# 第三次大作业使用说明书

班级：18级软件工程1班 姓名：许航 学号：201830663360

**一、引言**

本文档用于指导用户了解本次大作业操作流程。

**二、系统配置**

软件：JDK 1.8、IntelliJ IDEA 2020.1.1(Commuity Editon)、flink dependencies.zip、scala-2.11.8

操作系统：windows 10

下载项目所包含的两个工程导入IntelliJ IDEA中使用。

homework3工程需要在创建maven项目和配置maven环境且项目可以成功bulid后导入使用。

local\_to\_s3工程为一般的java工程。

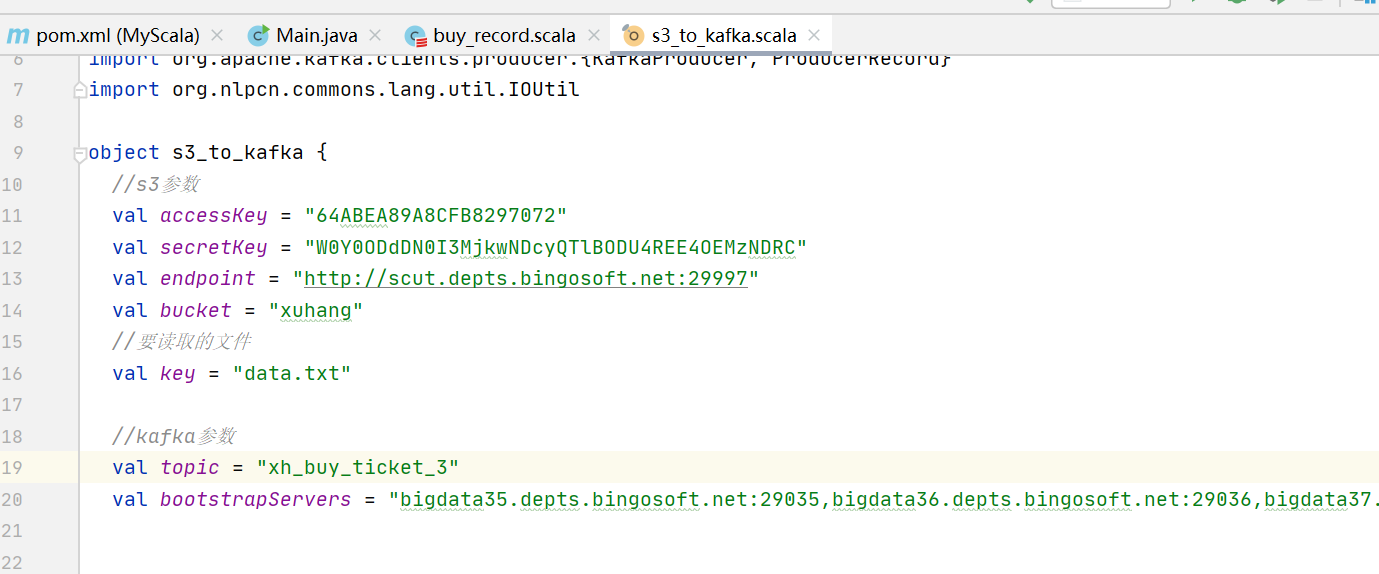
两个工程均需要导入flink dependencies.zip中的依赖，homework3需要添加scala-sdk-2.11.8。

**三、操作流程**

**\*前置要求：需要配置好maven环境**

1、将s3数据接入到kafka生产者，通过flink程序计算归类后将计算结果保存于本地。

（1）打开maven项目homework3中修改s3\_to\_kafka.scala中的参数



1. //s3参数
2. val accessKey = "64ABEA89A8CFB8297072"
3. val secretKey = "W0Y0ODdDN0I3MjkwNDcyQTlBODU4REE4OEMzNDRC"
4. val endpoint = "http://scut.depts.bingosoft.net:29997"
5. val bucket = "xuhang"
6. //要读取的文件
7. val key = "data.txt"
9. //kafka参数
10. val topic = "xh\_buy\_ticket\_3"

s3参数修改为用户自己的授权信息

bucket为s3桶中需要读取的数据所存放桶名称

topic为想要生产数据的kafka topic（确保该topic无其他的内容）

data.txt为事先上传于s3桶中的文件（已将文件备份保存与该文档同一目录下）

本使用说明所使用的data数据集为如下格式：

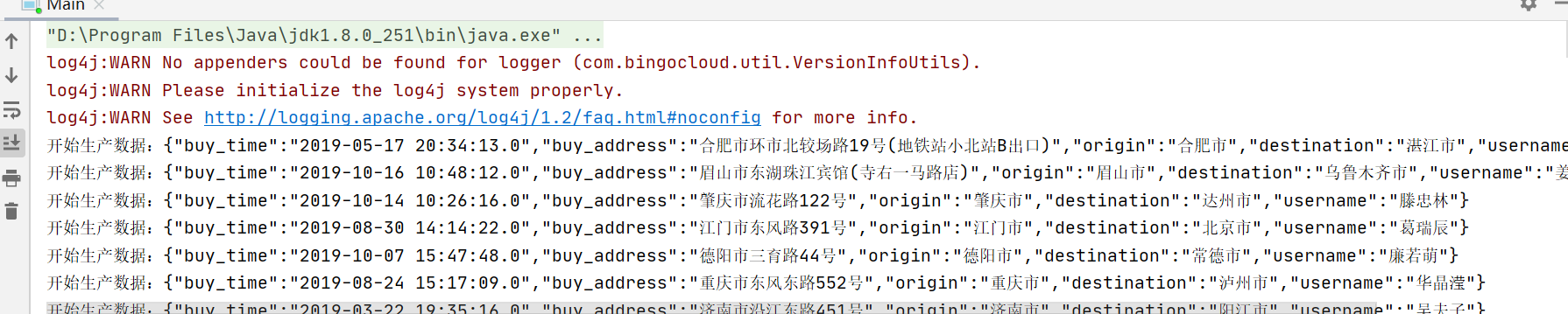


（2）修改Main.java的topic，修改为和上一步一致的topic，以确保消费的数据确实是s3导入的数据。

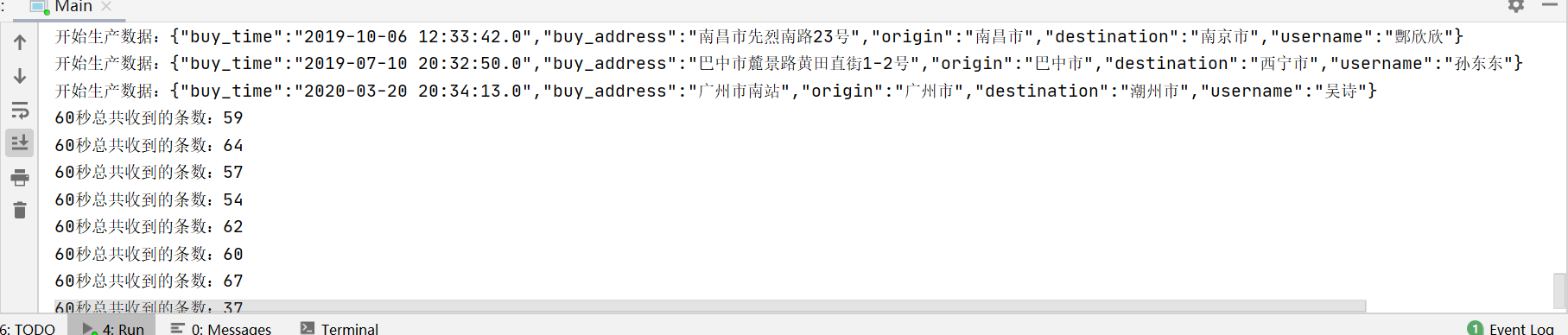
同时修改想要保存的文件路径（确保本地目录下不存在该目录）



（3）运行Main.java

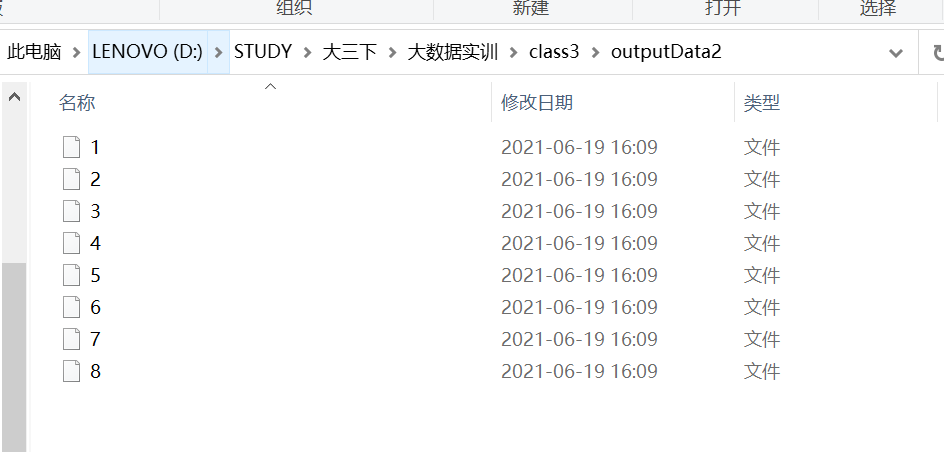


可以看到程序先将数据生产到相应kafka topic中。

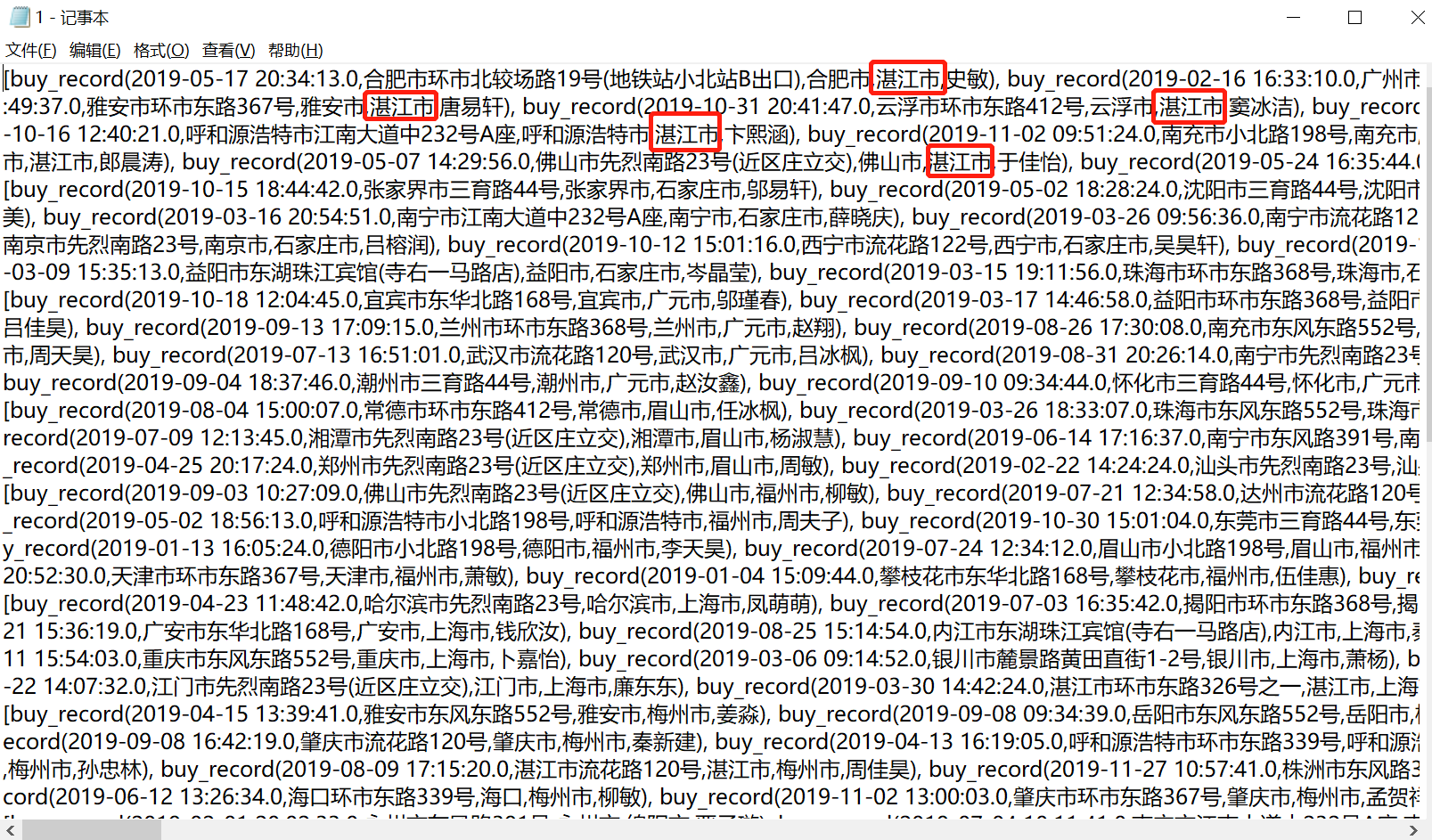


然后flink开始分组归类后写入本地文件目录。

到本地目录查看到outputData2内文件生成之后可以关闭当前程序。



检查分组的结果：

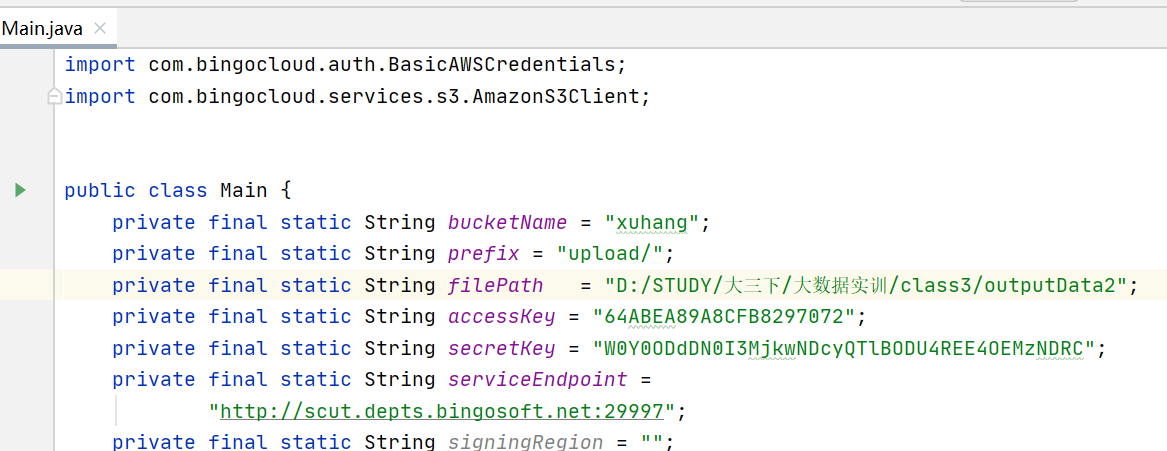


可以看到相同destination的购买记录被分在了同一个[]中。

输出8个分区的原因是flink根据keyby之后自动进行的分区为了方便其多线程进行运算操作。

2、将本地归类结果写入S3存储

（1）打开java项目Main.java修改参数



1. **private** **final** **static** String bucketName = "xuhang";
2. **private** **final** **static** String prefix = "upload/";
3. **private** **final** **static** String filePath   = "D:/STUDY/大三下/大数据实训/class3/outputData2";
4. **private** **final** **static** String accessKey = "64ABEA89A8CFB8297072";
5. **private** **final** **static** String secretKey = "W0Y0ODdDN0I3MjkwNDcyQTlBODU4REE4OEMzNDRC";
6. **private** **final** **static** String serviceEndpoint =
7. "http://scut.depts.bingosoft.net:29997";

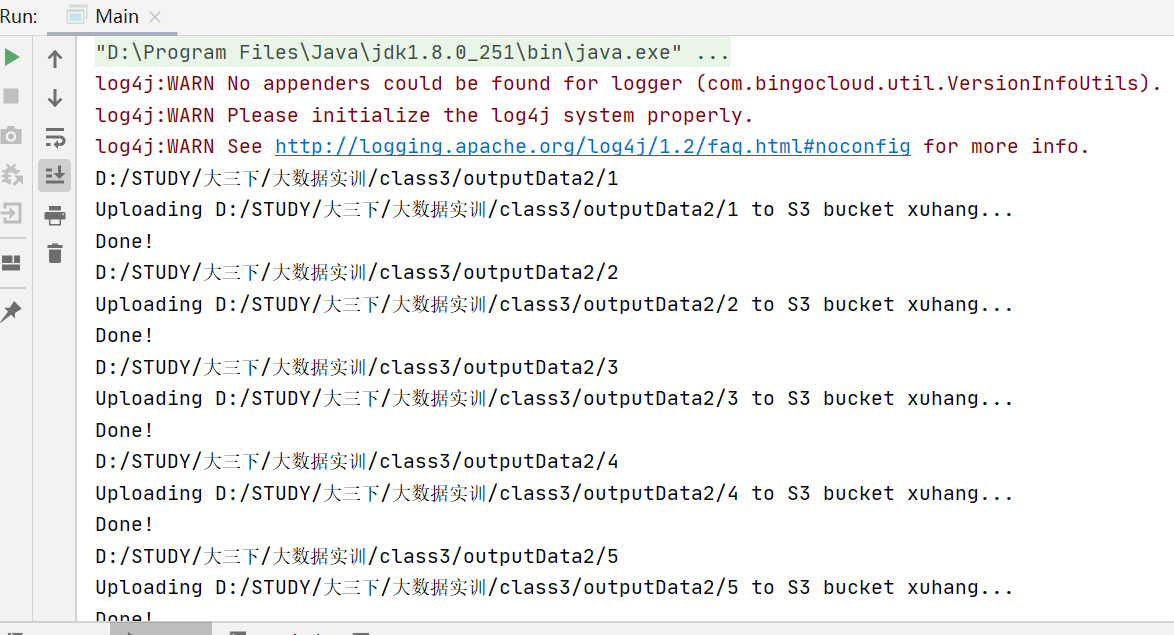
将accessKey、secretKey修改成用户自己的授权信息

filePath 设置成上一个项目中保存结果的文件路径

prefix 前缀想要上传结果的桶内的文件夹名

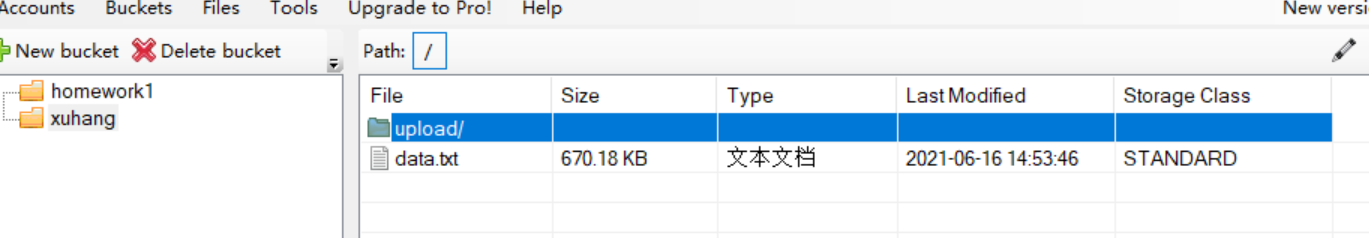
bucketName 想要上传结果的桶内

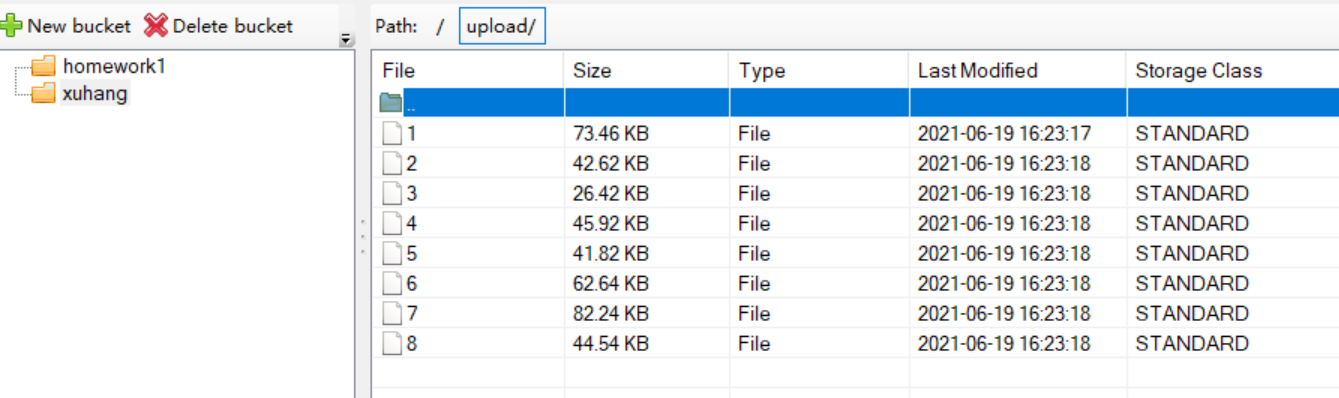
（2）运行Main.java



可以看到程序打印了上传文件的结果。

（3）在s3对应桶中检查上传结果





可以看到文件成功上传了。