**CustomPartition:**

**MapperCode:**

import java.io.IOException;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.MapReduceBase;

import org.apache.hadoop.mapred.OutputCollector;

import org.apache.hadoop.mapred.Reporter;

import org.apache.hadoop.mapred.Mapper;

public class PartitonerMapper extends MapReduceBase implements

Mapper<LongWritable, Text, Text, Text> {

/\*

\* Input line looks like: 96.7.4.14 - - [24/Apr/2011:04:20:11 -0400]

\* "GET /cat.jpg HTTP/1.1" 200 12433

\*/

@Override

public void map(LongWritable key, Text value,

OutputCollector<Text, Text> output, Reporter reporter)

throws IOException {

String[] fields = value.toString().split(" ");

if (fields.length > 3) {

String ip = fields[0];

String[] dtFields = fields[3].split("/");

if (dtFields.length > 1) {

String theMonth = dtFields[1];

output.collect(new Text(ip), new Text(theMonth));

}

}

}

}

**ReducerCode:**

import java.io.IOException;

import java.util.Iterator;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.OutputCollector;

import org.apache.hadoop.mapred.MapReduceBase;

import org.apache.hadoop.mapred.Reducer;

import org.apache.hadoop.mapred.Reporter;

public class PartitionerReducer extends MapReduceBase implements

Reducer<Text, Text, Text, IntWritable> {

@Override

public void reduce(Text key, Iterator<Text> values,

OutputCollector<Text, IntWritable> output, Reporter reporter)

throws IOException {

int wordCount = 0;

while (values.hasNext()) {

Text value = values.next();

wordCount ++;

}

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

}

}

package com.pc;

import java.util.HashMap;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.JobConf;

import org.apache.hadoop.mapred.Partitioner;

public class PartitionerMid<K2, V2> implements Partitioner<Text, Text> {

HashMap months = new HashMap();

public void configure(JobConf job) {

months.put("Jan", 0);

months.put("Feb", 1);

months.put("Mar", 2);

months.put("Apr", 3);

months.put("May", 4);

months.put("Jun", 5);

months.put("Jul", 6);

months.put("Aug", 7);

months.put("Sep", 8);

months.put("Oct", 9);

months.put("Nov", 10);

months.put("Dec", 11);

}

public int getPartition(Text key, Text value,

int numReduceTasks) {

return (Integer) (months.get(value.toString()));

***DriverClass:***

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.FileOutputFormat;

import org.apache.hadoop.mapred.JobClient;

import org.apache.hadoop.mapred.JobConf;

import org.apache.hadoop.conf.Configured;

import org.apache.hadoop.util.Tool;

import org.apache.hadoop.util.ToolRunner;

public class PartitionerDriver extends Configured implements Tool {

@Override

public int run(String[] args) throws Exception {

JobConf conf = new JobConf(getConf(), PartitionerDriver.class);

conf.setJobName(this.getClass().getName());

FileInputFormat.setInputPaths(conf, new Path(args[0]));

FileOutputFormat.setOutputPath(conf, new Path(args[1]));

conf.setMapperClass(PartitonerMapper.class);

conf.setReducerClass(PartitionerReducer.class);

conf.setMapOutputKeyClass(Text.class);

conf.setMapOutputValueClass(Text.class);

conf.setOutputKeyClass(Text.class);

conf.setOutputValueClass(IntWritable.class);

conf.setNumReduceTasks(12);

conf.setPartitionerClass(PartitionerMid.class);

JobClient.runJob(conf);

return 0;

}

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

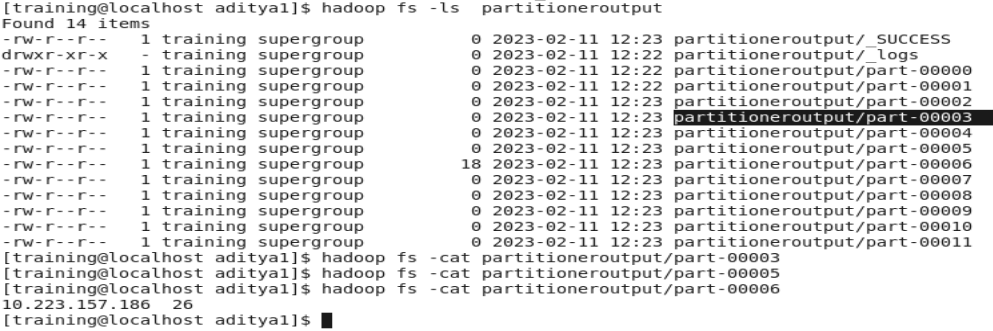
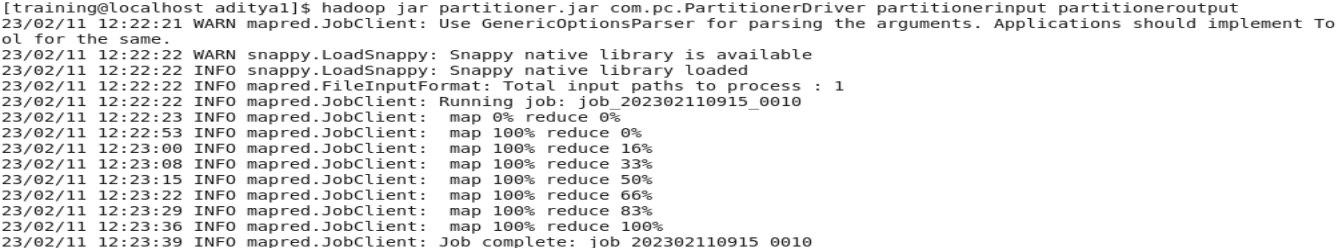
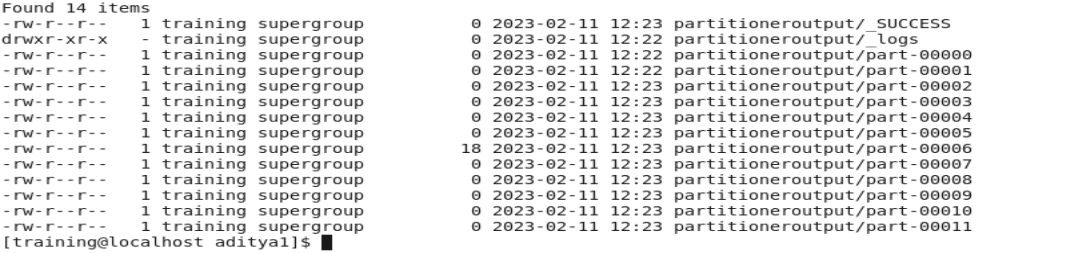
int exitCode = ToolRunner.run(new PartitionerDriver(), args);

System.exit(exitCode);

}

}

**Output:**

****