM-
    dd").format(new Date());
    // 用户上传文件时指定的前缀。
    String dir = "gulimall-basc/" + format + "/";

    Map<String, String> respMap = null;
    try {
        long expireTime = 30;
        long expireEndTime = System.currentTimeMillis() + expireTime *
    1000;
        Date expiration = new Date(expireEndTime);
        PolicyConditions policyConds = new PolicyConditions();
        policyConds.addConditionItem(PolicyConditions.COND_CONTENT LENG
    TH_RANGE, 0, 1048576000);
        policyConds.addConditionItem(MatchMode.StartWith,
    PolicyConditions.COND_KEY, dir);

        String postPolicy = ossClient.generatePostPolicy(expiration,
    policyConds);
        byte[] binaryData =
    postPolicy.getBytes(StandardCharsets.UTF_8);
        String encodedPolicy = BinaryUtil.toBase64String(binaryData);
        String postSignature =
    ossClient.calculatePostSignature(postPolicy);

        respMap = new LinkedHashMap<String, String>();
        respMap.put("accessid", accessId);
        respMap.put("policy", encodedPolicy);
        respMap.put("signature", postSignature);
        respMap.put("dir", dir);
        respMap.put("host", host);
        respMap.put("expire", String.valueOf(expireEndTime / 1000));
        // respMap.put("expire", formatISO8601Date(expiration));

    } catch (Exception e) {
        // Assert.fail(e.getMessage());
        System.out.println(e.getMessage());
    }

    return R.ok().put("data", respMap);
}

```

```
}  
}  
...
```


第六部分：前端页面实现

在前端页面中添加文件上传的表单，使用Ajax异步上传文件：

```
...  
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta http-equiv="X-UA-Compatible" content="IE=edge">  
  <meta name="viewport" content="width=device-width, initial-  
scale=1.0">  
  <title>File Upload</title>  
</head>  
<body>  
  <form id="fileUploadForm" enctype="multipart/form-data">  
    <input type="file" name="file" />  
    <button type="button" onclick="uploadFile()">Upload</button>  
</form>  
  
  <script>  
    function uploadFile() {  
      var form = document.getElementById('fileUploadForm');  
      var formData = new FormData(form);  
  
      fetch('/api/files/upload', {  
        method: 'POST',  
        body: formData  
      })  
      .then(response => response.text())  
      .then(data => {  
        console.log('File uploaded:', data);  
      })  
      .catch(error => {  
        console.error('Error uploading file:', error);  
      });  
    }  
  }  
</script>  
</body>  
</html>
```

```
</script>  
</body>  
</html>
```

...

结语:

通过以上步骤，我们成功地在Spring Boot应用中集成了阿里云OSS文件上传功能。用户可以通过前端页面选择文件并

原文链接: <https://juejin.cn/post/7356525427564101666>