

```

package it.polito.bigdata.hadoop.exercise20;

import java.io.IOException;

import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.lib.output.MultipleOutputs;

/**
 * Mapper of a map-only job
 * */
class MapperBigData extends Mapper<
    LongWritable, // Input key type
    Text,          // Input value type
    Text,          // Output key type
    NullWritable> { // Output value type

    // Define a MultiOutputs object
    private MultipleOutputs<Text, NullWritable> mos = null;

    protected void setup(Context context)
    {
        // Create a new MultiOutputs using the context object
        mos = new MultipleOutputs<Text, NullWritable>(context);
    }

    protected void map(
        LongWritable key,    // Input key type
        Text value,          // Input value type
        Context context) throws IOException, InterruptedException {

        String record=value.toString();
        // Split each record by using the field separator
        // fields[0]= sensor id
        // fields[1]= date
        // fields[2]= hour:minute
        // fields[3]= temperature
        String[] fields = record.split(",");

        float temperature=Float.parseFloat(fields[3]);

        if (temperature>30.0)
            mos.write("hightemp", value, NullWritable.get());
        else
            mos.write("normaltemp", value, NullWritable.get());
    }

    protected void cleanup(Context context) throws IOException,
    InterruptedException
    {
        // Close the MultiOutputs
        // If you do not close the MultiOutputs object the content of the
        output
        // files will not be correct
        mos.close();
    }
}

```

}

}