# Assignment no. 11

**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);
  }
}
```

**Output:**

```
$ echo -e "Hello World\nBye World" > input.txt

$ hadoop com.sun.tools.javac.Main WordCount.java

$ jar cf WordCount.jar WordCount*.class

$ hadoop jar WordCount.jar WordCount input.txt output

2025-04-20 17:53:21,402 INFO client.RMProxy: Connecting to ResourceManager at
localhost/127.0.0.1:8032
2025-04-20 17:53:21,925 INFO mapreduce.JobSubmitter: number of splits:1
2025-04-20 17:53:21,964 INFO mapreduce.JobSubmitter: Submitting tokens for job:
job_1713626600243_0001
2025-04-20 17:53:22,317 INFO impl.YarnClientImpl: Submitted application
application_1713626600243_0001
2025-04-20 17:53:22,402 INFO mapreduce.Job: The url to track the job:
http://localhost:8088/proxy/application_1713626600243_0001/
2025-04-20 17:53:22,402 INFO mapreduce.Job: Running job: job_1713626600243_0001
2025-04-20 17:53:28,402 INFO mapreduce.Job: Job job_1713626600243_0001 completed successfully
2025-04-20 17:53:28,417 INFO mapreduce.Job: Counters: 34
        File System Counters
                FILE: Number of bytes read=66
                FILE: Number of bytes written=260843
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
        Map-Reduce Framework
                Map input records=2
                Map output records=4
                Map output bytes=36
                Map output materialized bytes=42
                Input split bytes=100
                Combine input records=4
                Combine output records=3
                Reduce input groups=3
                Reduce shuffle bytes=42
                Reduce input records=3
                Reduce output records=3
                Spilled Records=6
                Shuffled Maps = 1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=50
                Total committed heap usage (bytes)=314572800

$ cat output/part-r-00000
Bye     1
Hello   1
World   2
```