Rao Nauman

p19-0073

map reducer-hadoop

**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);

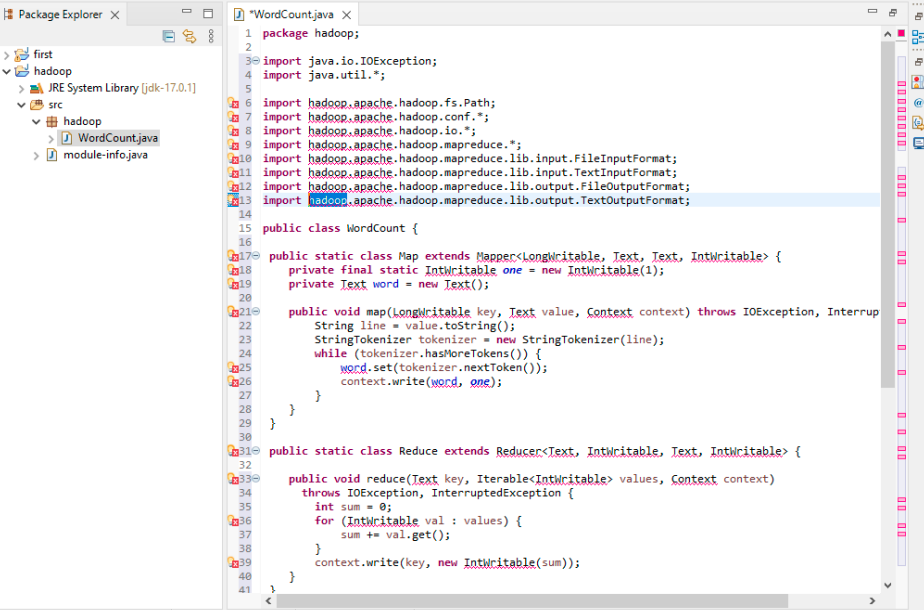
FileInputFormat.addInputPath(job, new Path(args[0]));

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

job.waitForCompletion(true);

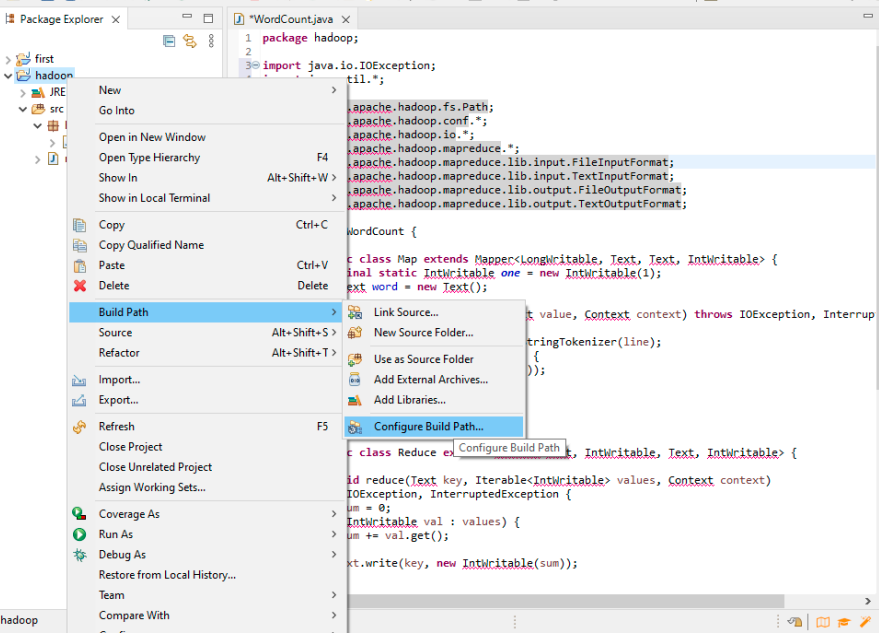
}

}

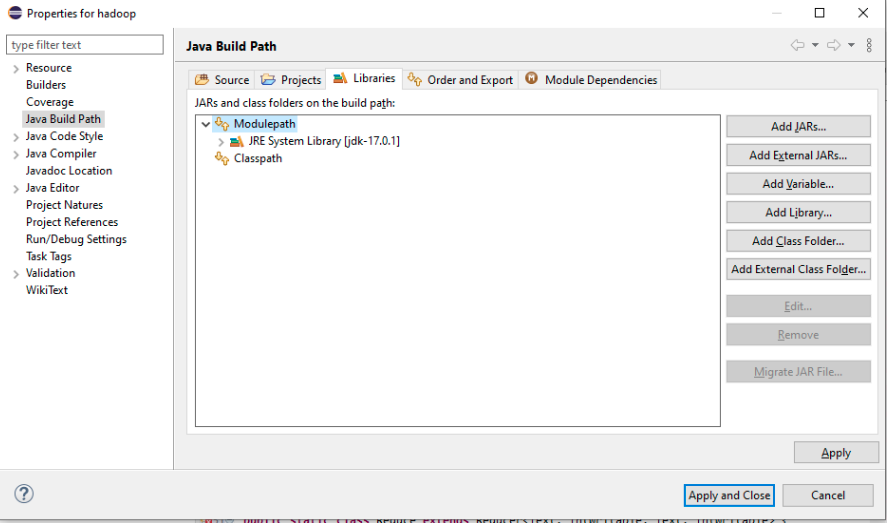


* Go to https://github.com/jijim/HADOOPWindows10/blob/master/Eclipse\_JAR\_Imports\_Final.rar and download Jar Files required to successfully run the program and after downloading, extract them.
* On left side of window of Eclipse, right click on Hadoop, click on build Path and click on

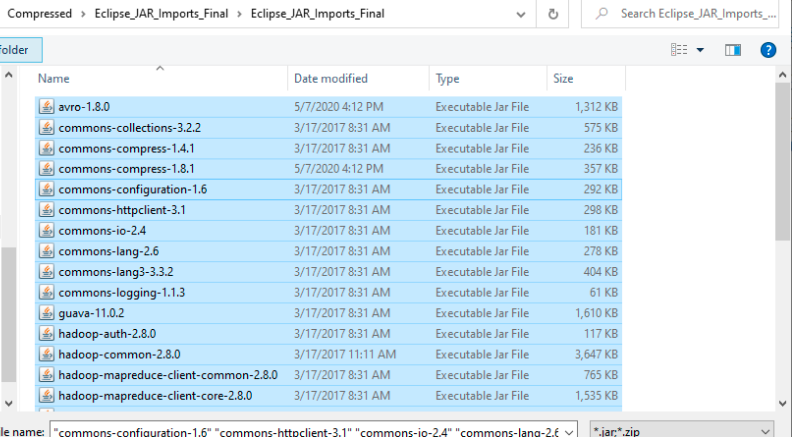
Configure build path.



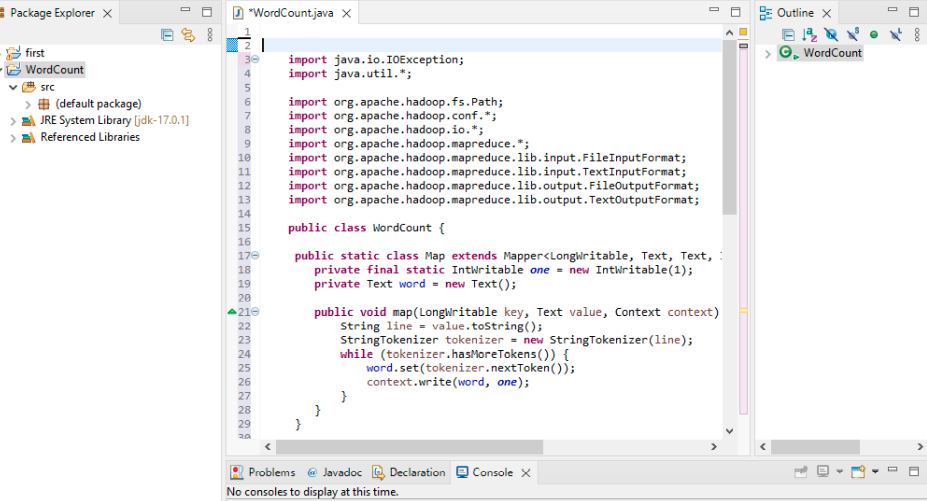
A WINDOW WILL BE OPENED



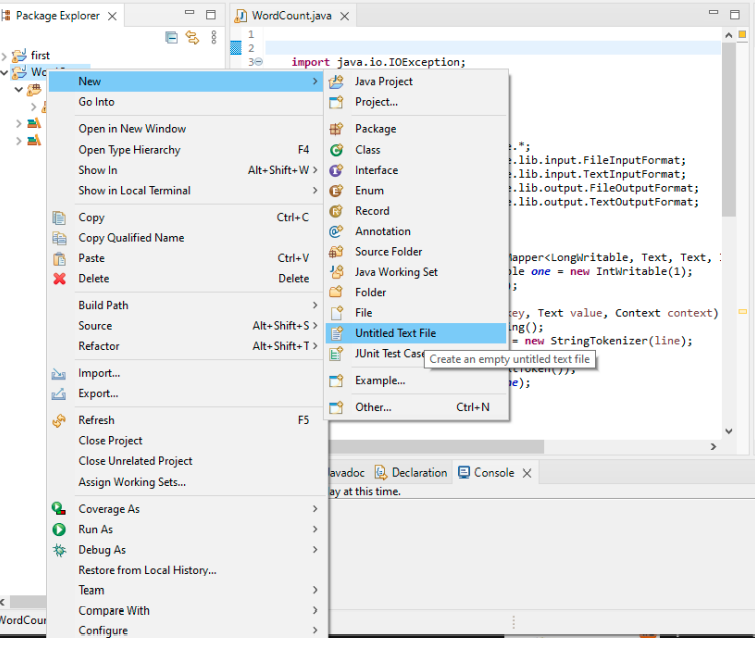
* Click on Add External JARs.. and select all jar files which you downloaded earlier



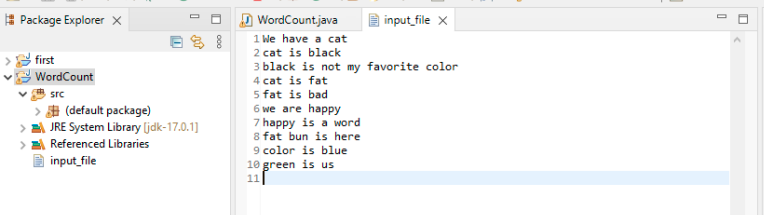
* . After adding, click on apply and you’ll see all the errors will be removed



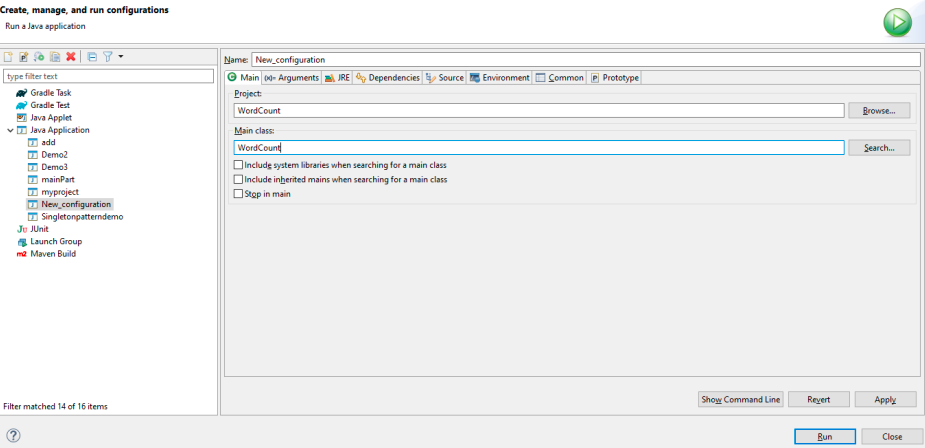
* . Now in order to create an input file for WordCount program, right click on WordCount, Click on new and then click on Untitled Text File



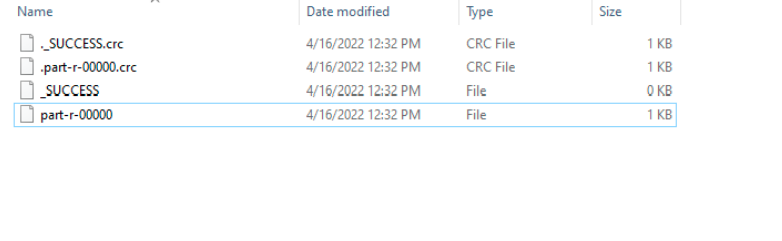
* In that text file, add any random text and save it.



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



* 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.



* Result

