Development Environment Tips

This is a living document and will be updated as we go. If you have tips, submit them to this post on Piazza.

Eclipse with MapReduce

Setting Up Your Project

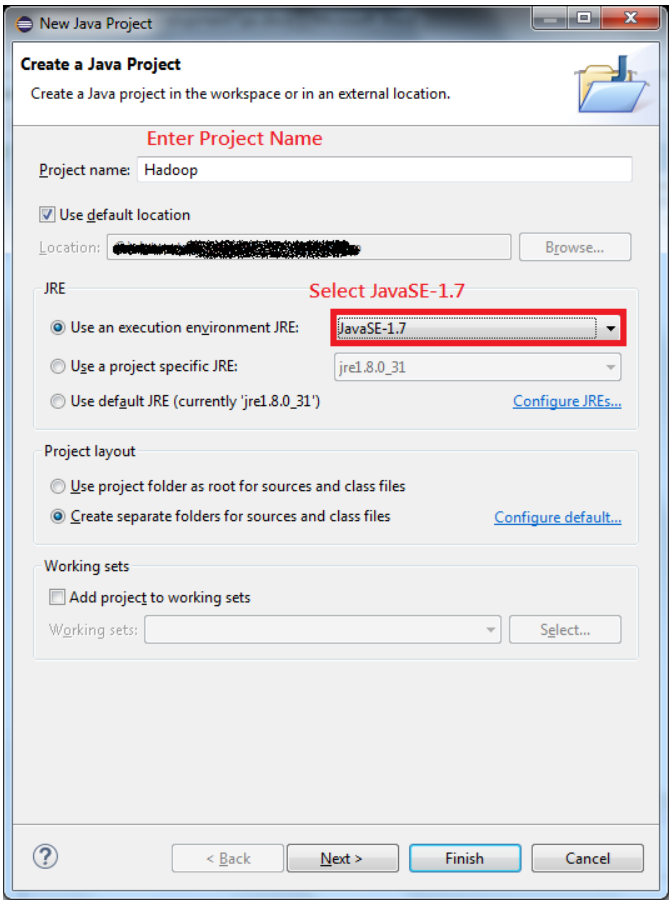
You can write everything in Notepad++ or Sublime if you'd like. However, if you want to use Eclipse without getting annoyed by all the compilation errors, you'll need to add the Hadoop jars to your projects build path.

You can download the Hadoop jars from the following links:

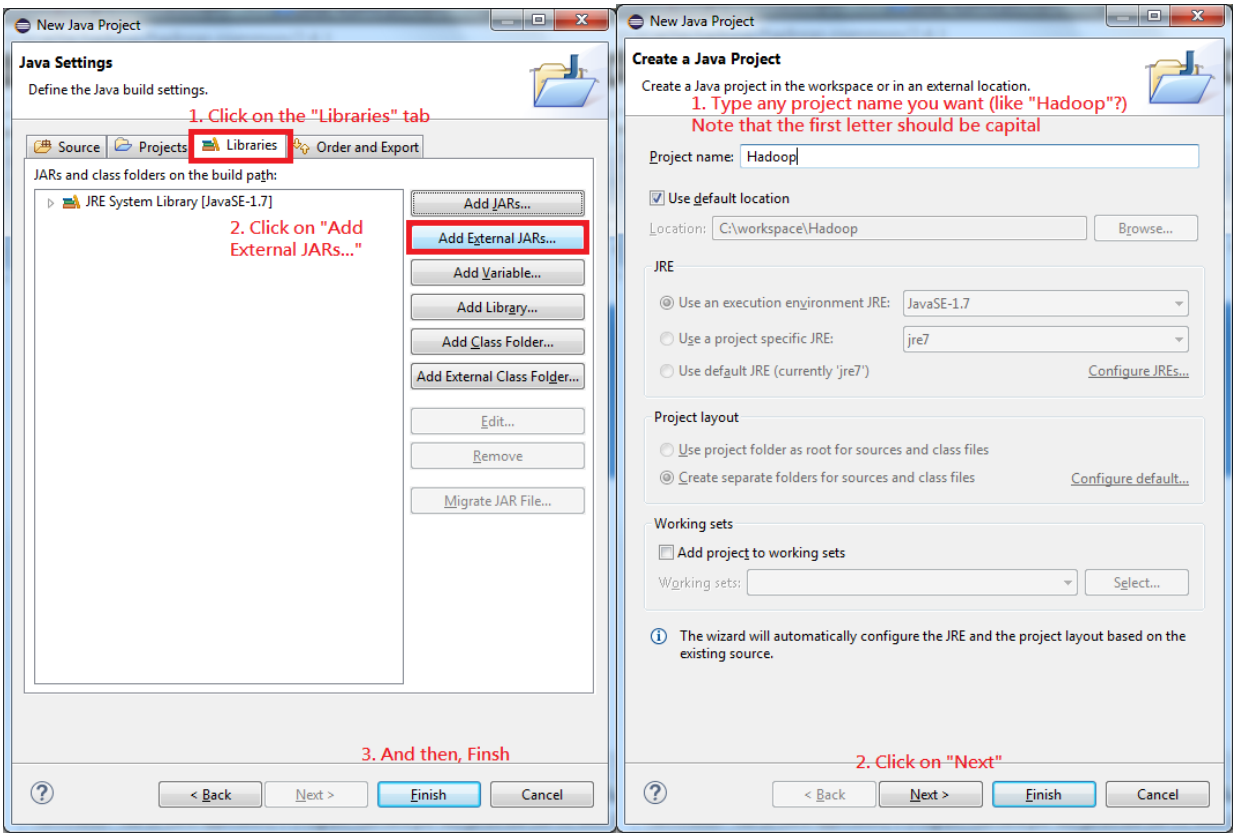
- http://mvnrepository.com/artifact/org.apache.hadoop/hadoop-common/2.4.1
- http://mvnrepository.com/artifact/org.apache.hadoop/hadoop-mapreduce-client-core/2.4.1
- http://mvnrepository.com/artifact/org.apache.hadoop/hadoop-annotations/2.4.1

Put them to somewhere on your U: drive so you can access it later. (U:\Documents\hadoop-2.4.1 might be good?)

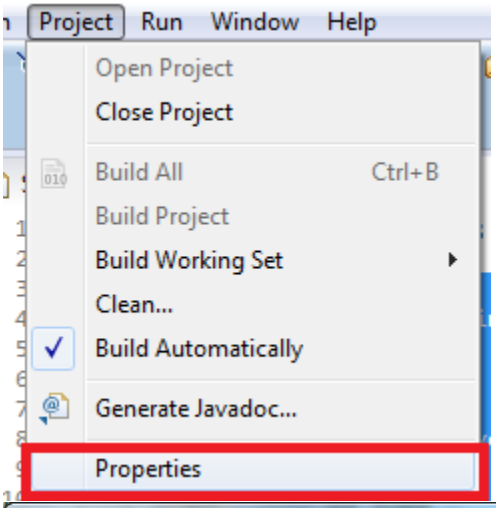
From there, launch Eclipse. Click File -> New -> New Java Project. This will launch the wizard. Give the project a name and select “JavaSE-1.7” as execution environment.

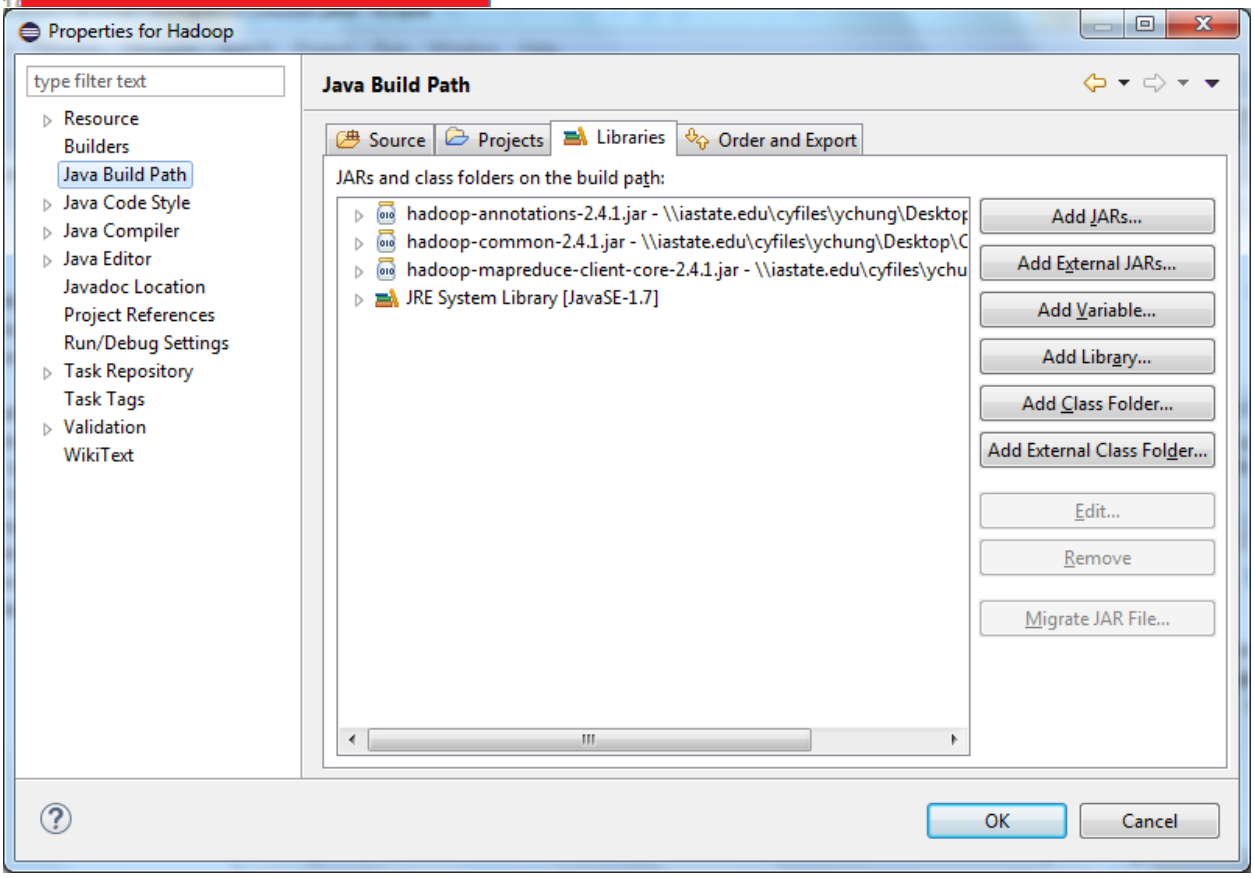


Click NEXT instead of FINISH. Select the Libraries tab and then click “Add External Jars”. Navigate to wherever you extracted the jars “U:\Documents\hadoop-2.4.1” and select all the jars in this folder.



In case that you forget to include these JAR files into your library or that you need to add more JARs into your project in the later labs, you can add more JARs into your project library. Click Project->Properties. In the Properties window, you can find out the Add External JARs button in Java Build Path->Libraries.

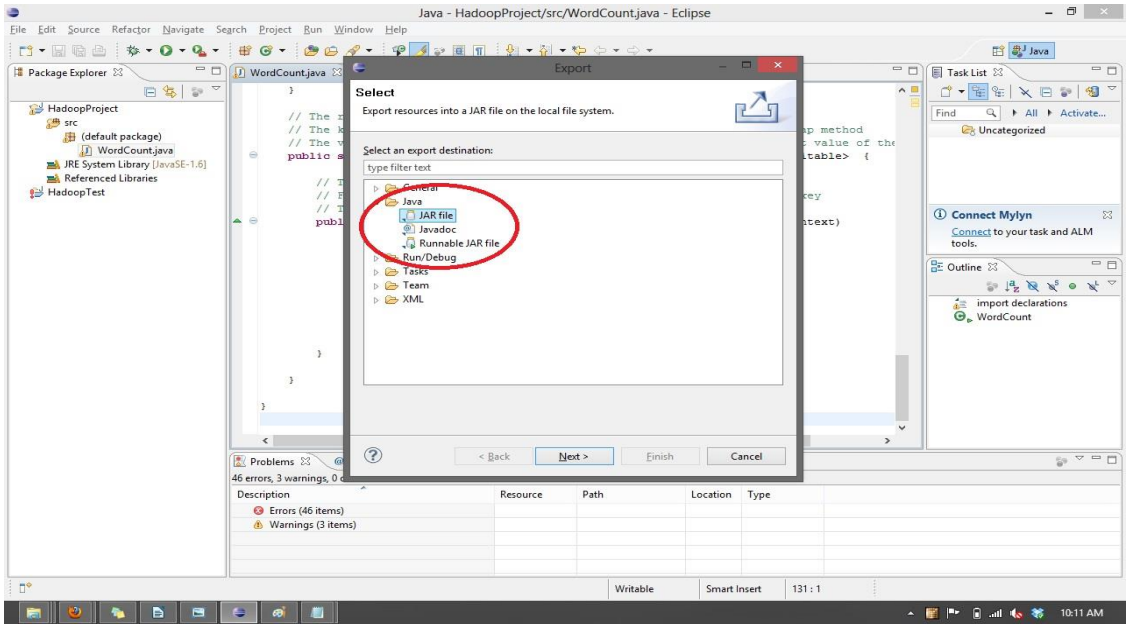


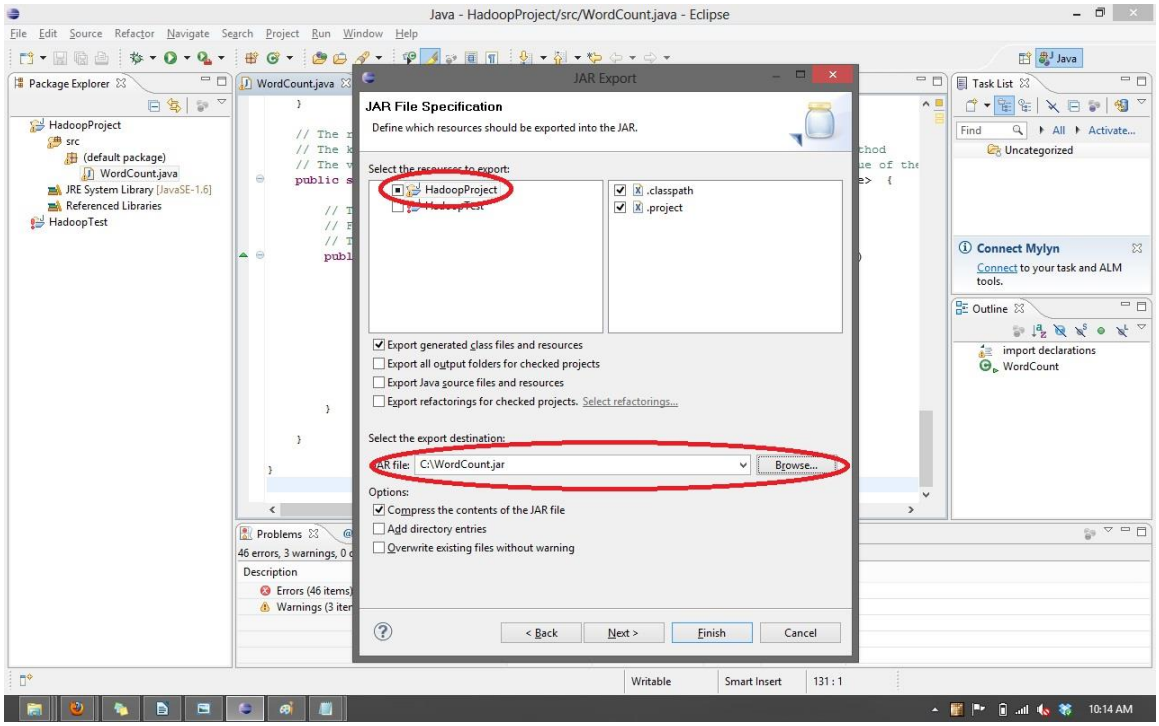


You can now copy and paste the code provided in the lab without compilation errors. You will also be able to explore the API's using the auto-complete features of eclipse (available by pressing ctrl+space).

Exporting to Jar

Select “export” from the file menu. Select Java-> JAR file (NOT Runnable JAR file). Click next and then check the box next to the project you want to export. Save the file somewhere on your U: drive so you can access it later. You will scp this JAR to Cystorm.





Bash Alias

You can reduce the amount you need to type by adding aliases to your .bashrc file. In your CyStorm home directory you can type nano ~/.bashrc to open an editor and add aliases. You may then copy the following below the line # User specific aliases and functions

```
alias hls='hdfs dfs -ls'

alias hmkdir='hdfs dfs -mkdir'

alias hrmr='hdfs dfs -rmr'

alias hc2l='hdfs dfs -copyToLocal'

alias hcfl='hdfs dfs -copyFromLocal'
```

This will map the right-hand side hdfs dfs –ls and hdfs dfs –mkdir to be accessible by shorter commands hls, hmkdir, etc.