SalesCountryDriver.java

```
package SalesCountry;

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

public class SalesCountryDriver {
        public static void main(String[] args) {
                JobClient my_client = new JobClient();
                // Create a configuration object for the job
                JobConf job_conf = new JobConf(SalesCountryDriver.class);

                // Set a name of the Job
                job_conf.setJobName("SalePerCountry");

                // Specify data type of output key and value
                job_conf.setOutputKeyClass(Text.class);
                job_conf.setOutputValueClass(IntWritable.class);

                // Specify names of Mapper and Reducer Class
                job_conf.setMapperClass(SalesCountry.SalesMapper.class);
                job_conf.setReducerClass(SalesCountry.SalesCountryReducer.class);

                // Specify formats of the data type of Input and output
                job_conf.setInputFormat(TextInputFormat.class);
                job_conf.setOutputFormat(TextOutputFormat.class);

                // Set input and output directories using command line arguments,
                //arg[0] = name of input directory on HDFS, and arg[1] =  name of output directory to be created to store the
output file.

                FileInputFormat.setInputPaths(job_conf, new Path(args[0]));
                FileOutputFormat.setOutputPath(job_conf, new Path(args[1]));

                my_client.setConf(job_conf);
                try {
                        // Run the job
                        JobClient.runJob(job_conf);
                } catch (Exception e) {
                        e.printStackTrace();
                }
        }
}
```

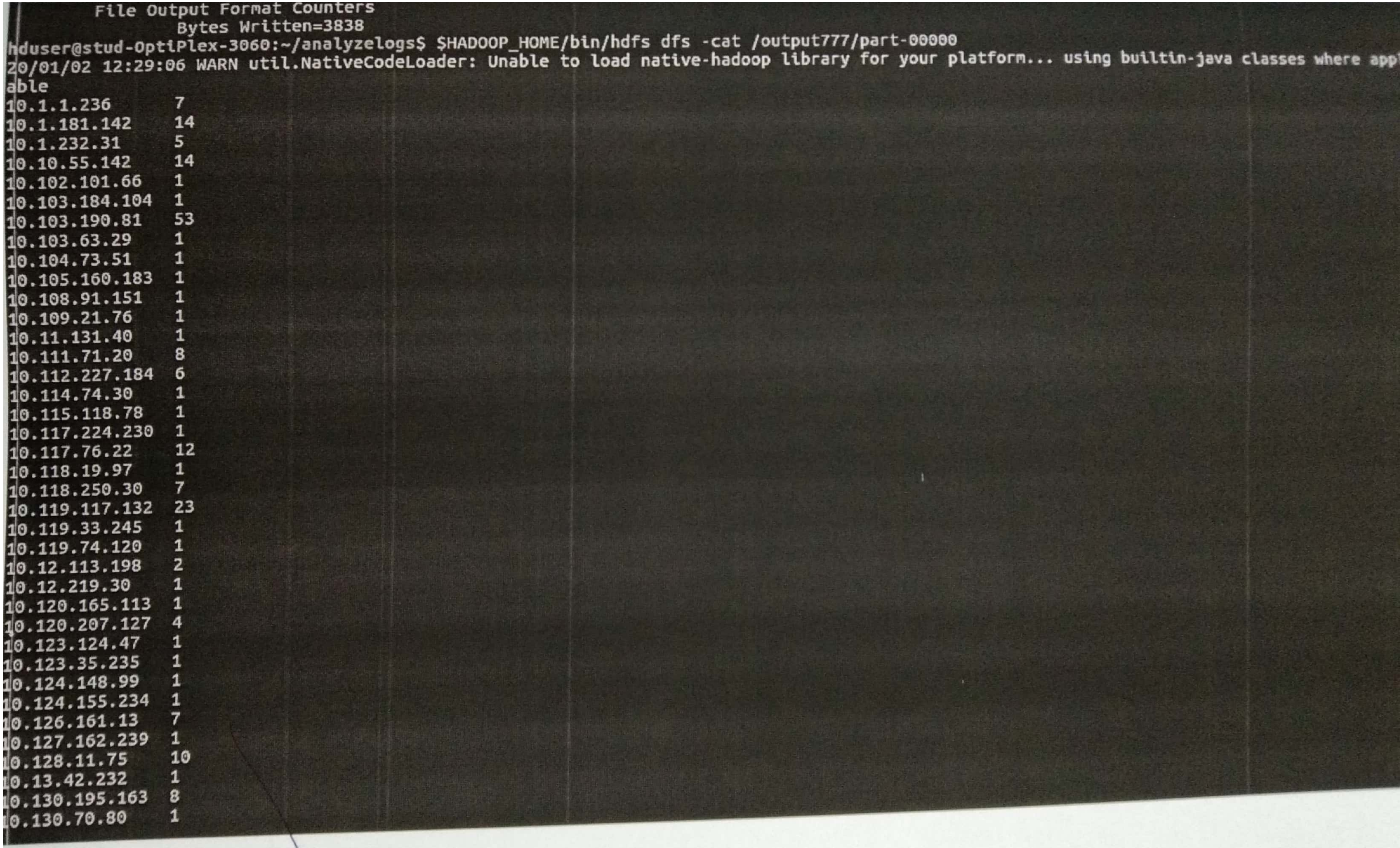
SalesMapper.java

```
package SalesCountry;

import java.io.IOException;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.*;

public class SalesMapper extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable> {
        private final static IntWritable one = new IntWritable(1);

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

                String valueString = value.toString();
                String[] SingleCountryData = valueString.split("-");
                output.collect(new Text(SingleCountryData[0]), one);
        }
}
```

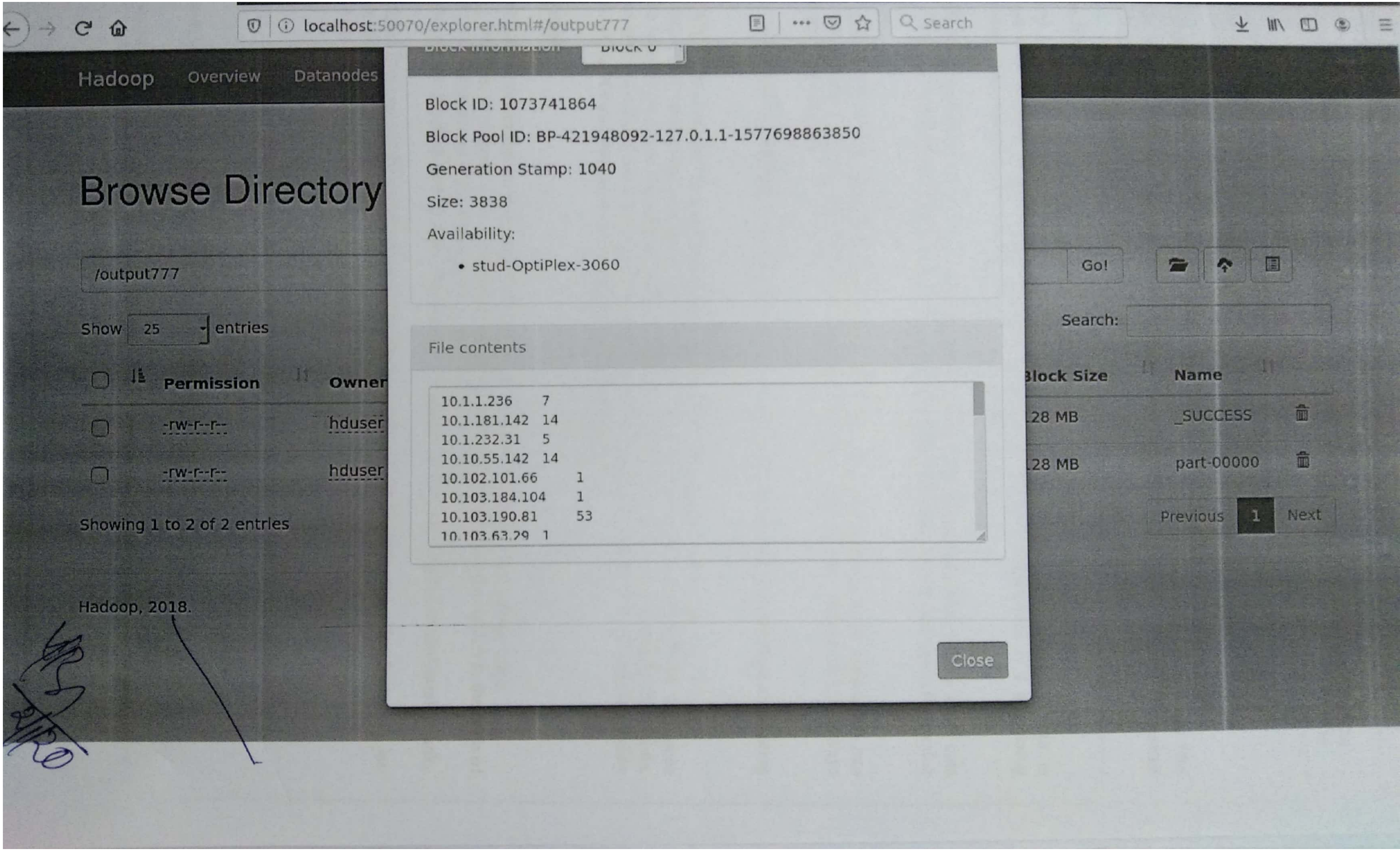
SalesCountryReducer.java

```
package SalesCountry;

import java.io.IOException;
import java.util.*;

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

public class SalesCountryReducer extends MapReduceBase implements Reducer<Text, IntWritable, Text, IntWritable> {

	public void reduce(Text t_key, Iterator<IntWritable> values, OutputCollector<Text,IntWritable> output, Reporter reporter)
throws IOException {
			Text key = t_key;
			int frequencyForCountry = 0;
			while (values.hasNext()) {
				// replace type of value with the actual type of our value
				IntWritable value = (IntWritable) values.next();
				frequencyForCountry += value.get();

			}
			output.collect(key, new IntWritable(frequencyForCountry));
	}
}
```

20/1/20



```
        File Output Format Counters
                Bytes Written=3838
hduser@stud-OptiPlex-3060:~/analyzelogs$ $HADOOP_HOME/bin/hdfs dfs -cat /output777/part-00000
20/01/02 12:29:06 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where app
able
10.1.1.236      7
10.1.181.142    14
10.1.232.31     5
10.10.55.142    14
10.102.101.66   1
10.103.184.104  1
10.103.190.81   53
10.103.63.29    1
10.104.73.51    1
10.105.160.183  1
10.108.91.151   1
10.109.21.76    1
10.11.131.40    1
10.111.71.20    8
10.112.227.184  6
10.114.74.30    1
10.115.118.78   1
10.117.224.230  1
10.117.76.22    12
10.118.19.97    1
10.118.250.30   7
10.119.117.132  23
10.119.33.245   1
10.119.74.120   1
10.12.113.198   2
10.12.219.30    1
10.120.165.113  1
10.120.207.127  4
10.123.124.47   1
10.123.35.235   1
10.124.148.99   1
10.124.155.234  1
10.126.161.13   7
10.127.162.239  1
10.128.11.75    10
10.13.42.232    1
10.130.195.163  8
10.130.70.80    1
```



Block ID: 1073741864

Block Pool ID: BP-421948092-127.0.1.1-1577698863850

Generation Stamp: 1040

Size: 3838

Availability:

- stud-OptiPlex-3060

File contents

```
10.1.1.236      7
10.1.181.142  14
10.1.232.31    5
10.10.55.142  14
10.102.101.66       1
10.103.184.104      1
10.103.190.81       53
10.103.63.29   1
```

Browse Directory

/output777

Show 25 entries

Search:

□	Permission	Owner	Block Size	Name
□	-rw-r--r--	hduser	128 MB	_SUCCESS
□	-rw-r--r--	hduser	128 MB	part-00000

Showing 1 to 2 of 2 entries

Hadoop, 2018.