**作业6**

**RDD编程初级实践**

1. **编程题目**

使用Scala语言编写Spark应用程序实现数据去重（需要使用编译打包工具Maven或sbt进行编译打包）。

对于两个输入文件A和B，编写Spark独立应用程序，对两个文件进行合并，并剔除其中重复的内容，得到一个新文件C。下面是输入文件和输出文件的一个样例，供参考。

输入文件A的样例如下：

20180901 x

20180902 y

20180903 x

20180904 y

20180905 z

20180906 z

输入文件B的样例如下：

20180901 y

20180902 y

20180903 x

20180904 z

20180905 y

根据输入的文件A和B合并得到的输出文件C的样例如下：

20180901 x

20180901 y

20180902 y

20180903 x

20180904 y

20180904 z

20180905 y

20180905 z

20180906 z

**二、实验报告**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验报告 | | | | |
| 题目 | RDD编程初级实践 | 姓名 | 邓语苏 | 日期 |
| 实验环境：Hadoop 3.3.6、Java jdk1.8.0\_381、Spark 3.5.0 | | | | |
| 实验内容与完成情况：   1. 在本地新建a.txt、b.txt   运行vim a.txt、b.txt  输入文件a.txt  20180901 x  20180902 y  20180903 x  20180904 y  20180905 z  20180906 z  输入文件b.txt  20180901 y  20180902 y  20180903 x  20180904 z  20180905 y   1. 编写merge.py文件 2. from pyspark.sql import SparkSession 3. from pyspark.sql.functions import concat\_ws 4. def merge\_and\_deduplicate(file\_a\_path, file\_b\_path, output\_path): 5. *# 创建Spark会话，并指定Hadoop配置* 6. spark = SparkSession.builder \ 7. .appName("MergeAndDeduplicate") \ 8. .config("spark.hadoop.fs.defaultFS", "hdfs://localhost:9000") \ 9. .getOrCreate() 10. *# 读取输入文件A和B* 11. df\_a = spark.read.text(file\_a\_path) 12. df\_b = spark.read.text(file\_b\_path) 13. *# 将数据按行拆分成键值对 (key, value)* 14. pairs\_a = df\_a.rdd.map(lambda row: tuple(row.value.split())) 15. pairs\_b = df\_b.rdd.map(lambda row: tuple(row.value.split())) 16. *# 合并文件A和文件B的内容* 17. merged\_rdd = pairs\_a.union(pairs\_b) 18. *# 去除重复项* 19. deduplicated\_rdd = merged\_rdd.distinct() 20. print("Merged RDD:") 21. for item in deduplicated\_rdd.collect(): 22. print(item) 23. *# 将结果保存到输出文件C* 24. deduplicated\_rdd.map(lambda x: "{}\t{}".format(x[0], x[1])).coalesce(1).saveAsTextFile(output\_path) 25. *# 停止Spark会话* 26. spark.stop() 27. if \_\_name\_\_ == "\_\_main\_\_": 28. file\_a\_path = "file:///home/hadoop/Downloads/a.txt" 29. file\_b\_path = "file:///home/hadoop/Downloads/b.txt" 30. output\_path = "file:///home/hadoop/Downloads/output" 31. merge\_and\_deduplicate(file\_a\_path, file\_b\_path, output\_path) 32. 使用spark-submit运行   输入/usr/local/spark/bin/spark-submit ~/Downloads/merge.py运行merge.py文件   1. 运行结果     在~/Downloads/output文件夹内生成了txt文件 | | | | |
| 出现的问题：  一开始无法正确生成文本文件。 | | | | |
| 解决方案（列出遇到的问题和解决办法，列出没有解决的问题）：  发现是a.txt和b.txt都有空行。删除空行后即可正常运行 | | | | |

（备注：相关实验过程可以截图放入实验报告）