



Name: Aaditya Gautam
Roll No: 122ad0022

Assignment 4 : HMRP ChainWordCount

Required Code Files:

1.) ChainWordCountDriver.java

```
import org.apache.hadoop.conf.Configured;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileSystem;
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.lib.chain.ChainMapper;
import org.apache.hadoop.mapreduce.lib.chain.ChainReducer;
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.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.util.Tool;
import org.apache.hadoop.util.ToolRunner;

import java.net.URI;

public class ChainWordCountDriver extends Configured implements Tool {

    public int run(String[] args) throws Exception {
        Configuration conf = getConf();
        Job job = Job.getInstance(conf, "Customer Word Count");
        job.setJarByClass(ChainWordCountDriver.class);

        // Output path (modify for your system or HDFS)
        Path outputPath = new Path(args[1]);
        FileSystem fs = FileSystem.get(new URI(outputPath.toString()), conf);
        fs.delete(outputPath, true);
```

```

// Set input and output formats
job.setInputFormatClass(TextInputFormat.class);
job.setOutputFormatClass(TextOutputFormat.class);

// Input and Output paths
FileInputFormat.setInputPaths(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, outputPath);

// Set up chain of Mappers and Reducers
Configuration mapAConf = new Configuration(false);
ChainMapper.addMapper(job, TokenizerMapper.class, LongWritable.class,
Text.class, Text.class, IntWritable.class, mapAConf);

Configuration mapBConf = new Configuration(false);
ChainMapper.addMapper(job, UpperCaserMapper.class, Text.class,
IntWritable.class, Text.class, IntWritable.class, mapBConf);

Configuration reduceConf = new Configuration(false);
ChainReducer.setReducer(job, WordCountReducer.class, Text.class,
IntWritable.class, Text.class, IntWritable.class, reduceConf);

Configuration mapCConf = new Configuration(false);
ChainReducer.addMapper(job, LastMapper.class, Text.class, IntWritable.class,
Text.class, IntWritable.class, mapCConf);

return job.waitForCompletion(true) ? 0 : 1;
}

public static void main(String[] args) throws Exception {
    int res = ToolRunner.run(new Configuration(), new ChainWordCountDriver(),
args);
    System.exit(res);
}
}

```

2). **TokenizerMapper.java**

```

import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;

```

```

import java.io.IOException;
import java.util.StringTokenizer;

public class TokenizerMapper extends Mapper<LongWritable, Text, Text,
IntWritable> {

    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(LongWritable key, Text value, Context context) throws
IOException, InterruptedException {
        String line = value.toString();
        StringTokenizer itr = new StringTokenizer(line, ",");
        while (itr.hasMoreTokens()) {
            word.set(itr.nextToken());
            context.write(word, one);
        }
    }
}

```

3). UpperCaserMapper.java

```

import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;

import java.io.IOException;

public class UpperCaserMapper extends Mapper<Text, IntWritable, Text,
IntWritable> {

    public void map(Text key, IntWritable value, Context context) throws
IOException, InterruptedException {
        String upperKey = key.toString().toUpperCase();
        context.write(new Text(upperKey), value);
    }
}

```

4). WordCountReducer.java

```
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;

import java.io.IOException;

public class WordCountReducer extends Reducer<Text, IntWritable, Text,
IntWritable> {

    public void reduce(Text key, Iterable<IntWritable> values, Context context) throws
IOException, InterruptedException {
        int sum = 0;
        for (IntWritable val : values) {
            sum += val.get();
        }
        context.write(key, new IntWritable(sum));
    }
}
```

5). LastMapper.java

```
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;

import java.io.IOException;

public class LastMapper extends Mapper<Text, IntWritable, Text, IntWritable> {

    public void map(Text key, IntWritable value, Context context) throws
IOException, InterruptedException {
        String[] parts = key.toString().split(" ");
        if (parts.length > 1) {
            // Assuming the format: "FirstName LastName"
            String lastName = parts[parts.length - 1]; // Extract last name
            context.write(new Text(lastName), value);
        }
    }
}
```

Input Dataset (Customer.csv):

text

customerId,customerName,contactNumber
1,Stephanie Leung,555-555-5555
2,Edward Kim,123-456-7890
3,Jose Madriz,281-330-8004
4,David Stork,408-555-0000
5,Emily Chen,987-654-3210
.....

OUTPUTs:

Browse Directory

Show

25

entries

Search:

<input type="checkbox"/>		Permission		Owner		Group		Size		Last Modified		Replication		Block Size		Name	
<input type="checkbox"/>		drwxr-xr-x		hadoop		supergroup		0 B		Sep 24 19:19		0		0 B		input_directory	
<input type="checkbox"/>		drwxr-xr-x		hadoop		supergroup		0 B		Sep 24 19:34		0		0 B		output_directory	

Showing 1 to 2 of 2 entries

Previous

1

Next

Hadoop, 2023.

Browse Directory

Show

25

entries

Search:

<input type="checkbox"/>		Permission		Owner		Group		Size		Last Modified		Replication		Block Size		Name	
<input type="checkbox"/>		-rw-r--r--		hadoop		supergroup		1.51 KB		Sep 24 19:19		1		128 MB		Customer.csv	

Showing 1 to 1 of 1 entries

Previous

1

Next

Hadoop, 2023.

Browse Directory

Show 25 entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rw-r--r--	hadoop	supergroup	0 B	Sep 24 19:34	1	128 MB	_SUCCESS	
<input type="checkbox"/>	-rw-r--r--	hadoop	supergroup	440 B	Sep 24 19:34	1	128 MB	part-r-00000	

Showing 1 to 2 of 2 entries

Previous

1

Next

```
hadoop@aadi-Vostro-3578:~/hadoop_4$ hadoop fs -cat /user/aadi/output_directory/p
art-r-00000
LEWIS      1
DIAZ      1
REED      1
PHILLIPS      1
JOHNSON 1
JAMES      1
TAYLOR    1
MOORE      1
ADAMS      1
KING      1
FLORES    1
STORK      1
WILSON     1
KIM        1
HARRIS     1
CHEN       1
WHITE      1
PEREZ      1
WOOD       1
YOUNG      1
MITCHELL      1
COOPER     1
BROOKS     1
CLARK      1
CAMPBELL      1
BAKER      1
BROWN      1
```

File information - part-r-00000

[Download](#)[Head the file \(first 32K\)](#)[Tail the file \(last 32K\)](#)

Block information --

Block 0 ▾

Block ID: 1073741845

Block Pool ID: BP-60072424-127.0.1.1-1726503367100

Generation Stamp: 1021

Size: 440

Availability:

- aadi-Vostro-3578

File contents

```
LEWIS 1
DIAZ 1
REED 1
PHILLIPS 1
JOHNSON 1
JAMES 1
TAYLOR 1
MOORE 1
```

Close