**EAST WEST UNIVERSITY**

# Big Data Analysis

**Course Code:** CSE **488**

**Section**: **01**

**Lab Assignment-** **2**

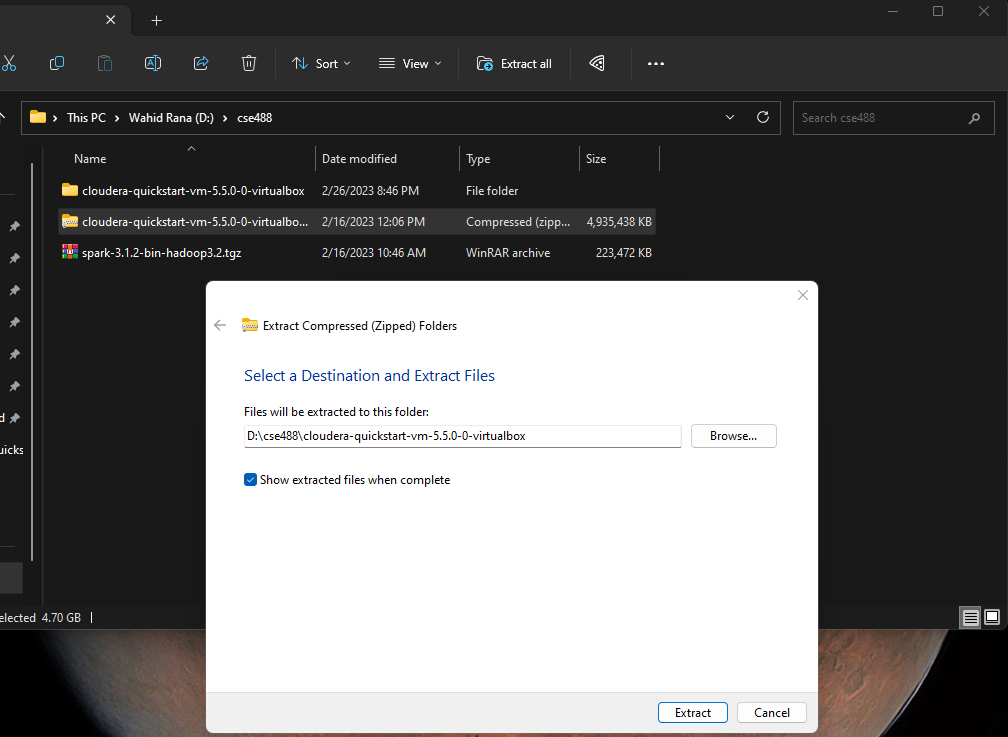
# **SUBMITTED TO:**

**Name:** Mohammad Rezwanul Huq

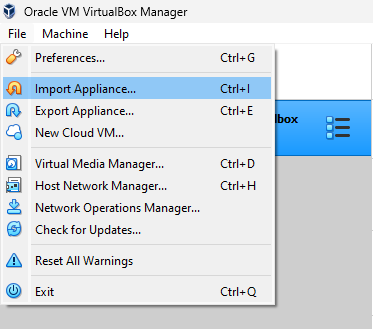
# **SUBMITTED BY:**

**Name:** MD. WAHIDUZZAMAN

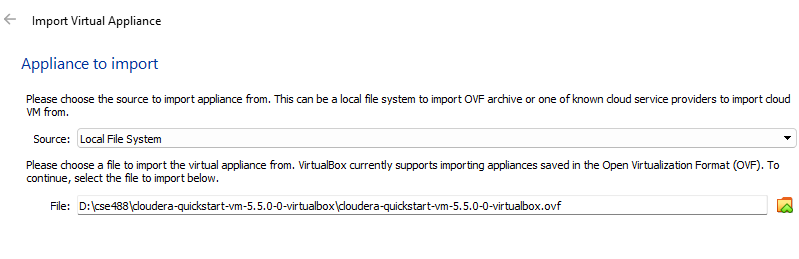
**ID No:** **2019-2-60-048**



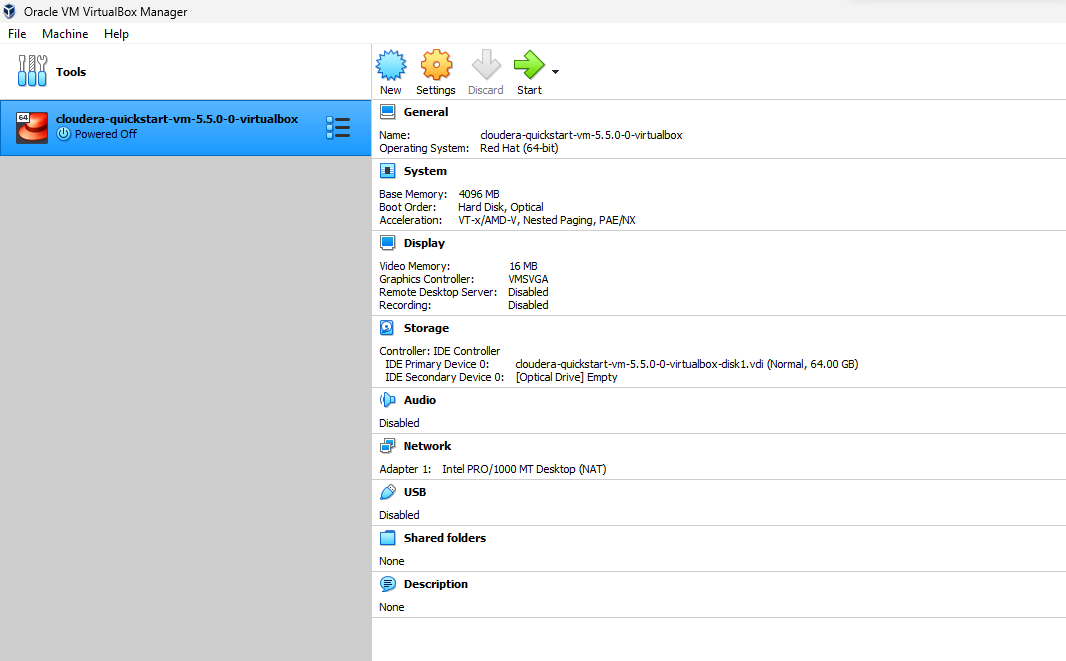
**Figure 1:** Extracting File



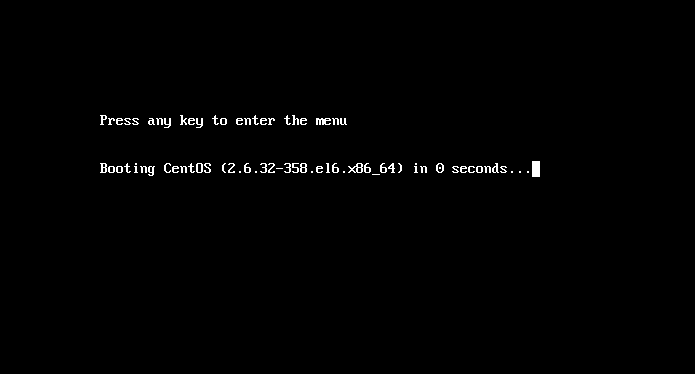
**Figure 2:** Importing OVF file



**Figure 3:** Locating the file

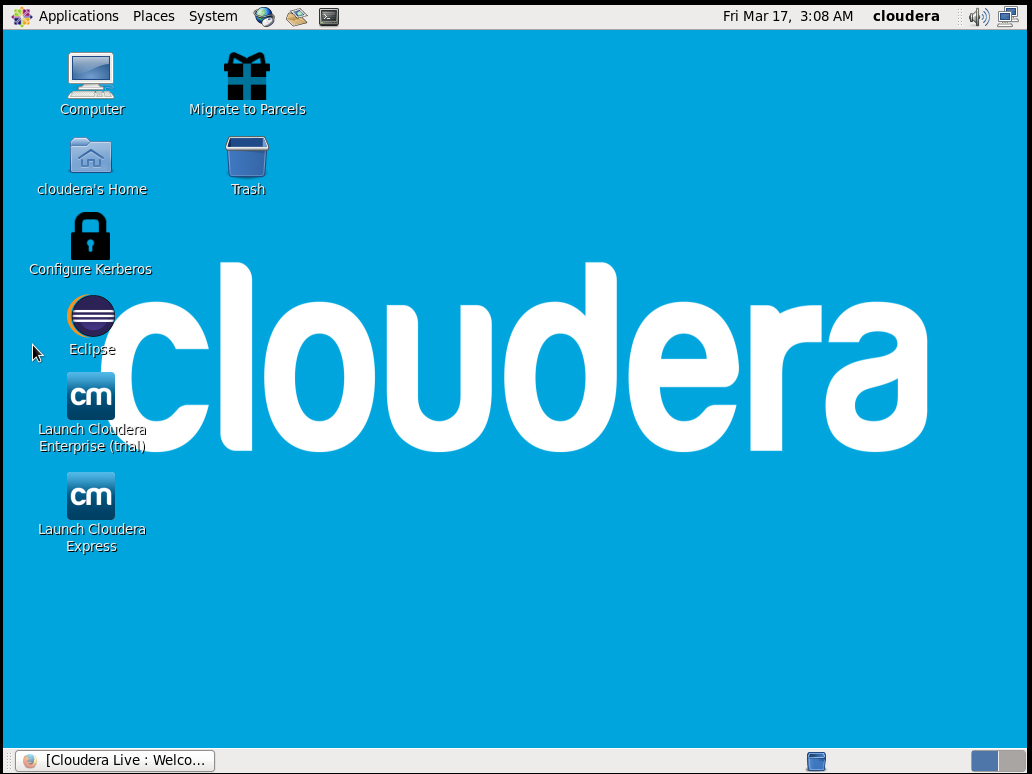


**Figure 4:** Successfully imported



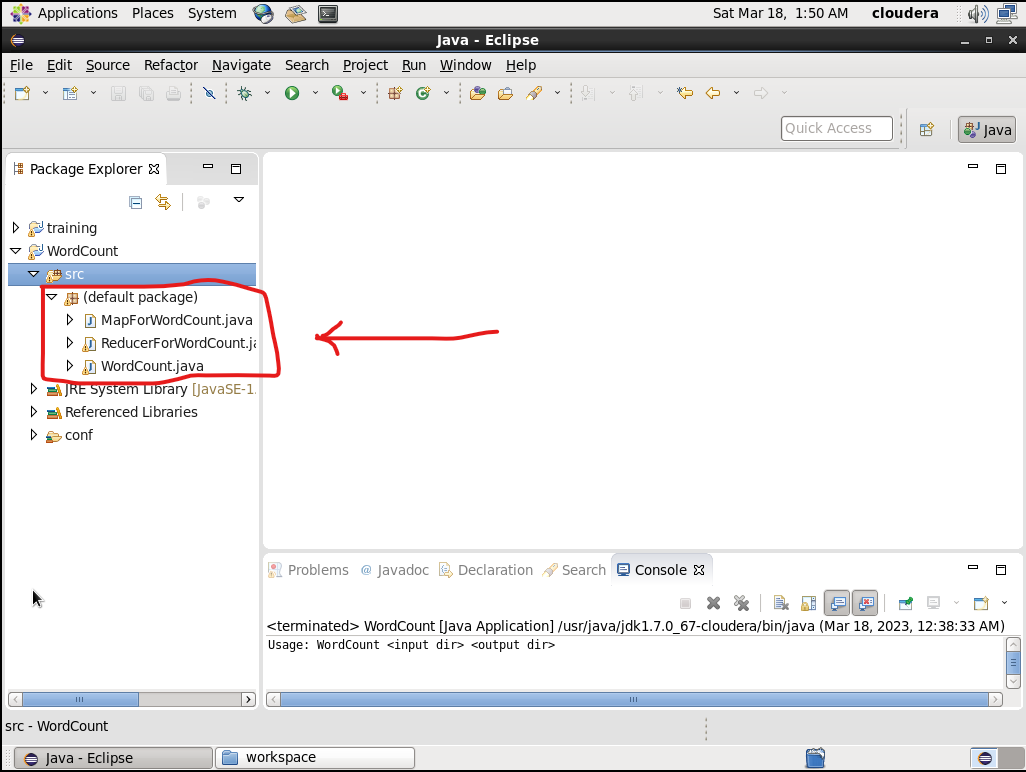


**Figure 5:** Starting the machine



**Figure 6:** Machine successfully started

**Creating Hadoop project in Eclipse:**

****

**Figure 7:** Creating project

**Implanting the given codes in the file**

**WordCount.java**

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.conf.Configuration;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

public class WordCount {

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

if (args.length != 2) {

System.out.printf("Usage: WordCount <input dir> <output dir>\n");

System.exit(-1);

}

Configuration config = new Configuration();

Path input = new Path(args[0]);

Path output = new Path(args[1]);

@SuppressWarnings("deprecation")

Job job = new Job();

job.setJarByClass(WordCount.class);

job.setJarByClass(WordCount.class);

job.setMapperClass(MapForWordCount.class);

job.setReducerClass(ReducerForWordCount.class);

job.setOutputKeyClass(Text.class);

job.setOutputValueClass(IntWritable.class);

FileInputFormat.addInputPath(job, input);

FileOutputFormat.setOutputPath(job,output);

boolean success = job.waitForCompletion(true);

System.exit(success ? 0 : 1);

}

}

**MapForWordCount.java**

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 MapForWordCount extends Mapper<LongWritable, Text, Text, IntWritable> {

@Override

public void map(LongWritable key, Text value, Context context)

throws IOException, InterruptedException {

String line = value.toString();

String[] words = line.split(",");

for (String word: words){

Text outputKey = new Text(word.toUpperCase().trim());

IntWritable outputValue = new IntWritable(1);

context.write(outputKey, outputValue);

}

}

}

**ReducerForWordCount.java**

import java.io.IOException;

import org.apache.hadoop.io.DoubleWritable;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Reducer;

public class ReducerForWordCount extends Reducer<Text, IntWritable, Text, IntWritable> {

@Override

public void reduce(Text key, Iterable<IntWritable> values, Context context)

throws IOException, InterruptedException {

int sum = 0;

for(IntWritable value: values){

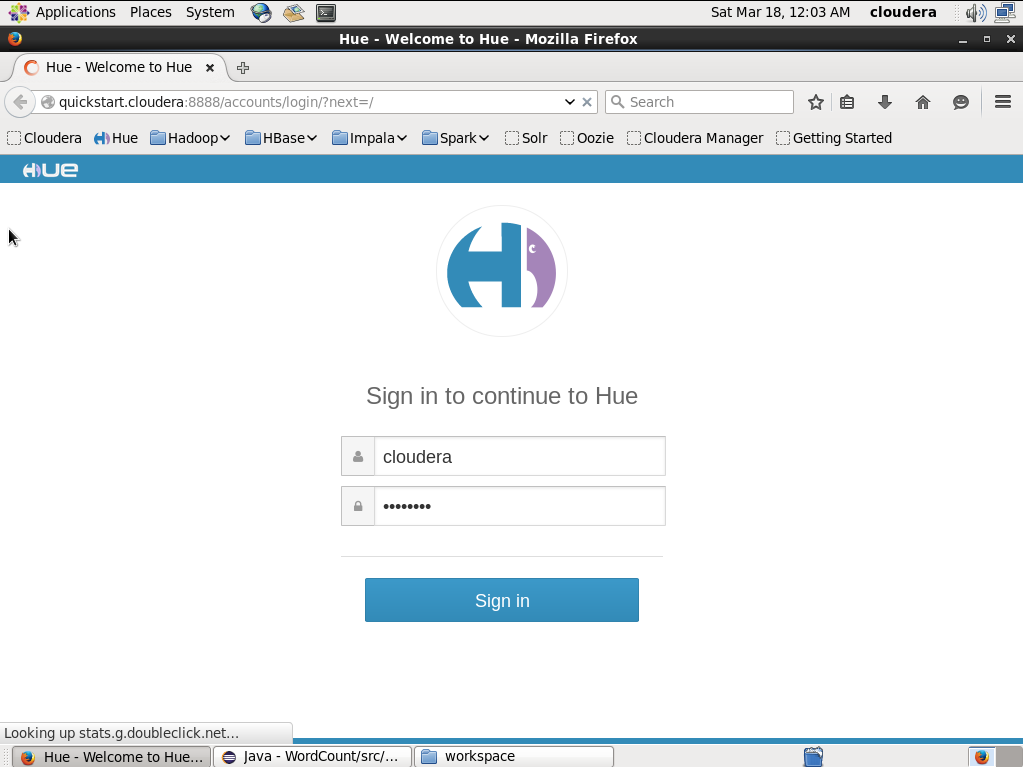
sum += value.get();

}

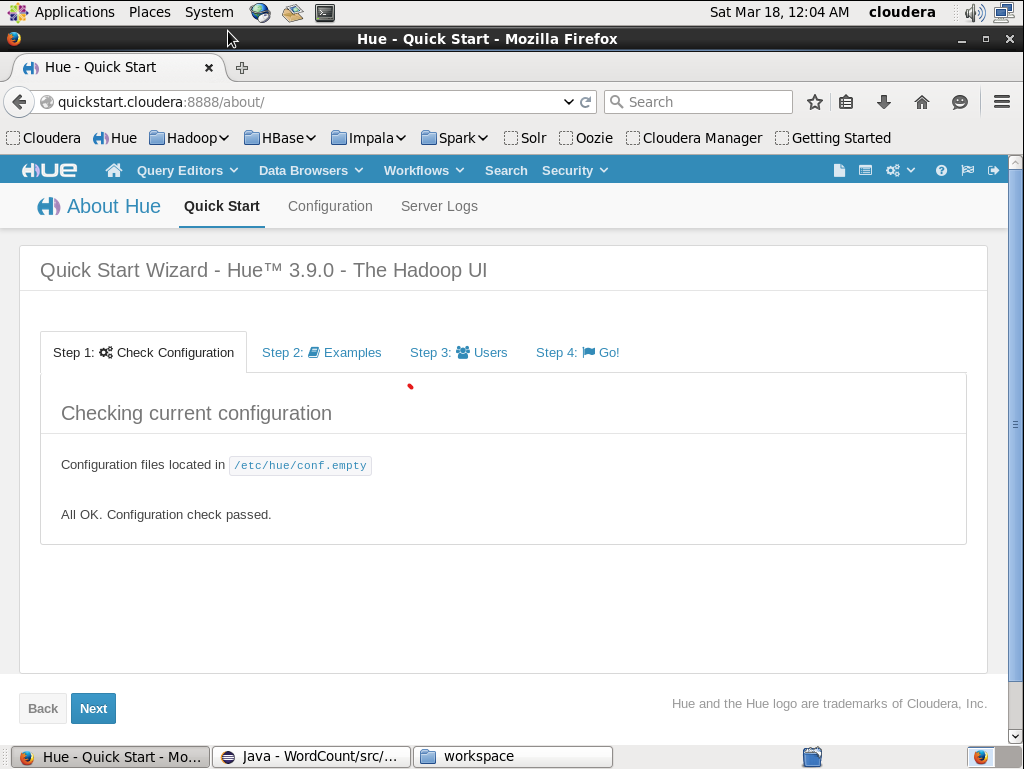
context.write(key, new IntWritable(sum));

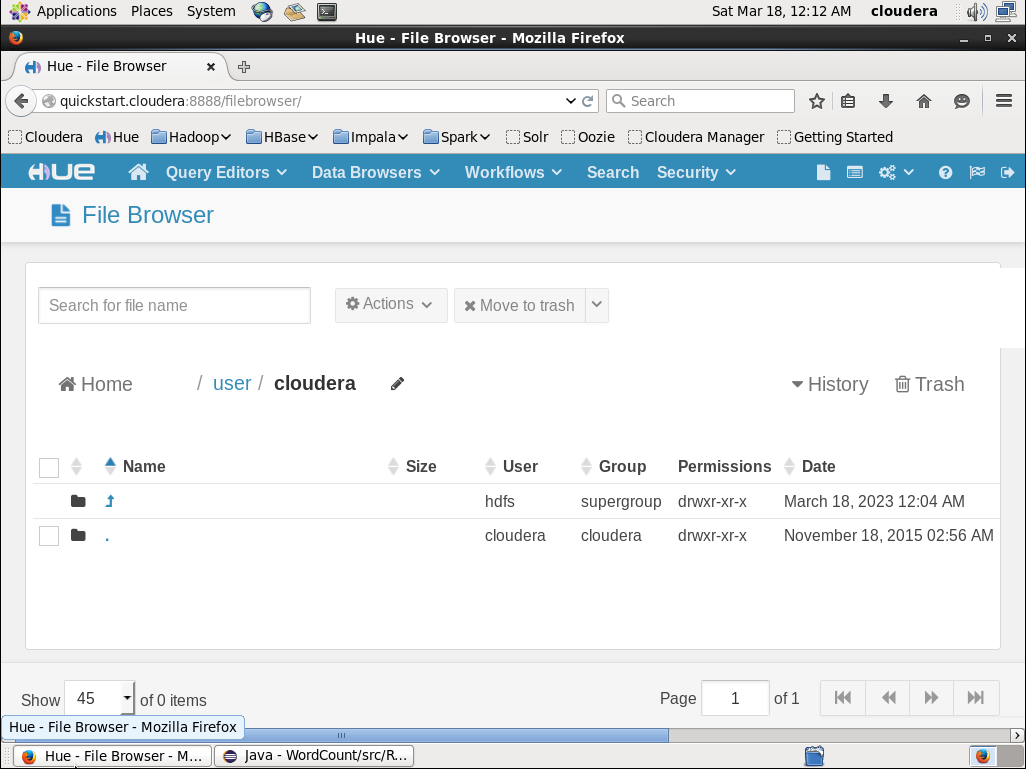
}

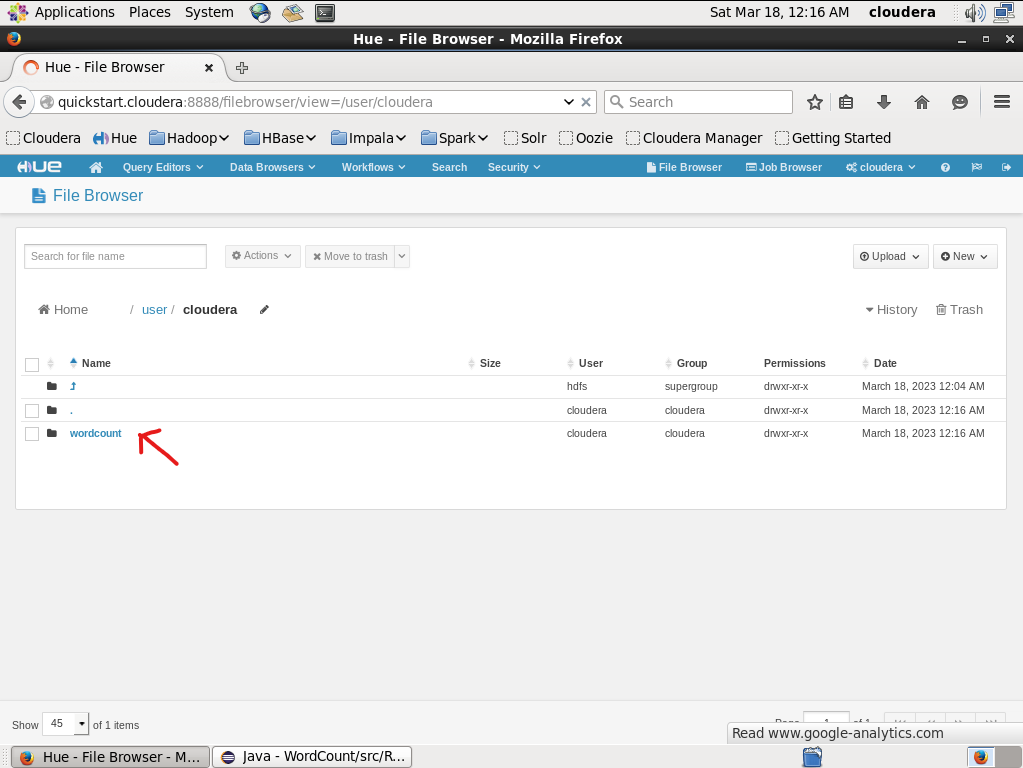
}

****

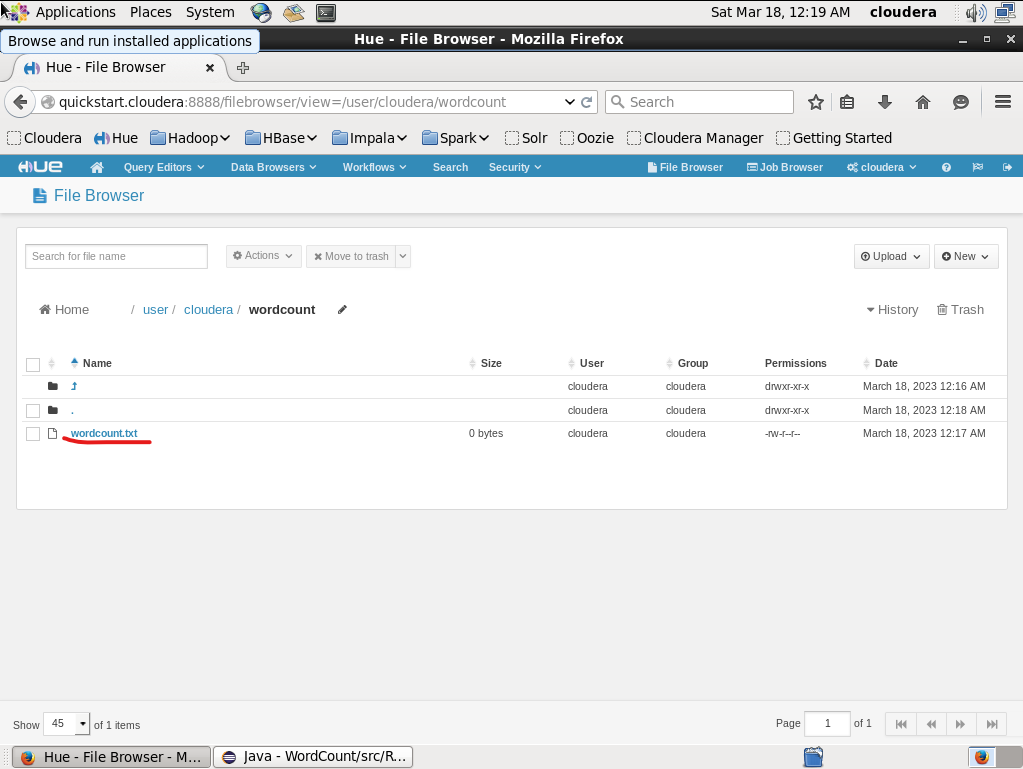
**Figure 8:** login to <http://quickstart.cloudera:8888>

****

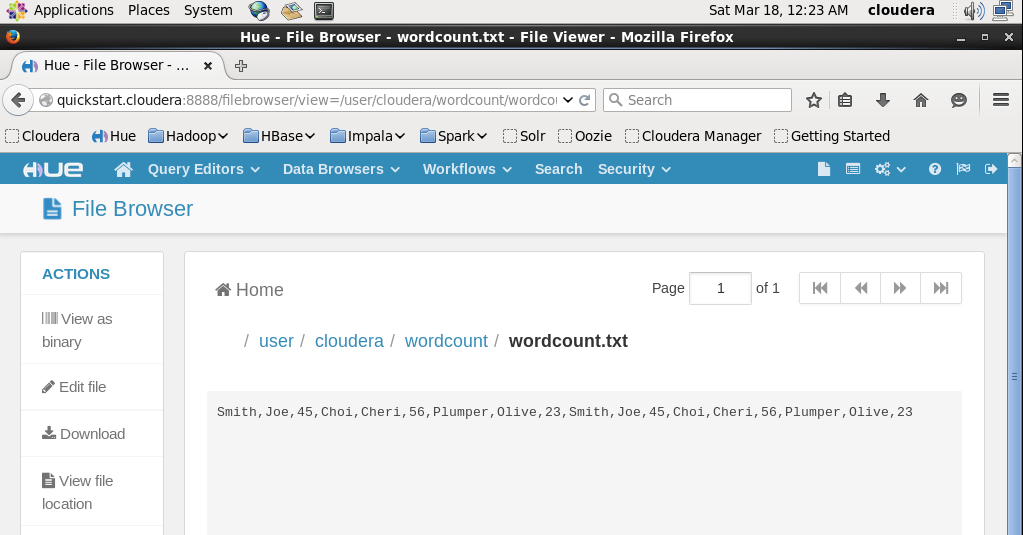
****



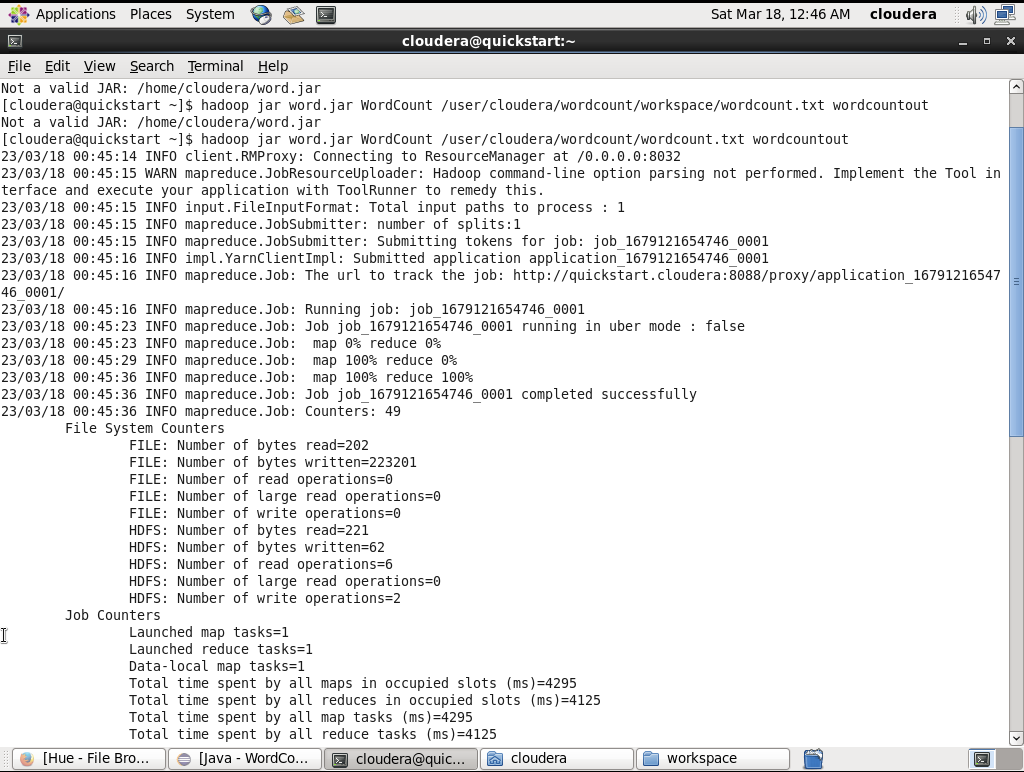
**Figure 9:** Creating a new directory



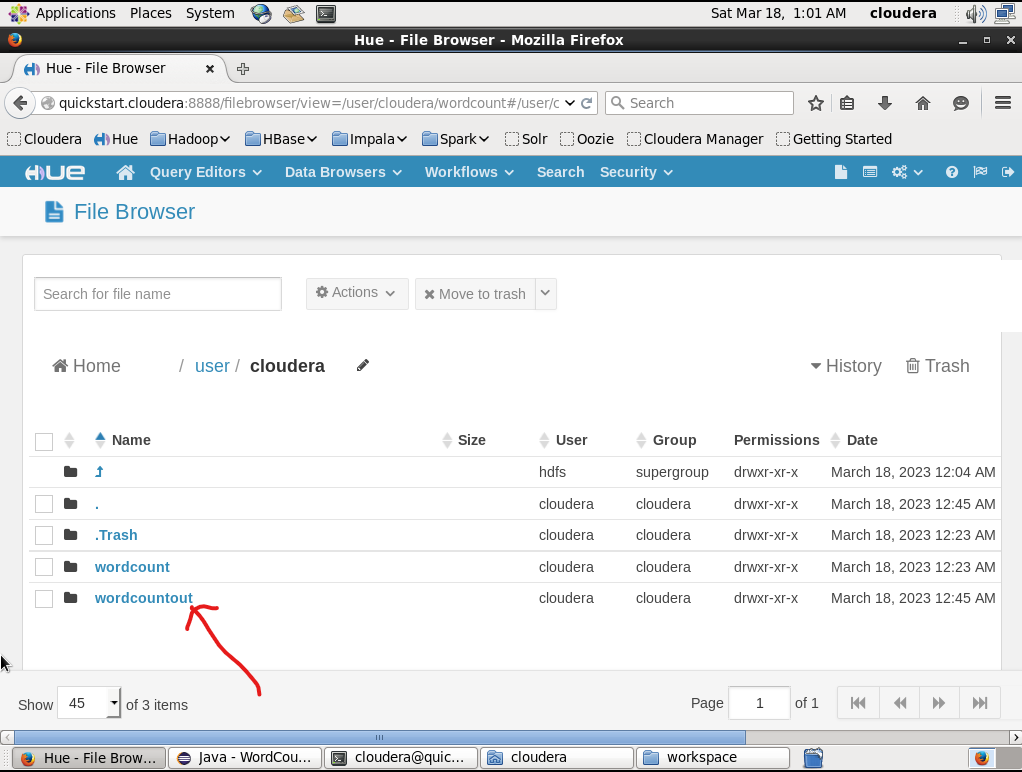
**Figure 10:** Creating new txt file for CSV values



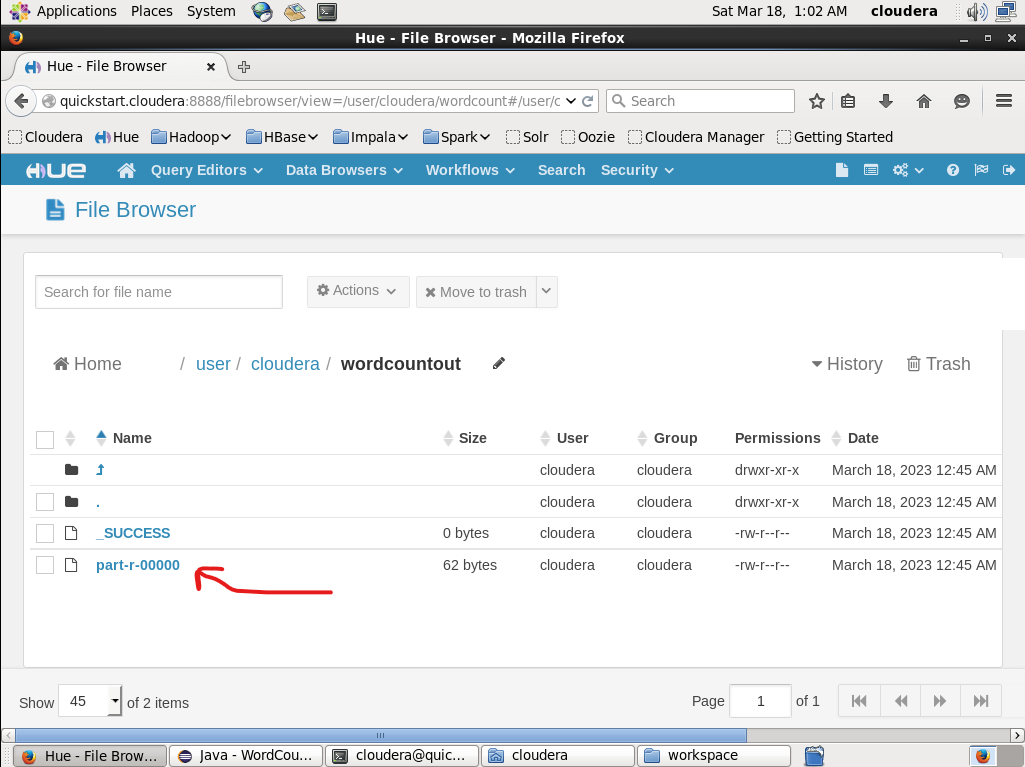
**Figure 11:** Editing the file



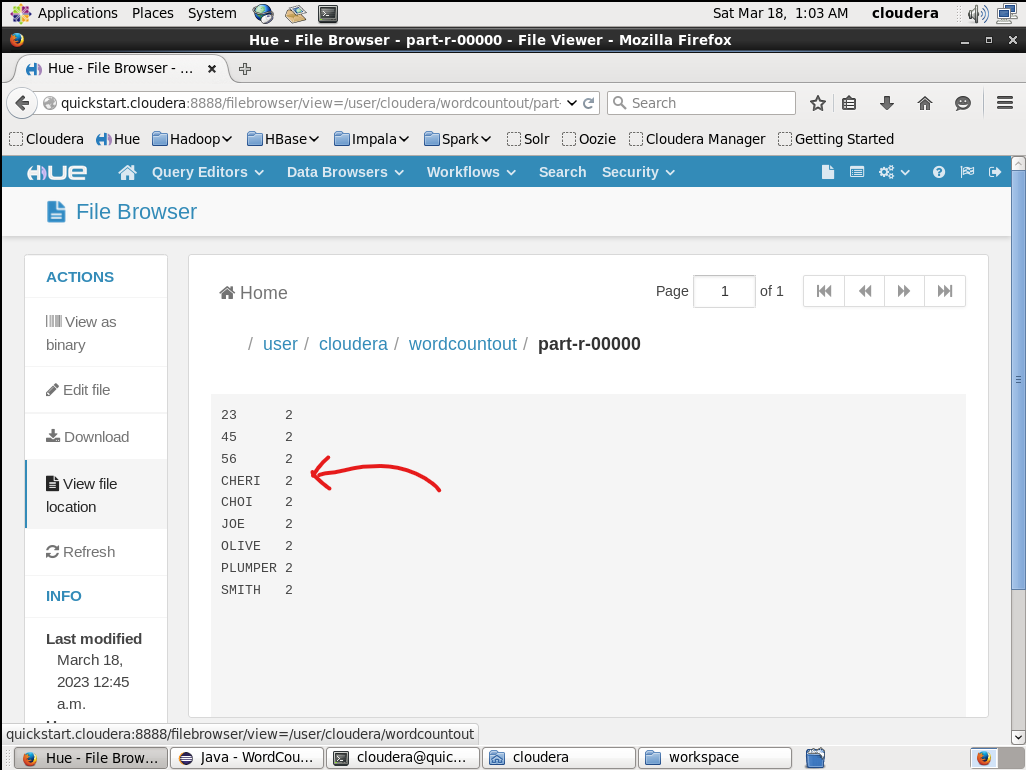
**Figure 12:** In terminal entering the command to run the project



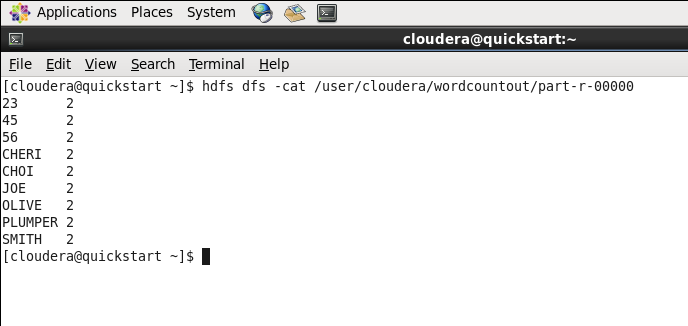
**Figure 13:** Output directory successfully generated



**Figure 14:** Output file



**Figure 15:** The output file



**Figure 16:** Reading the output file using terminal