**Session 3: EXPLORING MAPREDUCE**

Assignment 3.2

Student Name: Abarajithan SA

Course: Big Data Hadoop & Spark Training

Start Date:  2017-09-09

End Date:  2017-11-26

**Assignment 3.2** –We have a dataset of sales of different TV sets across different locations. Records look like:

**Samsung|Optima|14|Madhya Pradesh|132401|14200**

The fields are arranged like:

Company Name|Product Name|Size in inches|State|Pin Code|Price

There are some invalid records which contain 'NA' in either Company Name or Product Name.

1. Write a Map Reduce program to filter out the invalid records. Map only job will fit for this

Context.

Contents

[Introduction: 2](#_Toc495596351)

[Associated Data Files 2](#_Toc495596352)

[Problem Statement 2](#_Toc495596353)

[Actual Data Set: 2 records have ‘NA’ value 2](#_Toc495596354)

[Map only Java Program 3](#_Toc495596355)

[Driver Code 3](#_Toc495596356)

[Mapper Code 4](#_Toc495596357)

[Expected Output 4](#_Toc495596358)

[Input and Output Comparison 5](#_Toc495596359)

[Input 5](#_Toc495596360)

[Output 5](#_Toc495596361)

# Introduction:

In this assignment, I’m going to write a **Map Only** java program to filter out the invalid records provided in the dataset file.

We are using **Map only** program since we are going to filter a few records.

# Associated Data Files

<https://drive.google.com/file/d/0Bxr27gVaXO5sVjQ5QW0wQ3RCTUU/view?usp=sharing>

# Problem Statement

We have a dataset of sales of different TV sets across different locations.

Records look like:

Samsung|Optima|14|Madhya Pradesh|132401|14200

The fields are arranged like:

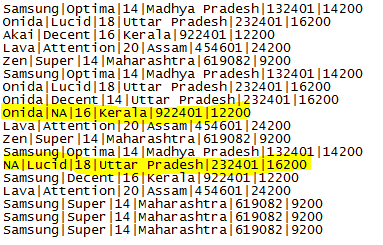
**Company Name|Product Name|Size in inches|State|Pin Code|Price**

There are some invalid records which contain 'NA' in either Company Name or Product Name.

1. Write a Map Reduce program to filter out the invalid records. Map only job will fit for this

Context.

### Actual Data Set: 2 records have ‘NA’ value



# Map only Java Program

## Driver Code

**package** InvalidRecords;

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

**import** org.apache.hadoop.fs.Path;

**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;

**import** org.w3c.dom.Text;

**public** **class** InvalidRecord

{

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

{

Configuration conf=**new** Configuration();

Job job=**new** Job(conf, "Invalid Data");

job.setJarByClass(InvalidRecord.**class**);

job.setMapOutputKeyClass(Text.**class**);

job.setMapOutputValueClass(Text.**class**);

job.setMapperClass(InvalidRecordsMapper.**class**);

job.setNumReduceTasks(0);

job.setInputFormatClass(TextInputFormat.**class**);

job.setOutputFormatClass(TextOutputFormat.**class**);

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

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

job.waitForCompletion(**true**);

}

}

## Mapper Code

**package** InvalidRecords;

**import** java.io.IOException;

**import** org.apache.hadoop.io.LongWritable;

**import** org.apache.hadoop.io.Text;

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

**public** **class** InvalidRecordsMapper **extends** Mapper<LongWritable,Text,Text,Text>

{

**public** **void** map(LongWritable key,Text value,Context context) **throws** IOException, InterruptedException

{

String line=value.toString();

String[]linearray=line.split("\\|");

**if**(!(linearray[0].equals("NA")||linearray[1].equals("NA")))

{

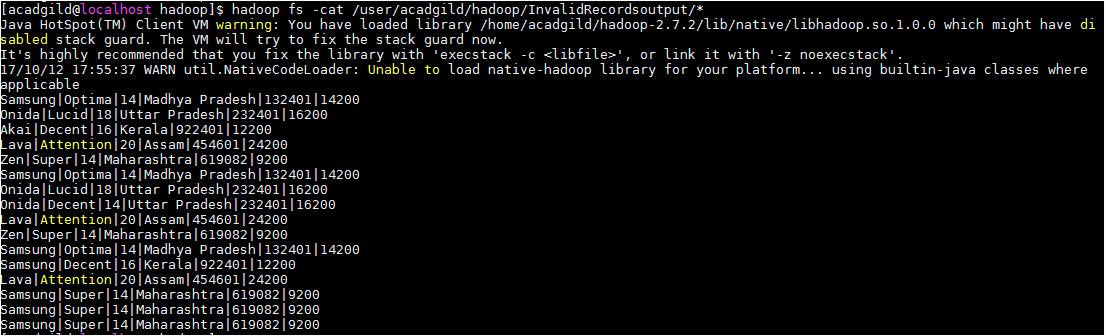
context.write(**new** Text(line), **new** Text());

}

}

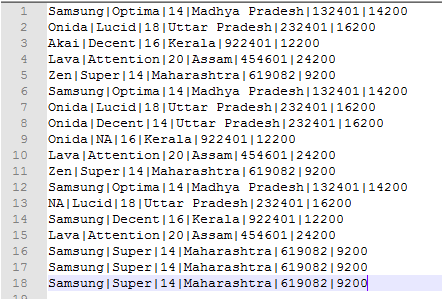
}

# Expected Output



## Input and Output Comparison

### Input



### Output

