Ques1:

Ans:

Mapper:

```
import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.io.DoubleWritable;
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 MyMapper extends Mapper<LongWritable,  Text, Text, IntWritable>

{

        @Override
        protected void map(LongWritable key, Text value,
                        org.apache.hadoop.mapreduce.Mapper.Context context)
                        throws IOException, InterruptedException {
String inputstring = value.toString();


for(String x : inputstring.split(" "))

{
        context.write(new Text(x),new IntWritable(1));

}
        }
}
```

Reducer:

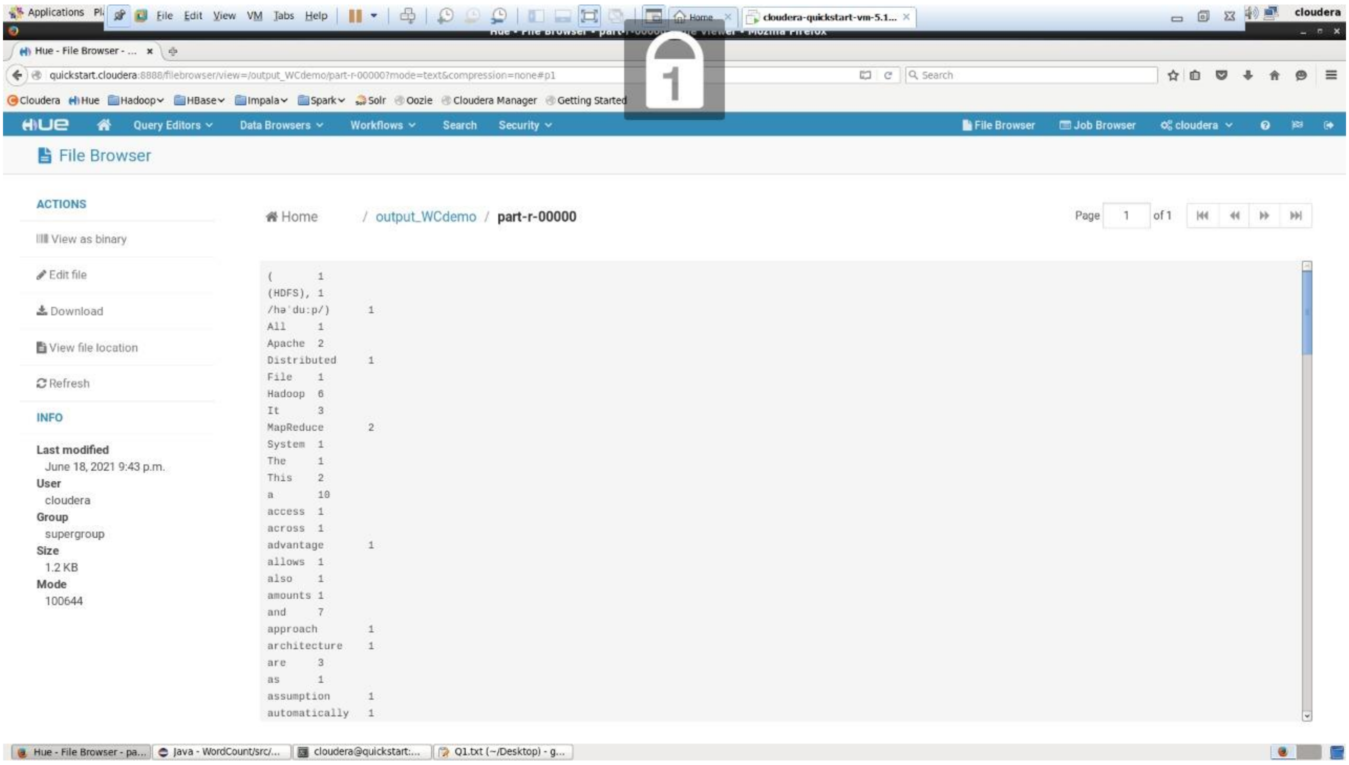
```
import java.io.IOException;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.Reducer.Context;

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

    @Override
    public void reduce(Text key, Iterable<IntWritable> values, Context context)
    throws IOException, InterruptedException {
        int y = 0;
        for(IntWritable x : values)
        {
            y ++;

        }
        context.write(key, new IntWritable(y) );

    }

}
```

Driver:

```
import java.io.IOException;
import java.net.URI;
import java.net.URISyntaxException;


import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileSystem;
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.FileInputFormat;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;


public class MyDriver {
        public static void main(String[] args) throws IOException, ClassNotFoundException,
InterruptedException, URISyntaxException
        {
                Configuration conf = new Configuration();
                j.setJobName("My First Job");
                j.setJarByClass(MyDriver.class );
                j.setMapperClass(MyMapper.class );


                j.setReducerClass(MyReducer.class);




                j.setOutputKeyClass(Text.class);
                j.setOutputValueClass(IntWritable.class);


                FileInputFormat.addInputPath(j, new Path(args[0]));
                FileOutputFormat.setOutputPath(j, new Path(args[1]));
```

```
                    URI uri = new URI(args[1].toString());

                    FileSystem fs =  FileSystem.get(uri, conf);

                    boolean x = fs.delete(new Path(uri),true);



                    int xxx =  j.waitForCompletion(true) ? 0 : 1;

               }

          }
```

OUTPUT



Applications
File Edit View VM Tabs Help
cloudera-quickstart-vm-5.1...
cloudera
Hue - File Browser - ...
quickstart.cloudera:8888/filebrowser/view=/output_WCdemo/part-r-00000?mode=text&compression=none#p1
Search
Cloudera Hue Hadoop HBase Impala Spark Solr Oozie Cloudera Manager Getting Started
HUE Query Editors Data Browsers Workflows Search Security File Browser Job Browser cloudera
File Browser
ACTIONS
View as binary
Edit file
Download
View file location
Refresh
INFO
Home / output_WCdemo / part-r-00000
Page 1 of 1
Hue - File Browser - pa...
Java - WordCount/src/...
cloudera@quickstart:...
Q1.txt (~/Desktop) - g...

Last modified
June 18, 2021 9:43 p.m.
User
cloudera
Group
supergroup
Size
1.2 KB
Mode
100644

```
(       1
(HDFS), 1
/həˈduːp/)      1
All     1
Apache  2
Distributed     1
File    1
Hadoop  6
It      3
MapReduce       2
System  1
The     1
This    2
a       10
access  1
across  1
advantage       1
allows  1
also    1
amounts 1
and     7
approach        1
architecture    1
are     3
as      1
assumption      1
automatically   1
```

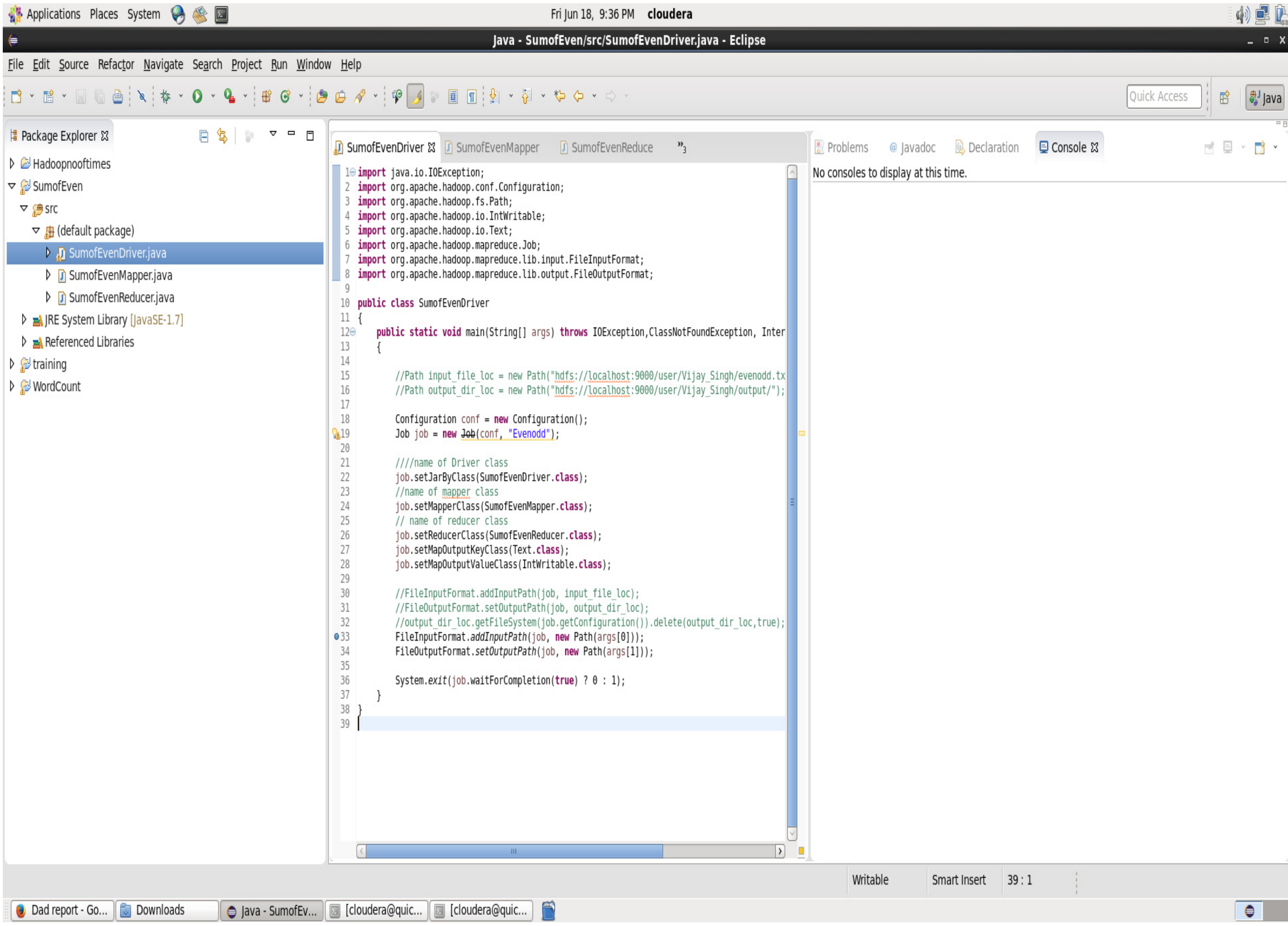
Ques2-

```
//Driver

import java.io.IOException;

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.Job;

import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;


public class SumofEvenDriver

{

	public static void main(String[] args) throws IOException,ClassNotFoundException,
InterruptedException

	{


		//Path input_file_loc = new
Path("hdfs://localhost:9000/user/Vijay_Singh/evenodd.txt/");   //hardcoded input location

		//Path output_dir_loc = new
Path("hdfs://localhost:9000/user/Vijay_Singh/output/");       //hardcoded output location


		Configuration conf = new Configuration();

		Job job = new Job(conf, "Evenodd");


		////name of Driver class

		job.setJarByClass(SumofEvenDriver.class);

		//name of mapper class

		job.setMapperClass(SumofEvenMapper.class);

		// name of reducer class
```

```
                    job.setReducerClass(SumofEvenReducer.class);
                    job.setMapOutputKeyClass(Text.class);
                    job.setMapOutputValueClass(IntWritable.class);

                    //FileInputFormat.addInputPath(job, input_file_loc);
                    //FileOutputFormat.setOutputPath(job, output_dir_loc);
                    //output_dir_loc.getFileSystem(job.getConfiguration()).delete(output_dir_loc,true);
                    FileInputFormat.addInputPath(job, new Path(args[0]));
                    FileOutputFormat.setOutputPath(job, new Path(args[1]));

                    System.exit(job.waitForCompletion(true) ? 0 : 1);
          }
}
//Mapper
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 SumofEvenMapper extends Mapper<LongWritable, Text, Text, IntWritable>
{


          public void map(LongWritable key, Text value, Context context)throws java.io.IOException,
InterruptedException
          {
                    String data[]=value.toString().split(",");    //data =  [85 131 993 392 689....]


                    for(String num:data)
                    {
```

```
                    int number=Integer.parseInt(num);
                    if((number%2)==0)
                    {
                              context.write(new Text("EVEN"), new IntWritable(number));   //
ODD 85  131  993
                    }


          }
     }
//Reducer
import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

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

     // reducer will recieve key value pair as  ODD [ 85 131 993 491 539 985 413.....
                    //        EVEN [ 392 870 240 888 184 494 704 996 408....

     public void reduce(Text key, Iterable<IntWritable> values, Context context)        throws
IOException, InterruptedException
     {
          int sum = 0;
          if(key.equals("EVEN"))
          {
                    for (IntWritable value : values)
                    {
                              sum += value.get();
                    }
          }
```

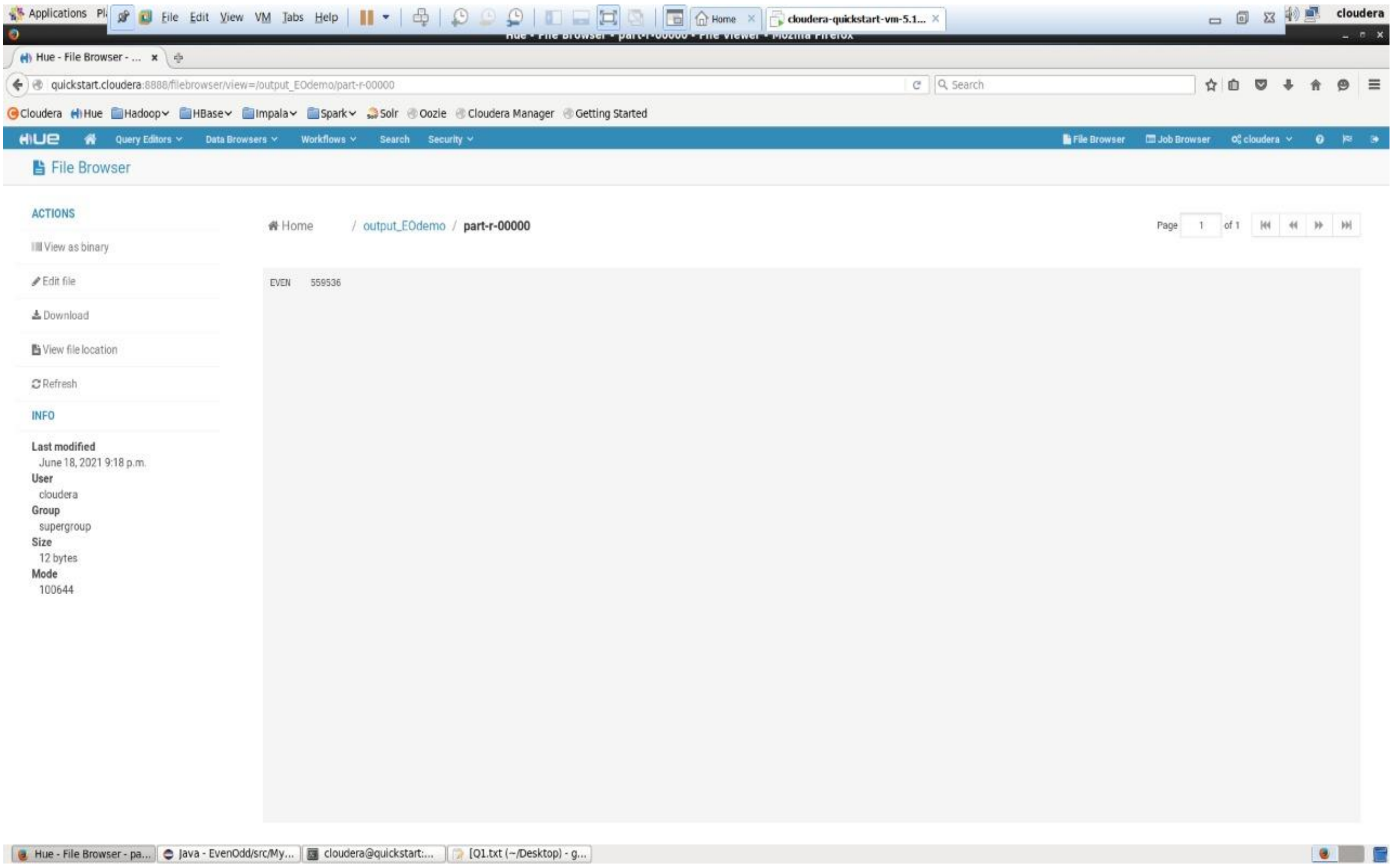
```
                context.write(key,new IntWritable(sum));;
        }
}
```



```
import java.io.IOException;
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.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class SumofEvenDriver
{
    public static void main(String[] args) throws IOException,ClassNotFoundException, Inter
    {

        //Path input_file_loc = new Path("hdfs://localhost:9000/user/Vijay_Singh/evenodd.tx
        //Path output_dir_loc = new Path("hdfs://localhost:9000/user/Vijay_Singh/output/");

        Configuration conf = new Configuration();
        Job job = new ~~Job~~(conf, "Evenodd");

        ////name of Driver class
        job.setJarByClass(SumofEvenDriver.class);
        //name of mapper class
        job.setMapperClass(SumofEvenMapper.class);
        // name of reducer class
        job.setReducerClass(SumofEvenReducer.class);
        job.setMapOutputKeyClass(Text.class);
        job.setMapOutputValueClass(IntWritable.class);

        //FileInputFormat.addInputPath(job, input_file_loc);
        //FileOutputFormat.setOutputPath(job, output_dir_loc);
        //output_dir_loc.getFileSystem(job.getConfiguration()).delete(output_dir_loc,true);
        FileInputFormat.addInputPath(job, new Path(args[0]));
        FileOutputFormat.setOutputPath(job, new Path(args[1]));

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

No consoles to display at this time.

Output



```
EVEN    559536
```

Last modified
June 18, 2021 9:18 p.m.
User
cloudera
Group
supergroup
Size
12 bytes
Mode
100644