Assignment 13

Locate dataset for working on weather data which reads the text input files and finds the average for temperature, dew point and wind speed.

```
// importing Libraries
import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.conf.Configuration;

public class MyMaxMin
{
	public static class MaxTemperatureMapper extends Mapper<LongWritable, Text, Text, Text>
	{
		public static final int MISSING = 9999;
		@Override
		public void map(LongWritable arg0, Text Value, Context context)throws IOException, InterruptedException
		{
			String line = Value.toString();
			if (!(line.length() == 0))
			{
				String date = line.substring(6, 14);
				float temp_Max = Float.parseFloat(line.substring(39, 45).trim());
				float temp_Min = Float.parseFloat(line.substring(47, 53).trim());
				if (temp_Max > 30.0)
				{
					context.write(new Text("The Day is Hot Day :" + date),
					new Text(String.valueOf(temp_Max)));
				}
				if (temp_Min < 15)
				{
					context.write(new Text("The Day is Cold Day :" + date),
					new Text(String.valueOf(temp_Min)));
				}
			}
		}
	}

	public static class MaxTemperatureReducer extends Reducer<Text, Text, Text, Text>
	{
		public void reduce(Text Key, Iterator<Text> Values, Context context)throws IOException, InterruptedException
		{
			String temperature = Values.next().toString();
			context.write(Key, new Text(temperature));
		}

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

		Configuration conf = new Configuration();
		Job job = new Job(conf, "weather example");
		job.setJarByClass(MyMaxMin.class);
		job.setMapOutputKeyClass(Text.class);
		job.setMapOutputValueClass(Text.class);
		job.setMapperClass(MaxTemperatureMapper.class);
		job.setReducerClass(MaxTemperatureReducer.class);
		job.setInputFormatClass(TextInputFormat.class);
		job.setOutputFormatClass(TextOutputFormat.class);
		Path OutputPath = new Path(args[1]);
		FileInputFormat.addInputPath(job, new Path(args[0]));
		FileOutputFormat.setOutputPath(job, new Path(args[1]));
		OutputPath.getFileSystem(conf).delete(OutputPath);
		System.exit(job.waitForCompletion(true) ? 0 : 1);
	}
}
```

```
The Day is Cold Day :20200101   -21.8
The Day is Cold Day :20200102   -23.4
The Day is Cold Day :20200103   -25.4
The Day is Cold Day :20200104   -26.8
The Day is Cold Day :20200105   -28.8
The Day is Cold Day :20200106   -30.0
The Day is Cold Day :20200107   -31.4
The Day is Cold Day :20200108   -33.6
The Day is Cold Day :20200109   -26.6
The Day is Cold Day :20200110   -24.3
```