

**MAP REDUCE JAVA**

cd

cat >WordCount.java

**//package org.myorg;**

**import java.io.IOException;**

**import java.util.\*;**

**import org.apache.hadoop.fs.Path;**

**import org.apache.hadoop.conf.\*;**

**import org.apache.hadoop.io.\*;**

**import org.apache.hadoop.mapred.\*;**

**import org.apache.hadoop.util.\*;**

**public class WordCount {**

**public static class Map extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable> {**

**private final static IntWritable one = new IntWritable(1);**

**private Text word = new Text();**

**public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter reporter) throws IOException {**

**String line = value.toString();**

**StringTokenizer tokenizer = new StringTokenizer(line);**

**while (tokenizer.hasMoreTokens()) {**

**word.set(tokenizer.nextToken());**

**output.collect(word, one);**

**}**

**}**

**}**

**public static class Reduce extends MapReduceBase implements Reducer<Text, IntWritable, Text, IntWritable> {**

**public void reduce(Text key, Iterator<IntWritable> values, OutputCollector<Text, IntWritable> output, Reporter reporter) throws IOException {**

**int sum = 0;**

**while (values.hasNext()) {**

**sum += values.next().get();**

**}**

**output.collect(key, new IntWritable(sum));**

**}**

**}**

**public static void main(String[] args) throws Exception {**

**JobConf conf = new JobConf(WordCount.class);**

**conf.setJobName("wordcount");**

**conf.setOutputKeyClass(Text.class);**

**conf.setOutputValueClass(IntWritable.class);**

**conf.setMapperClass(Map.class);**

**//conf.setCombinerClass(Reduce.class);**

**conf.setReducerClass(Reduce.class);**

**conf.setInputFormat(TextInputFormat.class);**

**conf.setOutputFormat(TextOutputFormat.class);**

**FileInputFormat.setInputPaths(conf, new Path(args[0]));**

**FileOutputFormat.setOutputPath(conf, new Path(args[1]));**

**JobClient.runJob(conf);**

**}**

**}**

**export CLASSPATH=/usr/local/hadoop/hadoop-core-1.2.1.jar**

**mkdir wordcount\_classes**

**javac -d wordcount\_classes/ WordCount.java**

**jar -cvf wordcount.jar -C wordcount\_classes/ .**

**hadoop jar wordcount.jar WordCount /user/ubuntu/wordcount/input /user/ubuntu/wordcount/output**