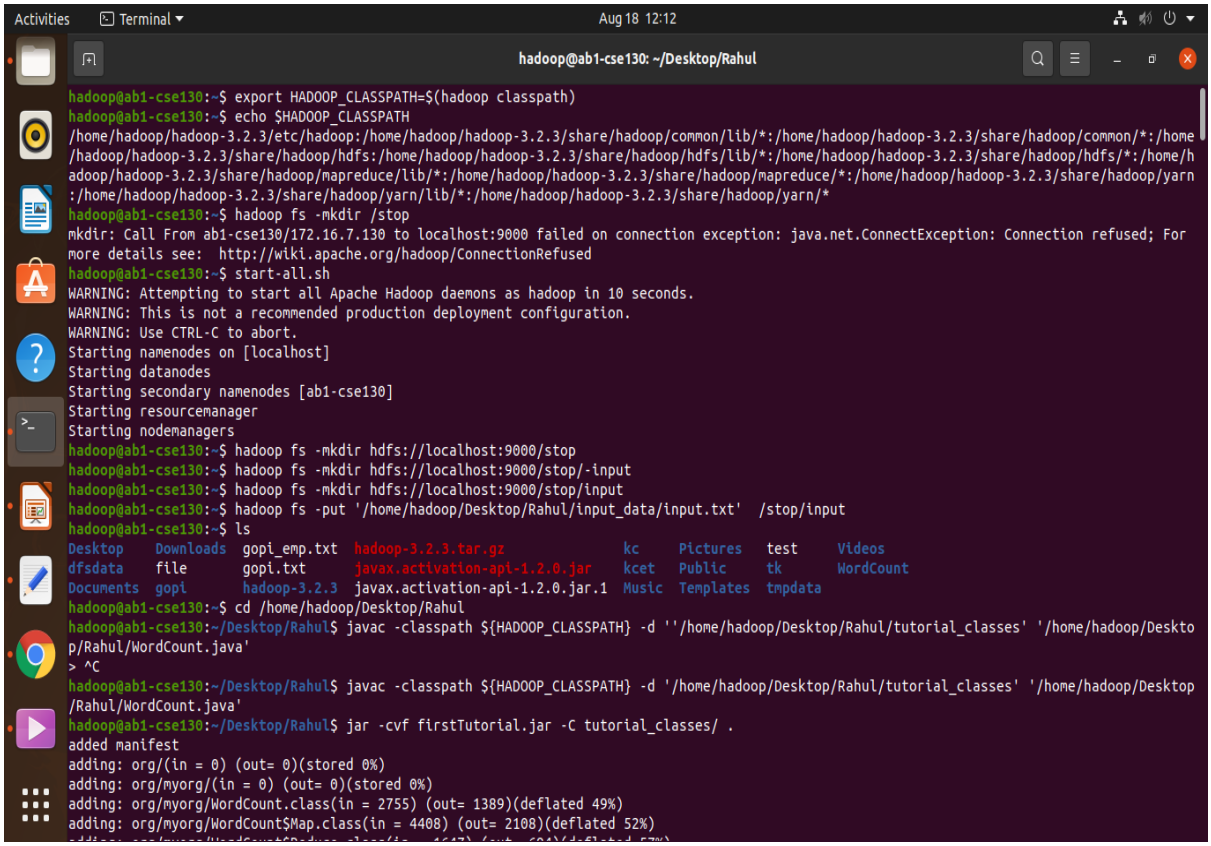
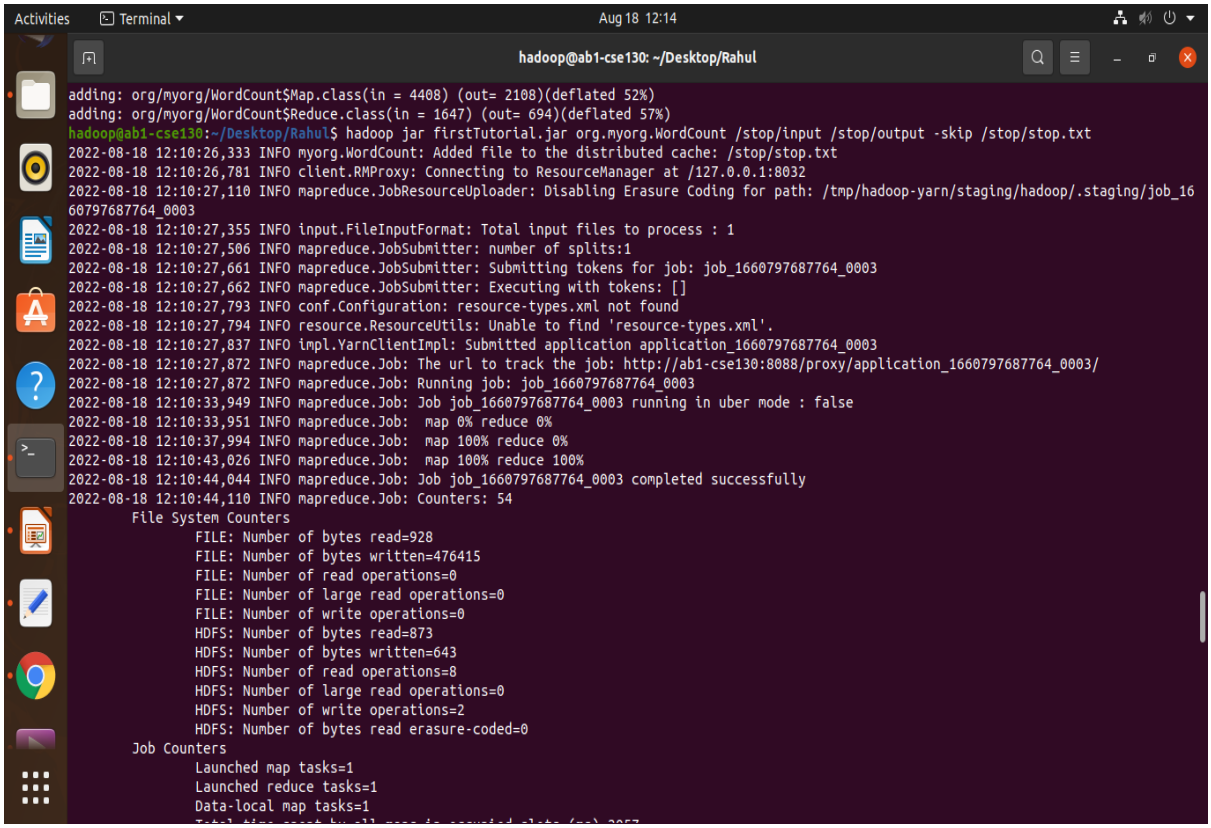
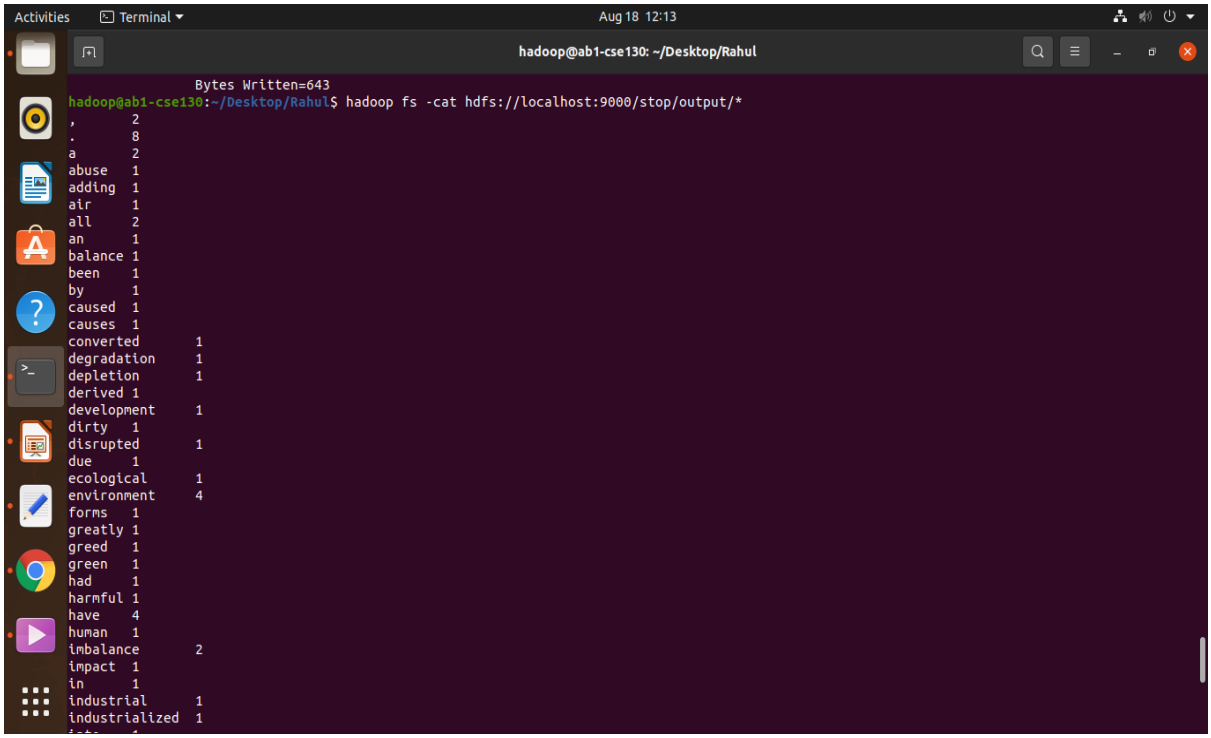
```
package org.myorg;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;   //Import classes for handling file I/O
import java.net.URI;
import java.util.HashSet;
import java.util.Set;
import java.io.IOException;
import java.util.regex.Pattern;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.FileSplit;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.util.StringUtils; //working with strings in Hadoop

import org.apache.log4j.Logger;

public class WordCount extends Configured implements Tool {

  private static final Logger LOG = Logger.getLogger(WordCount.class);

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

  public int run(String[] args) throws Exception {
    Job job = Job.getInstance(getConf(), "wordcount");
//Skip pattern configuration
    for (int i = 0; i < args.length; i += 1) {
      if ("-skip".equals(args[i])) {
        job.getConfiguration().setBoolean("wordcount.skip.patterns", true);
        i += 1;
        job.addCacheFile(new Path(args[i]).toUri());
        // this demonstrates logging
        LOG.info("Added file to the distributed cache: " + args[i]);
      }
    }
    job.setJarByClass(this.getClass());
    // Use TextInputFormat, the default unless job.setInputFormatClass is used
```

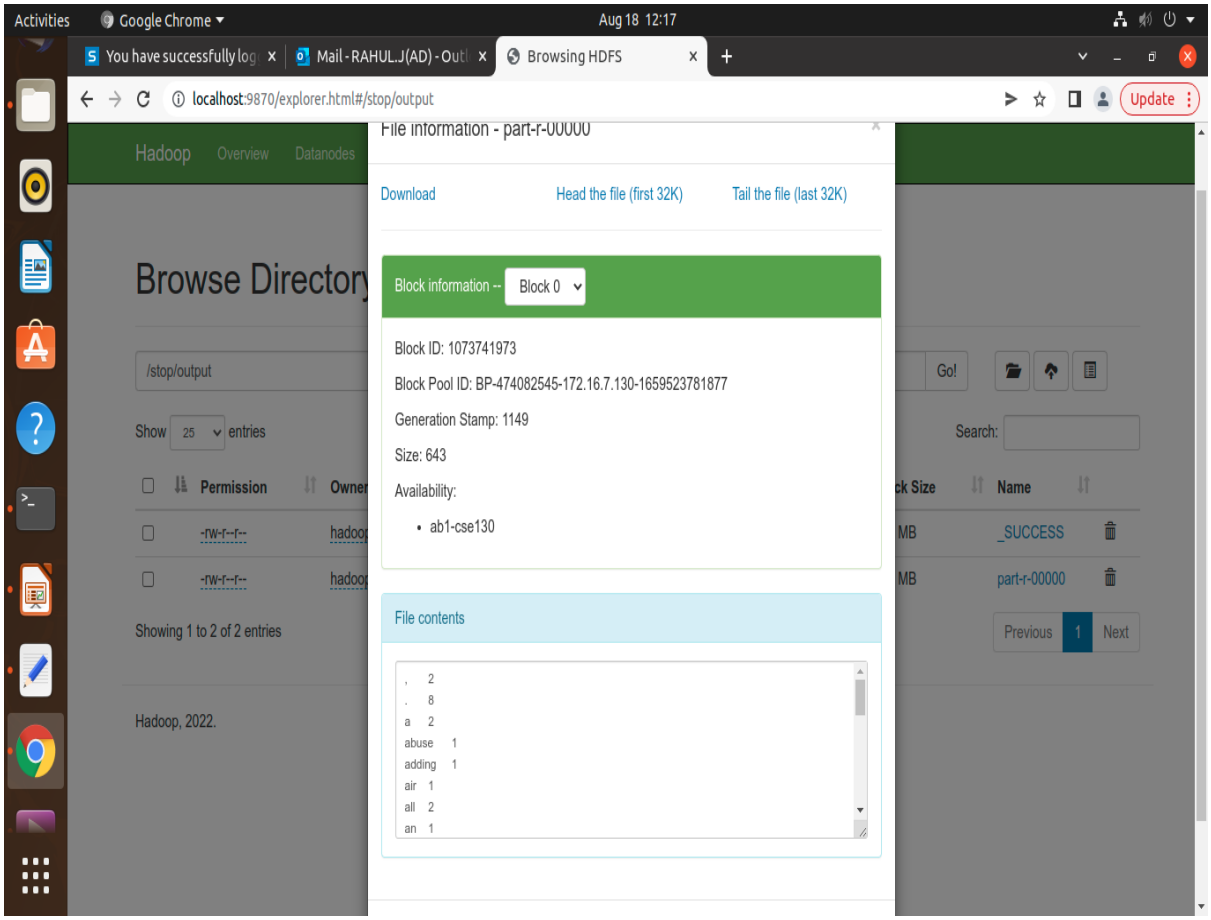
```
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    job.setMapperClass(Map.class);
    job.setCombinerClass(Reduce.class);
    job.setReducerClass(Reduce.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    return job.waitForCompletion(true) ? 0 : 1;
  }

  public static class Map extends Mapper<LongWritable, Text, Text, IntWritable> {
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();
    private boolean caseSensitive = false;
    private long numRecords = 0;
    private String input;
    private Set<String> patternsToSkip = new HashSet<String>(); //stop words to be removed
from the final result
    private static final Pattern WORD_BOUNDARY = Pattern.compile("\\s*\\b\\s*");

    protected void setup(Mapper.Context context)
        throws IOException,
        InterruptedException {
      if (context.getInputSplit() instanceof FileSplit) {
        this.input = ((FileSplit) context.getInputSplit()).getPath().toString();
      } else {
        this.input = context.getInputSplit().toString();
      }
      Configuration config = context.getConfiguration();
      this.caseSensitive = config.getBoolean("wordcount.case.sensitive", false);
//parseSkipFile method
      if (config.getBoolean("wordcount.skip.patterns", false)) {
        URI[] localPaths = context.getCacheFiles();
        parseSkipFile(localPaths[0]);
      }
    }
//Getting file from the HDFS and to read until EOL
    private void parseSkipFile(URI patternsURI) {
      LOG.info("Added file to the distributed cache: " + patternsURI);
      try {
        BufferedReader fis = new BufferedReader(new FileReader(new
File(patternsURI.getPath()).getName()));
        String pattern;
        while ((pattern = fis.readLine()) != null) {
          patternsToSkip.add(pattern);
        }
      } catch (IOException ioe) {
        System.err.println("Caught exception while parsing the cached file '"
            + patternsURI + "' : " + StringUtils.stringifyException(ioe));
      }
```

```
    }

    public void map(LongWritable offset, Text lineText, Context context)
        throws IOException, InterruptedException {
      String line = lineText.toString();
      if (!caseSensitive) {
        line = line.toLowerCase();
      }
      Text currentWord = new Text();
      for (String word : WORD_BOUNDARY.split(line)) {
        if (word.isEmpty() || patternsToSkip.contains(word)) {
            continue;
        }
            currentWord = new Text(word);
            context.write(currentWord,one);
        }
    }
  }

  public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable> {
    @Override
    public void reduce(Text word, Iterable<IntWritable> counts, Context context)
        throws IOException, InterruptedException {
      int sum = 0;
      for (IntWritable count : counts) {
        sum += count.get();
      }
      context.write(word, new IntWritable(sum));
    }
  }
}
```


Activities
Terminal
Aug 18 12:12
hadoop@ab1-cse130: ~/Desktop/Rahul
hadoop@ab1-cse130:~$ export HADOOP_CLASSPATH=$(hadoop classpath)
hadoop@ab1-cse130:~$ echo $HADOOP_CLASSPATH
/home/hadoop/hadoop-3.2.3/etc/hadoop:/home/hadoop/hadoop-3.2.3/share/hadoop/common/lib/*:/home/hadoop/hadoop-3.2.3/share/hadoop/common/*:/home/hadoop/hadoop-3.2.3/share/hadoop/hdfs:/home/hadoop/hadoop-3.2.3/share/hadoop/hdfs/lib/*:/home/hadoop/hadoop-3.2.3/share/hadoop/hdfs/*:/home/hadoop/hadoop-3.2.3/share/hadoop/mapreduce/lib/*:/home/hadoop/hadoop-3.2.3/share/hadoop/mapreduce/*:/home/hadoop/hadoop-3.2.3/share/hadoop/yarn:/home/hadoop/hadoop-3.2.3/share/hadoop/yarn/lib/*:/home/hadoop/hadoop-3.2.3/share/hadoop/yarn/*
hadoop@ab1-cse130:~$ hadoop fs -mkdir /stop
mkdir: Call From ab1-cse130/172.16.7.130 to localhost:9000 failed on connection exception: java.net.ConnectException: Connection refused; For more details see: http://wiki.apache.org/hadoop/ConnectionRefused
hadoop@ab1-cse130:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hadoop in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ab1-cse130]
Starting resourcemanager
Starting nodemanagers
hadoop@ab1-cse130:~$ hadoop fs -mkdir hdfs://localhost:9000/stop
hadoop@ab1-cse130:~$ hadoop fs -mkdir hdfs://localhost:9000/stop/-input
hadoop@ab1-cse130:~$ hadoop fs -mkdir hdfs://localhost:9000/stop/input
hadoop@ab1-cse130:~$ hadoop fs -put '/home/hadoop/Desktop/Rahul/input_data/input.txt' /stop/input
hadoop@ab1-cse130:~$ ls
Desktop Downloads gopi_emp.txt hadoop-3.2.3.tar.gz kc Pictures test Videos
dfsdata file gopi.txt javax.activation-api-1.2.0.jar kcet Public tk WordCount
Documents gopi hadoop-3.2.3 javax.activation-api-1.2.0.jar.1 Music Templates tmpdata
hadoop@ab1-cse130:~$ cd /home/hadoop/Desktop/Rahul
hadoop@ab1-cse130:~/Desktop/Rahul$ javac -classpath ${HADOOP_CLASSPATH} -d ''/home/hadoop/Desktop/Rahul/tutorial_classes' '/home/hadoop/Desktop/Rahul/WordCount.java'
> ^C
hadoop@ab1-cse130:~/Desktop/Rahul$ javac -classpath ${HADOOP_CLASSPATH} -d '/home/hadoop/Desktop/Rahul/tutorial_classes' '/home/hadoop/Desktop/Rahul/WordCount.java'
hadoop@ab1-cse130:~/Desktop/Rahul$ jar -cvf firstTutorial.jar -C tutorial_classes/ .
added manifest
adding: org/(in = 0) (out= 0)(stored 0%)
adding: org/myorg/(in = 0) (out= 0)(stored 0%)
adding: org/myorg/WordCount.class(in = 2755) (out= 1389)(deflated 49%)
adding: org/myorg/WordCount$Map.class(in = 4408) (out= 2108)(deflated 52%)


Activities
Terminal
Aug 18 12:14
hadoop@ab1-cse130: ~/Desktop/Rahul
adding: org/myorg/WordCount$Map.class(in = 4408) (out= 2108)(deflated 52%)
adding: org/myorg/WordCount$Reduce.class(in = 1647) (out= 694)(deflated 57%)
hadoop@ab1-cse130:~/Desktop/Rahul$ hadoop jar firstTutorial.jar org.myorg.WordCount /stop/input /stop/output -skip /stop/stop.txt
2022-08-18 12:10:26,333 INFO myorg.WordCount: Added file to the distributed cache: /stop/stop.txt
2022-08-18 12:10:26,781 INFO client.RMProxy: Connecting to ResourceManager at /127.0.0.1:8032
2022-08-18 12:10:27,110 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hadoop/.staging/job_1660797687764_0003
2022-08-18 12:10:27,355 INFO input.FileInputFormat: Total input files to process : 1
2022-08-18 12:10:27,506 INFO mapreduce.JobSubmitter: number of splits:1
2022-08-18 12:10:27,661 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1660797687764_0003
2022-08-18 12:10:27,662 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-08-18 12:10:27,793 INFO conf.Configuration: resource-types.xml not found
2022-08-18 12:10:27,794 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-08-18 12:10:27,837 INFO impl.YarnClientImpl: Submitted application application_1660797687764_0003
2022-08-18 12:10:27,872 INFO mapreduce.Job: The url to track the job: http://ab1-cse130:8088/proxy/application_1660797687764_0003/
2022-08-18 12:10:27,872 INFO mapreduce.Job: Running job: job_1660797687764_0003
2022-08-18 12:10:33,949 INFO mapreduce.Job: Job job_1660797687764_0003 running in uber mode : false
2022-08-18 12:10:33,951 INFO mapreduce.Job: map 0% reduce 0%
2022-08-18 12:10:37,994 INFO mapreduce.Job: map 100% reduce 0%
2022-08-18 12:10:43,026 INFO mapreduce.Job: map 100% reduce 100%
2022-08-18 12:10:44,044 INFO mapreduce.Job: Job job_1660797687764_0003 completed successfully
2022-08-18 12:10:44,110 INFO mapreduce.Job: Counters: 54
File System Counters
FILE: Number of bytes read=928
FILE: Number of bytes written=476415
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=873
HDFS: Number of bytes written=643
HDFS: Number of read operations=8
HDFS: Number of large read operations=0
HDFS: Number of write operations=2
HDFS: Number of bytes read erasure-coded=0
Job Counters
Launched map tasks=1
Launched reduce tasks=1
Data-local map tasks=1


Activities
Terminal
Aug 18 12:13
hadoop@ab1-cse130: ~/Desktop/Rahul
Bytes Written=643
hadoop@ab1-cse130:~/Desktop/Rahul$ hadoop fs -cat hdfs://localhost:9000/stop/output/*
, 2
. 8
a 2
abuse 1
adding 1
air 1
all 2
an 1
balance 1
been 1
by 1
caused 1
causes 1
converted 1
degradation 1
depletion 1
derived 1
development 1
dirty 1
disrupted 1
due 1
ecological 1
environment 4
forms 1
greatly 1
greed 1
green 1
had 1
harmful 1
have 4
human 1
imbalance 2
impact 1
in 1
industrial 1
industrialized 1


Activities
Google Chrome
Aug 18 12:17
You have successfully log
Mail - RAHUL.J(AD) - Outl
Browsing HDFS
localhost:9870/explorer.html#/stop/output
Update
Hadoop
Overview
Datanodes
Browse Directory
/stop/output
Go!
Show 25 entries
Search:
Permission
Owner
Size
Name
-rw-r--r--
hadoop
MB
_SUCCESS
-rw-r--r--
hadoop
MB
part-r-00000
Showing 1 to 2 of 2 entries
Previous
1
Next
Hadoop, 2022.
File information - part-r-00000
Download
Head the file (first 32K)
Tail the file (last 32K)
Block information --
Block 0
Block ID: 1073741973
Block Pool ID: BP-474082545-172.16.7.130-1659523781877
Generation Stamp: 1149
Size: 643
Availability:
ab1-cse130
File contents
, 2
. 8
a 2
abuse 1
adding 1
air 1
all 2
an 1