LARGE SCALE DATA PROCESSING (CSE 3025)
WIN SEMESTER 2017-18

NAME: B.SHUBANKAR

REGNO: 15BCE1123

PROFESSOR: DR. Maheswari N

LAB-4

AIM: To use combiner functionality.

PROGRAM:

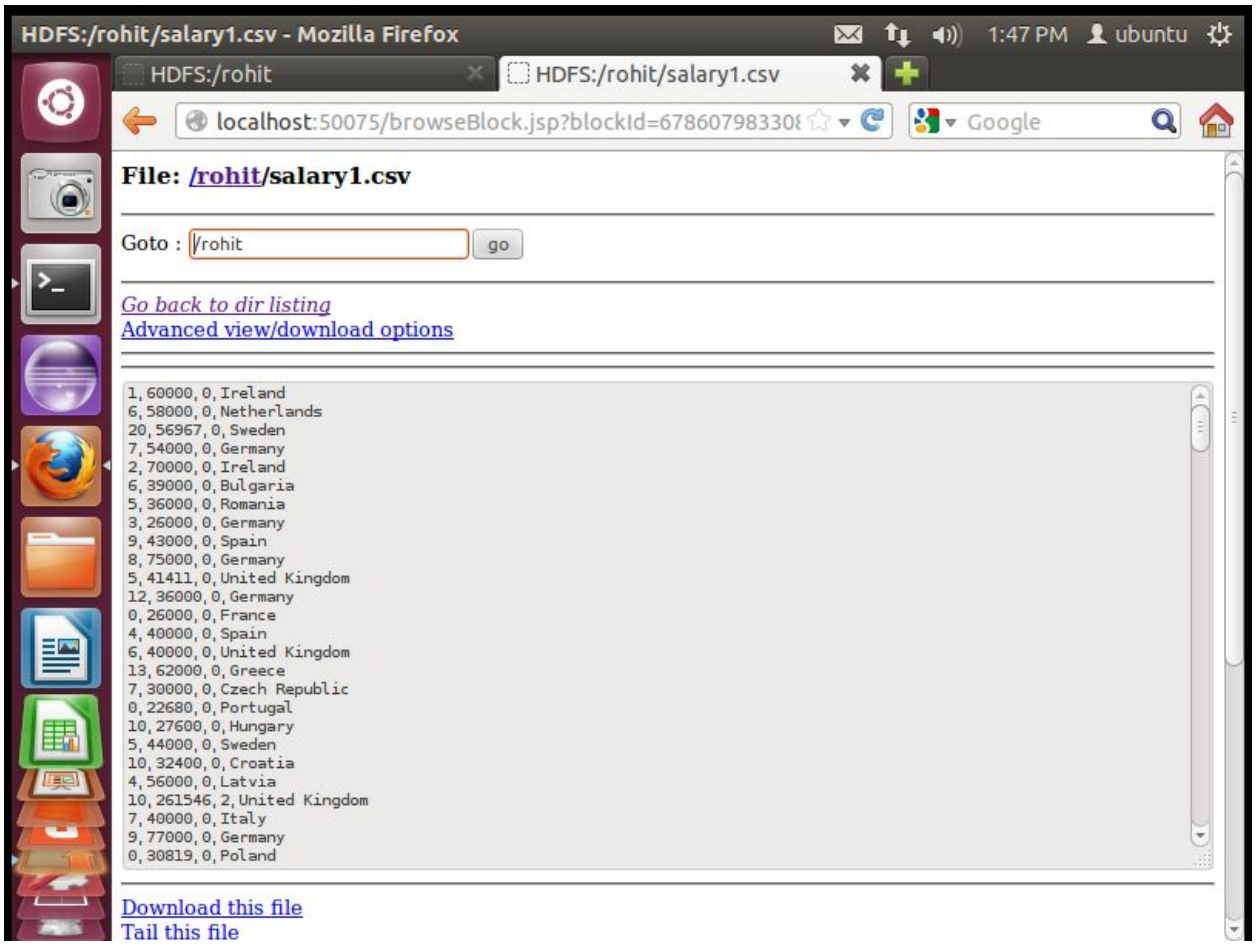
1. To add combiner to word count program.

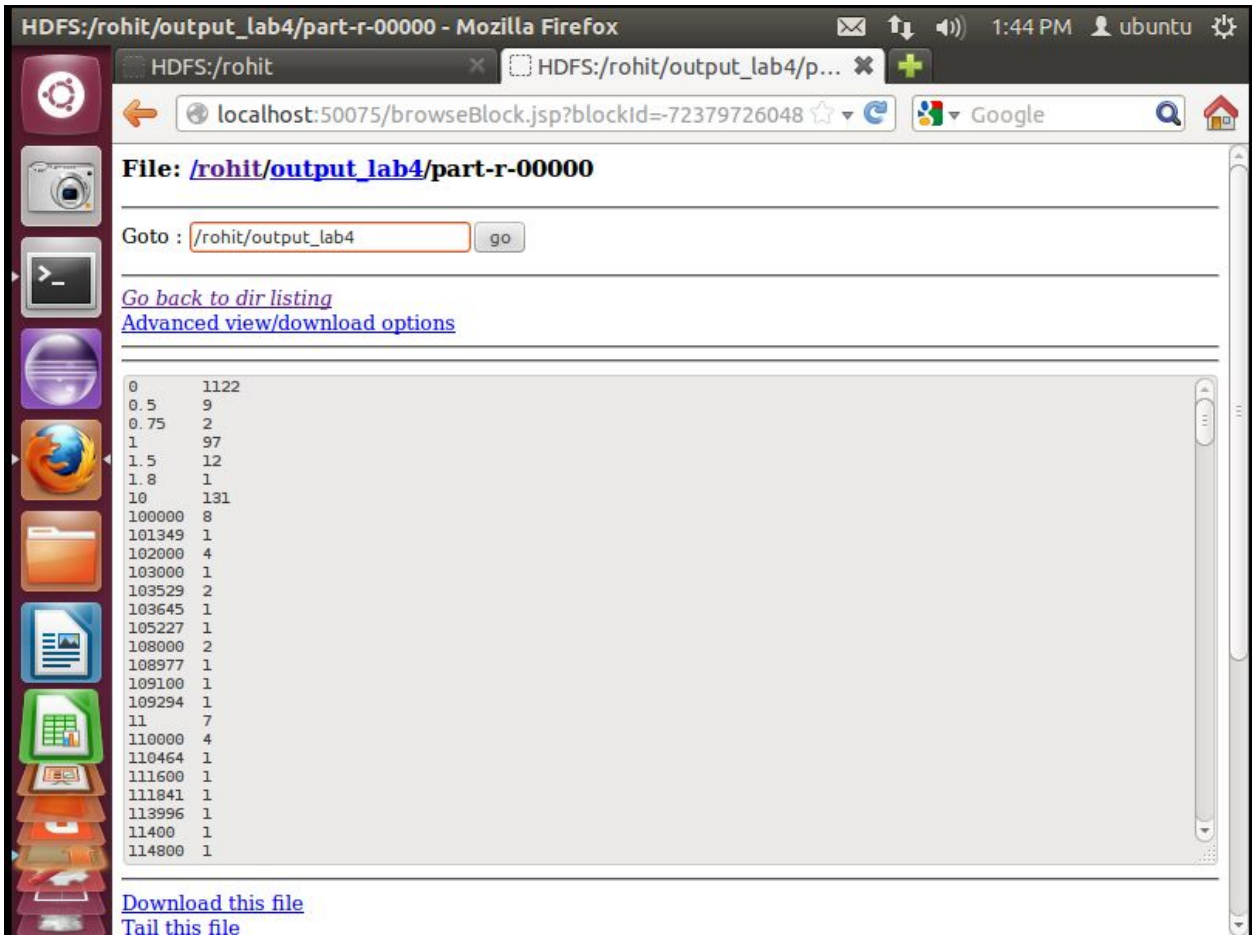
```
import java.io.IOException;
import java.util.*;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
public class WordCountmp1 {
```

```
public static class Map extends Mapper<LongWritable, Text, Text,
IntWritable> {
private final static IntWritable one = new IntWritable(1);
public void map(LongWritable key, Text value, Context context) throws
IOException, InterruptedException {
String line = value.toString();
StringTokenizer tokenizer = new StringTokenizer(line);
while (tokenizer.hasMoreTokens()) {
word.set(tokenizer.nextToken());
context.write(word, one);
}
}
}
public static class Reduce extends Reducer<Text, IntWritable, Text,
IntWritable> {
public void reduce(Text key, Iterable<IntWritable> values, Context
context)throws IOException, InterruptedException {
int sum = 0;
for (IntWritable val : values) {
sum += val.get();
}
context.write(key, new IntWritable(sum));
}
}
```

```
public static void main(String[] args) throws Exception {
Configuration conf = new Configuration();
Job job = new Job(conf, "wordcountmp1");
job.setJarByClass(WordCountmp1.class);
job.setOutputKeyClass(Text.class);
job.setOutputValueClass(IntWritable.class);
job.setMapperClass(Map.class);
job.setReducerClass(Reduce.class);
job.setCombinerClass(Reduce.class);
job.setInputFormatClass(TextInputFormat.class);
job.setOutputFormatClass(TextOutputFormat.class);
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
job.waitForCompletion(true);
}
}
```

OUTPUT :


HDFS:/rohit/salary1.csv - Mozilla Firefox
1:47 PM
ubuntu
HDFS:/rohit
HDFS:/rohit/salary1.csv
localhost:50075/browseBlock.jsp?blockId=67860798330
Google
File: /rohit/salary1.csv
Goto : /rohit
go
Go back to dir listing
Advanced view/download options
1,60000,0,Ireland
6,58000,0,Netherlands
20,56967,0,Sweden
7,54000,0,Germany
2,70000,0,Ireland
6,39000,0,Bulgaria
5,36000,0,Romania
3,26000,0,Germany
9,43000,0,Spain
8,75000,0,Germany
5,41411,0,United Kingdom
12,36000,0,Germany
0,26000,0,France
4,40000,0,Spain
6,40000,0,United Kingdom
13,62000,0,Greece
7,30000,0,Czech Republic
0,22680,0,Portugal
10,27600,0,Hungary
5,44000,0,Sweden
10,32400,0,Croatia
4,56000,0,Latvia
10,261546,2,United Kingdom
7,40000,0,Italy
9,77000,0,Germany
0,30819,0,Poland
Download this file
Tail this file



2) To display records with salary greater than 1,00,000.

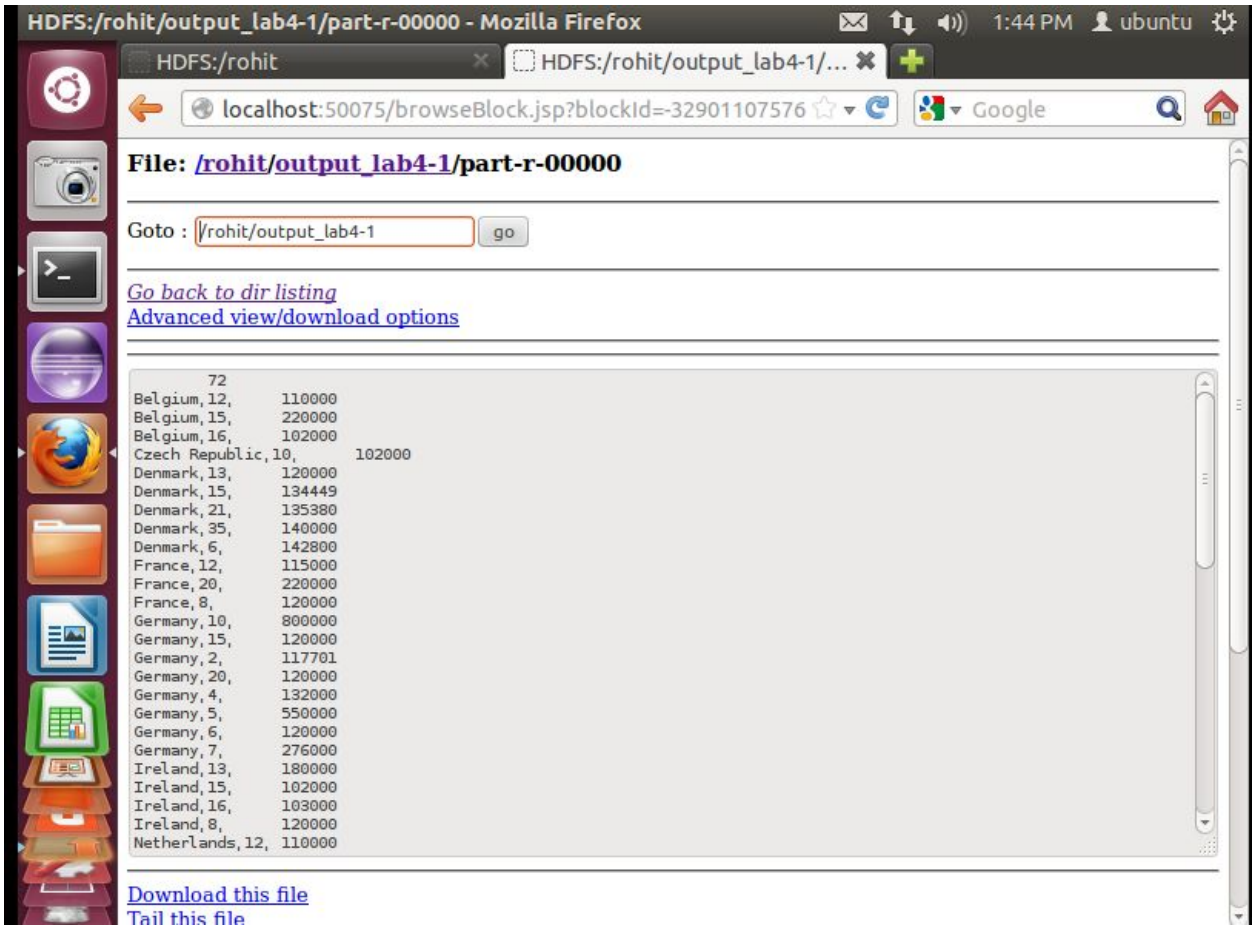
PROGRAM:

```
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.mapreduce.*;
```

```
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
public class Salary {
public static class Map extends Mapper<LongWritable, Text, Text,
IntWritable> {
private Text w= new Text("Total no of words");
int count=0;
public void map(LongWritable key, Text value, Context context) throws
IOException, InterruptedException {
 String[] line =value.toString().split(",");
 int i= Integer.parseInt(line[1]);
 if(i>100000)
{
 context.write(new Text(line[3]+","+line[0]),new IntWritable(i));
 count=count+1;
}
}
```

```
 public void cleanup(Context context)throws IOException, InterruptedException {
 context.write(w, new IntWritable(count));
}
}
public static void main(String[] args) throws Exception {
Configuration conf = new Configuration();
Job job = new Job(conf, "Salary");
job.setJarByClass(Salary.class);
job.setOutputKeyClass(Text.class);
job.setOutputValueClass(IntWritable.class);
job.setMapperClass(Map.class);
job.setInputFormatClass(TextInputFormat.class);
job.setOutputFormatClass(TextOutputFormat.class);
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
job.waitForCompletion(true);
}
}
```

OUTPUT:


HDFS:/rohit/output_lab4-1/part-r-00000 - Mozilla Firefox
1:44 PM
ubuntu
HDFS:/rohit
HDFS:/rohit/output_lab4-1/...
localhost:50075/browseBlock.jsp?blockId=-32901107576
Google
File: /rohit/output_lab4-1/part-r-00000
Goto : /rohit/output_lab4-1
go
Go back to dir listing
Advanced view/download options
72
Belgium,12, 110000
Belgium,15, 220000
Belgium,16, 102000
Czech Republic,10, 102000
Denmark,13, 120000
Denmark,15, 134449
Denmark,21, 135380
Denmark,35, 140000
Denmark,6, 142800
France,12, 115000
France,20, 220000
France,8, 120000
Germany,10, 800000
Germany,15, 120000
Germany,2, 117701
Germany,20, 120000
Germany,4, 132000
Germany,5, 550000
Germany,6, 120000
Germany,7, 276000
Ireland,13, 180000
Ireland,15, 102000
Ireland,16, 103000
Ireland,8, 120000
Netherlands,12, 110000
Download this file
Tail this file