Suryanarayan.B

CB.EN.U4CSE19056

Big Data Lab Evaluation

Code:

**import** java.io.IOException;

**import** java.util.StringTokenizer;

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

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

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

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

**import** org.apache.hadoop.mapreduce.Job;

**import** org.apache.hadoop.mapreduce.Mapper;

**import** org.apache.hadoop.mapreduce.Reducer;

**import** org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

**import** org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

**public** **class** StartsWith{

**public** **static** **class** MapperClass

**extends** Mapper<Object, Text, Text, IntWritable>{

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

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

**public** **void** map(Object key, Text value, Context context

) **throws** IOException, InterruptedException {

StringTokenizer itr = **new** StringTokenizer(value.toString());

**while** (itr.hasMoreTokens()) {

word.set(itr.nextToken());

**if**(word.toString().startsWith("b") && word.toString().endsWith("ing")){

context.write(word, ***one***);

}

}

}

}

**public** **static** **class** ReducerClass

**extends** Reducer<Text,IntWritable,Text,IntWritable> {

**private** IntWritable result = **new** IntWritable();

**public** **void** reduce(Text key, Iterable<IntWritable> values,

Context context

) **throws** IOException, InterruptedException {

**int** sum = 0;

**for** (IntWritable val : values) {

sum += val.get();

}

result.set(sum);

context.write(key, result);

}

}

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

Configuration conf = **new** Configuration();

Job job = Job.*getInstance*(conf, "word count bing");

job.setJarByClass(StartsWith.**class**);

job.setMapperClass(MapperClass.**class**);

job.setCombinerClass(ReducerClass.**class**);

job.setReducerClass(ReducerClass.**class**);

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

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

FileInputFormat.*addInputPath*(job, **new** Path(args[0]));

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

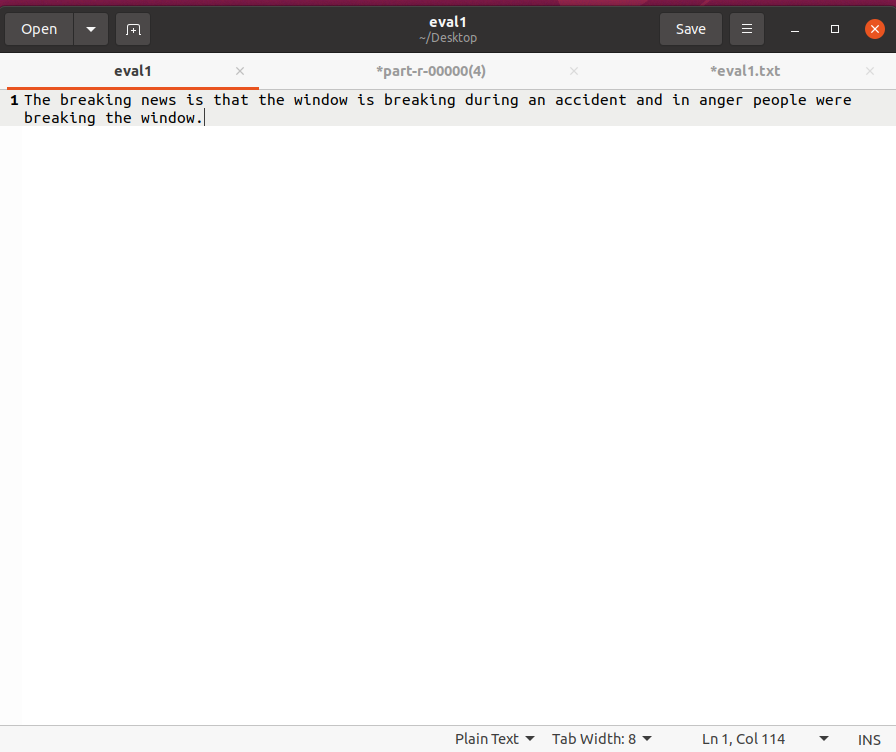
System.*exit*(job.waitForCompletion(**true**) ? 0 : 1);

}

}

Input File:

The breaking news is that the window is breaking during an accident and in anger people were breaking the window.



Output File:

