

# 单词计数

作者: [mubai](#)

原文链接: <https://ld246.com/article/1478885385535>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

```
//编写Mapper类 TokenizerMapper.java
```

```
import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

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, Interr
    ptedException {

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

```
//编写Reducer类 IntSumReducer.java
```

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

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

    private IntWritable result = new IntWritable();

    public void reduce(Text key, Iterable<IntWritable> values, Context context) throws IOExcep
    tion, InterruptedException {

        int sum = 0;
        for (IntWritable val : values) {
            sum += val.get();
        }
        result.set(sum);
        context.write(key, result);
    }
}
```

```
//编写main函数 WordCount.java
```

```
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.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class WordCount {

    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {

        Configuration conf = new Configuration();
        if (args.length != 2) {
            System.err.println("Usage:wordcount<in> <out>");
            System.exit(2);
        }

        Job job = new Job(conf, "word count");
        job.setJarByClass(WordCount.class);
        job.setMapperClass(TokenizerMapper.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);
    }
}

```

//测试数据 input.txt

```

hello world
hello world
hello hadoop

```

#在HDFS上创建文件夹

```
hadoop fs -mkdir -p /class
```

#复制本地数据到HDFS

```
hadoop fs -copyFromLocal input.txt /class/input.txt
```

#编译源码

```
javac -classpath ../hadoop-core-1.2.1.jar *.java
```

#将类文件打包成jar

```
jar cvf WordCount.jar *.class
```

#运行程序，并将结果输出到HDFS上的/class/out文件夹

```
hadoop jar WordCount.jar WordCount /class/input.txt /class/out
```

#罗列HDFS上文件夹里的文件

```
hadoop fs -ls /class/out
```

#查看运行结果

```
hadoop fs -cat /class/out/part-r-00000
```

#运行结果

```
hadoop 1  
hello 3  
world 2
```