|  |
| --- |
|  |
| |  | | --- | |  | | |  | | --- | | import java.io.IOException; | |  | import java.util.TreeMap; | |  |  | |  | import org.apache.hadoop.conf.Configuration; | |  | import org.apache.hadoop.fs.Path; | |  | import org.apache.hadoop.io.LongWritable; | |  | import org.apache.hadoop.io.NullWritable; | |  | import org.apache.hadoop.io.Text; | |  | import org.apache.hadoop.mapreduce.Job; | |  | import org.apache.hadoop.mapreduce.Mapper; | |  | import org.apache.hadoop.mapreduce.Partitioner; | |  | import org.apache.hadoop.mapreduce.Reducer; | |  | import org.apache.hadoop.mapreduce.lib.input.FileInputFormat; | |  | import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat; | |  |  | |  |  | |  | //use retail data D11,D12,D01 and D02 | |  |  | |  | public class AgeWiseTop5Products { | |  |  | |  | // Mapper Class | |  |  | |  | public static class AgeWiseTop5ProductMapperClass extends Mapper<LongWritable,Text,Text,Text> | |  | { | |  | public void map(LongWritable key, Text value, Context context) | |  | { | |  | try{ | |  | String[] str = value.toString().split(";"); | |  | String prodid = str[5]; | |  | String sales = str[8]; | |  | String age = str[2]; | |  | String myValue = age + ',' + sales; | |  | context.write(new Text(prodid), new Text(myValue)); | |  | } | |  | catch(Exception e) | |  | { | |  | System.out.println(e.getMessage()); | |  | } | |  | } | |  | } | |  |  | |  | //Partitioner class | |  |  | |  | public static class AgeWiseTop5ProductPartitioner extends | |  | Partitioner < Text, Text > | |  | { | |  | @Override | |  | public int getPartition(Text key, Text value, int numReduceTasks) | |  | { | |  | String[] str = value.toString().split(","); | |  | String age = str[0].trim(); | |  |  | |  | if (age.equals("A")) | |  | { | |  | return 0; | |  | } | |  | if(age.equals("B")) | |  | { | |  | return 1 ; | |  | } | |  | if(age.equals("C")) | |  | { | |  | return 2 ; | |  | } | |  |  | |  | if(age.equals("D")) | |  | { | |  | return 3 ; | |  | } | |  |  | |  | if(age.equals("E")) | |  | { | |  | return 4 ; | |  | } | |  |  | |  | if(age.equals("F")) | |  | { | |  | return 5 ; | |  | } | |  |  | |  | if(age.equals("G")) | |  | { | |  | return 6 ; | |  | } | |  |  | |  | if(age.equals("H")) | |  | { | |  | return 7 ; | |  | } | |  |  | |  | if(age.equals("I")) | |  | { | |  | return 8 ; | |  | } | |  | else | |  | { | |  | return 9; | |  | } | |  | } | |  | } | |  |  | |  |  | |  | //Reducer class | |  |  | |  | public static class AgeWiseTop5ProductReducerClass extends Reducer<Text,Text,NullWritable,Text> | |  | { | |  | private TreeMap<Long, Text> repToRecordMap = new TreeMap<Long, Text>(); | |  |  | |  | public void reduce(Text key, Iterable <Text> values, Context context) throws IOException, InterruptedException | |  | { | |  | long totalsales = 0; | |  | String age = ""; | |  |  | |  | for (Text val : values) | |  | { | |  | String[] token = val.toString().split(","); | |  | totalsales = totalsales + Long.parseLong(token[1]); | |  | age = token[0]; | |  | } | |  |  | |  | String myValue = key.toString(); | |  | String mytotal = String.format("%d", totalsales); | |  | myValue = myValue + ',' + age + ',' + mytotal ; | |  |  | |  | repToRecordMap.put(new Long(totalsales), new Text(myValue)); | |  | if (repToRecordMap.size() > 5) | |  | { | |  | repToRecordMap.remove(repToRecordMap.firstKey()); | |  | } | |  | } | |  |  | |  | protected void cleanup(Context context) throws IOException, | |  | InterruptedException | |  | { | |  |  | |  | for (Text t : repToRecordMap.descendingMap().values()) | |  | { | |  | context.write(NullWritable.get(), t); | |  | } | |  | } | |  |  | |  | } | |  |  | |  | //Main class | |  |  | |  | public static void main(String[] args) throws Exception { | |  |  | |  | Configuration conf = new Configuration(); | |  | Job job = Job.getInstance(conf, "Top 5 products Sold Age wise"); | |  | job.setJarByClass(AgeWiseTop5Products.class); | |  | job.setMapperClass(AgeWiseTop5ProductMapperClass.class); | |  | job.setPartitionerClass(AgeWiseTop5ProductPartitioner.class); | |  | job.setReducerClass(AgeWiseTop5ProductReducerClass.class); | |  | job.setNumReduceTasks(10); | |  | job.setMapOutputKeyClass(Text.class); | |  | job.setMapOutputValueClass(Text.class); | |  | job.setOutputKeyClass(NullWritable.class); | |  | job.setOutputValueClass(Text.class); | |  | FileInputFormat.addInputPath(job, new Path(args[0])); | |  | FileOutputFormat.setOutputPath(job, new Path(args[1])); | |  | System.exit(job.waitForCompletion(true) ? 0 : 1); | |  | } | |  | } | | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |
|  | |  |