**《Java技术》实验报告**

**实验二： 2021年7月16日**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学院** | | 计算机与信息学院 | | **专业班级** | | 计算机20-01 | **姓名** | 叶春亭 | |  | | **成绩** | |  |
| **课程**  **名称** | | Java技术 | | | **实验项目**  **名 称** | 实验二 货物进销管理系统 | | |  | | **指导 教师** | | 薛峰 | |
| 教师评语 |  | | 教师签名：  年 月 日 | | | | | | | | | | | |

**一．实验目的**

1．掌握Java中文件的读写操作。

2．学会使用Java提供的实用类（Vector, ArrayList）来完成特定的功能。

3．掌握字符串类（String, StringBuffer）的使用。

4．掌握用面向对象的方法分析和解决复杂问题。

**二、实验原理**

1、类与对象的应用

2、文件流的 应用

3、字符串类string的一些必要函数的应用

4、一些 arraylist类的数组在类中的应用 实例

**三、使用硬件、软件环境**

PC 计算机一台，配置为CPU为1.5G，内存为512M，硬盘为40G，安装Windows10操作系统。

另外，使用JCreator，JDK1.6.0等软件

**四、实验过程、步骤及原始记录(算法、原程序、测试结果，分析等)**

编写一个Inventory.java完成以下功能：

1．程序首先打开并读取Inventory.txt中记录的所有库存记录，然后读取Transactions.txt，处理这个文件中包含的事务，记录发货记录到Shipping.txt，并记录错误信息到Errors.txt中。最后更新库存到另外一个文件NewInventory.txt中。

2．文件Inventory.txt和NewInventory.txt的每行包含一个存货记录，没条记录包含下面一些字段息，这些字段之间用一个tab分开（见后面的文件格式）：

|  |  |
| --- | --- |
| 字段 | 格式和含义 |
| Item number | 字符串型，货物编号 |
| Quantity | 整型，货物数量 |
| Supplier | 字符串型，供应商编号 |
| Description | 字符串型，货物描述 |

3．字段Items按照从小到大的顺序写入文件的。注意Item号不必连续，如Item号为752的后面可能是800。

4．文件Transactions.txt包含几个不同的处理记录（每行一条记录）。每条记录前面以一个大写字母开头，表示录是什if么类型的事务。在不同的大写字母后面是不同的信息格式。所有的字段也是以tab键分开的（见Transactions.txt文件格式）。

5．以'O'开头（Order的首字母）的事务表示这是一个发货订单，即某一种货物应该发给特定的客户。Item number和Quantity的格式如上面表格定义。Custom编号和上面的Supplier编号一致。处理一条定单记录（以'O'开头的事务）意味着从库存记录中减少相应货物的数量（减少的数量=发货单中的数量），记录发货信息到Shipping.txt中。注意：Inventory.txt中的quantity不应该小于0，如果对于某一种货物，库存的数量小于发货单的数量的话，系统应该停止处理发货单，并记录出错信息到Errors.txt。如果对于某一种货物有多个发货单，而且库存总量小于这些发货单的总和的话，系统应该按照发货单中的数量从小到大的有限原则满足客户。也就是说，对于某一种货物如果一个数量Quantity少的发货单没有处理之前，数量Quantity多的发货单永远不会被处理。（这种处理原则不受发货单记录在Transactions.txt的先后顺序影响）

6．以'R'开头的事务表示这是一个到货单记录，在'R'后面是Item number和它的数量Quanlity。处理一条到货单意味着增加库存中相应货物的数量（增加的数量=到货单中的数量）。注意：如果在Transactions.txt文件中，到货单出现在发货单之后，到货单中的货物数量可以用来填补发货单中的数量（可以理解成在Transactions.txt中，优先处理到货单）。

7．以'A'开头的事务表示向库存中增加一种新的货物（即这种货物以前库存中没有），在'A'后面是Item number，供应商supplier以及货物的描述description。处理一个新增货物记录意味着向库存中增加一个数量Quantity为0的新的Item。你可以假设在一个Transactions.txt中，新增货物记录总是出现在第一个到货单之前。

8．以'D'开头的事务表示从库存中删除一种货物，在'D'后面是Item号。删除操作总是在所有的事物处理之后才被处理，以保证对于可能出现的同一种货物的发货单的操作能在删除之前被正确处理。如果要删除的某种货物的库存量Quantity不为0的话，系统应该向Errors.txt记录出错信息。

9．文件Shipping.txt中的每一行代表给某一客户的发货信息。Shipping.txt中的每一行分别是客户编号、Item号、货物数量，它们之间用tab键分隔。如果发货单中有两条客户编号和Item编号一样的记录，在Shipping.txt中应该将这两条发货信息合并（即将它们的数量相加）。

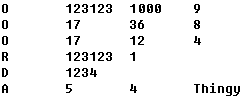
10．Errors.txt文件包含未发送的发货记录和库存量大于0的删除记录。Errors.txt每一行包含Custom编号、Item编号以及发货单上的数量Quantity。对于删除操作，Custom编号为0，数量Quntity为库存中的Quantity.

11．实验测试数据:

Inventory.txt



Transactions.txt



实验思路：

1、首先将题目中的两个文件Inventory.txt 和transaction.txt储存在该java文件的包中一遍读取。

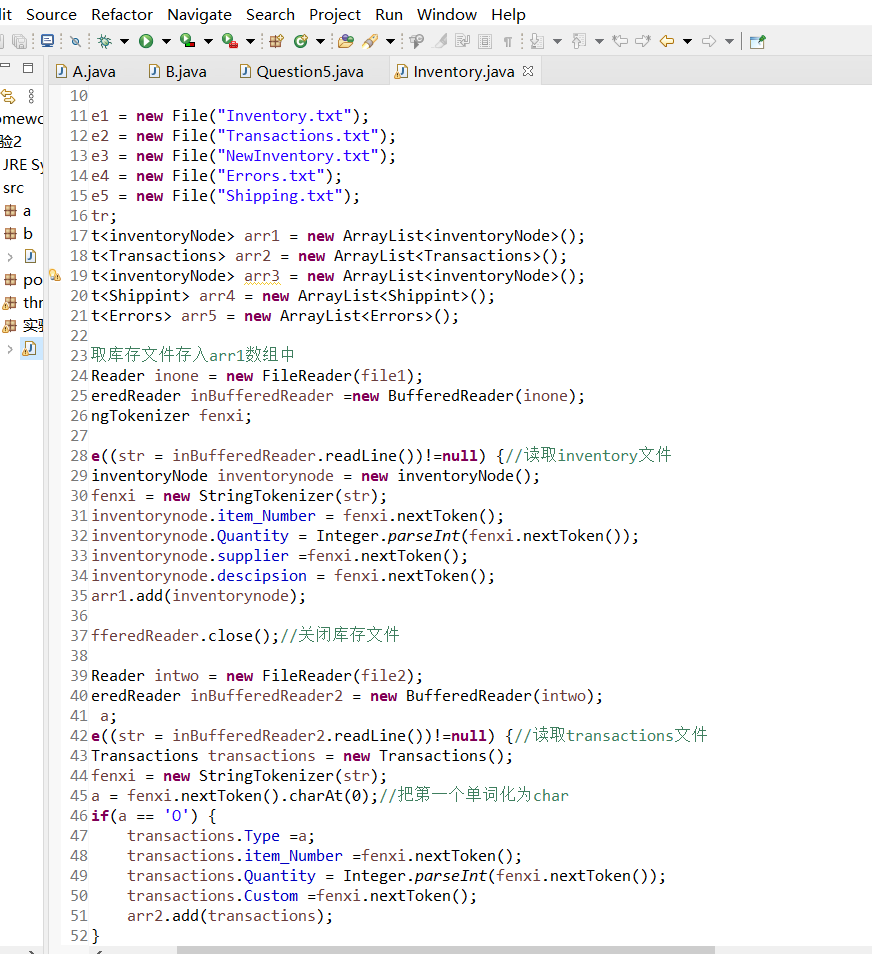
2、创建每一个文件每一条多需要的元素，并将每一行文件的所有类型数据封装成一个类，总共有五个文件，也就是有五种结点，每个结点代表一行文件的数据，如下图：

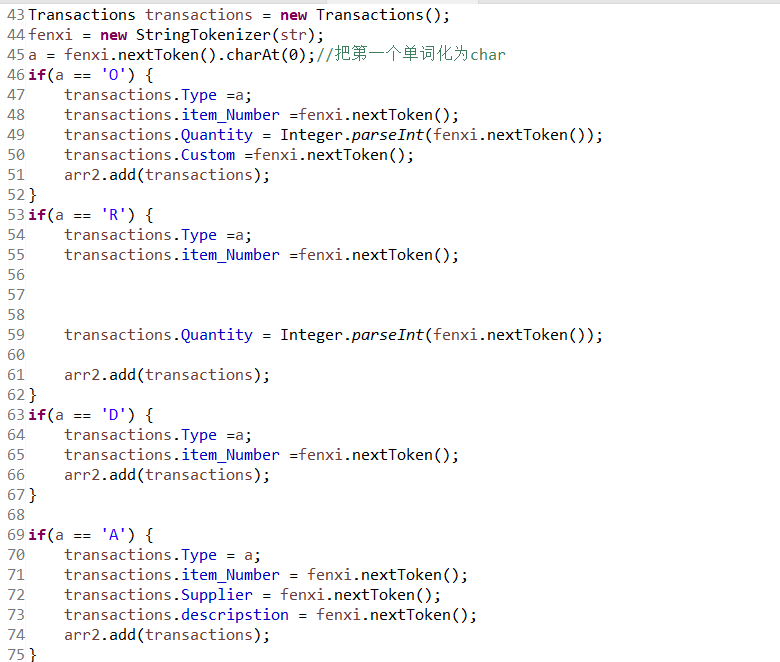


3、然后用File类读取五个文件，(新建三个)。用五个arraylist类封装五个文件的内容。

之后读取两个文件中的内容，transaction中的类型，并依ORAD四种操作分类。存进两个链表中。

如图





4、在事件和 库存存入数组中后进行每一个事件的处理

并结合实际情况对所有时间做一个先后顺序的排序：

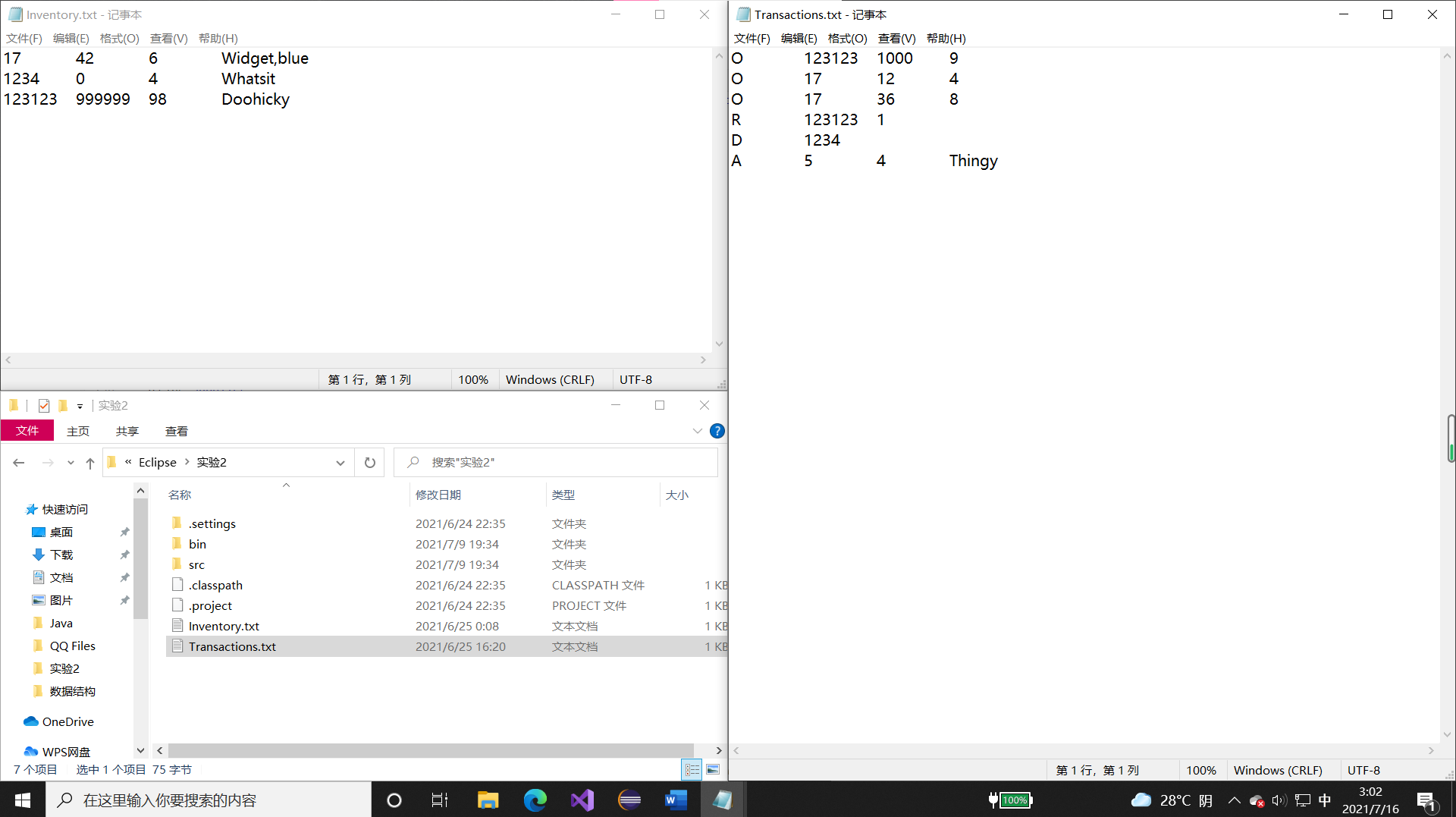
AROD





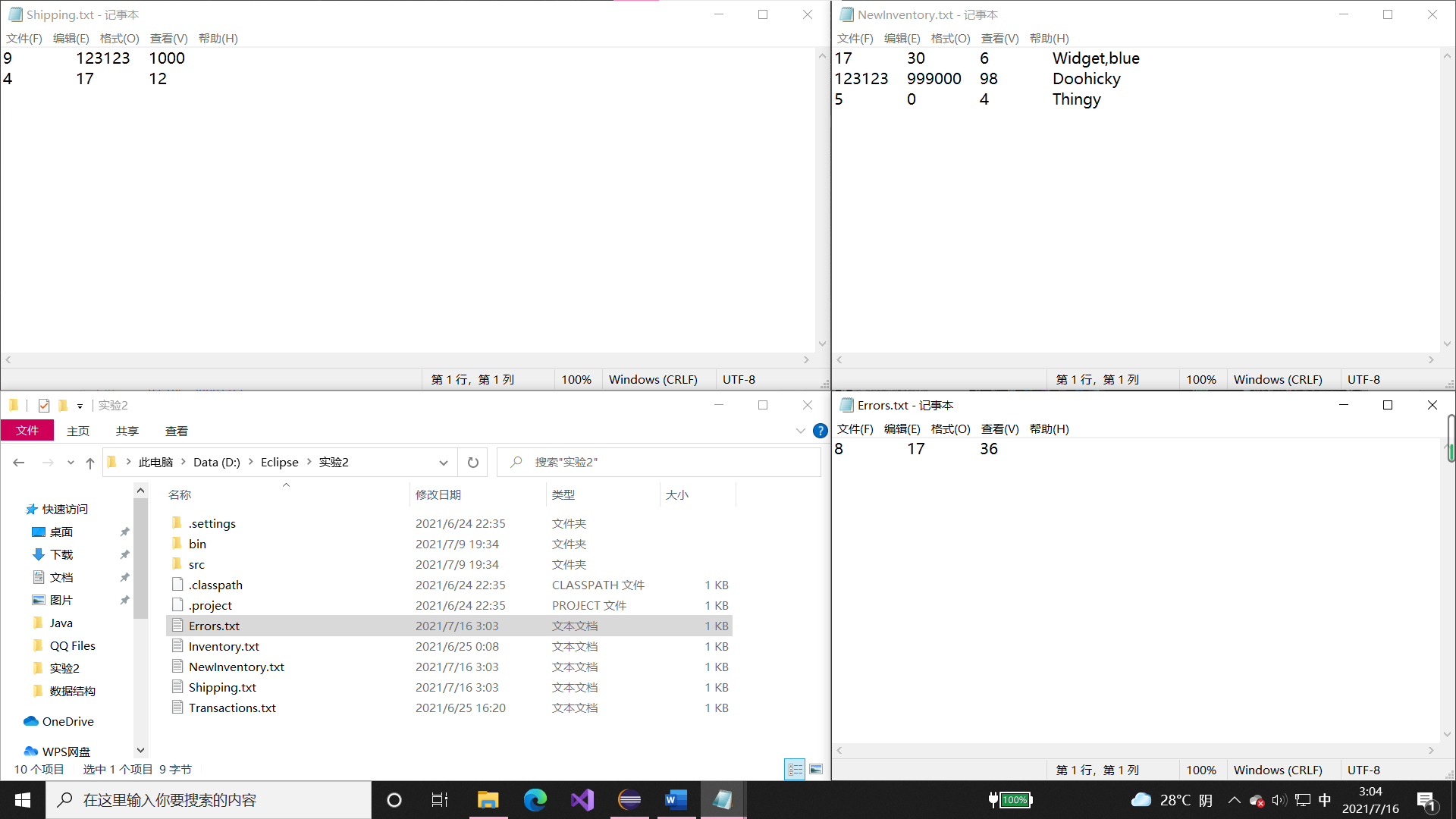
最后在把处理后的 数据存入其余三个数组，再把这三个数组里的数据写入文件，程序完成！

运行结果：



两个初始文件

运行完程序后出现了三个文件如图：



**五、实验结论、分析、思考题与心得体会**

本实验让我学会了如何使用文件流缓冲流读写文件，进一步加深了我对类的理解以及能够慢慢熟练运用类解决实际问题。

还学会了string类里的一些函数，比如切割函数