**NLine Text Input Format - Mapper Class**

import java.io.IOException;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Mapper;

public class MapperNLine extends Mapper<LongWritable, Text, LongWritable, Text> {

@Override

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

throws IOException, InterruptedException {

context.write(key, value);

}

}

----------------

**Driver Class**

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.conf.Configured;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.mapreduce.Job;

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

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

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

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

import org.apache.hadoop.util.Tool;

import org.apache.hadoop.util.ToolRunner;

public class Driver extends Configured implements Tool {

@Override

public int run(String[] args) throws Exception {

if (args.length != 2) {

System.out

.printf("Two parameters are required for DriverNLineInputFormat- <input dir> <output dir>\n");

return -1;

}

Job job = new Job(getConf());

job.setJobName("NLineInputFormat example");

job.setJarByClass(Driver.class);

job.setInputFormatClass(NLineInputFormat.class);

NLineInputFormat.addInputPath(job, new Path(args[0]));

LazyOutputFormat.setOutputFormatClass(job, TextOutputFormat.class);

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

job.setMapperClass(MapperNLine.class);

job.setNumReduceTasks(0);

boolean success = job.waitForCompletion(true);

return success ? 0 : 1;

}

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

int exitCode = ToolRunner.run(new Configuration(), new Driver(), args);

System.exit(exitCode);

}

}