|  |
| --- |
| import java.io.IOException; |
|  |  |
|  | import org.apache.hadoop.conf.Configuration; |
|  | import org.apache.hadoop.fs.Path; |
|  | import org.apache.hadoop.io.Text; |
|  | import org.apache.hadoop.mapreduce.Job; |
|  | import org.apache.hadoop.mapreduce.Mapper; |
|  | import org.apache.hadoop.mapreduce.lib.input.FileInputFormat; |
|  | import org.apache.hadoop.mapreduce.lib.input.KeyValueTextInputFormat; |
|  | import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat; |
|  | import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat; |
|  |  |
|  |  |
|  | public class KeyValueToText { |
|  | public static class KeyValueMapper extends |
|  | Mapper<Text, Text, Text, Text> { |
|  | public void map(Text key, Text value, Context context) |
|  | throws IOException, InterruptedException { |
|  | context.write(key, value); |
|  | } |
|  | } |
|  | public static void main(String[] args) throws Exception { |
|  |  |
|  | Configuration conf = new Configuration(); |
|  | conf.set("mapreduce.input.keyvaluelinerecordreader.key.value.separator", ","); |
|  | //conf.set("mapreduce.output.textoutputformat.separator",","); |
|  | Job job = Job.getInstance(conf, "Break String into 2 parts i.e. Key and Value"); |
|  | job.setJarByClass(KeyValueToText.class); |
|  | job.setMapperClass(KeyValueMapper.class); |
|  | job.setNumReduceTasks(0); |
|  | job.setOutputKeyClass(Text.class); |
|  | job.setOutputValueClass(Text.class); |
|  | job.setInputFormatClass(KeyValueTextInputFormat.class); |
|  | job.setOutputFormatClass(TextOutputFormat.class); |
|  | FileInputFormat.addInputPath(job, new Path(args[0])); |
|  | FileOutputFormat.setOutputPath(job, new Path(args[1])); |
|  | System.exit(job.waitForCompletion(true) ? 0 : 1); |
|  | } |
|  |  |
|  |  |
|  |  |
|  |  |
|  | } |