

**DSBDA MOCK – Finding the minimum and maximum bills in each year from 2011 to 2025 using the hadoop.**

**Roll no : 33320**

```
package PackageDemo;
```

```
import java.io.IOException;
```

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

```
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.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;
```

```
import org.apache.hadoop.util.GenericOptionsParser;
```

```
public class WordCount {
```

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

```
        // Set up configuration
```

```
        Configuration c = new Configuration();
```

```
        String[] files = new GenericOptionsParser(c, args).getRemainingArgs();
```

```
        // Input and output paths
```

```

Path input = new Path(files[0]);
Path output = new Path(files[1]);

// Set up the job
Job j = new Job(c, "WordCount");
j.setJarByClass(WordCount.class);
j.setMapperClass(MapForWordCount.class);
j.setReducerClass(ReduceForWordCount.class);

// Set output types
j.setOutputKeyClass(Text.class);
j.setOutputValueClass(Text.class);

// Set input and output paths
FileInputFormat.addInputPath(j, input);
FileOutputFormat.setOutputPath(j, output);

// Run the job
System.exit(j.waitForCompletion(true) ? 0 : 1);
}

```

```

// Mapper for calculating min and max

public static class MapForWordCount extends Mapper<LongWritable, Text, Text, Text> {

    public void map(LongWritable key, Text value, Context con) throws IOException,
        InterruptedException {

        // Convert the line into a string and split by commas
        String line = value.toString();
        String[] tokens = line.split(",");

        // Extract the year (first entry)
        String year = tokens[0].replaceAll("\\\"", "").trim();

        // Initialize variables for min and max values
        int minBill = Integer.MAX_VALUE;

```

```

int maxBill = Integer.MIN_VALUE;

// Process the monthly bills (skip the year, process the rest)
for (int i = 1; i < tokens.length; i++) {
    try {
        int bill = Integer.parseInt(tokens[i]);
        // Update min and max values
        if (bill < minBill) {
            minBill = bill;
        }
        if (bill > maxBill) {
            maxBill = bill;
        }
    } catch (NumberFormatException e) {
        // If there's an invalid number, just skip it
        continue;
    }
}

// Output the year along with min and max values
con.write(new Text(year), new Text(minBill + "," + maxBill));
}
}

// Reducer for passing the min/max values
public static class ReduceForWordCount extends Reducer<Text, Text, Text, Text> {

    public void reduce(Text key, Iterable<Text> values, Context con) throws IOException,
        InterruptedException {

        // Since each year will have only one result from the mapper, we can directly pass it through
        for (Text val : values) {

```

```
con.write(key, val);  
}  
}  
}  
}
```

Terminal Commnds:

```
[cloudera@quickstart Desktop]$ hadoop fs -put elecbills.txt elecbills1
```

```
[cloudera@quickstart Desktop]$ hadoop jar MRProgramsDemo2.jar  
PackageDemo.WordCount elecbills1 MRDir101
```

```
25/03/26 00:26:24 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
```

```
25/03/26 00:26:25 WARN security.UserGroupInformation: PriviledgedActionException  
as:cloudera (auth:SIMPLE) cause:org.apache.hadoop.mapred.FileAlreadyExistsException:  
Output directory hdfs://quickstart.cloudera:8020/user/cloudera/MRDir101 already exists
```

```
Exception in thread "main" org.apache.hadoop.mapred.FileAlreadyExistsException: Output  
directory hdfs://quickstart.cloudera:8020/user/cloudera/MRDir101 already exists
```

```
    at  
org.apache.hadoop.mapreduce.lib.output.FileOutputFormat.checkOutputSpecs(FileOutputF  
ormat.java:146)
```

```
    at org.apache.hadoop.mapreduce.JobSubmitter.checkSpecs(JobSubmitter.java:562)
```

```
    at  
org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:432)
```

```
    at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1306)
```

```
    at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1303)
```

```
    at java.security.AccessController.doPrivileged(Native Method)
```

```
    at javax.security.auth.Subject.doAs(Subject.java:415)
```

```
    at  
org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1671)
```

```
    at org.apache.hadoop.mapreduce.Job.submit(Job.java:1303)
```

```
at org.apache.hadoop.mapreduce.Job.waitForCompletion(Job.java:1324)
at PackageDemo.WordCount.main(WordCount.java:41)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:606)
at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
```

```
[cloudera@quickstart Desktop]$ hadoop jar MRProgramsDemo2.jar
PackageDemo.WordCount elec bills1 MRDir102
```

25/03/26 00:26:33 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032

25/03/26 00:26:35 INFO input.FileInputFormat: Total input paths to process : 1

25/03/26 00:26:35 INFO mapreduce.JobSubmitter: number of splits:1

25/03/26 00:26:35 INFO mapreduce.JobSubmitter: Submitting tokens for job:  
job\_1742968849704\_0003

25/03/26 00:26:35 INFO impl.YarnClientImpl: Submitted application  
application\_1742968849704\_0003

25/03/26 00:26:36 INFO mapreduce.Job: The url to track the job:  
[http://quickstart.cloudera:8088/proxy/application\\_1742968849704\\_0003/](http://quickstart.cloudera:8088/proxy/application_1742968849704_0003/)

25/03/26 00:26:36 INFO mapreduce.Job: Running job: job\_1742968849704\_0003

25/03/26 00:26:47 INFO mapreduce.Job: Job job\_1742968849704\_0003 running in uber  
mode : false

25/03/26 00:26:47 INFO mapreduce.Job: map 0% reduce 0%

25/03/26 00:26:55 INFO mapreduce.Job: map 100% reduce 0%

25/03/26 00:27:07 INFO mapreduce.Job: map 100% reduce 100%

25/03/26 00:27:08 INFO mapreduce.Job: Job job\_1742968849704\_0003 completed  
successfully

25/03/26 00:27:08 INFO mapreduce.Job: Counters: 49

File System Counters

FILE: Number of bytes read=195

FILE: Number of bytes written=221061

FILE: Number of read operations=0

FILE: Number of large read operations=0

FILE: Number of write operations=0

HDFS: Number of bytes read=808

HDFS: Number of bytes written=159

HDFS: Number of read operations=6

HDFS: Number of large read operations=0

HDFS: Number of write operations=2

#### Job Counters

Launched map tasks=1

Launched reduce tasks=1

Data-local map tasks=1

Total time spent by all maps in occupied slots (ms)=6796

Total time spent by all reduces in occupied slots (ms)=8070

Total time spent by all map tasks (ms)=6796

Total time spent by all reduce tasks (ms)=8070

Total vcore-seconds taken by all map tasks=6796

Total vcore-seconds taken by all reduce tasks=8070

Total megabyte-seconds taken by all map tasks=6959104

Total megabyte-seconds taken by all reduce tasks=8263680

#### Map-Reduce Framework

Map input records=15

Map output records=15

Map output bytes=159

Map output materialized bytes=195

Input split bytes=121

Combine input records=0

Combine output records=0

Reduce input groups=15  
Reduce shuffle bytes=195  
Reduce input records=15  
Reduce output records=15  
Spilled Records=30  
Shuffled Maps =1  
Failed Shuffles=0  
Merged Map outputs=1  
GC time elapsed (ms)=140  
CPU time spent (ms)=1230  
Physical memory (bytes) snapshot=348332032  
Virtual memory (bytes) snapshot=3007369216  
Total committed heap usage (bytes)=226365440

#### Shuffle Errors

BAD\_ID=0  
CONNECTION=0  
IO\_ERROR=0  
WRONG\_LENGTH=0  
WRONG\_MAP=0  
WRONG\_REDUCE=0

#### File Input Format Counters

Bytes Read=687

#### File Output Format Counters

Bytes Written=159

[cloudera@quickstart Desktop]\$ hadoop fs -ls MRDir102

Found 2 items

-rw-r--r-- 1 cloudera cloudera 0 2025-03-26 00:27 MRDir102/\_SUCCESS

-rw-r--r-- 1 cloudera cloudera 159 2025-03-26 00:27 MRDir102/part-r-00000

```
[cloudera@quickstart Desktop]$ hadoop fs -cat MRDir102/part-r-000002011 2,43
```

```
2012 26,31
```

```
2013 31,36
```

```
2014 38,43
```

```
2015 0,45
```

```
2016 2,43
```

```
2017 26,31
```

```
2018 31,36
```

```
2019 38,43
```

```
2020 0,45
```

```
2021 2,43
```

```
2022 26,31
```

```
2023 31,36
```

```
2024 38,43
```

```
2025 0,45
```

```
[cloudera@quickstart Desktop]$
```