MPLab3

小组成员：

201830059 陈思睿 1310459235@qq.com

201220098 杨林 1479016371@qq.com

201830122 邹忠奇 617397223@qq.com

Task1: Nation

`KEYOUT=Text, VALUEOUT=DoubleWritable`

Map

使用 `value.toString().split("\\|")` 将.tbl文件中的每个元素分开，取出 `c_nationkey` 项和 `c_acctbal` 项（并转换为 `DoubleWritable` ），存入 `context` 中。

code:

```
public static class Map extends Mapper<Object, Text, Text, DoubleWritable> {
    @Override
    public void map(Object key, Text value, Context context) throws IOException, InterruptedException {
        String[] info=value.toString().split("\\|");
        Text nation = new Text(info[3]);
        DoubleWritable balance = new DoubleWritable(Double.parseDouble(info[5]));
        context.write(nation,balance);
    }
}
```

Reduce

`for` 循环求和 `values` ，写入 `context` 。

code:

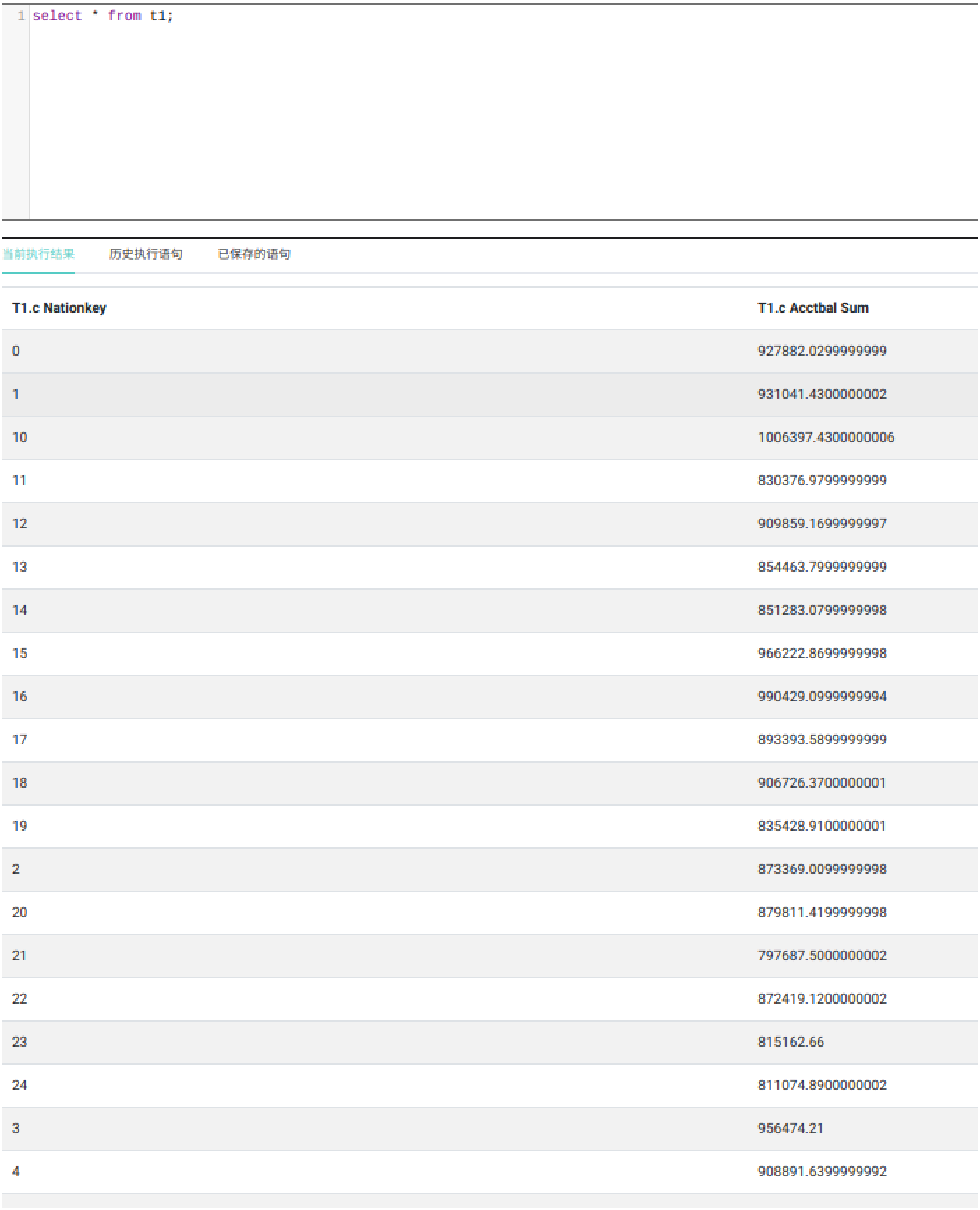
```
public static class Reduce extends Reducer<Text, DoubleWritable, Text, DoubleWritable>
{
    @Override
    public void reduce(Text key, Iterable<DoubleWritable> values, Context context)
throws IOException, InterruptedException {
        double ans=0.0;
        for(DoubleWritable dw:values){
            ans+=dw.get();
        }
        context.write(key,new DoubleWritable(ans));
    }
}
```

Hive create table

输出文件在 `/user/2023fg08/lab3/output1/` 中

```
create table t1 (c_nationkey string, c_acctbal_sum string) row format delimited fields
terminated by '\t' location '/user/2023fg08/lab3/output1/';
```

Result



```sql
select * from t1;
```

T1.c Nationkey	T1.c Acctbal Sum
0	927882.0299999999
1	931041.4300000002
10	1006397.4300000006
11	830376.9799999999
12	909859.1699999997
13	854463.7999999999
14	851283.0799999998
15	966222.8699999998
16	990429.0999999994
17	893393.5899999999
18	906726.3700000001
19	835428.9100000001
2	873369.0099999998
20	879811.4199999998
21	797687.5000000002
22	872419.1200000002
23	815162.66
24	811074.8900000002
3	956474.21
4	908891.6399999992

Task2

`KEYOUT=Text, VALUEOUT=Text`

Map

使用`value.toString().split("\\|")`将.tbl文件中的每个元素分开，取出`order_key,` `order_priority`和`ship_priority`项，以`order_priority`为`Key`， `ship_priority + "\t" + order_key`为`Value`写入`context`。

code:

```java
public static class Map extends Mapper<Object, Text, Text, Text> {

    @Override
    public void map(Object key, Text value, Context context) throws IOException, InterruptedException {
        String[] info=value.toString().split("\\|");
        String orderKey=info[0];
        Text orderPriority=new Text(info[5]);
        String shipPriority=info[7];
        Text msg= new Text(shipPriority+"\t"+orderKey);
        context.write(orderPriority,msg);
    }
}
```

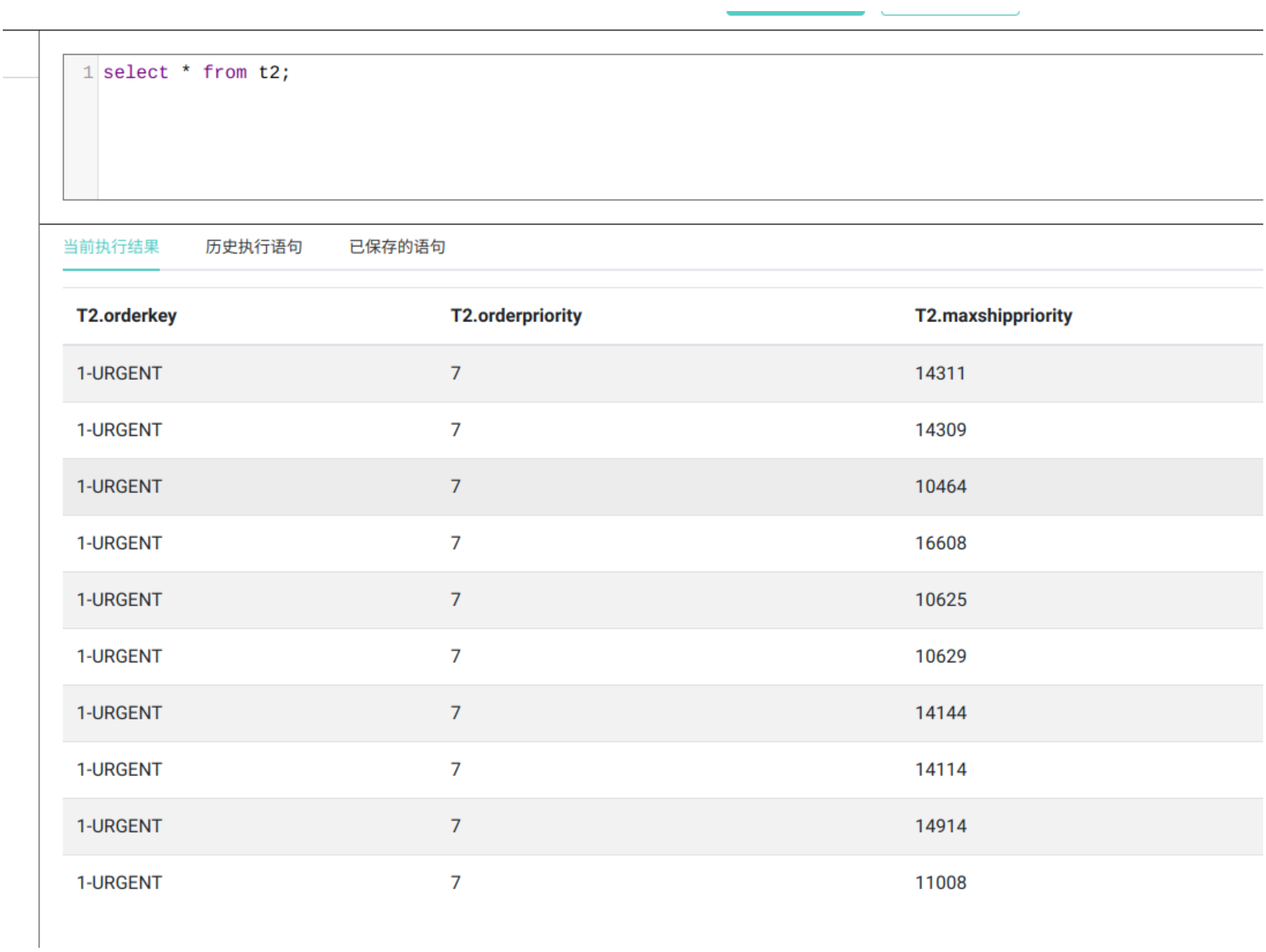
Reduce

for循环遍历values，找到`ship_priority`的最大值，将`ship_priority`最大的`Text`存入`ArrayList`，写入`context`。

code:

```java
public static class Reduce extends Reducer<Text, Text, Text, Text> {
    @Override
    public void reduce(Text key, Iterable<Text> values, Context context) throws
IOException, InterruptedException {
        int maxShip=-99999;
        ArrayList<Text> maxList=new ArrayList<>();
        for(Text v : values){
            String[] s=v.toString().split("\t");
            int shipPriority=Integer.parseInt(s[0]);
            if(shipPriority>maxShip){
                maxShip=shipPriority;
                maxList.clear();
                maxList.add(new Text(v));
            }
            else if(shipPriority==maxShip){
                maxList.add(new Text(v));
            }
        }
        for(Text o:maxList){
            context.write(key,o);
        }
    }
}
```

Hive Result:

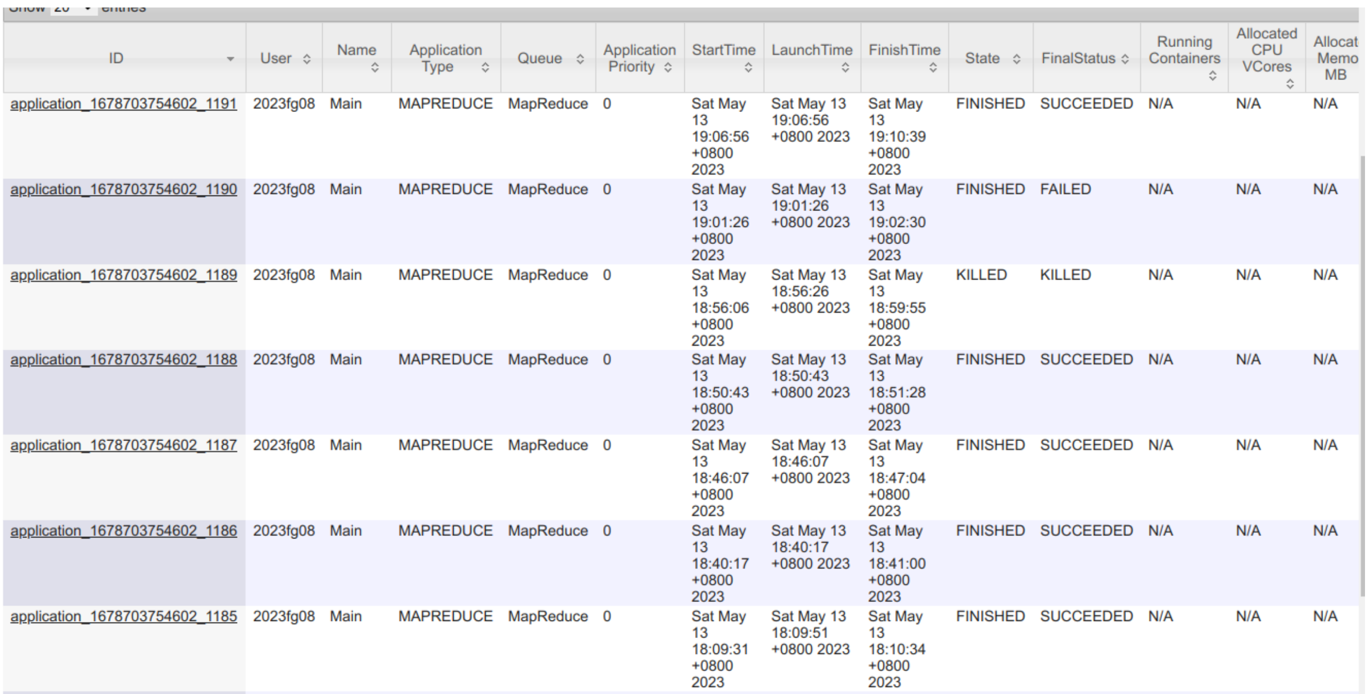


```
select * from t2;
```

当前执行结果　历史执行语句　已保存的语句

T2.orderkey	T2.orderpriority	T2.maxshippriority
1-URGENT	7	14311
1-URGENT	7	14309
1-URGENT	7	10464
1-URGENT	7	16608
1-URGENT	7	10625
1-URGENT	7	10629
1-URGENT	7	14144
1-URGENT	7	14114
1-URGENT	7	14914
1-URGENT	7	11008

WebUI



Show 20 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocat Memo MB
application_1678703754602_1191	2023fg08	Main	MAPREDUCE	MapReduce	0	Sat May 13 19:06:56 +0800 2023	Sat May 13 19:06:56 +0800 2023	Sat May 13 19:10:39 +0800 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A
application_1678703754602_1190	2023fg08	Main	MAPREDUCE	MapReduce	0	Sat May 13 19:01:26 +0800 2023	Sat May 13 19:01:26 +0800 2023	Sat May 13 19:02:30 +0800 2023	FINISHED	FAILED	N/A	N/A	N/A
application_1678703754602_1189	2023fg08	Main	MAPREDUCE	MapReduce	0	Sat May 13 18:56:06 +0800 2023	Sat May 13 18:56:26 +0800 2023	Sat May 13 18:59:55 +0800 2023	KILLED	KILLED	N/A	N/A	N/A
application_1678703754602_1188	2023fg08	Main	MAPREDUCE	MapReduce	0	Sat May 13 18:50:43 +0800 2023	Sat May 13 18:50:43 +0800 2023	Sat May 13 18:51:28 +0800 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A
application_1678703754602_1187	2023fg08	Main	MAPREDUCE	MapReduce	0	Sat May 13 18:46:07 +0800 2023	Sat May 13 18:46:07 +0800 2023	Sat May 13 18:47:04 +0800 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A
application_1678703754602_1186	2023fg08	Main	MAPREDUCE	MapReduce	0	Sat May 13 18:40:17 +0800 2023	Sat May 13 18:40:17 +0800 2023	Sat May 13 18:41:00 +0800 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A
application_1678703754602_1185	2023fg08	Main	MAPREDUCE	MapReduce	0	Sat May 13 18:09:31 +0800 2023	Sat May 13 18:09:51 +0800 2023	Sat May 13 18:10:34 +0800 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A

Task1:1185

Task2:1191