Lab 1

Roll No.: J019 – Ayush Hendre, J031—Rohit Mittal

Aim: Word Count Using Map Reduce

Objectives:

1.To run Java command.

2. Copy Data file from Local to HDFS.

3. Generate a Word count query.

4. Display Word count of the file

Code & Output:

WCDriver

```
//Driver:
// Importing libraries
import
java.io.IOException;
import
org.apache.hadoop.conf.Configured;
import org.apache.hadoop.fs.Path;
import
org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import
org.apache.hadoop.mapred.FileInputFormat;
import
org.apache.hadoop.mapred.FileOutputFormat;
import org.apache.hadoop.mapred.JobClient;
import
org.apache.hadoop.mapred.JobConf;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;

public class WCDriver extends Configured implements Tool {

        public int run(String args[]) throws IOException
        {
                if (args.length < 2)
                {
                        System.out.println("Please give valid
                        inputs"); return -1;
```

```
}
```

```
                JobConf conf = new JobConf(WCDriver.class);
                FileInputFormat.setInputPaths(conf, new
                Path(args[0]));
                FileOutputFormat.setOutputPath(conf, new
                Path(args[1]));
                conf.setMapperClass(WCMapper.class);
                conf.setReducerClass(WCReducer.class);
                conf.setMapOutputKeyClass(Text.class);
                conf.setMapOutputValueClass(IntWritable.class);
                conf.setOutputKeyClass(Text.class);
                conf.setOutputValueClass(IntWritable.class);
                JobClient.runJob(conf);
                return 0;
        }

        // Main Method
        public static void main(String args[]) throws Exception
        {
                int exitCode = ToolRunner.run(new WCDriver(),
                args); System.out.println(exitCode);
        }
}
```

WCMapper

Mapper:

```
// Importing libraries
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.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
import
org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;

public class WCMapper extends MapReduceBase implements Mapper<LongWritable,
```

Text, Text, IntWritable> {

```
  // Map function
  public void map(LongWritable key, Text value,
          OutputCollector<Text, IntWritable> output, Reporter
          rep) throws IOException
  {
    String line = value.toString();
    // Splitting the line on
    spaces for (String word :
    line.split(" "))
    {
      if (word.length() > 0)
      {
        output.collect(new Text(word), new IntWritable(1));
      }
    }
  }
}
```

WCReducer

```
//Reducer:

// Importing libraries
import
java.io.IOException;
import
java.util.Iterator;
import
org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
i       m       p       o       r       t
org.apache.hadoop.mapred.MapReduceBase;
i       m       p       o       r       t
org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;

public class WCReducer extends MapReduceBase implements Reducer<Text,
                                                    IntWritable, Text, IntWritable> {
        // Reduce function
        public void reduce(Text key, Iterator<IntWritable> value,
```

OutputCollector<Text, IntWritable>
output,

Reporter rep) throws IOException

```
        {
                int count = 0;
                // Counting the frequency of each
                words while (value.hasNext())
                {
                        IntWritable i =
                        value.next(); count +=
                        i.get();
                }
                output.collect(key, new IntWritable(count));
        }
}
```

```
hive> CREATE TABLE FILES1 (line STRING);
OK
Time taken: 0.099 seconds
hive> LOAD DATA INPATH 'random2.txt' OVERWRITE INTO TABLE FILES1;
Loading data to table default.files1
chgrp: changing ownership of 'hdfs://quickstart.cloudera:8020/user/hive/warehous
e/files1/random2.txt': User does not belong to supergroup
Table default.files1 stats: [numFiles=1, numRows=0, totalSize=152, rawDataSize=0
]
OK
Time taken: 0.507 seconds
```