Kaustubh Shrikant Kabra

ERP Number :- 38

TE Comp 1

Weather Analyse (average Temperature, dew point, wind speed)

Weather.java

```
import java.io.IOException;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.LongWritable;
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.mapred.KeyValueTextInputFormat;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;

/**
```

```
* This is an Hadoop Map/Reduce application for Working on weather data It reads
* the text input files, breaks each line into stations weather data and finds
* average for temperature , dew point , wind speed. The output is a locally
* sorted list of stations and its 12 attribute vector of average temp , dew ,
* wind speed of 4 sections for each month.
*
* To run: bin/hadoop jar target/weather-1.0.jar [-m <i>maps</i>] [-r
* <i>reduces</i>] <i>in-dir for job 1</i> <i>out-dir for job 1</i> <i>out-dir
* for job 2</i>
*/
public class Weather extends Configured implements Tool {
        final long DEFAULT_SPLIT_SIZE = 128 * 1024 * 1024;

        /**
         * Map Class for Job 1
         *
         * For each line of input, emits key value pair with
         * station_yearmonth_sectionno as key and 3 attribute vector with
         * temperature , dew point , wind speed as value.Map method will strip the
         * day and hour from field and replace it with section no (
         * <b>station_yearmonth_sectionno</b>, <b><temperature,dew point , wind
         * speed></b>).
         */
        public static class MapClass extends MapReduceBase
                        implements Mapper<LongWritable, Text, Text, Text> {

                private Text word = new Text();
                private Text values = new Text();

                public void map(LongWritable key, Text value,
                                                OutputCollector<Text, Text> output,
```

```
                                        Reporter reporter) throws IOException {
                String line = value.toString();
                StringTokenizer itr = new StringTokenizer(line);
                int counter = 0;
                String key_out = null;
                String value_str = null;
                boolean skip = false;
                loop:while (itr.hasMoreTokens() && counter<13) {
                        String str = itr.nextToken();
                        switch (counter) {
                        case 0:
                                key_out = str;
                                if(str.contains("STN")){//Ignoring rows where station id is all
9
                                        skip = true;
                                        break loop;
                                }else{
                                        break;
                                }
                        case 2:
                                int                     hour                    =
Integer.valueOf(str.substring(str.lastIndexOf("_")+1, str.length()));
                                str = str.substring(4,str.lastIndexOf("_")-2);
                                /*if(hour<=5){
                                        str = str.concat("_section4");
                                }else if(hour>5 && hour<=11){
                                        str = str.concat("_section1");
                                }else if(hour>11 && hour<=17){
                                        str = str.concat("_section2");
                                }else if(hour>17 && hour<=23){
                                        str = str.concat("_section3");
                                }*/
```

```
                if(hour>4 && hour<=10){
                        str = str.concat("_section1");
                }else if(hour>10 && hour<=16){
                        str = str.concat("_section2");
                }else if(hour>16 && hour<=22){
                        str = str.concat("_section3");
                }else{
                        str = str.concat("_section4");
                }


                key_out = key_out.concat("_").concat(str);
                break;
        case 3://Temperature
                if(str.equals("9999.9")){//Ignoring rows where temperature
is all 9
                        skip = true;
                        break loop;
                }else{
                        value_str = str.concat(" ");
                        break;
                }
        case 4://Dew point
                if(str.equals("9999.9")){//Ignoring rows where dew point is
all 9
                        skip = true;
                        break loop;
                }else{
                        value_str = value_str.concat(str).concat(" ");
                        break;
                }
```

```
                                case 12://Wind speed
                                        if(str.equals("999.9")){//Ignoring rows where wind speed is
all 9
                                                skip = true;
                                                break loop;
                                        }else{
                                                value_str = value_str.concat(str).concat(" ");
                                                break;
                                        }
                                default:
                                        break;
                                }
                                counter++;
                        }
                        if(!skip){
                                word.set(key_out);
                                values.set(value_str);
                                output.collect(word, values);
                        }
                }
        }
        /**
         * Reducer Class for Job 1
         *
         * A reducer class that just emits 3 attribute vector with average
         * temperature , dew point , wind speed for each of the section of the month
         * for each input
         */
        public static class Reduce extends MapReduceBase
                        implements Reducer<Text, Text, Text, Text> {
                private Text value_out_text = new Text();
```

```
        public void reduce(Text key, Iterator<Text> values,
                        OutputCollector<Text, Text> output, Reporter reporter) throws
IOException {
                double sum_temp = 0;
                double sum_dew = 0;
                double sum_wind = 0;
                int count = 0;

                while (values.hasNext()) {

                        String str = values.next().toString();

                        StringTokenizer itr = new StringTokenizer(str);
                        int count_vector = 0;
                        while (itr.hasMoreTokens()) {
                                String nextToken = itr.nextToken(" ");
                                if(count_vector==0){
                                        sum_temp += Double.valueOf(nextToken);
                                }
                                if(count_vector==1){
                                        sum_dew += Double.valueOf(nextToken);
                                }
                                if(count_vector==2){
                                        sum_wind += Double.valueOf(nextToken);
                                }
                                count_vector++;
                        }
                        count++;
                }
                double avg_tmp = sum_temp / count;
                double avg_dew = sum_dew / count;
```

```
                double avg_wind = sum_wind / count;
                System.out.println(key.toString()+" count is "+count+" sum of temp is "+sum_temp+" sum of dew is "+sum_dew+" sum of wind is "+sum_wind+"\n");
                String value_out = String.valueOf(avg_tmp).concat(" ").concat(String.valueOf(avg_dew)).concat(" ").concat(String.valueOf(avg_wind));
                value_out_text.set(value_out);
                output.collect(key, value_out_text);
            }
        }
        static int printUsage() {
            System.out.println("weather [-m <maps>] [-r <reduces>] <job_1 input> <job_1 output> <job_2 output>");
            ToolRunner.printGenericCommandUsage(System.out);
            return -1;
        }
        /**
         * The main driver for weather map/reduce program.
         * Invoke this method to submit the map/reduce job.
         * @throws IOException When there is communication problems with the
         *                     job tracker.
         */
        public int run(String[] args) throws Exception {
            Configuration config = getConf();

            // We need to lower input block size by factor of two.

            JobConf conf = new JobConf(config, Weather.class);
            conf.setJobName("Weather Job1");

            // the keys are words (strings)
            conf.setOutputKeyClass(Text.class);
            // the values are counts (ints)
```

```
conf.setOutputValueClass(Text.class);

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

conf.setMapperClass(MapClass.class);
//conf.setCombinerClass(Combiner.class);
conf.setReducerClass(Reduce.class);
List<String> other_args = new ArrayList<String>();
for(int i=0; i < args.length; ++i) {
        try {
                if ("-m".equals(args[i])) {
                        conf.setNumMapTasks(Integer.parseInt(args[++i]));
                } else if ("-r".equals(args[i])) {
                        conf.setNumReduceTasks(Integer.parseInt(args[++i]));
                } else {
                        other_args.add(args[i]);
                }
        } catch (NumberFormatException except) {
                System.out.println("ERROR: Integer expected instead of " + args[i]);
                return printUsage();
        } catch (ArrayIndexOutOfBoundsException except) {
                System.out.println("ERROR: Required parameter missing from " +
                        args[i-1]);
                return printUsage();
        }
}
// Make sure there are exactly 2 parameters left.
FileInputFormat.setInputPaths(conf, other_args.get(0));
FileOutputFormat.setOutputPath(conf, new Path(other_args.get(1)));
JobClient.runJob(conf);
```

```
        return 0;
    }
    public static void main(String[] args) throws Exception {
        int res = ToolRunner.run(new Configuration(), new Weather(), args);
        System.exit(res);
    }
}
```

Input: sample_weather.txt (sample)

```
690190 13910  20060201_0 51.75   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  000000
690190 13910  20060201_1 54.74   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  00000
690190 13910  20060201_2 50.59   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  00000
690190 13910  20060201_3 51.67   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  000000
690190 13910  20060201_4 65.67   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  000000
690190 13910  20060201_5 55.37   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  000000
690190 13910  20060201_6 49.26   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  000000
690190 13910  20060201_7 55.44   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  000000
690190 13910  20060201_8 64.05   33.0 24  1006.3 24  943.9 24  15.0 24  10.7 24  22.0  28.9   0.00I 999.9  000000
```

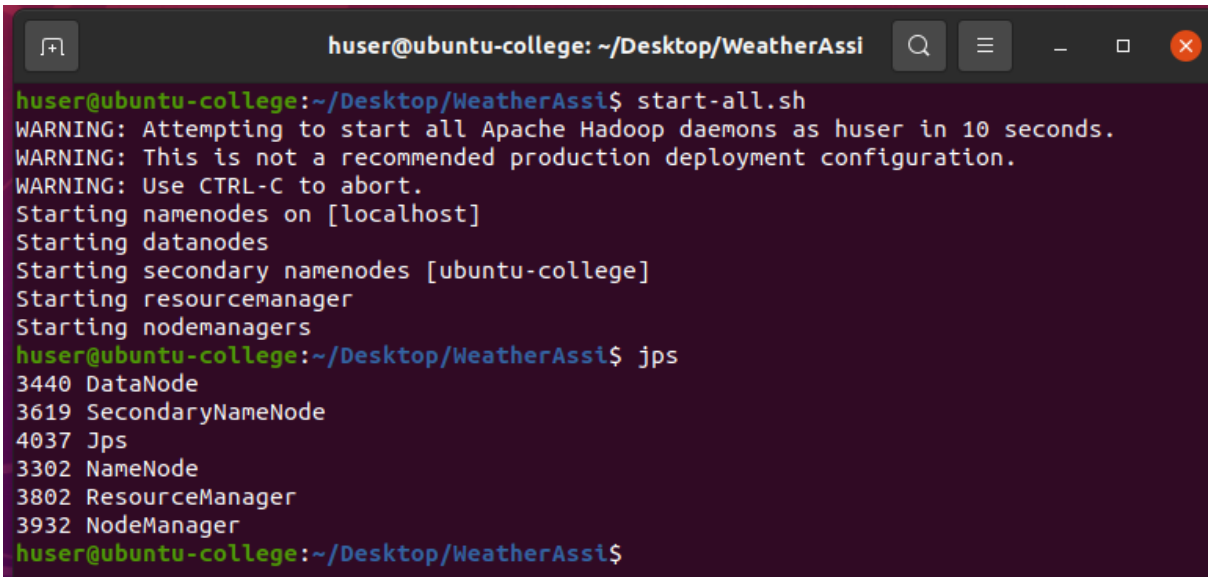
Output: part-00000.txt (on Hadoop)

```
690190_02_section1  53.87166666666666 25.899999999999995 7.774999999999998
690190_02_section2  54.76125000000001 25.900000000000006 7.774999999999999
690190_02_section3  53.25041666666667 25.899999999999995 7.774999999999996
690190_02_section4  52.44708333333333 25.900000000000006 7.774999999999999
```

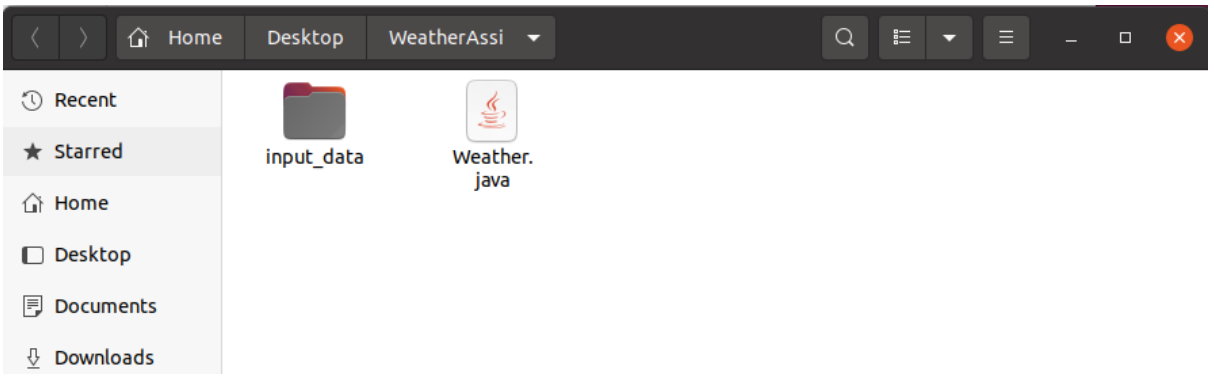
Weather Data Analysis Steps to run:

1. Starting Hadoop

   start-all.sh

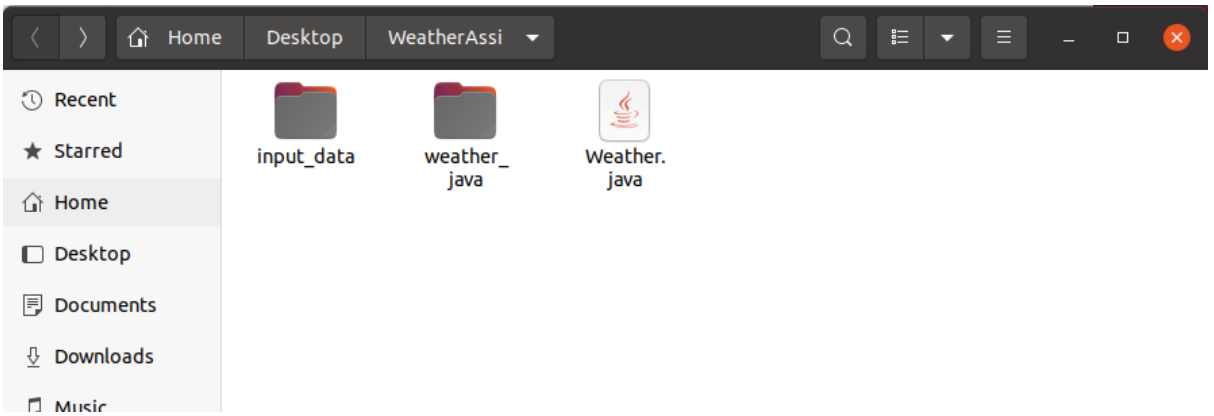


2. Made A folder “WeatherAssi” and write Weather.java code.
3. Create new folder for input data.



4. Add input text file in the input data folder.

5. Create new folder to hold java class files.



6. Set HADOOP_CLASSPATH environment variable.

   export HADOOP_CLASSPATH=$(hadoop classpath)

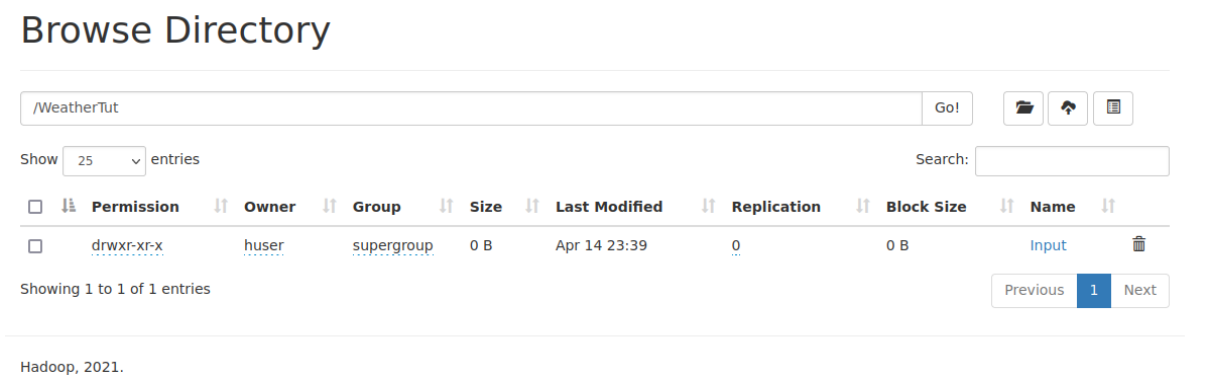
```
huser@ubuntu-college:~/Desktop/WeatherAssi$ export HADOOP_CLASSPATH=$(hadoop classpa
th)
huser@ubuntu-college:~/Desktop/WeatherAssi$ echo $HADOOP_CLASSPATH
/home/huser/hadoop/hadoop-3.2.2/etc/hadoop:/home/huser/hadoop/hadoop-3.2.2/share/had
oop/common/lib/*:/home/huser/hadoop/hadoop-3.2.2/share/hadoop/common/*:/home/huser/h
adoop/hadoop-3.2.2/share/hadoop/hdfs:/home/huser/hadoop/hadoop-3.2.2/share/hadoop/hd
fs/lib/*:/home/huser/hadoop/hadoop-3.2.2/share/hadoop/hdfs/*:/home/huser/hadoop/hado
op-3.2.2/share/hadoop/mapreduce/lib/*:/home/huser/hadoop/hadoop-3.2.2/share/hadoop/m
apreduce/*:/home/huser/hadoop/hadoop-3.2.2/share/hadoop/yarn:/home/huser/hadoop/hado
op-3.2.2/share/hadoop/yarn/lib/*:/home/huser/hadoop/hadoop-3.2.2/share/hadoop/yarn/*
huser@ubuntu-college:~/Desktop/WeatherAssi$
```

7. Create a directory on HDFS

   hdfs dfs -mkdir /WeatherTut

   hdfs dfs -mkdir /WeatherTut/Input

8. Checking on localhost:9870



9. Upload the input file (device) to that directory.

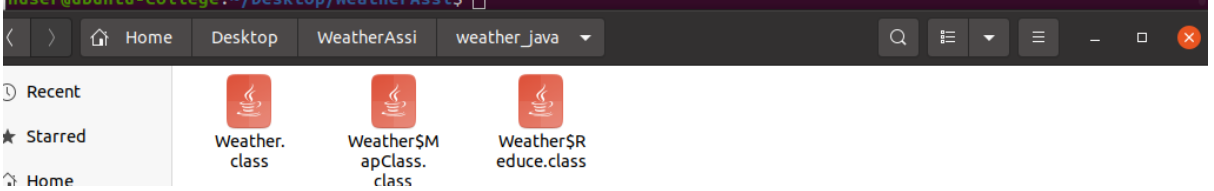
   hdfs dfs -put <Input file > <hdfs input dir>

```
huser@ubuntu-college:~/Desktop/WeatherAssi$ hdfs dfs -mkdir /WeatherTut
huser@ubuntu-college:~/Desktop/WeatherAssi$ hdfs dfs -mkdir /WeatherTut/Input
huser@ubuntu-college:~/Desktop/WeatherAssi$ hdfs dfs -put input_data/sample_weather.txt /WeatherTut/Input
huser@ubuntu-college:~/Desktop/WeatherAssi$
```

10. Compile the java code:

javac -classpath $(HADOOP_CLASSPATH) -d <Classes_folder> <java file>

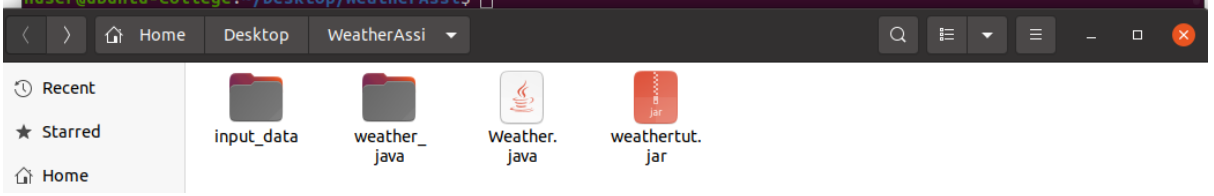
```
huser@ubuntu-college:~/Desktop/WeatherAssi$ hdfs dfs -put input_data/sample_weather.txt /WeatherTut/Input
huser@ubuntu-college:~/Desktop/WeatherAssi$ javac -classpath ${HADOOP_CLASSPATH} -d '/home/huser/Desktop/Weather
Assi/weather_java' /home/huser/Desktop/WeatherAssi/Weather.java
huser@ubuntu-college:~/Desktop/WeatherAssi$
```



11. Creation .jar file of classes:

jar -cvf <jar file name > -C <classes folder> .

```
huser@ubuntu-college:~/Desktop/WeatherAssi$ jar -cvf weathertut.jar -C /home/huser/Desktop/WeatherAssi/weather_j
ava/ .
added manifest
adding: Weather$Reduce.class(in = 2782) (out= 1285)(deflated 53%)
adding: Weather$MapClass.class(in = 2897) (out= 1378)(deflated 52%)
adding: Weather.class(in = 3267) (out= 1703)(deflated 47%)
huser@ubuntu-college:~/Desktop/WeatherAssi$
```



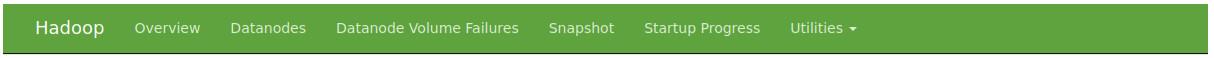
12. Running the jar file on Hadoop

hadoop jar <jar file> <class name> <hdfs input dir> <hdfs output dir>

```
huser@ubuntu-college:~/Desktop/WeatherAssi$ hadoop jar weathertut.jar Weather /WeatherTut/Input /WeatherTut/Outp
ut
2022-04-14 23:46:46,131 INFO client.RMProxy: Connecting to ResourceManager at /127.0.0.1:8032
2022-04-14 23:46:46,859 INFO client.RMProxy: Connecting to ResourceManager at /127.0.0.1:8032
2022-04-14 23:46:47,402 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/
staging/huser/.staging/job_1649959472953_0001
2022-04-14 23:46:47,836 INFO mapred.FileInputFormat: Total input files to process : 1
2022-04-14 23:46:47,988 INFO mapreduce.JobSubmitter: number of splits:2
2022-04-14 23:46:48,804 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1649959472953_0001
2022-04-14 23:46:48,806 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-04-14 23:46:49,499 INFO conf.Configuration: resource-types.xml not found
2022-04-14 23:46:49,500 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-04-14 23:46:50,212 INFO impl.YarnClientImpl: Submitted application application_1649959472953_0001
2022-04-14 23:46:50,314 INFO mapreduce.Job: The url to track the job: http://ubuntu-college:8088/proxy/applicati
on_1649959472953_0001/
2022-04-14 23:46:50,325 INFO mapreduce.Job: Running job: job_1649959472953_0001
```

```
                Peak Map Physical memory (bytes)=253128704
                Peak Map Virtual memory (bytes)=2524299264
                Peak Reduce Physical memory (bytes)=146530304
                Peak Reduce Virtual memory (bytes)=2532839424
        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=16149
        File Output Format Counters
                Bytes Written=296
huser@ubuntu-college:~/Desktop/WeatherAssi$
```

13. Check output on localhost:9870 /localhost:50070



Browse Directory

/WeatherTut Go!

Show 25 entries Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	huser	supergroup	0 B	Apr 14 23:40	0	0 B	Input
drwxr-xr-x	huser	supergroup	0 B	Apr 14 23:48	0	0 B	Output

Showing 1 to 2 of 2 entries Previous 1 Next

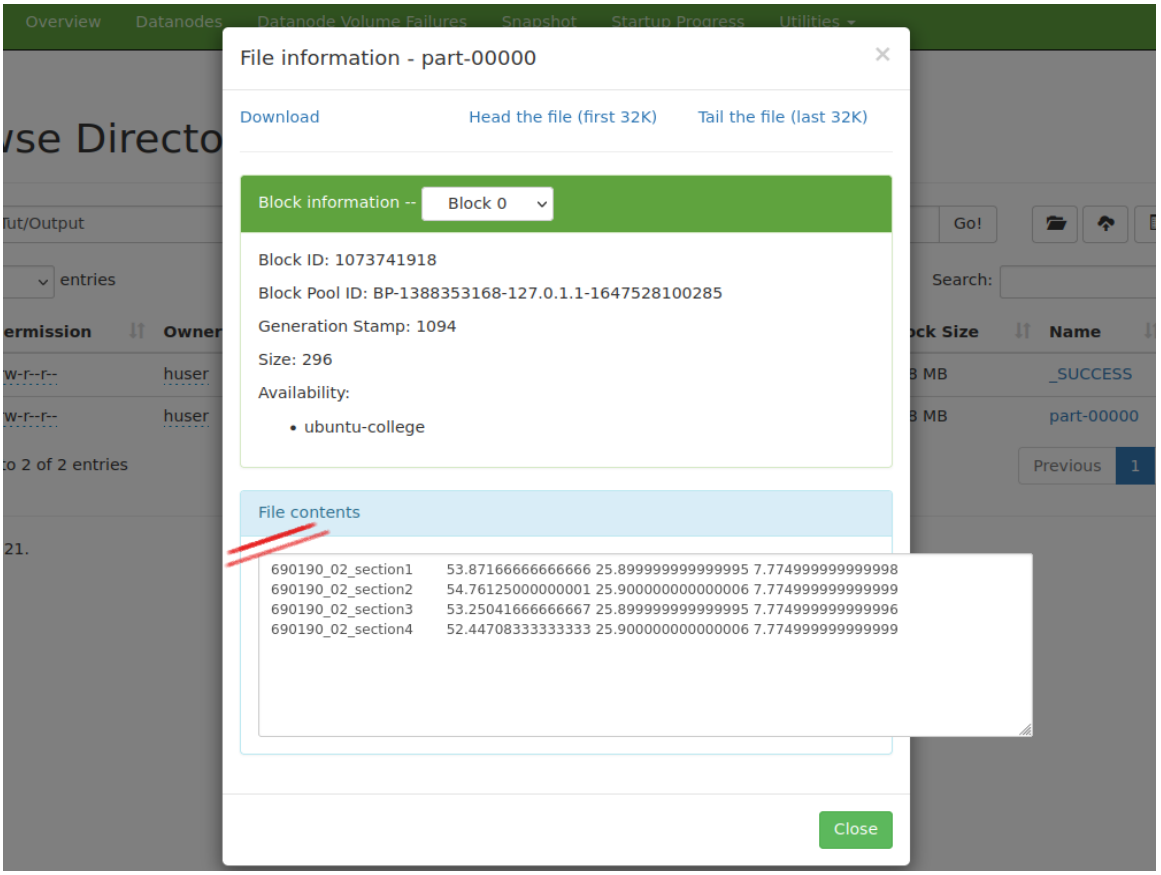
Browse Directory

/WeatherTut/Output Go!

Show 25 entries Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	huser	supergroup	0 B	Apr 14 23:48	1	128 MB	_SUCCESS
-rw-r--r--	huser	supergroup	296 B	Apr 14 23:48	1	128 MB	part-00000

Showing 1 to 2 of 2 entries Previous 1 Next



14. Stop Hadoop services:

stop-all.sh

```
huser@ubuntu-college:~/Desktop/WeatherAssi$ stop-all.sh
WARNING: Stopping all Apache Hadoop daemons as huser in 10 seconds.
WARNING: Use CTRL-C to abort.
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [ubuntu-college]
Stopping nodemanagers
localhost: WARNING: nodemanager did not stop gracefully after 5 seconds: Trying to kill with kill -9
Stopping resourcemanager
WARNING: resourcemanager did not stop gracefully after 5 seconds: Trying to kill with kill -9
huser@ubuntu-college:~/Desktop/WeatherAssi$
```