Row Count Program

```
import java.io.IOException;

import java.util.*;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.*;

import org.apache.hadoop.mapred.*;

public class LineCount
{
  public static class Map extends MapReduceBase implements
    Mapper < LongWritable, Text, Text, IntWritable > //MAPPER CLASS
  {
    private final static IntWritable one = new IntWritable (1);
    private Text word = new Text ("Total Number of Rows : “); //word is a variable which will be
used for printing the output

    public void map (LongWritable key, Text value,
                     OutputCollector < Text, IntWritable > output,
                     Reporter reporter) throws IOException
    {
      output.collect (word, one); //one is the variable which consists ROWCOUNT
    }
  }

  public static class Reduce extends MapReduceBase implements
    Reducer < Text, IntWritable, Text, IntWritable > //REDUCER CLASS
  {
    public void reduce (Text key, Iterator < IntWritable > values,
                        OutputCollector < Text, IntWritable > output,
                        Reporter reporter) throws IOException
    {
      int sum = 0;
      while (values.hasNext ()) //Parsing file
          {
            sum += values.next ().get (); //Counting Number of Rows
          }
      output.collect (key, new IntWritable (sum));
    }
  }

  public static void main (String[]args) throws Exception
  {
      JobConf conf = new JobConf (LineCount.class);
      conf.setJobName (“LineCount"); //LineCount is the CLASS name
      conf.setNumReduceTasks (5);
      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])); //Takes argument[0] as input file in
command prompt
```

```
    FileOutputFormat.setOutputPath (conf, new Path (args[1])); //Takes argument[1] as output file
in command prompt


    JobClient.runJob (conf);
  }
}
```