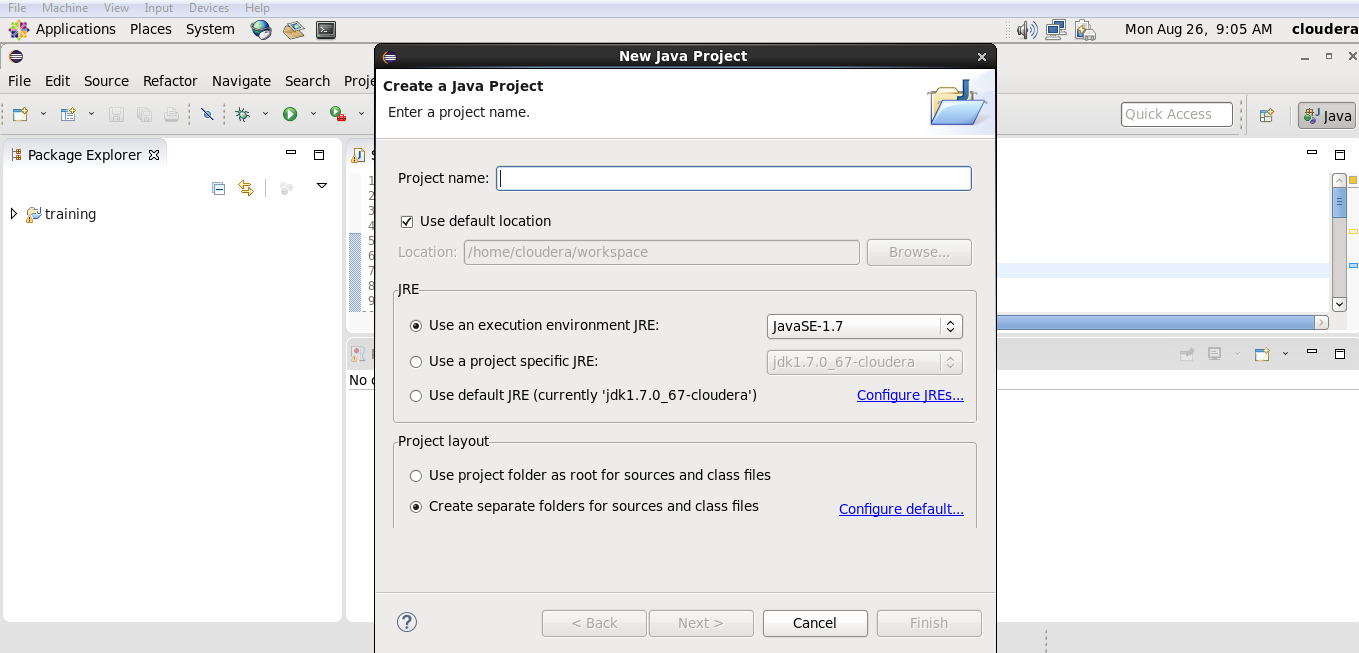
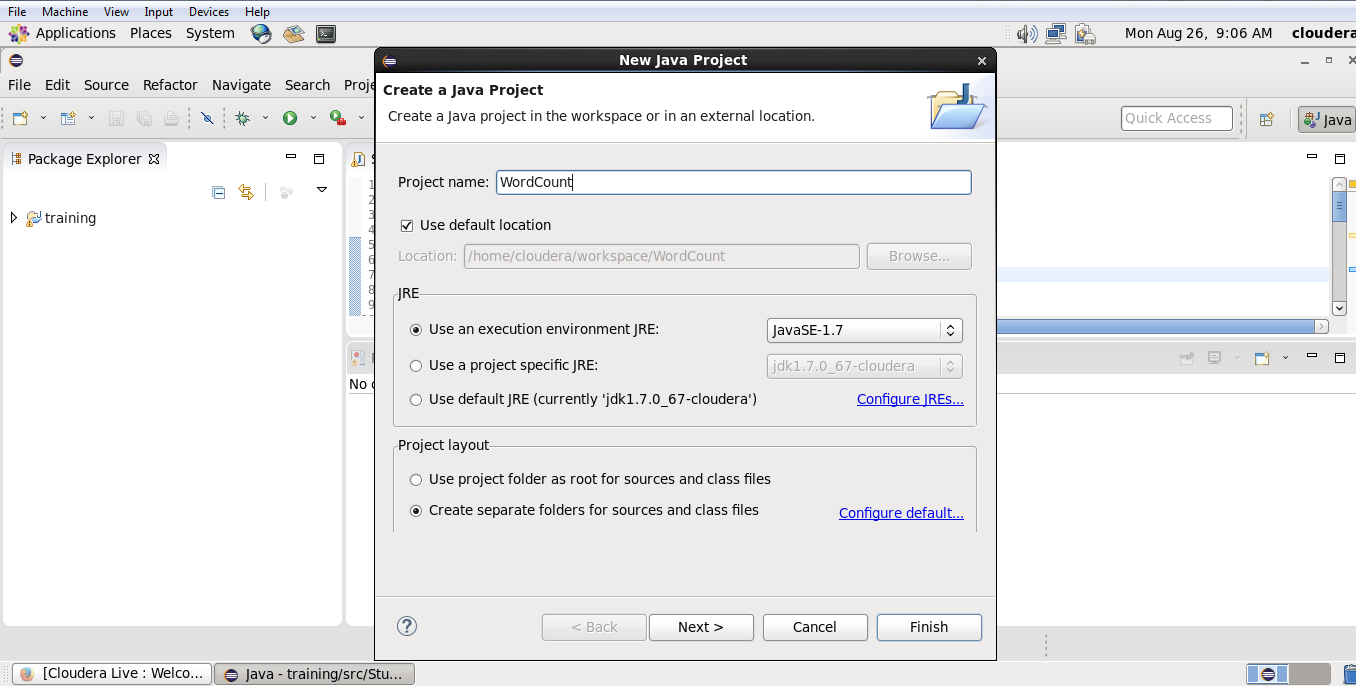
Eclipse WordCount Project:

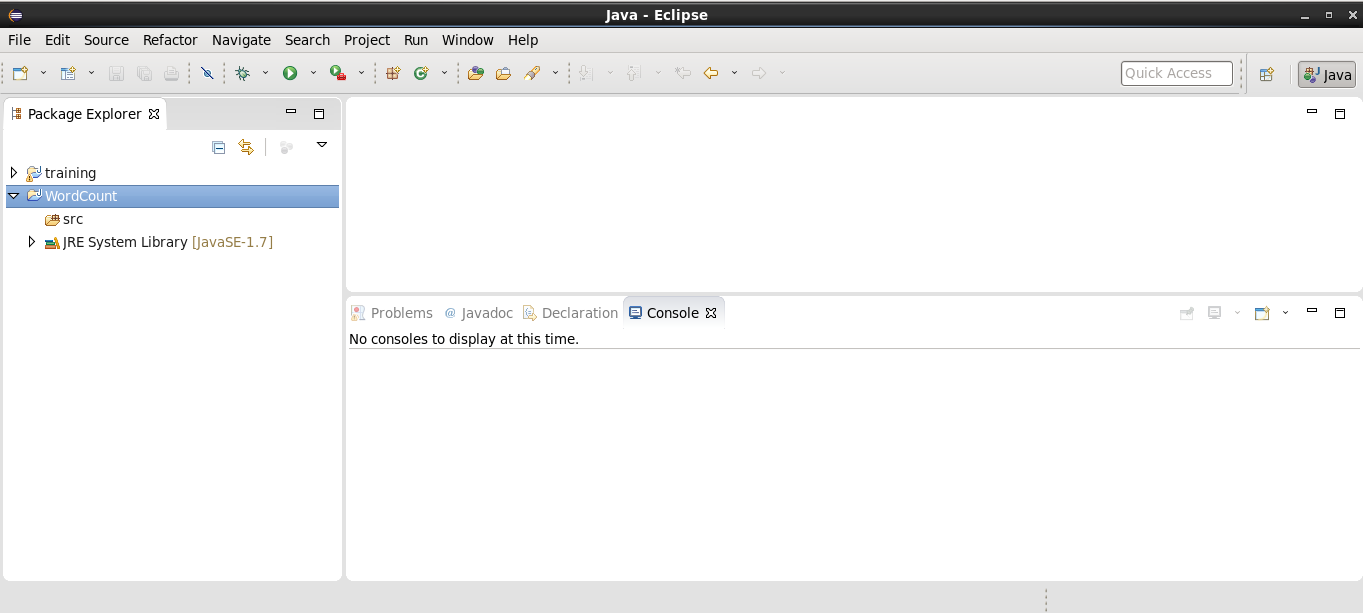
**Step by step instructions:**

File > New > Java Project > Next.



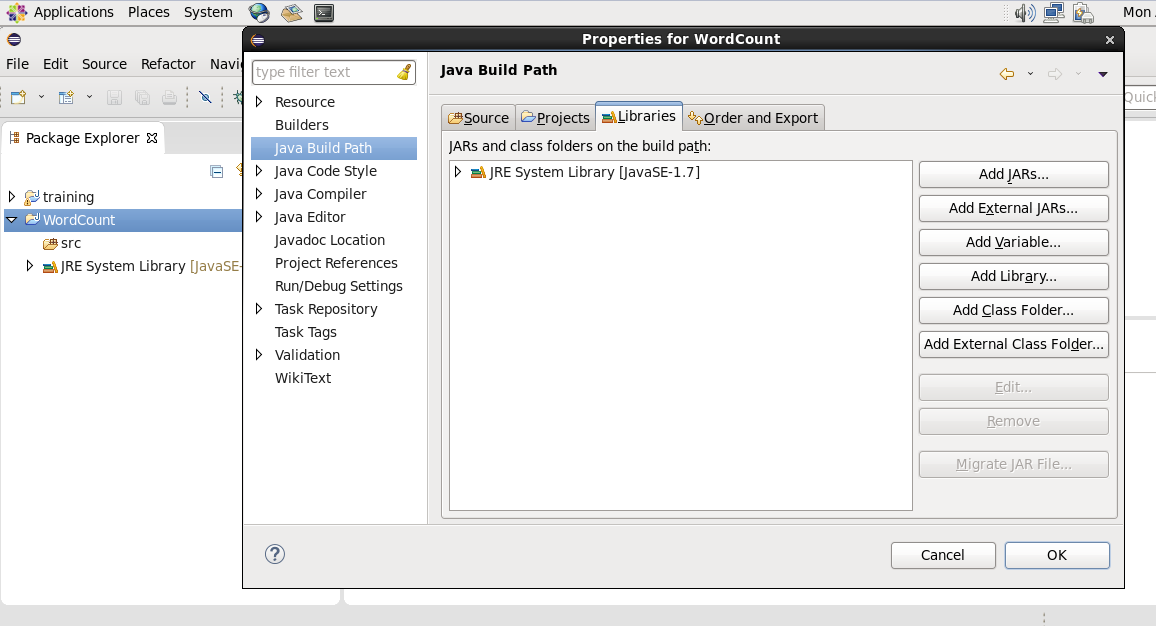
"WordCount" as our project name and click "Finish":



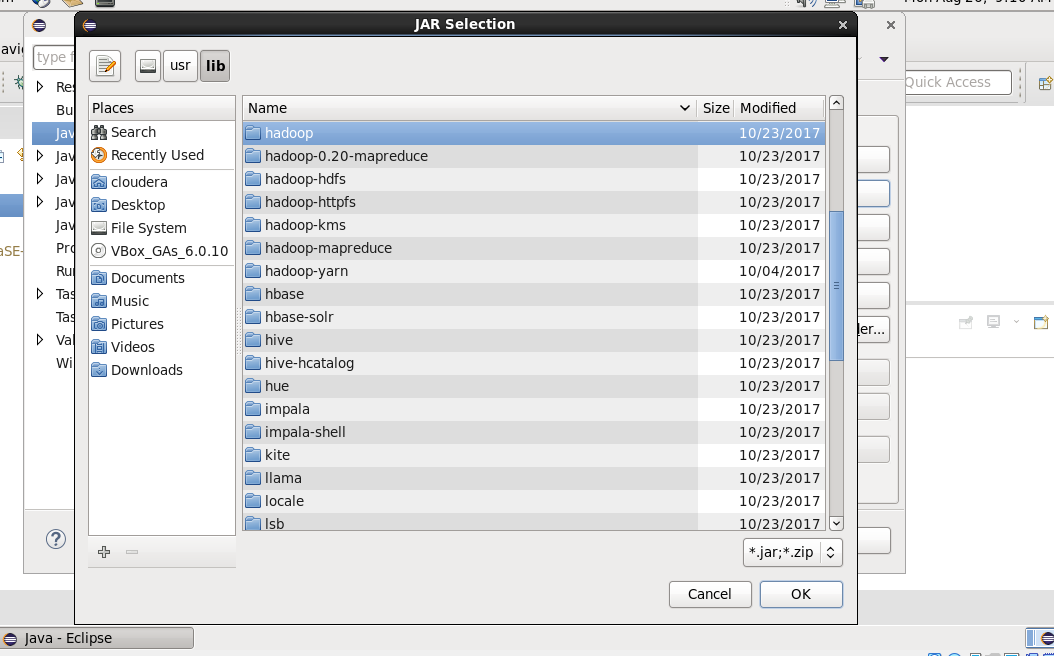


Getting references to hadoop libraries

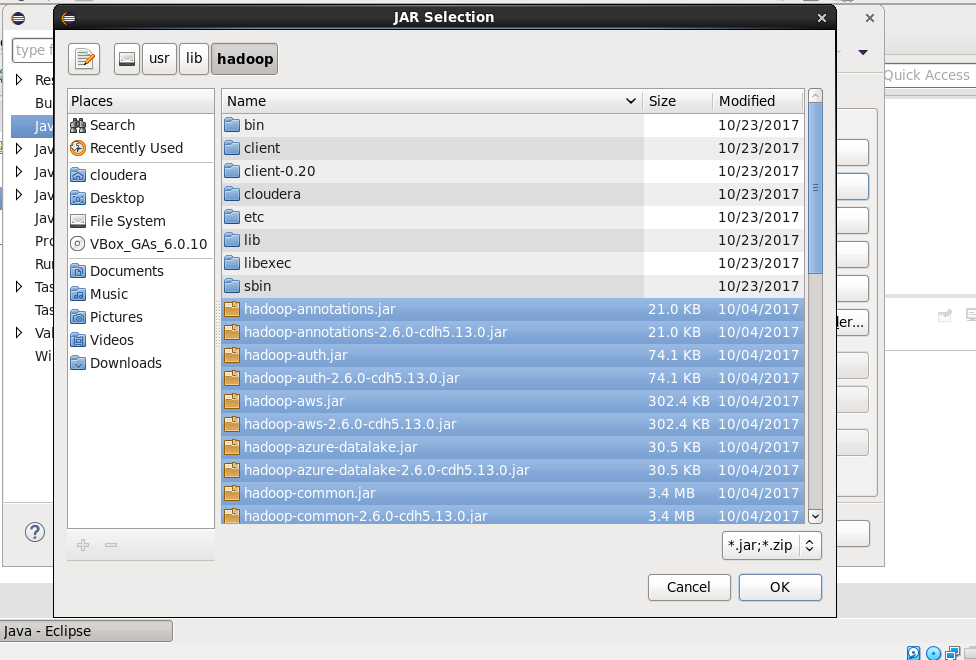
Right click on WordCount project and select "Properties":

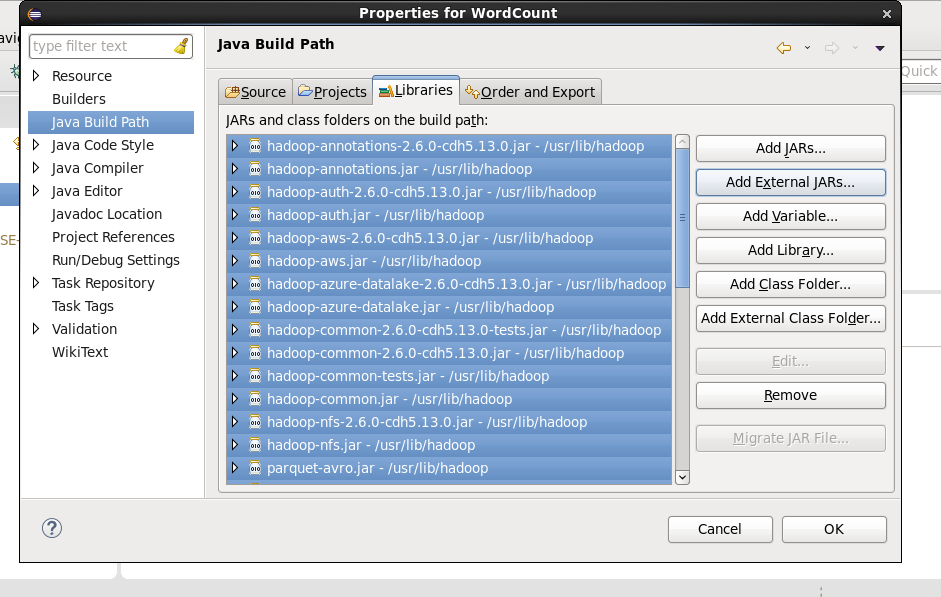


Hit "Add External JARs...", then, File System > usr > lib > hadoop :

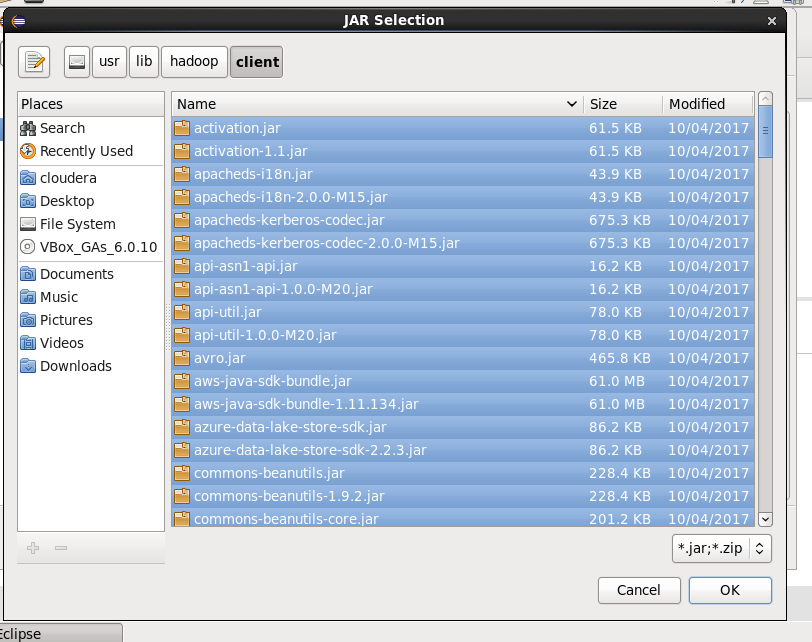


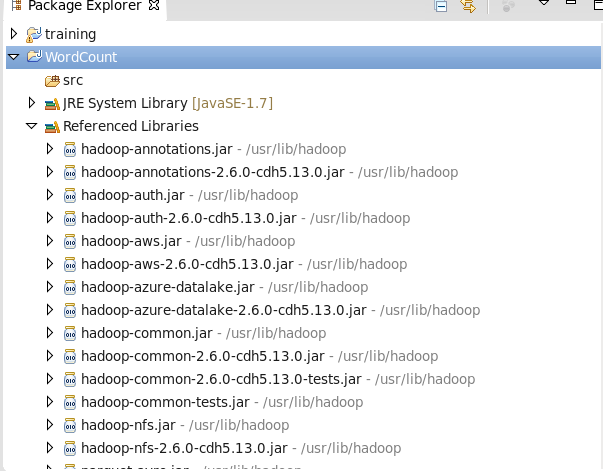
We may want to select all jars, and click OK:





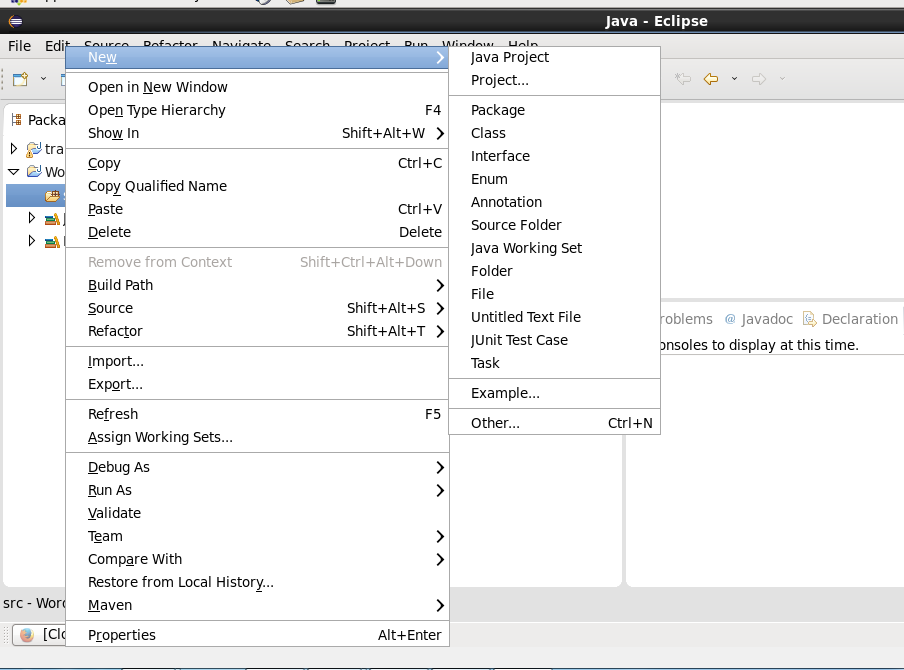
We need to add more external libs. Go to "Add External JARs..." again, then grab all libs in "client": Then, hit "OK"

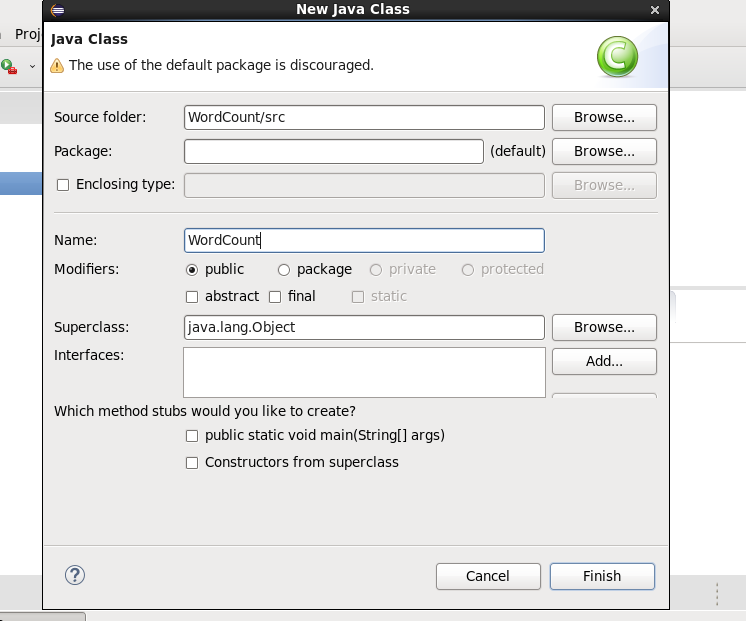




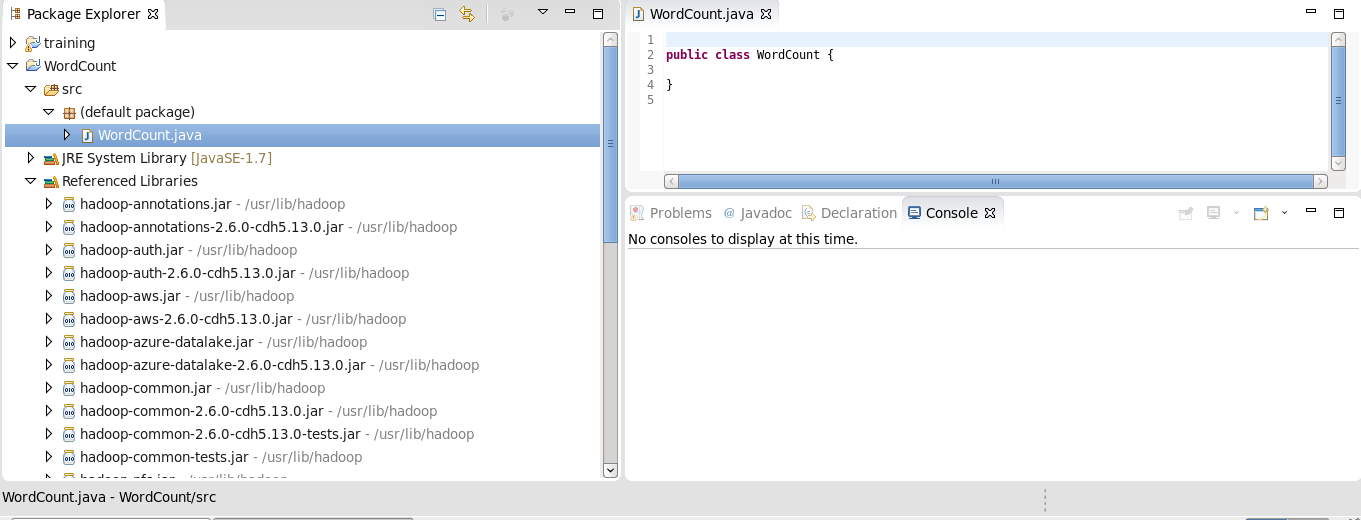
Creating class files

Right click on source, New > Class:



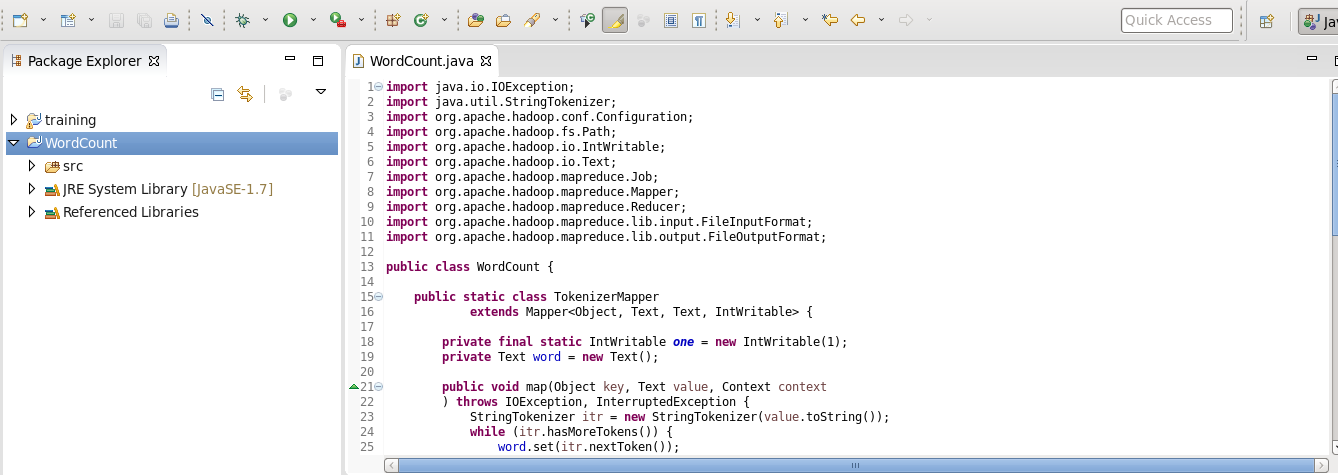


Click "Finish".



Copy and paste wordcount code:

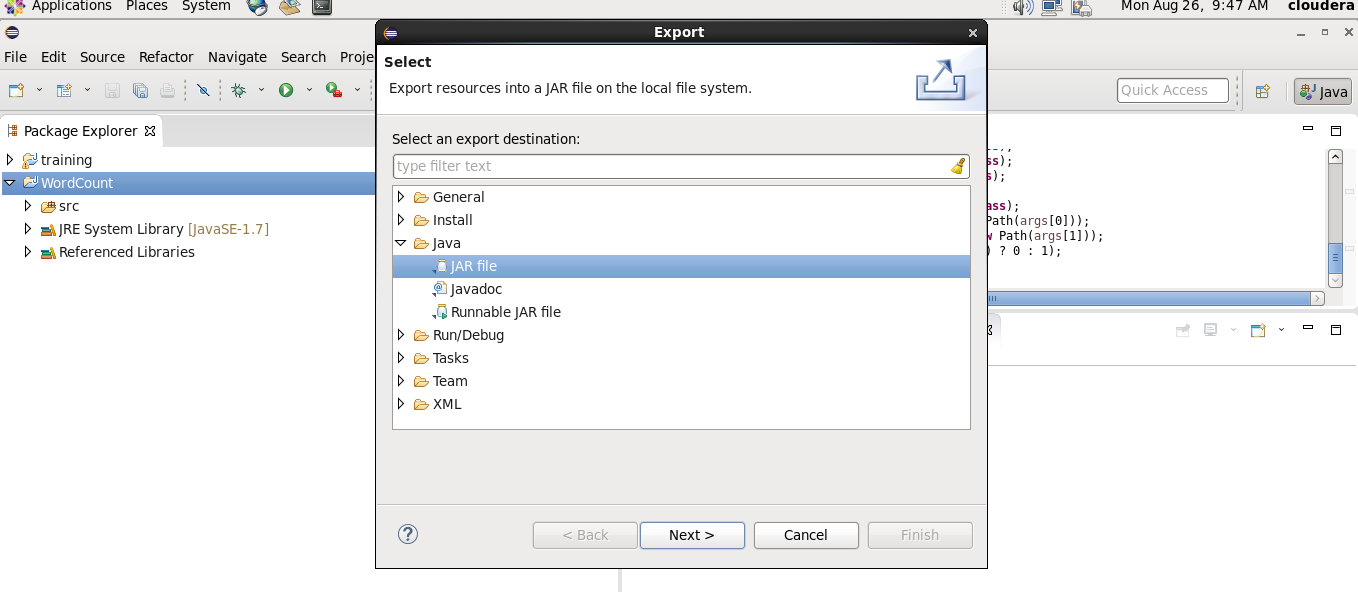
import java.io.IOException;  
import java.util.StringTokenizer;  
import org.apache.hadoop.conf.Configuration;  
import org.apache.hadoop.fs.Path;  
import [org.apache.hadoop.io](http://org.apache.hadoop.io/" \t "_blank).IntWritable;  
import org.apache.hadoop.io.Text;  
import org.apache.hadoop.mapreduce.Job;  
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);  
    }  
}

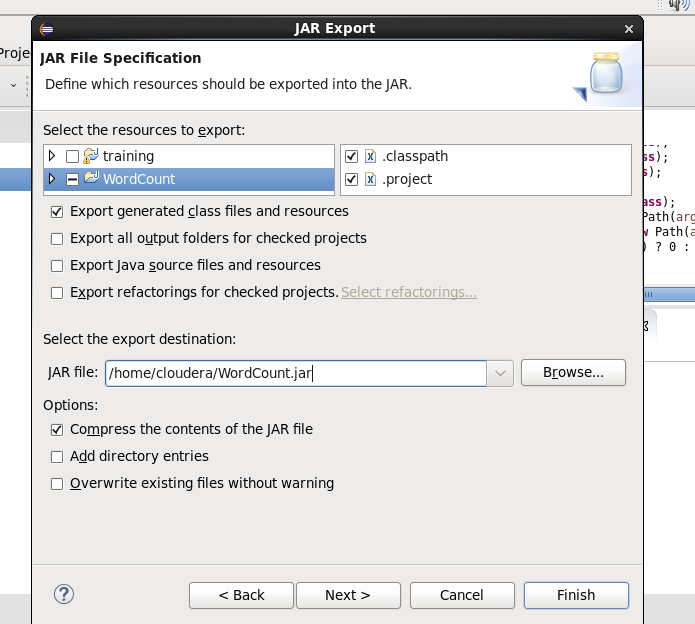


Exporting the Jar

Now we want to export it as a jar.

Right click on WordCount project and select "Export...":





Input file

Here is our input:

[cloudera@quickstart ~]$ cat /home/cloudera/wordcount.txt

Copy paste input text here:

[cloudera@quickstart ~]$ hadoop fs -mkdir input

[cloudera@quickstart ~]$ hadoop fs -put /home/cloudera/wordcount.txt input/

[cloudera@quickstart ~]$ hadoop fs -ls input

Found 1 items

-rw-r--r-- 1 cloudera cloudera input/wordcount.txt

[cloudera@quickstart ~]$ hadoop fs -cat input/wordcount.txt

Your input text

[cloudera@quickstart ~]$

run

Time to run MapReduce job:

$ hadoop jar /home/cloudera/WordCount.jar WordCount input/wordcount.txt output

Output

Here is our output from the MR run:

[cloudera@quickstart ~]$ hadoop fs -ls output

Found 2 items -rw-r--r-- 1 cloudera cloudera output/\_SUCCESS -rw-r--r-- 1 cloudera cloudera output/part-00000

[cloudera@quickstart ~]$ hadoop fs -cat output/part-00000

Issues found:

Connection issues to connect to cloudera

Restart all services using cloudera manager