**Assignment 3.2**

**Job program:**

package invalided;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.conf.\*;

import org.apache.hadoop.mapreduce.Job;

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

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

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

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

public class FilterInvalidRec{

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

Configuration conf= new Configuration();

Job job = new Job(conf, " Demo of Assignment3.1");

job.setJarByClass(FilterInvalidRec.class);

Job.setMapOutputKeyClass(Text.class);

job.setMapoutputValueClass(Text.class);

Job.setOutputKeyClass(Text.class);

job.setOutputValueClass(Text.class);

Job.setMapperClass(FilterInvalidMapper.class);

job.setNumReduceTasks(0);

Job.setInputFormatClass(textInputFormat.class);

job.setOutputFormatClass (textOutputFormat.class);

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

FileoutputFormat.addoutputPath(job, new Path (args[1]});

job.waitForCompletion(true);

}

}

**Mapper Program:**

package invalidrec;

import java.io.IOException;

import org.apache.hadoop.io.NullWritable;

import org.apache.hadoop.io.LongWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.\*;

public class FilterInvalidMapper extends Mapper <LongWritable,Text,NullWritable,Text> {

NullWritable key;

Text line;

public void setup(Context context) {

//key = new NullWritable();

line = new Text ();

}

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

throws IOException, InterruptedException {

String lineArray = value.toString();

line = new Text(lineArray);

if(!lineArray.contains("NA")) {

context.write(key,line);

}

}

}