

Log File Processing

Roll Number: 33310

Name: Rishikesh Chaudhari

```
package MockExam;

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.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.util.GenericOptionsParser;

import java.io.IOException;

public class MacFileClass {
    public static void main(String[] args) throws Exception {
        Configuration conf = new Configuration();
        String[] files = new GenericOptionsParser(conf,
args).getRemainingArgs();
        Path input = new Path(files[0]);
        Path output = new Path(files[1]);

        Job job = Job.getInstance(conf, "MAC Address Count");
        job.setJarByClass(MacFileClass.class);
        job.setMapperClass(MapForLogProcessing.class);
        job.setReducerClass(ReduceForLogProcessing.class);
        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);

        FileInputFormat.addInputPath(job, input);
        FileOutputFormat.setOutputPath(job, output);

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

```

        public static class MapForLogProcessing extends
Mapper<LongWritable, Text, Text, IntWritable> {
    public void map(LongWritable key, Text value, Context context)
throws IOException, InterruptedException {
        String line = value.toString();
        String[] fields = line.split(",");

        if (fields.length > 1) { // Ensure there are enough
fields
            String macAddress = fields[1]; // MAC address from
the 2nd column
            context.write(new Text(macAddress), new
IntWritable(1));
        } else {
            System.out.println("Skipping invalid line: " + line);
        }
    }
}

        public static class ReduceForLogProcessing extends Reducer<Text,
IntWritable, Text, IntWritable> {
    public void reduce(Text key, Iterable<IntWritable> values,
Context context) throws IOException, InterruptedException {
        int macCount = 0;

        for (IntWritable value : values) {
            macCount += value.get();
        }

        context.write(key, new IntWritable(macCount));
    }
}
}

```

Output

30-75-12-0D-CE-4C	436
30-F7-72-05-D4-67	190
30-F7-72-1E-CF-6F	572
30-F9-ED-A7-6A-9E	216
34-78-D7-86-52-25	1887
34-78-D7-C7-13-D8	3942
34-BB-26-18-CB-AD	212
34-DE-1A-A5-7B-1B	316
34-F6-4B-F3-9F-F8	938
38-A4-ED-5A-1F-E5	375
38-A4-ED-5E-8A-3D	97
38-A4-ED-5F-E7-C7	78
38-A4-ED-C7-3D-95	1237
38-A4-ED-E8-8B-D9	127
38-B1-DB-C6-DA-4D	772
38-D5-47-8A-A7-49	2527
3C-A0-67-DE-C7-BB	575
3C-BB-FD-A3-37-72	820
40-40-A7-4A-3A-DC	24
40-4D-7F-B6-C7-1C	2117
40-B8-9A-47-A0-37	1876
40-F2-E9-36-9A-58	97
44-03-2C-7C-A0-D3	904
44-04-44-E3-EF-7B	191
44-04-44-ED-1B-9E	614
44-04-44-F0-61-3E	380
48-88-CA-6C-77-62	1057
48-88-CA-6F-82-5B	159
48-88-CA-BD-06-3D	660
48-EE-0C-45-CA-27	946
4C-32-75-AA-7B-E0	98
4C-49-E3-62-EA-CB	883
4C-49-E3-69-8A-27	999
4C-49-E3-70-2A-47	1722
4C-49-E3-70-5D-29	417
4C-49-E3-70-C1-33	456
4C-49-E3-76-C4-91	109

4C-49-E3-7A-42-8B	77
4C-49-E3-7B-6B-D7	115
50-04-B8-43-EB-E7	28
50-82-D5-84-43-52	29
50-82-D5-91-FE-46	303
50-8F-4C-9A-F1-F9	230
50-8F-4C-9F-E0-63	3843
50-8F-4C-AE-94-39	572
50-C7-BF-7C-2E-C9	1413
54-8C-A0-91-BE-F1	293
54-DC-1D-7C-E1-F4	169
58-00-E3-D9-02-11	675
58-00-E3-E9-A1-5D	19
58-48-22-5C-F9-DA	79
58-6D-8F-7D-DF-1B	725
58-FB-84-7B-5B-6F	978
5C-99-60-20-A5-87	2744
5C-99-60-8B-FD-C1	429
5C-99-60-FB-63-58	18
5C-AC-4C-27-8A-97	4075
5C-F7-C3-D7-C5-A8	1567
NatedMachine	16