# Oss上传与下载

## OSS上传

### maven依赖

<!-- 阿里云OSS-->

<dependency>

<groupId>com.aliyun.oss</groupId>

<artifactId>aliyun-sdk-oss</artifactId>

<version>2.4.0</version>

</dependency>

### AliyunOSSUtil.java

@Component

public class AliyunOSSUtil {

private static final org.slf4j.Logger logger = LoggerFactory.getLogger(AliyunOSSUtil.class);

/\*\*

\* 上传文件

\*/

public String upLoad(File file) {

logger.info("------OSS文件上传开始--------" + file.getName());

String endpoint = "你的endpoint ";

//这里endpoint 在你的bucket列表->点击你的bucket->点击概览中间就有，下面有截图

System.out.println("获取到的Point为:" + endpoint);

String accessKeyId = "你的accessKeyId "; //accessKeyId 、accessKeySecret 上面有说到哪里获取

String accessKeySecret = "你的accessKeySecret ";

String bucketName = "你的bucketName "; //刚才新建的bucket名称

String fileHost = "你的fileHost "; //在刚才新建的bucket下面新建一个目录，这就是那个目录的名称

SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");

String dateStr = format.format(new Date());

// 判断文件

if (file == null) {

return null;

}

OSSClient client = new OSSClient(endpoint, accessKeyId, accessKeySecret);

try {

// 判断容器是否存在,不存在就创建

if (!client.doesBucketExist(bucketName)) {

client.createBucket(bucketName);

CreateBucketRequest createBucketRequest = new CreateBucketRequest(bucketName);

createBucketRequest.setCannedACL(CannedAccessControlList.PublicRead);

client.createBucket(createBucketRequest);

}

// 设置文件路径和名称

String fileUrl = fileHost + "/" + (dateStr + "/" + UUID.randomUUID().toString().replace("-", "") + "-" + file.getName());

// 上传文件

PutObjectResult result = client.putObject(new PutObjectRequest(bucketName, fileUrl, file));

// 设置权限(公开读)

client.setBucketAcl(bucketName, CannedAccessControlList.PublicRead);

if (result != null) {

logger.info("------OSS文件上传成功------" + "https://makeromance.oss-cn-hangzhou.aliyuncs.com/" + fileUrl);

}

} catch (OSSException oe) {

logger.error(oe.getMessage());

} catch (ClientException ce) {

logger.error(ce.getErrorMessage());

} finally {

if (client != null) {

client.shutdown();

}

}

return null;

}

}

### UpLoadController.java

@Controller

public class UpLoadController {

private static final String TO\_PATH = "upload";

private static final String RETURN\_PATH = "success";

@Autowired

private AliyunOSSUtil aliyunOSSUtil;

@RequestMapping("/toUpLoadFile")

public String toUpLoadFile() {

return TO\_PATH;

}

/\*\*

\* 文件上传

\*/

@RequestMapping(value = "/uploadFile")

public String uploadBlog(@RequestParam("file") MultipartFile file) {

String filename = file.getOriginalFilename();

System.out.println(filename + "==filename");

try {

if (file != null) {

if (!"".equals(filename.trim())) {

File newFile = new File(filename);

FileOutputStream os = new FileOutputStream(newFile);

os.write(file.getBytes());

os.close();

file.transferTo(newFile);

// 上传到OSS

String uploadUrl = aliyunOSSUtil.upLoad(newFile);

}

}

} catch (Exception ex) {

ex.printStackTrace();

}

return RETURN\_PATH;

}

}

注释：Bucket、Access Key,来获取accessKeyId、accessKeySecret这两个参数