7) Create a bar graph to depict the number of applications for each year

package h1b;

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.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapred.TestMiniMRClientCluster.MyReducer;

import org.apache.hadoop.mapreduce.Job;

import org.apache.hadoop.mapreduce.Mapper;

import org.apache.hadoop.mapreduce.Reducer;

import org.apache.hadoop.mapreduce.Mapper.Context;

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

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

public class Question7 {

public static class MyMapper extends Mapper<LongWritable,Text,Text,Text>{

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

{

String[] each = value.toString().split("\t");

String app =each[1];

String year =each[7];

context.write(new Text(year),new Text(app));

}

}

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

public void reduce(Text key, Iterable<Text> values,Context context) throws IOException, InterruptedException{

int sum = 0;

for (Text val : values) {

sum ++;

}

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

}

}

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

{

Configuration con=new Configuration();

Job job=Job.getInstance(con,"Q7");

job.setJarByClass(Question7.class);

job.setMapperClass(MyMapper.class);

job.setMapOutputKeyClass(Text.class);

job.setMapOutputValueClass(Text.class);

job.setReducerClass(MyReducer.class);

job.setOutputKeyClass(Text.class);

job.setOutputValueClass(IntWritable.class);

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

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

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

}

}