|  |  |
| --- | --- |
|  | |
| import java.io.\*; | |
|  | | import org.apache.hadoop.io.Text; | |
|  | | import org.apache.hadoop.io.LongWritable; | |
|  | | import org.apache.hadoop.mapreduce.Job; | |
|  | | import org.apache.hadoop.mapreduce.Mapper; | |
|  | | import org.apache.hadoop.mapreduce.Reducer; | |
|  | | import org.apache.hadoop.conf.\*; | |
|  | | import org.apache.hadoop.fs.\*; | |
|  | | import org.apache.hadoop.mapreduce.lib.input.\*; | |
|  | | import org.apache.hadoop.mapreduce.lib.output.\*; | |
|  | |  | |
|  | |  | |
|  | | public class StockVolume { | |
|  | |  | |
|  | | public static class MapClass extends Mapper<LongWritable,Text,Text,LongWritable> | |
|  | | { | |
|  | | public void map(LongWritable key, Text value, Context context) | |
|  | | { | |
|  | | try{ | |
|  | | String[] str = value.toString().split(","); | |
|  | | long vol = Long.parseLong(str[7]); | |
|  | | context.write(new Text(str[1]),new LongWritable(vol)); | |
|  | |  | |
|  | | } | |
|  | | catch(Exception e) | |
|  | | { | |
|  | | System.out.println(e.getMessage()); | |
|  | | } | |
|  | | } | |
|  | | } | |
|  | |  | |
|  | | public static class ReduceClass extends Reducer<Text,LongWritable,Text,LongWritable> | |
|  | | { | |
|  | | private LongWritable result = new LongWritable(); | |
|  | |  | |
|  | | public void reduce(Text key, Iterable<LongWritable> values,Context context) throws IOException, InterruptedException { | |
|  | | long sum = 0; | |
|  | |  | |
|  | | for (LongWritable val : values) | |
|  | | { | |
|  | | sum += val.get(); | |
|  | | } | |
|  | |  | |
|  | | result.set(sum); | |
|  | | context.write(key, result); | |
|  | |  | |
|  | | } | |
|  | | } | |
|  | | public static void main(String[] args) throws Exception { | |
|  | | Configuration conf = new Configuration(); | |
|  | | //conf.set("name", "value") | |
|  | | Job job = Job.getInstance(conf, "Volume Count"); | |
|  | | job.setJarByClass(StockVolume.class); | |
|  | | job.setMapperClass(MapClass.class); | |
|  | | //job.setCombinerClass(ReduceClass.class); | |
|  | | job.setReducerClass(ReduceClass.class); | |
|  | | //job.setNumReduceTasks(2); | |
|  | | job.setOutputKeyClass(Text.class); | |
|  | | job.setOutputValueClass(LongWritable.class); | |
|  | | FileInputFormat.addInputPath(job, new Path(args[0])); | |
|  | | FileOutputFormat.setOutputPath(job, new Path(args[1])); | |
|  | | System.exit(job.waitForCompletion(true) ? 0 : 1); | |
|  | | } | |
|  | | } | |