```java
// WC_Runner.java

package com.wc;

import java.io.IOException;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.FileInputFormat;
import org.apache.hadoop.mapred.FileOutputFormat;
import org.apache.hadoop.mapred.JobClient;
import org.apache.hadoop.mapred.JobConf;
import org.apache.hadoop.mapred.TextInputFormat;
import org.apache.hadoop.mapred.TextOutputFormat;

public class WC_Runner {
   public static void main(String[] args) throws IOException {
        JobConf conf = new JobConf(WC_Runner.class);
        conf.setJobName("WordCount");
        conf.setOutputKeyClass(Text.class);
        conf.setOutputValueClass(IntWritable.class);
        conf.setMapperClass(WC_Mapper.class);
        conf.setCombinerClass(WC_Reducer.class);
        conf.setReducerClass(WC_Reducer.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);
   }
}
```

```java
// WC_Mapper.java

package com.wc;

import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;

public class WC_Mapper 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);
       }
   }
}
```

```java
// WC_Reducer.java

package com.wc;

import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;

public class WC_Reducer  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));
   }
}
```

Input:

```
HDFS is a storage unit of Hadoop
MapReduce is a processing tool for Hadoop
```

Output:

```
HDFS            1
Hadoop          2
MapReduce       1
a               2
for             1
is              2
of              1
processing      1
storage         1
tool            1
unit            1
```