

In the MapReduce wordcount example, we find out the frequency of each word. Here, the role of the Mapper is to map the keys to the existing values and the role of the Reducer is to aggregate the keys of common values. So, everything is represented in the form of key-value pair.

Steps to execute MapReduce word count example

- Create a text file in your local machine and write some text into it.
\$ nano data.txt
- Check the text written in the data.txt file.
\$ cat data.txt

For Mapper.

```
package com.javatpoint;
```

```
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.mapred.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;

public class WC_Mapper extends MapReduceBase implements
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,OutputCollector<Text,IntWritable> output,
    Reporter reporter) throws IOException{
    String line = value.toString();
    StringTokenizer tokenizer = new StringTokenizer(line);
    while (tokenizer.hasMoreTokens()){
        word.set(tokenizer.nextToken());
        output.collect(word, one);
    }
}
}

```

For reducer:

```

package com.javatpoint;

import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;

```

```
public class WC_Reducer extends MapReduceBase implements
Reducer<Text,IntWritable,Text,IntWritable> {

    public void reduce(Text key, Iterator<IntWritable>
values,OutputCollector<Text,IntWritable> output,
    Reporter reporter) throws IOException {
        int sum=0;
        while (values.hasNext()) {
            sum+=values.next().get();
        }
        output.collect(key,new IntWritable(sum));
    }
}
```