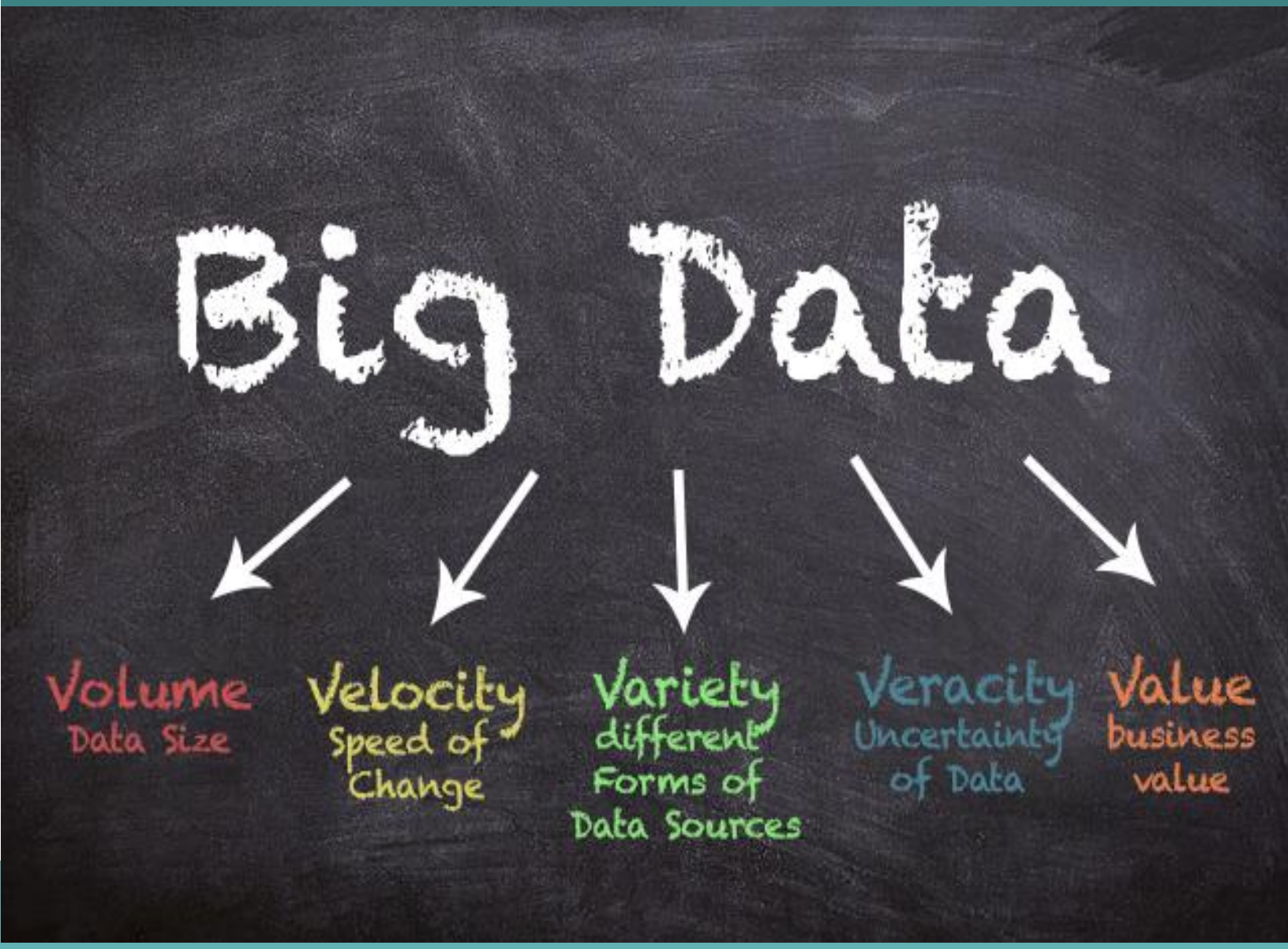
07/31/2018 | By: Ajit K Prasad



Big Data Engineering with Hadoop & Spark

Assignment on Advance MapReduce
&
Introduction to Unix Concepts

Acadgild

Session 5: Assignment 5.1

This assignment is aimed at consolidating the concepts that was learned during the Advance MapReduce & Unix Introduction session of the course.

Problem Statement

- Dataset as sample data of songs heard by users on an online streaming platform is provided in a file name ***"musicdata.txt"***.
  111115|222|0|1|0
  111113|225|1|0|0
  111117|223|0|1|1
  111115|225|1|0|0
- The Description of data set attached in musicdata.txt is as follows: -
  - 1st Column - UserId
  - 2nd Column - TrackId
  - 3rd Column - Songs Share status (1 for shared, 0 for not shared)
  - 4th Column - Listening Platform (Radio or Web - 0 for radio, 1 for web)
  - 5th Column - Song Listening Status (0 for skipped, 1 for fully heard)
- Write Map Reduce program for following tasks.

Solution:

- MapReduce codes were written for each task in Java using Eclipse IDE
- Java Jar files for each MapReduce code was generated which was used to execute MR jobs for the tasks.

Task 1:

– Find the number of unique listeners in the data set.

Solution:

DRIVER CODE:

```
package task1;
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.lib.output.TextOutputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.NullWritable;
public class uniqueListeners
        {
        public static void main(String[] args) throws Exception
        {
                if (args.length != 2)
                {
                System.err.println("Usage: uniqueListeners <input path> <output path>");
                System.exit(-1);
                }
                //Job Related Configurations
                Configuration conf = new Configuration();
                Job job = new Job(conf, "Unique_Listeners_Job");
                job.setJarByClass (uniqueListeners.class);

                //number of reducers set to 1
                job.setNumReduceTasks(1);
                //Provide paths to pick the input file for the job
                FileInputFormat.setInputPaths(job, new Path(args[0]));
                //Provide paths to pick the output file for the job, and delete it if already
                present
                Path outputPath = new Path(args[1]);
                FileOutputFormat.setOutputPath(job, outputPath);
                outputPath.getFileSystem(conf).delete(outputPath, true);
                //To set the Mapper and Reducer of this job
                job.setMapperClass(uniqueListenersMapper.class);
                job.setReducerClass(uniqueListenersReducer.class);
                //set the input and output format class
                job.setInputFormatClass(TextInputFormat.class);
                job.setOutputFormatClass(TextOutputFormat.class);
```

```
		//We set output key class as Text and as output value class as IntWritable
		job.setOutputKeyClass(Text.class);
		job.setOutputValueClass(IntWritable.class);
		//execute the job
		System.exit(job.waitForCompletion(true) ? 0 : 1);
	}
}
```

MAPPER CODE:

```
package task1;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
import java.util.*;
public class uniqueListenersMapper
	extends Mapper<LongWritable, Text, Text, IntWritable>
	{
	private final static IntWritable one = new IntWritable(1);
	@Override
	public void map(LongWritable key, Text value, Context context)
		throws IOException, InterruptedException
		{
			// Here we are converting Text to String
			String content = value.toString();
			String[] linesArray = content.split(" ");
			for(String line : linesArray)
			{
				//we are splitting line by pipe (|)
				String[] word = line.split("\\|");
				//we are assigning listener values from word
				Text listeners = new Text(word[0]);
				context.write(listeners,one);
			}
		}
	}
```

REDUCER CODE:

```
package task1;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
import java.util.HashSet;
public class uniqueListenersReducer
	extends Reducer<Text, IntWritable, Text, IntWritable>
```

```
{
private int sum;
@Override
protected void setup(Context context)
{
        sum = 0;
}
@Override
public void reduce(Text key, Iterable<IntWritable> values,
        Context context) throws IOException, InterruptedException
        {
                System.out.println("From The Reducer=>"+key);
                //we are using HashSet to have unique count of listeners
                HashSet<Integer> set = new HashSet<Integer>();
                for (IntWritable value : values)
                 {
                        if(set.add(value.get()))
                        sum+=value.get();
                }
        }
        @Override
protected void cleanup(Context context) throws IOException,
InterruptedException
{
        context.write(new Text("Number of unique listeners"), new
        IntWritable(sum));
}
}
```

Command to execute:

- *$ hadoop jar Assignment5_task1.jar /musicdata.txt /Assignment5_task1output*
- *$ hadoop fs -cat /Assignmennt5_task1output/part-r-00000*

```
[acadgild@localhost ~]$ hadoop jar Assignment5 task1.jar /musicdata.txt /Assignment5 taskloutput
18/07/28 20:18:13 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
18/07/28 20:18:15 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1:8032
18/07/28 20:18:17 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and
execute your application with ToolRunner to remedy this.
18/07/28 20:18:18 INFO input.FileInputFormat: Total input paths to process : 1
18/07/28 20:18:18 INFO mapreduce.JobSubmitter: number of splits:1
18/07/28 20:18:18 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1532787724818_0002
18/07/28 20:18:19 INFO impl.YarnClientImpl: Submitted application application_1532787724818_0002
18/07/28 20:18:19 INFO mapreduce.Job: The url to track the job: http://localhost:8088/proxy/application_1532787724818_0002/
18/07/28 20:18:19 INFO mapreduce.Job: Running job: job_1532787724818_0002
18/07/28 20:18:33 INFO mapreduce.Job: Job job_1532787724818_0002 running in uber mode : false
18/07/28 20:18:33 INFO mapreduce.Job:  map 0% reduce 0%
18/07/28 20:18:43 INFO mapreduce.Job:  map 100% reduce 0%
18/07/28 20:18:56 INFO mapreduce.Job:  map 100% reduce 100%
18/07/28 20:18:57 INFO mapreduce.Job: Job job_1532787724818_0002 completed successfully
18/07/28 20:18:57 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=58
                FILE: Number of bytes written=215459
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=172
                HDFS: Number of bytes written=29
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=7170
                Total time spent by all reduces in occupied slots (ms)=10660
                Total time spent by all map tasks (ms)=7170
                Total time spent by all reduce tasks (ms)=10660
                Total vcore-milliseconds taken by all map tasks=7170
                Total vcore-milliseconds taken by all reduce tasks=10660
                Total megabyte-milliseconds taken by all map tasks=7342080
                Total megabyte-milliseconds taken by all reduce tasks=10915840
        Map-Reduce Framework
                Map input records=4
                Map output records=4
                Map output bytes=44
                Map output materialized bytes=58
                Input split bytes=100
                Combine input records=0
                Combine output records=0
                Reduce input groups=3
                Reduce shuffle bytes=58
                Reduce input records=4
                Reduce output records=1
                Spilled Records=8
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=217
                CPU time spent (ms)=2470
                Physical memory (bytes) snapshot=296804352
                Virtual memory (bytes) snapshot=4118200320
                Total committed heap usage (bytes)=170004480
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=72
        File Output Format Counters
                Bytes Written=29
You have new mail in /var/spool/mail/acadgild
```

```
[acadgild@localhost ~]$ hadoop fs -ls /Assignment5_taskloutput
18/07/28 20:23:00 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup          0 2018-07-28 20:18 /Assignment5_taskloutput/_SUCCESS
-rw-r--r--   1 acadgild supergroup         29 2018-07-28 20:18 /Assignment5_taskloutput/part-r-00000
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ hadoop fs -cat /Assignment5_taskloutput/part-r-00000
18/07/28 20:23:07 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
Number of unique listeners      3
```

Task 2:

– What are the number of times a song was heard fully.

Solution:

DRIVER CODE:

```
package task2;
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.lib.output.TextOutputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.NullWritable;
public class fullSongCount
{
	public static void main(String[] args) throws Exception
	{
		if (args.length != 2)
		{
		System.err.println("Usage: uniqueListeners <input path> <output path>");
		System.exit(-1);
		}
	 //Job Related Configurations
	Configuration conf = new Configuration();
	Job job = new Job(conf, "Full_Song_Count_Job");
	job.setJarByClass(fullSongCount.class);
	//number of reducers set to 1
	job.setNumReduceTasks(1);
	//Provide paths to pick the input file for the job
	FileInputFormat.setInputPaths(job, new Path(args[0]));
	//Provide paths to pick the output file for the job, and delete it if already present
	Path outputPath = new Path(args[1]);
	FileOutputFormat.setOutputPath(job, outputPath);
	outputPath.getFileSystem(conf).delete(outputPath, true);
	//To set the mapper and Reducer of this job
	job.setMapperClass(fullSongCountMapper.class);
	job.setReducerClass(fullSongCountReducer.class);
	//set the input and output format class
	job.setInputFormatClass(TextInputFormat.class);
	job.setOutputFormatClass(TextOutputFormat.class);
	//We set output key as NullWritable as we are not returning key
	job.setOutputKeyClass(Text.class);
```

```
        job.setOutputValueClass(IntWritable.class);
        //execute the job
        System.exit(job.waitForCompletion(true) ? 0 : 1);
        }
}
```

MAPPER CODE:

```
package task2;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
import java.util.*;
public class fullSongCountMapper
        extends Mapper<LongWritable, Text, Text, IntWritable>
        {
                private final static IntWritable one = new IntWritable(1);
                @Override
                public void map(LongWritable key, Text value, Context context)
                        throws IOException, InterruptedException
                        {
                                String content = value.toString();
                                String[] linesArray = content.split(" ");
                                for(String line : linesArray){
                                //we are splitting line by pipe (|)
                                String[] word = line.split("\\|");
                                // Store 1st and 5th column values
                                Text fully_heard = new Text(word[4]);
                                Text listener = new Text(word[0]);
                                // Select only those lines which have 5th column value as "1"
                                if(fully_heard.equals(new Text("1")))
                                context.write(listener,one);
                        }
        }
}
```

REDUCER CODE:

```
package task2;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
public class fullSongCountReducer
        extends Reducer<Text, IntWritable, Text, IntWritable>
        {
```

```
		private int sum;
		@Override
		protected void setup(Context context)
		{
			sum = 0;
		}
		@Override
		public void reduce(Text key, Iterable<IntWritable> values, Context context)
			throws IOException, InterruptedException
			{
				for (IntWritable value : values) {
				sum+=value.get();
			}
		}
@Override
		protected void cleanup(Context context)
			throws IOException, InterruptedException
			{
				context.write(new Text("number of times a song was heard
				fully"), new IntWritable(sum));
			}
	}
```

Command to execute and pull output:

- *$ hadoop jar Assignment5_task2.jar /musicdata.txt /Assignment5_task2output*
- *$ hadoop fs -cat /Assignmennt5_task2output/part-r-00000*

```
[acadgild@localhost ~]$ hadoop jar Assignment5_task2.jar /musicdata.txt /Assignment5_task2output
18/07/28 20:41:48 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
18/07/28 20:41:50 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1:8032
18/07/28 20:41:52 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and
execute your application with ToolRunner to remedy this.
18/07/28 20:41:53 INFO input.FileInputFormat: Total input paths to process : 1
18/07/28 20:41:53 INFO mapreduce.JobSubmitter: number of splits:1
18/07/28 20:41:53 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1532787724818_0003
18/07/28 20:41:54 INFO impl.YarnClientImpl: Submitted application application_1532787724818_0003
18/07/28 20:41:54 INFO mapreduce.Job: The url to track the job: http://localhost:8088/proxy/application_1532787724818_0003/
18/07/28 20:41:54 INFO mapreduce.Job: Running job: job_1532787724818_0003
18/07/28 20:42:11 INFO mapreduce.Job: Job job_1532787724818_0003 running in uber mode : false
18/07/28 20:42:11 INFO mapreduce.Job:  map 0% reduce 0%
18/07/28 20:42:22 INFO mapreduce.Job:  map 100% reduce 0%
18/07/28 20:42:36 INFO mapreduce.Job:  map 100% reduce 100%
18/07/28 20:42:37 INFO mapreduce.Job: Job job_1532787724818_0003 completed successfully
18/07/28 20:42:38 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=45
                FILE: Number of bytes written=215423
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=172
                HDFS: Number of bytes written=41
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=8410
                Total time spent by all reduces in occupied slots (ms)=11301
                Total time spent by all map tasks (ms)=8410
                Total time spent by all reduce tasks (ms)=11301
                Total megabyte-milliseconds taken by all map tasks=8611840
                Total megabyte-milliseconds taken by all reduce tasks=11572224
        Map-Reduce Framework
                Map input records=4
                Map output records=3
                Map output bytes=33
                Map output materialized bytes=45
                Input split bytes=100
                Combine input records=0
                Combine output records=0
                Reduce input groups=2
                Reduce shuffle bytes=45
                Reduce input records=3
                Reduce output records=1
                Spilled Records=6
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=274
                CPU time spent (ms)=4000
                Physical memory (bytes) snapshot=300085248
                Virtual memory (bytes) snapshot=4118192128
                Total committed heap usage (bytes)=170004480
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=72
        File Output Format Counters
                Bytes Written=41
You have new mail in /var/spool/mail/acadgild
```

```
[acadgild@localhost ~]$ hadoop fs -ls /Assignment5_task2output
18/07/29 00:19:08 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup          0 2018-07-29 00:16 /Assignment5_task2output/_SUCCESS
-rw-r--r--   1 acadgild supergroup         41 2018-07-29 00:16 /Assignment5_task2output/part-r-00000
[acadgild@localhost ~]$ hadoop fs -cat /Assignment5_task2output/part-r-00000
18/07/29 00:19:26 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
number of times a song was heard fully  1
```

Task 3:

– What are the number of times a song was shared.

Solution:

DRIVER CODE:

```
package task3;
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.lib.output.TextOutputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.io.LongWritable;
public class shareSongCount
{
	public static void main(String[] args) throws Exception
	{
		if (args.length != 2)
		{
		System.err.println("Usage: uniqueListeners <input path> <output path>");
		System.exit(-1);
		}
	//Job Related Configurations
	Configuration conf = new Configuration();
	Job job = new Job(conf, "Full_Song_Count_Job");
	job.setJarByClass(shareSongCount.class);
	//number of reducers set to 1
	job.setNumReduceTasks(1);
	//Provide paths to pick the input file for the job
	FileInputFormat.setInputPaths(job, new Path(args[0]));
	//Provide paths to pick the output file for the job, and delete it if already present
	Path outputPath = new Path(args[1]);
	FileOutputFormat.setOutputPath(job, outputPath);
	outputPath.getFileSystem(conf).delete(outputPath, true);
	//To set the mapper of this job and there is no Reducer
	job.setMapperClass(shareSongCountMapper.class);
	job.setReducerClass(shareSongCountReducer.class);
	//set the input and output format class
	job.setInputFormatClass(TextInputFormat.class);
	job.setOutputFormatClass(TextOutputFormat.class);
	//We set output key as NullWritable as we are not returning key
	job.setOutputKeyClass(Text.class);
	job.setOutputValueClass(IntWritable.class);
```

```
        //execute the job
        System.exit(job.waitForCompletion(true) ? 0 : 1);
        }
}
```

MAPPER CODE:

```
package task3;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
import java.util.*;
public class shareSongCountMapper
        extends Mapper<LongWritable, Text, Text, IntWritable>
        {
                private final static IntWritable one = new IntWritable(1);
                @Override
                public void map(LongWritable key, Text value, Context context)
                        throws IOException, InterruptedException
                        {
                                String content = value.toString();
                                String[] linesArray = content.split(" ");
                                for(String line : linesArray){
                                //we are splitting line by pipe (|)
                                String[] word = line.split("\\|");
                                // Store 1st and 3rd column values
                                Text song_shared = new Text(word[2]);
                                Text listener = new Text(word[0]);
                                // Select only those lines which have 3rd column value as "1"
                                if(song_shared.equals(new Text("1")))
                                context.write(listener,one);
                        }
        }
```

REDUCER CODE:

```
package task3;
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
public class shareSongCountReducer
        extends Reducer<Text, IntWritable, Text, IntWritable>
        {
                private int sum;
                @Override
```

```
        protected void setup(Context context)
        {
                sum = 0;
        }
        @Override
        public void reduce(Text key, Iterable<IntWritable> values, Context context)
                throws IOException, InterruptedException
                {
                        System.out.println("From The Reducer=>"+key);
                        for (IntWritable value : values)
                        {
                                sum+=value.get();
                        }
                }
                @Override
        protected void cleanup(Context context)
                throws IOException, InterruptedException
                {
                        context.write(new Text("Number of times a song was shared"),
                        new IntWritable(sum));
                }
}
```

Command to execute:

- *$ hadoop jar Assignment5_task3.jar /musicdata.txt /Assignment5_task3output*
- *$ hadoop fs -cat /Assignmennt5_task3output/part-r-00000*

```
[acadgild@localhost ~]$ hadoop jar Assignment5_task3.jar /musicdata.txt /Assignment5_task3output
18/07/29 00:33:41 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
18/07/29 00:33:43 INFO client.RMProxy: Connecting to ResourceManager at localhost/127.0.0.1:8032
18/07/29 00:33:44 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and
execute your application with ToolRunner to remedy this.
18/07/29 00:33:46 INFO input.FileInputFormat: Total input paths to process : 1
18/07/29 00:33:46 INFO mapreduce.JobSubmitter: number of splits:1
18/07/29 00:33:46 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1532821119785_0010
18/07/29 00:33:46 INFO impl.YarnClientImpl: Submitted application application_1532821119785_0010
18/07/29 00:33:47 INFO mapreduce.Job: The url to track the job: http://localhost:8088/proxy/application_1532821119785_0010/
18/07/29 00:33:47 INFO mapreduce.Job: Running job: job_1532821119785_0010
18/07/29 00:34:00 INFO mapreduce.Job: Job job_1532821119785_0010 running in uber mode : false
18/07/29 00:34:00 INFO mapreduce.Job:  map 0% reduce 0%
18/07/29 00:34:14 INFO mapreduce.Job:  map 100% reduce 0%
18/07/29 00:34:28 INFO mapreduce.Job:  map 100% reduce 100%
18/07/29 00:34:28 INFO mapreduce.Job: Job job_1532821119785_0010 completed successfully
18/07/29 00:34:28 INFO mapreduce.Job: Counters: 49
        File System Counters
                FILE: Number of bytes read=32
                FILE: Number of bytes written=215419
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=172
                HDFS: Number of bytes written=36
                HDFS: Number of read operations=6
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=10999
                Total time spent by all reduces in occupied slots (ms)=11108
                Total time spent by all map tasks (ms)=10999
                Total time spent by all reduce tasks (ms)=11108
                Total vcore-milliseconds taken by all map tasks=10999
                Total vcore-milliseconds taken by all reduce tasks=11108
                Total megabyte-milliseconds taken by all map tasks=11262976
                Total megabyte-milliseconds taken by all reduce tasks=11374592
        Map-Reduce Framework
                Map input records=4
                Map output records=2
                Map output bytes=22
                Map output materialized bytes=32
                Input split bytes=100
                Combine input records=0
                Combine output records=0
                Reduce input groups=2
                Reduce shuffle bytes=32
                Reduce input records=2
                Reduce output records=1
                Spilled Records=4
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=323
                CPU time spent (ms)=2570
                Physical memory (bytes) snapshot=296923136
                Virtual memory (bytes) snapshot=4118192128
                Total committed heap usage (bytes)=170004480
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=72
        File Output Format Counters
                Bytes Written=36
```

```
[acadgild@localhost ~]$ hadoop fs -ls /Assignment5_task3output
18/07/29 00:34:37 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup          0 2018-07-29 00:34 /Assignment5_task3output/_SUCCESS
-rw-r--r--   1 acadgild supergroup         36 2018-07-29 00:34 /Assignment5_task3output/part-r-00000
[acadgild@localhost ~]$ hadoop fs -cat /Assignment5_task3output/part-r-00000
18/07/29 00:34:45 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
applicable
Number of times a song was shared       2
```