MAP REDUCER-HADOOP

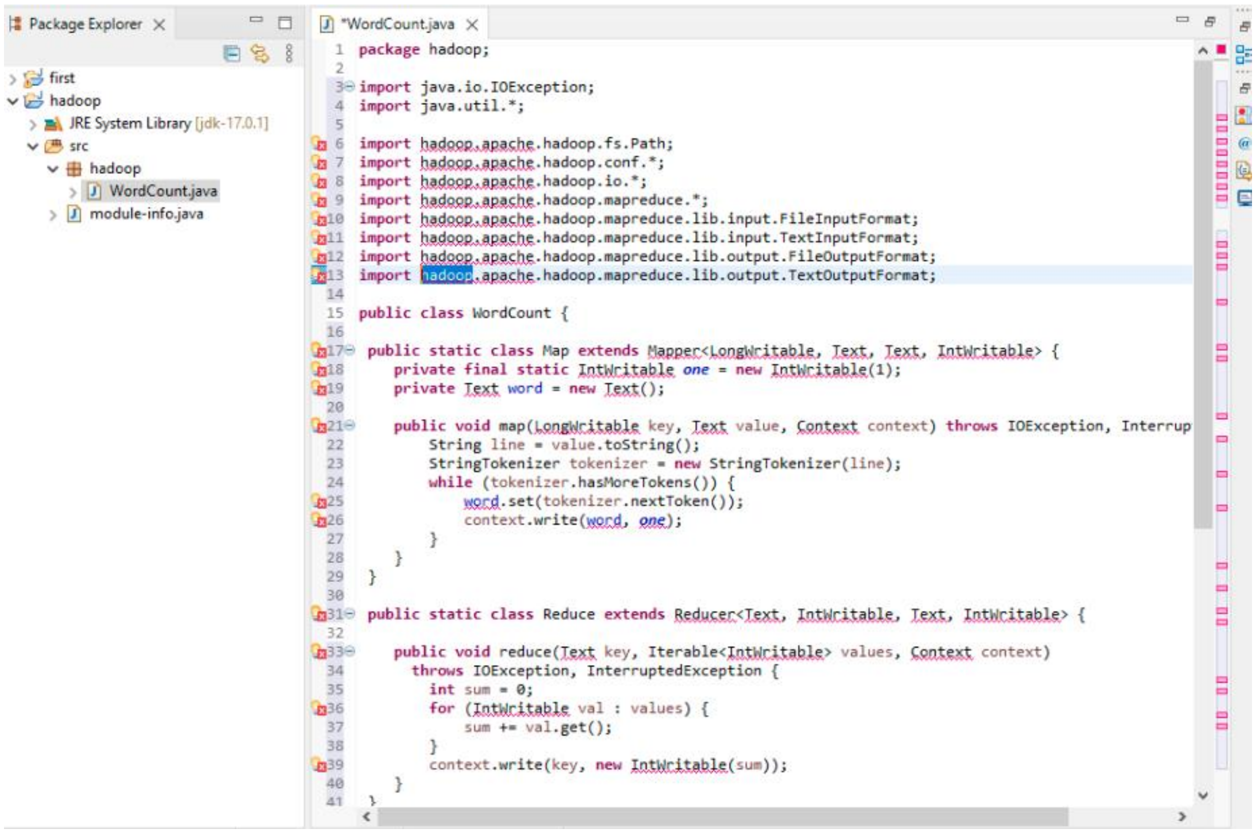
Rao Nauman

P19-0073

Steps to do:

- ✓ Open any code editor (I have used Eclipse) compatible with compiling Java.
- ✓ Create a project WordCount or give it any name and create a class named WordCount.
- ✓ Inside the class type in the following code of MapReduce

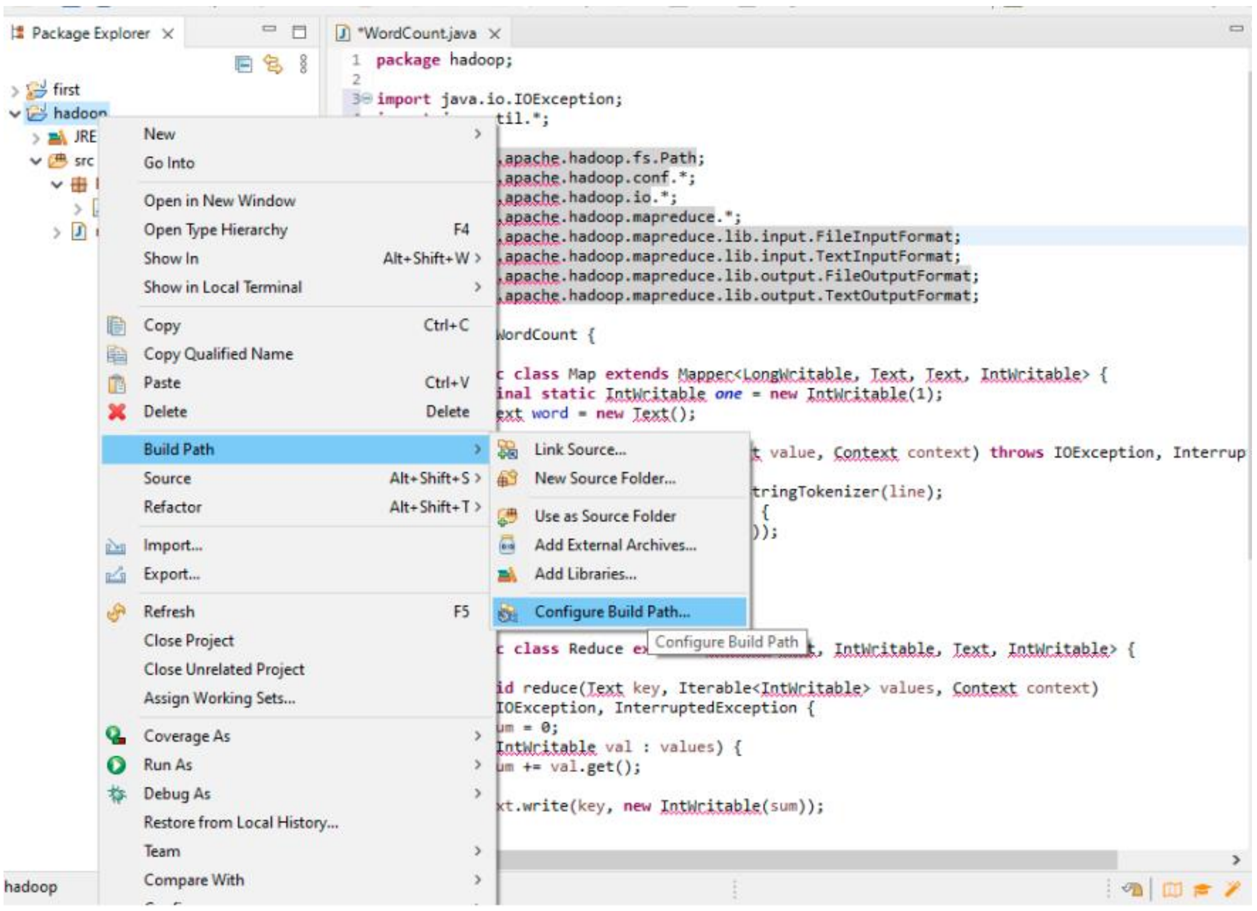
```
mport java.io.IOException;

import java.util.*;


import org.apache.hadoop.fs.Path;

import org.apache.hadoop.conf.*;

import org.apache.hadoop.io.*;

import org.apache.hadoop.mapreduce.*;

import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;

import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;


public class WordCount {


public static class Map 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, InterruptedException {

 String line = value.toString();

 StringTokenizer tokenizer = new StringTokenizer(line);

 while (tokenizer.hasMoreTokens()) {

 word.set(tokenizer.nextToken());
 context.write(word, one);
 }
 }
}

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

public static void main(String[] args) throws Exception {
 Configuration conf = new Configuration();

 Job job = new Job(conf, "wordcount");

 job.setOutputKeyClass(Text.class);
 job.setOutputValueClass(IntWritable.class);

 job.setMapperClass(Map.class);
 job.setReducerClass(Reduce.class);
 job.setInputFormatClass(TextInputFormat.class);

 job.setOutputFormatClass(TextOutputFormat.class);


 FileInputFormat.addInputPath(job, new Path(args[0]));

 FileOutputFormat.setOutputPath(job, new Path(args[1]));


 job.waitForCompletion(true);

}


}
```



```
package hadoop;

import java.io.IOException;
import java.util.*;

import hadoop.apache.hadoop.fs.Path;
import hadoop.apache.hadoop.conf.*;
import hadoop.apache.hadoop.io.*;
import hadoop.apache.hadoop.mapreduce.*;
import hadoop.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import hadoop.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import hadoop.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import hadoop.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

public class WordCount {

 public static class Map 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, Interrup
        String line = value.toString();
        StringTokenizer tokenizer = new StringTokenizer(line);
        while (tokenizer.hasMoreTokens()) {
            word.set(tokenizer.nextToken());
            context.write(word, one);
        }
    }
 }

 public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable> {

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

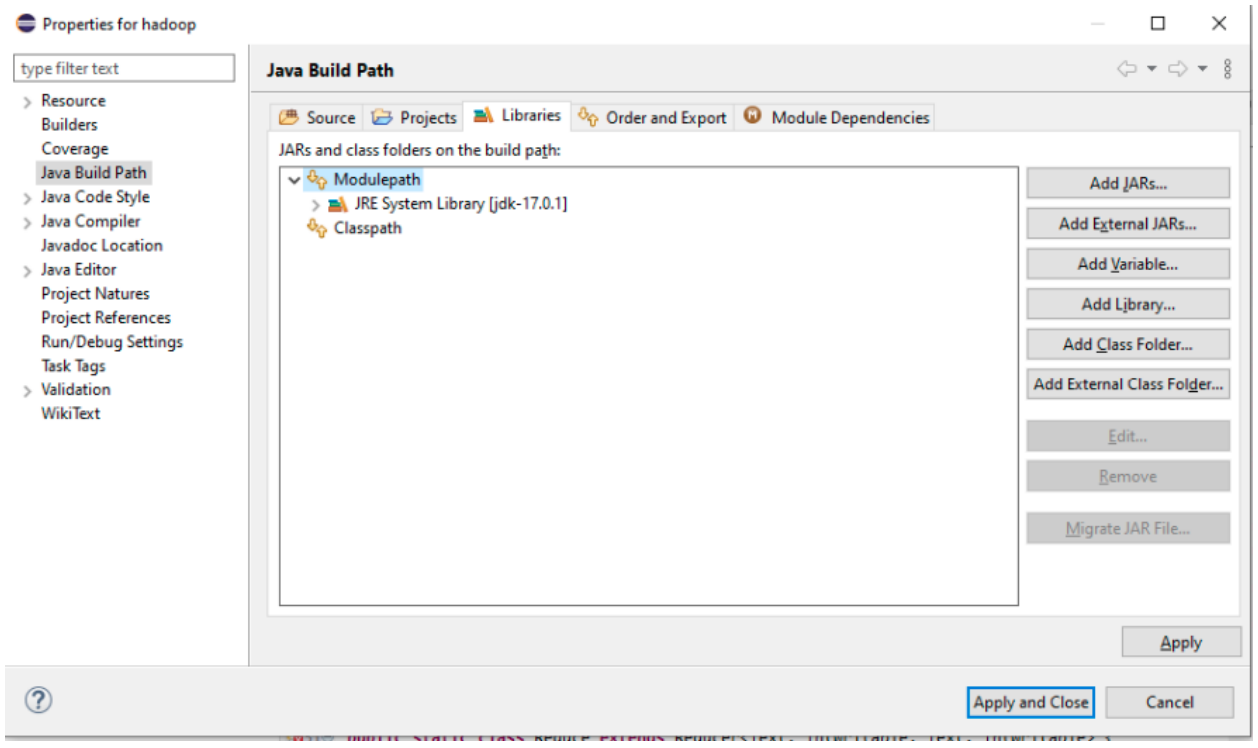
- ✓ Go to https://github.com/jijim/HADOOP￼Windows10/blob/master/Eclipse_JAR_Imports_Final.rar and download Jar Files required to successfully run the program and after downloading, extract them.

✓ On left side of window of Eclipse, right click on Hadoop, click on build Path and click on Configure build path.

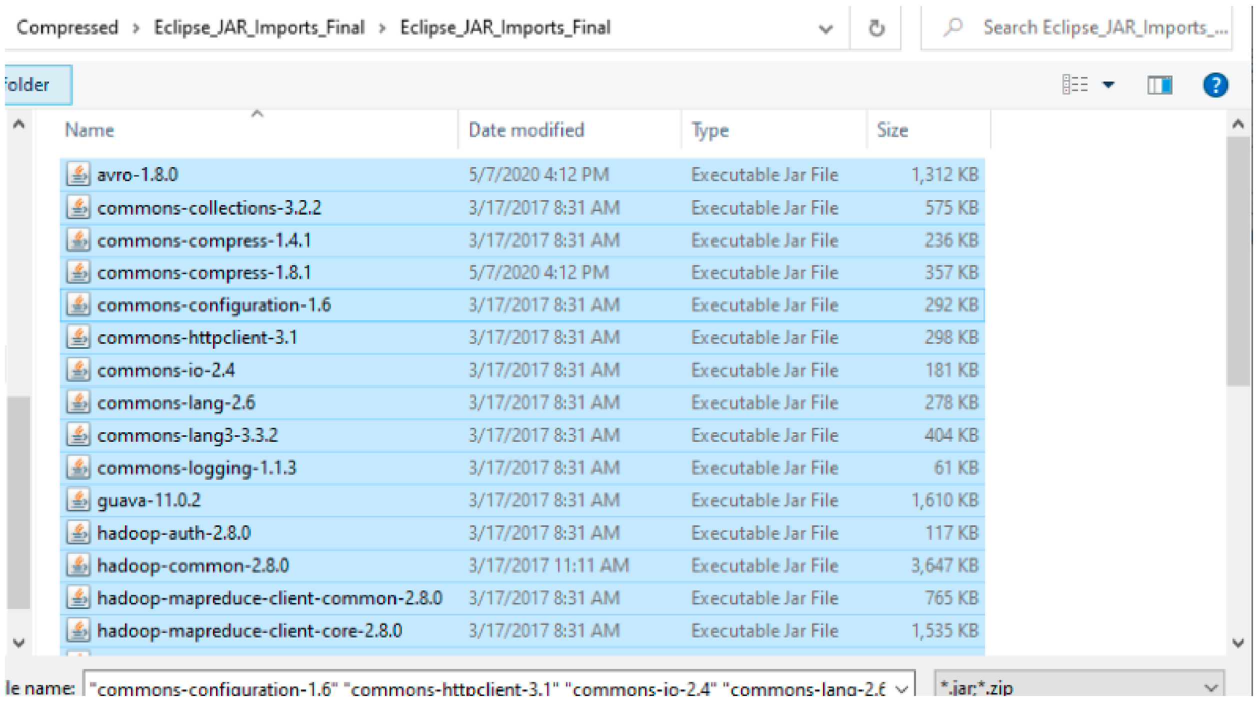


```
package hadoop;

import java.io.IOException;
                 til.*;

.apache.hadoop.fs.Path;
.apache.hadoop.conf.*;
.apache.hadoop.io.*;
.apache.hadoop.mapreduce.*;
.apache.hadoop.mapreduce.lib.input.FileInputFormat;
.apache.hadoop.mapreduce.lib.input.TextInputFormat;
.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

WordCount {

c class Map extends Mapper<LongWritable, Text, Text, IntWritable> {
inal static IntWritable one = new IntWritable(1);
ext word = new Text();

                                     t value, Context context) throws IOException, Interrup

                                     tringTokenizer(line);
                                      {
                                     ));




c class Reduce ex                        t, IntWritable, Text, IntWritable> {

id reduce(Text key, Iterable<IntWritable> values, Context context)
IOException, InterruptedException {
um = 0;
IntWritable val : values) {
um += val.get();

xt.write(key, new IntWritable(sum));
```

A WINDOW WILL BE OPENED



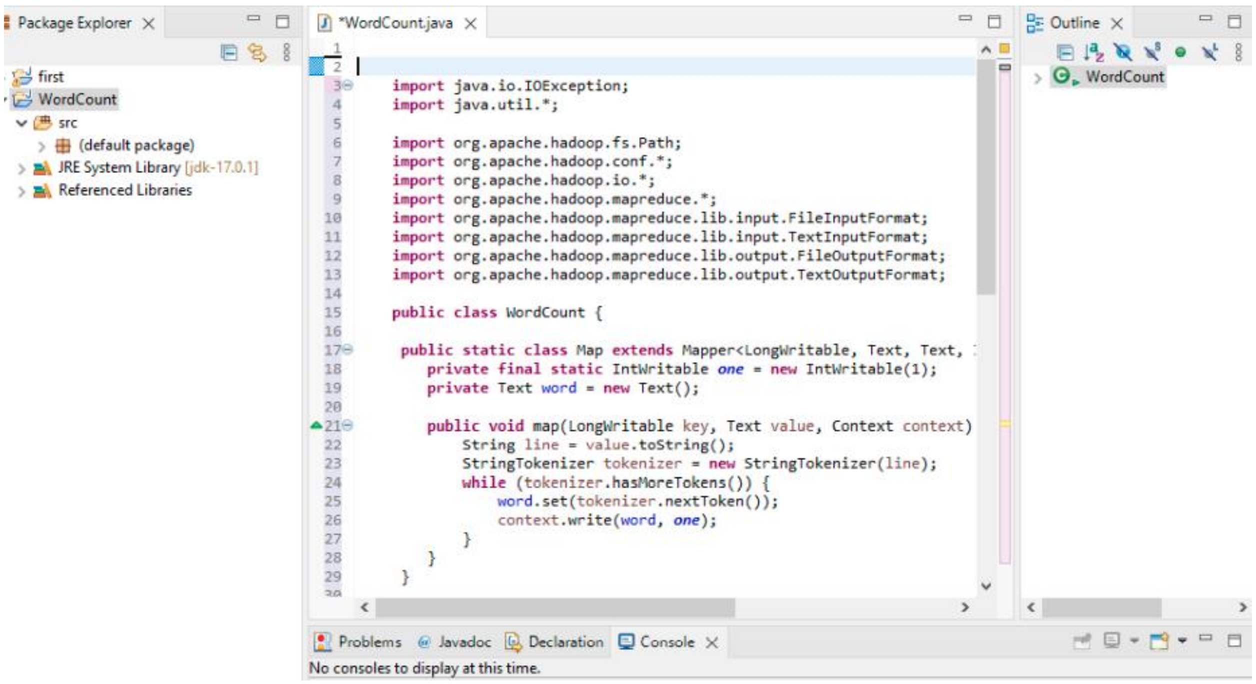
- ✓ Click on Add External JARs.. and select all jar files which you downloaded earlier


Compressed › Eclipse_JAR_Imports_Final › Eclipse_JAR_Imports_Final
Search Eclipse_JAR_Imports_...
older

Name	Date modified	Type	Size
avro-1.8.0	5/7/2020 4:12 PM	Executable Jar File	1,312 KB
commons-collections-3.2.2	3/17/2017 8:31 AM	Executable Jar File	575 KB
commons-compress-1.4.1	3/17/2017 8:31 AM	Executable Jar File	236 KB
commons-compress-1.8.1	5/7/2020 4:12 PM	Executable Jar File	357 KB
commons-configuration-1.6	3/17/2017 8:31 AM	Executable Jar File	292 KB
commons-httpclient-3.1	3/17/2017 8:31 AM	Executable Jar File	298 KB
commons-io-2.4	3/17/2017 8:31 AM	Executable Jar File	181 KB
commons-lang-2.6	3/17/2017 8:31 AM	Executable Jar File	278 KB
commons-lang3-3.3.2	3/17/2017 8:31 AM	Executable Jar File	404 KB
commons-logging-1.1.3	3/17/2017 8:31 AM	Executable Jar File	61 KB
guava-11.0.2	3/17/2017 8:31 AM	Executable Jar File	1,610 KB
hadoop-auth-2.8.0	3/17/2017 8:31 AM	Executable Jar File	117 KB
hadoop-common-2.8.0	3/17/2017 11:11 AM	Executable Jar File	3,647 KB
hadoop-mapreduce-client-common-2.8.0	3/17/2017 8:31 AM	Executable Jar File	765 KB
hadoop-mapreduce-client-core-2.8.0	3/17/2017 8:31 AM	Executable Jar File	1,535 KB

le name: "commons-configuration-1.6" "commons-httpclient-3.1" "commons-io-2.4" "commons-lang-2.6 *.jar;*.zip

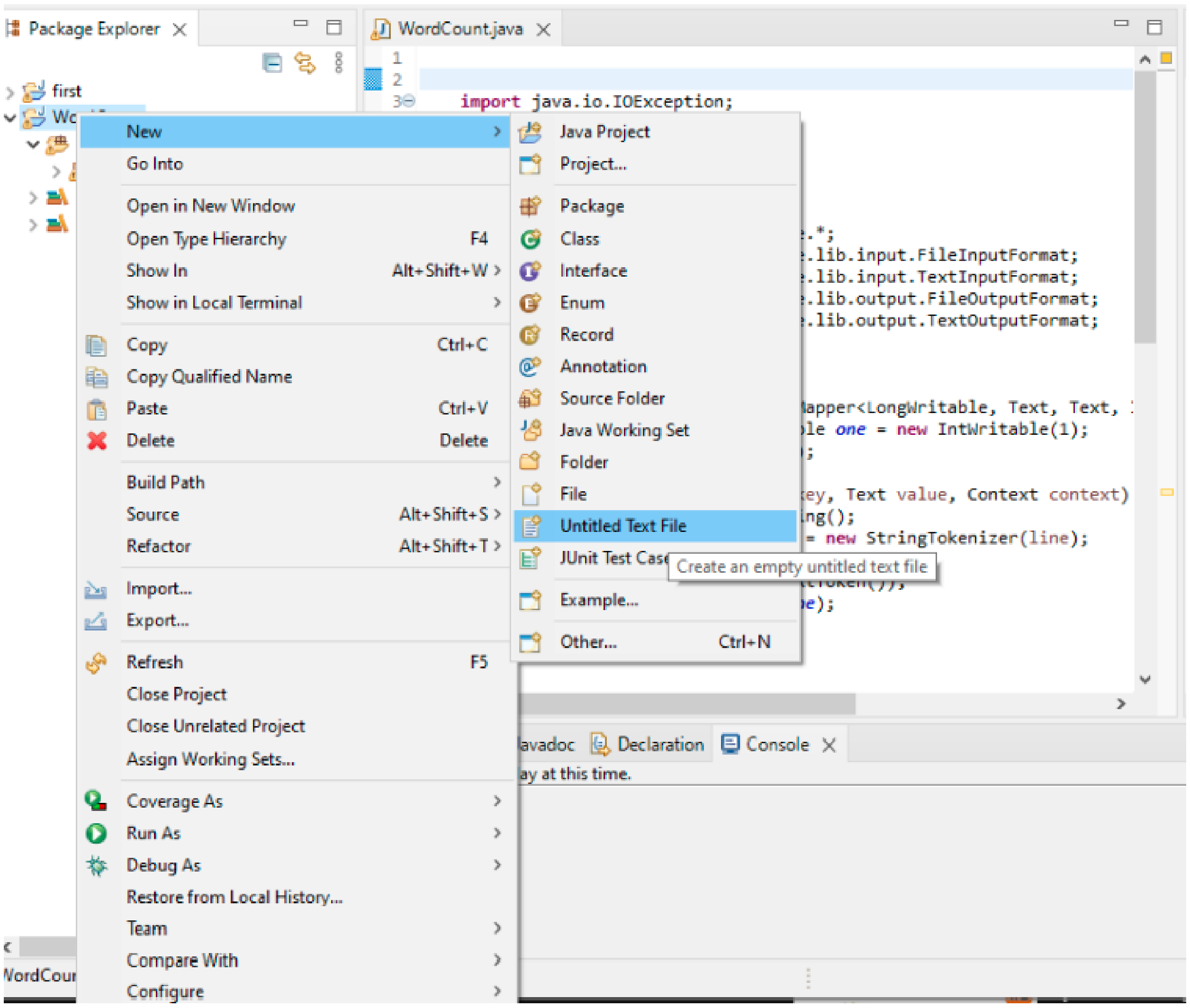
- ✓ . After adding, click on apply and you'll see all the errors will be removed



```java

import java.io.IOException;
import java.util.*;

import org.apache.hadoop.fs.Path;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.io.*;
import org.apache.hadoop.mapreduce.*;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

public class WordCount {

 public static class Map extends Mapper<LongWritable, Text, Text, 
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(LongWritable key, Text value, Context context)
        String line = value.toString();
        StringTokenizer tokenizer = new StringTokenizer(line);
        while (tokenizer.hasMoreTokens()) {
            word.set(tokenizer.nextToken());
            context.write(word, one);
        }
    }
 }
```

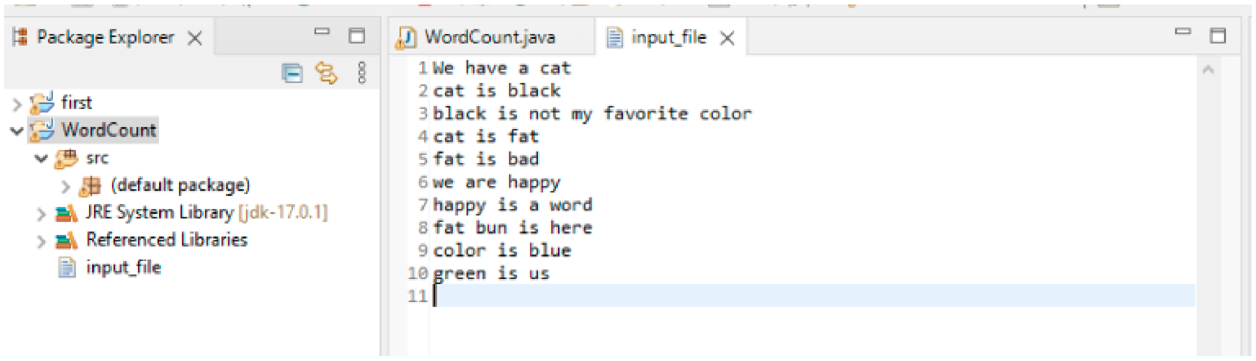
No consoles to display at this time.

✓ . Now in order to create an input file for WordCount program, right click on WordCount, Click on new and then click on Untitled Text File



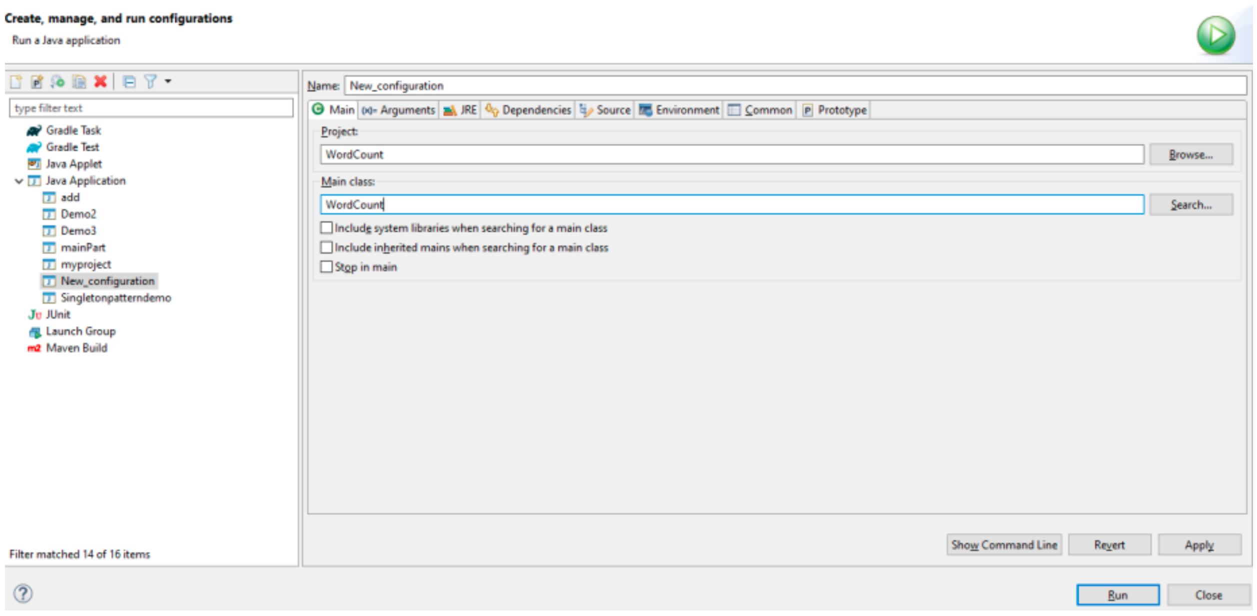
```
import java.io.IOException;
```

✓ In that text file, add any random text and save it.



```
We have a cat
cat is black
black is not my favorite color
cat is fat
fat is bad
we are happy
happy is a word
fat bun is here
color is blue
green is us
```

✓ After that click on run in menu bar and then click on run configurations. A window will be opened. Give the main Class name.


Create, manage, and run configurations

Run a Java application

Name: New_configuration

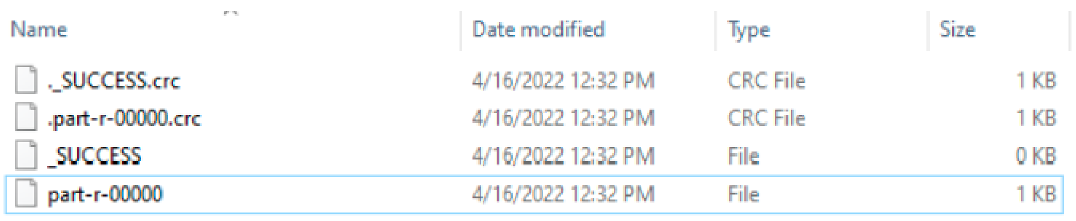
Project: WordCount

Main class: WordCount

- [ ] Include system libraries when searching for a main class
- [ ] Include inherited mains when searching for a main class
- [ ] Stop in main

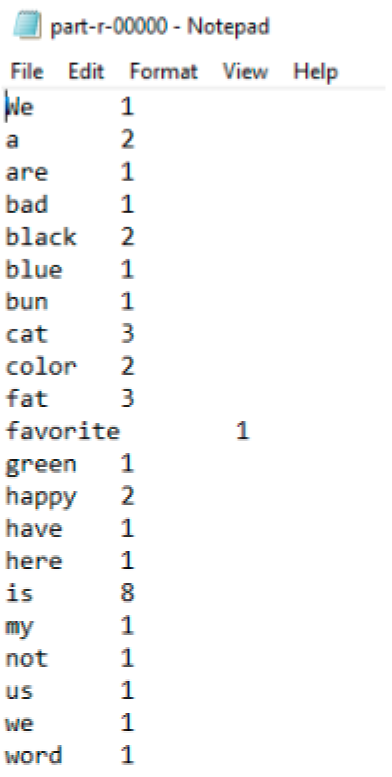
Filter matched 14 of 16 items

✓ Go to your directory: **eclipse-workspace\WordCount** and open the output directory which you created for program. There will be a file part-r-00000. Open it with text editor. You'll be able to see your output of word count.



Name	Date modified	Type	Size
._SUCCESS.crc	4/16/2022 12:32 PM	CRC File	1 KB
.part-r-00000.crc	4/16/2022 12:32 PM	CRC File	1 KB
_SUCCESS	4/16/2022 12:32 PM	File	0 KB
part-r-00000	4/16/2022 12:32 PM	File	1 KB

✓ Result



```
We      1
a       2
are     1
bad     1
black   2
blue    1
bun     1
cat     3
color   2
fat     3
favorite        1
green   1
happy   2
have    1
here    1
is      8
my      1
not     1
us      1
we      1
word    1
```