

Assignment No. 2

Subject: Data Science & Big Data Analytics Lab

Class: TE 09

Name: Samyak Jain

Roll No: 33134

Word Count Program

Code:

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

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
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;

public class WordCount {

    public static class TokenizerMapper
        extends Mapper<Object, Text, Text, IntWritable>{

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

        public void map(Object key, Text value, Context context
```

```

        ) throws IOException, InterruptedException {
StringTokenizer itr = new StringTokenizer(value.toString());
while (itr.hasMoreTokens()) {
    word.set(itr.nextToken());
    context.write(word, one);
}
}
}

```

```

public static class IntSumReducer
    extends Reducer<Text,IntWritable,Text,IntWritable> {
    private IntWritable result = new IntWritable();

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

```

```

public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
}

```

```

    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
}
}

```

TERMINAL INPUT COMMANDS AND OUTPUT:

```

[cloudera@quickstart ~]$ pwd
/home/cloudera
[cloudera@quickstart ~]$ cat > /home/cloudera/ProcessFile1.txt
car
truck
bike
truck
car
car
bike
car
truck
car
truck
truck
car
bike
^Z
[1]+  Stopped                  cat > /home/cloudera/ProcessFile1.txt
[cloudera@quickstart ~]$ hdfs dfs -ls
[cloudera@quickstart ~]$ hdfs dfs -ls /
Found 5 items
drwxr-xr-x  - hbase supergroup          0 2023-01-30 22:09 /hbase

```

```
drwxr-xr-x    - solr  solr                0 2015-06-09 03:38 /solr
drwxrwxrwx    - hdfs  supergroup          0 2023-01-29 22:26 /tmp
drwxr-xr-x    - hdfs  supergroup          0 2015-06-09 03:38 /user
drwxr-xr-x    - hdfs  supergroup          0 2015-06-09 03:36 /var

[cloudera@quickstart ~]$ hdfs dfs -put /home/cloudera/ProcessFile1.txt
/inputfolder1/

[cloudera@quickstart ~]$ hdfs dfs -ls /out1
Found 2 items
-rw-r--r--    1 cloudera supergroup          0 2023-01-30 23:27
/out1/_SUCCESS
-rw-r--r--    1 cloudera supergroup        21 2023-01-30 23:27
/out1/part-r-00000

[cloudera@quickstart ~]$ hdfs dfs -cat /out1/part-r-00000
bike 3
car 6
truck 5

[cloudera@quickstart ~]$
```