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 MaxTemperatureMapper

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

private static final int MISSING = 9999;

@Override

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

throws IOException, InterruptedException {

String line = value.toString();

String year = line.substring(15, 19);

int airTemperature;

try{

if (line.charAt(87) == '+') { // parseInt doesn't like leading plus signs

airTemperature = Integer.parseInt(line.substring(88, 92));

} else {

airTemperature = Integer.parseInt(line.substring(87, 92));

}

}catch(NumberFormatException e){

airTemperature = 000;

}

String quality = line.substring(92, 93);

if (airTemperature != MISSING && quality.matches("[01459]")) {

context.write(new Text(year), new IntWritable(airTemperature));

}

}

}