Assignment 4: Map Reduce Introduction

We have a dataset of sales of different TV sets across different locations.

Records look like:
Samsung|Optima|14|Madhya Pradesh|132401|14200

The fields are arranged like:
Company Name|Product Name|Size in inches|State|Pin Code|Price

There are some invalid records which contain 'NA' in either Company Name or Product Name.

1. Write a Map Reduce program to filter out the invalid records. Map only job will fit for this Context.

Solution: Actual Data:

```
Samsung|Optima|14|Madhya Pradesh|132401|14200
Onida|Lucid|18|Uttar Pradesh|232401|16200
Akai|Decent|16|Kerala|922401|12200
Lava|Attention|20|Assam|454601|24200
Zen|Super|14|Maharashtra|619082|9200
Samsung|Optima|14|Madhya Pradesh|132401|14200
Onida|Lucid|18|Uttar Pradesh|232401|16200
Onida|Decent|14|Uttar Pradesh|232401|16200
Onida|NA|16|Kerala|922401|12200
Lava|Attention|20|Assam|454601|24200
Zen|Super|14|Maharashtra|619082|9200
Samsung|Optima|14|Madhya Pradesh|132401|14200
NA|Lucid|18|Uttar Pradesh|232401|16200
Samsung|Decent|16|Kerala|922401|12200
Lava|Attention|20|Assam|454601|24200
Samsung|Super|14|Maharashtra|619082|9200
Samsung|Super|14|Maharashtra|619082|9200
Samsung|Super|14|Maharashtra|619082|9200
```

Driver Code:

```
package InvalidRecords;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.mapreduce.Job;
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;
import org.w3c.dom.Text;

public class InvalidRecord

{

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

```
        {
                Configuration conf=new Configuration();
                Job job=new Job(conf, "Invalid Data");

                job.setJarByClass(InvalidRecord.class);

                job.setMapOutputKeyClass(Text.class);
                job.setMapOutputValueClass(Text.class);

                job.setMapperClass(InvalidRecordsMapper.class);
                job.setNumReduceTasks(0);

                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);

        }
}
```

Mapper Code:

```
package InvalidRecords;

import java.io.IOException;

import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class InvalidRecordsMapper extends Mapper<LongWritable,Text,Text,Text>
{
        public void map(LongWritable key,Text value,Context context) throws IOException,
InterruptedException
        {
                String line=value.toString();
                String[]linearray=line.split("\\|");

                if(!(linearray[0].equals("NA")||linearray[1].equals("NA")))
                {
                        context.write(new Text(line), new Text());
                }
        }

}
```

Output:

```
[acadgild@localhost hadoop]$ hadoop fs -cat /user/acadgild/hadoop/InvalidRecordsoutput/*
Java HotSpot(TM) Client VM warning: You have loaded library /home/acadgild/hadoop-2.7.2/lib/native/libhadoop.so.1.0.0 which might have di
sabled stack guard. The VM will try to fix the stack guard now.
It's highly recommended that you fix the library with 'execstack -c <libfile>', or link it with '-z noexecstack'.
17/10/12 17:55:37 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
Samsung|Optima|14|Madhya Pradesh|132401|14200
Onida|Lucid|18|Uttar Pradesh|232401|16200
Akai|Decent|16|Kerala|922401|12200
Lava|Attention|20|Assam|454601|24200
Zen|Super|14|Maharashtra|619082|9200
Samsung|Optima|14|Madhya Pradesh|132401|14200
Onida|Lucid|18|Uttar Pradesh|232401|16200
Onida|Decent|14|Uttar Pradesh|232401|16200
Lava|Attention|20|Assam|454601|24200
Zen|Super|14|Maharashtra|619082|9200
Samsung|Optima|14|Madhya Pradesh|132401|14200
Samsung|Decent|16|Kerala|922401|12200
Lava|Attention|20|Assam|454601|24200
Samsung|Super|14|Maharashtra|619082|9200
Samsung|Super|14|Maharashtra|619082|9200
Samsung|Super|14|Maharashtra|619082|9200
```

2. Write a Map Reduce program to calculate the total units sold for each Company

Solution:

Driver Code:

```
package TotalUnitSale;

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.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 TotalUnitSale
{

        public static void main(String[] args) throws Exception
        {
                Configuration conf = new Configuration();
```

```
                Job job = new Job(conf, "TV TotalUnitSale");// the job runs under this

                job.setJarByClass(TotalUnitSale.class);

                job.setMapOutputKeyClass(Text.class); //mapper key output
                job.setMapOutputValueClass(IntWritable.class); //mapper output value

                job.setOutputKeyClass(Text.class);// output key of the mapreduce
                job.setOutputValueClass(IntWritable.class);//output value of the mapreduce

                job.setMapperClass(TotalUnitSaleMapper.class);// Mapper class
                job.setReducerClass(TotalUnitSaleReducer.class);//reducer class

                job.setNumReduceTasks(2);

                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);

        }

}
```

Mapper Code:

```
package TotalUnitSale;


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.mapreduce.Mapper;

public class TotalUnitSaleMapper extends Mapper<LongWritable, Text, Text, IntWritable>
{
        private final static IntWritable unit = new IntWritable(1); // declaring the Mapper
value
        private Text CompanyName = new Text();                  //declaring      the
Mapper key

        public void map(LongWritable key, Text value, Context context ) throws IOException,
InterruptedException
        {
```

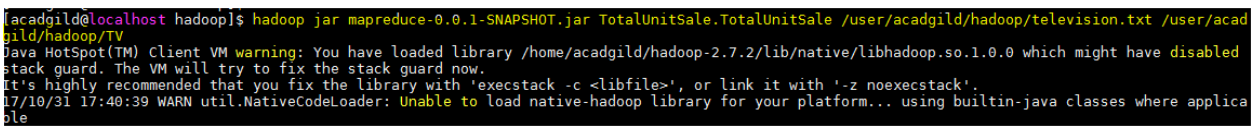
```
                String[] Linearray = value.toString().split("\\|");
                StringTokenizer tokenizer=new StringTokenizer(Linearray[0]); //we have used the String Tokenizer class which takes array into single word/token.
                while(tokenizer.hasMoreTokens()) // the while loop checks for the more tokens/words, if we have next token it will continue the loop
                {
                        CompanyName.set(tokenizer.nextToken());

        }
                        context.write(CompanyName, unit); // output of the Mapper Key and Value

}
}
```

Reducer Code:

```
        package TotalUnitSale;

        import java.io.IOException;

        import org.apache.hadoop.io.IntWritable;
        import org.apache.hadoop.io.Text;
        import org.apache.hadoop.mapreduce.Reducer;

        public class TotalUnitSaleReducer extends Reducer<Text, IntWritable, Text, IntWritable>
        {
                public void reduce(Text CompanyName, Iterable<IntWritable> values, Context context) throws IOException, InterruptedException
                {
                        int sum=0;              // declaring a variable sum
                        for(IntWritable value:values) // the for loop get the iterable values and counting the values
                        {
                                sum+=value.get();
                        }
                        context.write(CompanyName, new IntWritable(sum)); // output of the the Key and value
                }

        }
```

Command:

hadoop jar mapreduce-0.0.1-SNAPSHOT.jar TotalUnitSale.TotalUnitSale/user/acadgild/hadoop/television.txt /user/acadgild/hadoop/TV



```
17/10/31 17:41:56 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applica
ble
Found 11 items
drwxr-xr-x   - acadgild supergroup          0 2017-10-12 21:15 /user/acadgild/hadoop/InvalidDataMR
drwxr-xr-x   - acadgild supergroup          0 2017-10-12 18:46 /user/acadgild/hadoop/InvalidRecord2
drwxr-xr-x   - acadgild supergroup          0 2017-10-12 17:44 /user/acadgild/hadoop/InvalidRecordsoutput
drwxr-xr-x   - acadgild supergroup          0 2017-10-31 17:26 /user/acadgild/hadoop/OnidaTV
drwxr-xr-x   - acadgild supergroup          0 2017-10-31 17:41 /user/acadgild/hadoop/TV
-rw-r--r--   1 acadgild supergroup       1958 2017-10-13 18:56 /user/acadgild/hadoop/WordCount.txt
-rw-r-----   1 acadgild supergroup        237 2017-09-25 11:10 /user/acadgild/hadoop/max-temp.txt
drwxr-xr-x   - acadgild supergroup          0 2017-09-24 14:31 /user/acadgild/hadoop/maxout
-rw-r--r--   1 acadgild supergroup      21007 2017-09-24 14:25 /user/acadgild/hadoop/sample_temperature_dataset.csv
-rw-r--r--   1 acadgild supergroup        733 2017-10-12 10:25 /user/acadgild/hadoop/television.txt
-rw-r--r--   1 acadgild supergroup        300 2017-09-24 14:16 /user/acadgild/hadoop/word-count.txt
[acadgild@localhost hadoop]$
```

Output:

```
[acadgild@localhost hadoop]$ hadoop fs -cat /user/acadgild/hadoop/TV/*
Java HotSpot(TM) Client VM warning: You have loaded library /home/acadgild/hadoop-2.7.2/lib/native/libhadoop.so.1.0.0 which might have disabled
stack guard. The VM will try to fix the stack guard now.
It's highly recommended that you fix the library with 'execstack -c <libfile>', or link it with '-z noexecstack'.
17/10/31 17:42:31 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applica
ble
NA      1
Onida   4
Zen     2
Akai    1
Lava    3
Samsung 7
[acadgild@localhost hadoop]$
```

3. Write a Map Reduce program to calculate the total units sold in each state for Onida company.

Solution:

Driver Code:

```
package OnidaTotalUnit;

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.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 OnidaTotalUnit

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

    {

            Configuration conf = new Configuration();
            Job job = new Job(conf, "Onida Total Unit");// the job runs under this

            job.setJarByClass(OnidaTotalUnit.class);

            job.setMapOutputKeyClass(Text.class); //mapper key output
            job.setMapOutputValueClass(IntWritable.class); //mapper output value
```

```
        job.setOutputKeyClass(Text.class);//output key of the mapreduce
        job.setOutputValueClass(IntWritable.class); //output value of the mapreduce

        job.setMapperClass(OnidaMapper.class); // mapper class
        job.setReducerClass(OnidaReducer.class);// reducer class

        job.setNumReduceTasks(2);

        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);
    }

}
```

Mapper Code:

```
package OnidaTotalUnit;

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.mapreduce.Mapper;


public class OnidaMapper extends Mapper<LongWritable, Text, Text, IntWritable>
{
    public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException
    {

        String[] Linearray = value.toString().split("\\|"); //the array is split into string value and stored in Linearray
        if(Linearray[0].equals("Onida")) // checking the word Onida in the linearray[0], if it is Onida print the state name in linearray[3]and unit value
        {
            Text State = new Text(Linearray[3]);
            IntWritable unit= new IntWritable(1);
            context.write(State, unit);
        }
    }
```

```
    }
```

Reducer Code:

```
package OnidaTotalUnit;

import java.io.IOException;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

public class OnidaReducer extends Reducer<Text, IntWritable, Text, IntWritable>
{
    public void reduce(Text State, Iterable<IntWritable> values, Context context) throws IOException, InterruptedException
    {
        int sum = 0;  // declaring the variable sum
        for(IntWritable value:values)  // the for loop get the iterable values and counting the values
        {
            sum+= value.get();
        }

        context.write(State, new IntWritable(sum)); // print the state name which is the key and the number of units stored in the sum
    }

}
```

Command:

hadoop jar mapreduce-0.0.1-SNAPSHOT.jar OnidaTotalUnit.OnidaTotalUnit/user/acadgild/hadoop/television.txt/user/acadgild/hadoop/OnidaTV

```
[acadgild@localhost hadoop]$
[acadgild@localhost hadoop]$ hadoop jar mapreduce-0.0.1-SNAPSHOT.jar OnidaTotalUnit.OnidaTotalUnit /user/acadgild/hadoop/television.txt /user/acadgild/hadoop/OnidaTV
Java HotSpot(TM) Client VM warning: You have loaded library /home/acadgild/hadoop-2.7.2/lib/native/libhadoop.so.1.0.0 which might have disabled stack guard. The VM will try to fix the stack guard now.
It's highly recommended that you fix the library with 'execstack -c <libfile>', or link it with '-z noexecstack'.
17/10/31 17:25:21 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

```
17/10/31 17:27:51 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 13 items
drwxr-xr-x   - acadgild supergroup          0 2017-10-12 21:15 /user/acadgild/hadoop/InvalidDataMR
drwxr-xr-x   - acadgild supergroup          0 2017-10-12 18:46 /user/acadgild/hadoop/InvalidRecord2
drwxr-xr-x   - acadgild supergroup          0 2017-10-12 17:44 /user/acadgild/hadoop/InvalidRecordsoutput
drwxr-xr-x   - acadgild supergroup          0 2017-10-26 12:48 /user/acadgild/hadoop/Onida
drwxr-xr-x   - acadgild supergroup          0 2017-10-26 12:51 /user/acadgild/hadoop/Onida1
drwxr-xr-x   - acadgild supergroup          0 2017-10-26 16:15 /user/acadgild/hadoop/Onida4
drwxr-xr-x   - acadgild supergroup          0 2017-10-31 17:26 /user/acadgild/hadoop/OnidaTV
-rw-r--r--   1 acadgild supergroup       1958 2017-10-13 18:56 /user/acadgild/hadoop/WordCount.txt
-rw-r-----   1 acadgild supergroup        237 2017-09-25 11:10 /user/acadgild/hadoop/max-temp.txt
drwxr-xr-x   - acadgild supergroup          0 2017-09-24 14:31 /user/acadgild/hadoop/maxout
-rw-r--r--   1 acadgild supergroup      21007 2017-09-24 14:25 /user/acadgild/hadoop/sample_temperature_dataset.csv
-rw-r--r--   1 acadgild supergroup        733 2017-10-12 10:25 /user/acadgild/hadoop/television.txt
-rw-r--r--   1 acadgild supergroup        300 2017-09-24 14:16 /user/acadgild/hadoop/word-count.txt
[acadgild@localhost hadoop]$
[acadgild@localhost hadoop]$
[acadgild@localhost hadoop]$ hadoop fs -cat /user/acadgild/hadoop/OnidaTV/*
```

Output:

```
[acadgild@localhost hadoop]$ hadoop fs -cat /user/acadgild/hadoop/OnidaTV/*
Java HotSpot(TM) Client VM warning: You have loaded library /home/acadgild/hadoop-2.7.2/lib/native/libhadoop.so.1.0.0 which might have disabled
stack guard. The VM will try to fix the stack guard now.
It's highly recommended that you fix the library with 'execstack -c <libfile>', or link it with '-z noexecstack'.
17/10/31 17:28:21 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applica
ble
Uttar Pradesh   3
Kerala  1
[acadgild@localhost hadoop]$
```