Code:

1> LogFileMapper.java

(Use for mapping the IP addresses frominput csv file)

package LogFileCountry;

importjava.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.\*;

public class LogFileMapper extends MapReduceBase implements Mapper<LongWritable, Text, Text,

IntWritable> {

private final static IntWritable one = new IntWritable(1);

public void map(LongWritablekey, Text value, OutputCollector<Text, IntWritable> output, Reporter

reporter) throws IOException {

String valueString = value.toString();

String[] SingleIpData = valueString.split("-");

output.collect(new Text(SingleIpData[0]), one);

}

} 2>LogFileReduce.java

(Use for reducing data received from mapper process to final output)

package LogFileCountry;

import java.io.IOException;

import java.util.\*;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.\*;

public class LogFileReducer extends MapReduceBase implements Reducer<Text, IntWritable, Text,

IntWritable> {

public void reduce(Text t\_key, Iterator<IntWritable> values, OutputCollector<Text,IntWritable>

output, Reporter reporter) throws IOException {

Text key = t\_key;

int frequencyForIp = 0;

while (values.hasNext()) {

// replace type of value with the actual type of our value

IntWritable value = (IntWritable) values.next();

frequencyForIp += value.get();

}

output.collect(key, new IntWritable(frequencyForIp));

}

}

3>LogFileCountryDriver.java

package LogFileCountry;

(The driver code to run map-reduceon hdfs)

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.\*;

importorg.apache.hadoop.mapred.\*;

public class LogFileCountryDriver {

public static void main(String[] args) {

JobClient my\_client = new JobClient();

// Create a configuration object for the job

JobConf job\_conf = new JobConf(LogFileCountryDriver.class);

// Set a name of the Job

job\_conf.setJobName("LogFileIP");

// Specify data type of output key and value

job\_conf.setOutputKeyClass(Text.class);

job\_conf.setOutputValueClass(IntWritable.class);

// Specify names of Mapper and ReducerClass

job\_conf.setMapperClass(LogFileCountry.LogFileMapper.class);

job\_conf.setReducerClass(LogFileCountry.LogFileReducer.class);

// Specify formats of the data type of Input and output

job\_conf.setInputFormat(TextInputFormat.class);

job\_conf.setOutputFormat(TextOutputFormat.class);

// Set input and output directories using command line arguments,

//arg[0] = name of input directory on HDFS, and arg[1] = name of output directory to be

created to store the output file.

FileInputFormat.setInputPaths(job\_conf, new Path(args[0]));

FileOutputFormat.setOutputPath(job\_conf, new Path(args[1]));

my\_client.setConf(job\_conf);

try { // Runthejob

JobClient.runJob(job\_conf);

}catch(Exceptione){

e.printStackTrace();

}

}

}

4>log\_file.txt(Inputfilesample)

0.223.157.186--[15/Jul/2009:20:50:32-0700]"GET/assets/js/the-associates.jsHTTP/1.1"304

10.223.157.186--[15/Jul/2009:20:50:33-0700]"GET/assets/img/home-logo.png HTTP/1.1" 304

10.223.157.186--[15/Jul/2009:20:50:33-0700] "GET /assets/img/dummy/primary-news-2.jpgHTTP/1.1"304

10.223.157.186--[15/Jul/2009:20:50:33-0700]"GET /assets/img/dummy/primary-news-1.jpgHTTP1.1"304

10.223.157.186--[15/Jul/2009:20:50:33-0700]"GET/assets/img/home-media-block-placeholder.jpgHTTP/1.1"304

10.223.157.186--[15/Jul/2009:20:50:33-0700] "GET /assets/img/dummy/secondary-news-4.jpgHTTP/1.1"304

10.223.157.186--[15/Jul/2009:20:50:33-0700]"GET/assets/img/loading.gifHTTP/1.1"304

10.223.157.186--[15/Jul/2009:20:50:33-0700] "GET /assets/img/search-button.gifHTTP/1.1"304

5> Output(part-00000.txtOnHadoop)(sample)

10.1.1.236 7

10.1.181.142 14

10.1.232.31 5

10.10.55.142 14

10.102.101.66 1

10.103.184.104 1

10.103.190.81 53

10.103.63.29 1

10.104.73.51 1

10.105.160.183 1

10.108.91.151 15

10.109.21.76 1

10.11.131.40 1

10.111.71.20 8

10.112.227.184 6

10.114.74.30 1

10.115.118.78 1

10.117.224.230 1