

Generating JavaDoc in IntelliJ

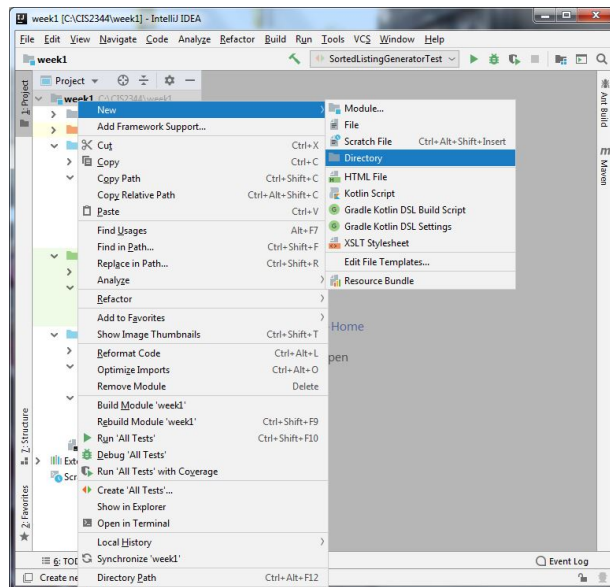
October 9, 2019

One way of checking the documentation of your code is to generate JavaDoc documentation from your documented code. JavaDoc will identify any areas where it feels that your documentation is lacking information.

Note: Even more so than with testing — the fact that JavaDoc reports no problems is *not* a sign that your documentation is perfect (or even good!). However, if JavaDoc *does* report problems, this is almost certainly an indication that your documentation could be improved.

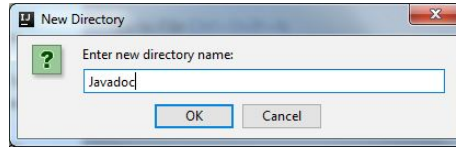
IntelliJ has a tool for generating JavaDoc documentation. This document briefly describes how to use this tool:

- First, you need to create a directory in which to keep your JavaDoc documentation. I normally create a `JavaDoc1` directory in each project:
 - Right click on the project root, and select “New→Directory”

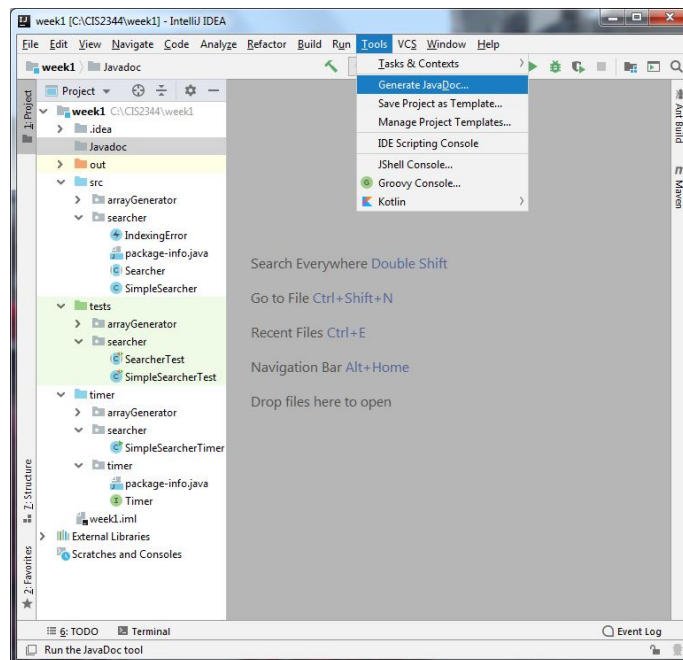


¹I have been doing this for each code bundle, so you should be able to find the JavaDoc for each of them.

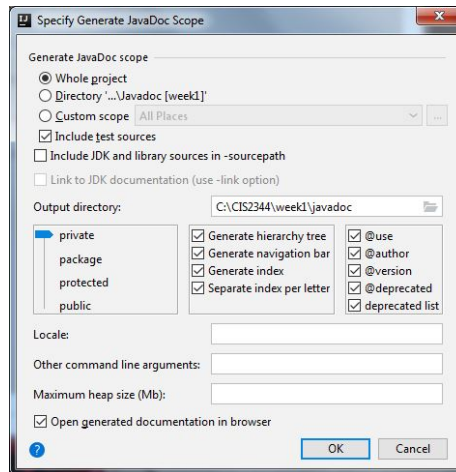
- Name your new directory



- Now you can run Javadoc:
 - Select “Tools→Generate Javadoc...”

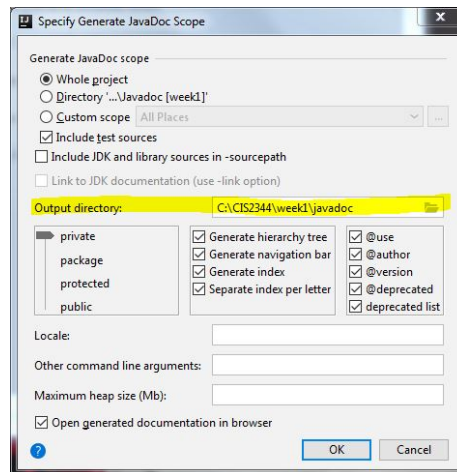


- A pop-up window appears.

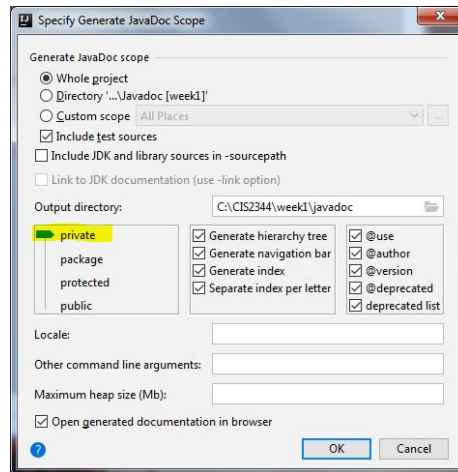


In this:

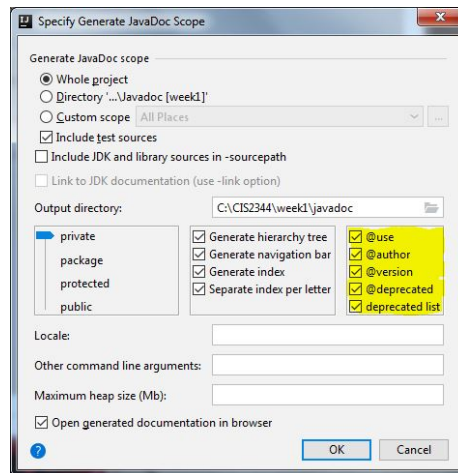
- * Use **Output directory** to specify the Javadoc directory you created above



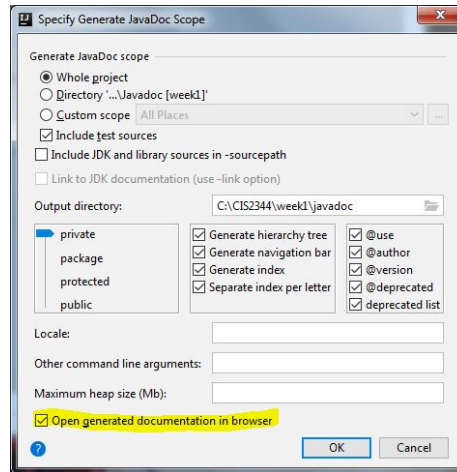
- * You probably want to set the level slider to “private” to ensure that Javadoc covers all of your code



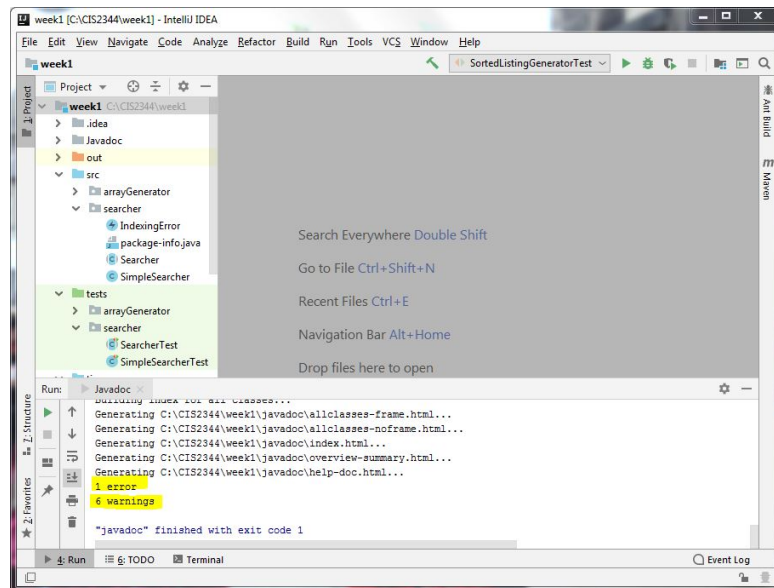
- * You probably also want to check all of the “annotation” check boxes



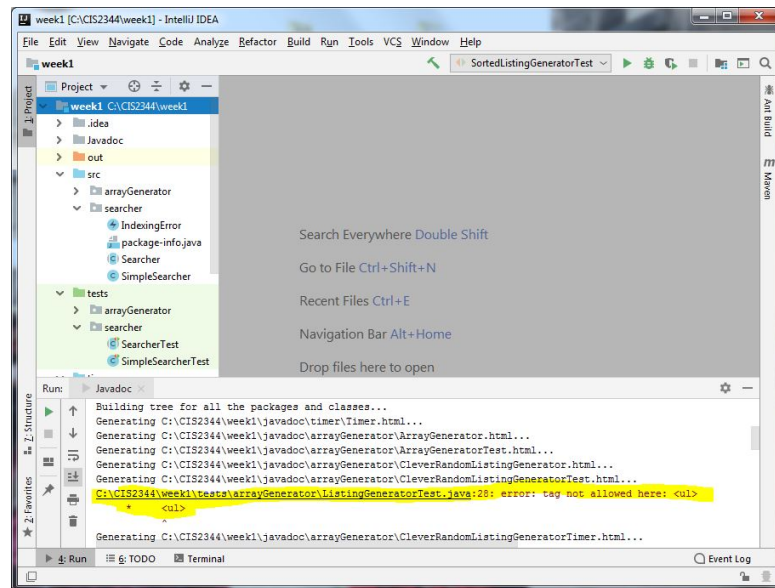
- * Finally, if you want to automatically open a browser to see your generated Javadoc documentation, tick the “Open generated documentation...” box



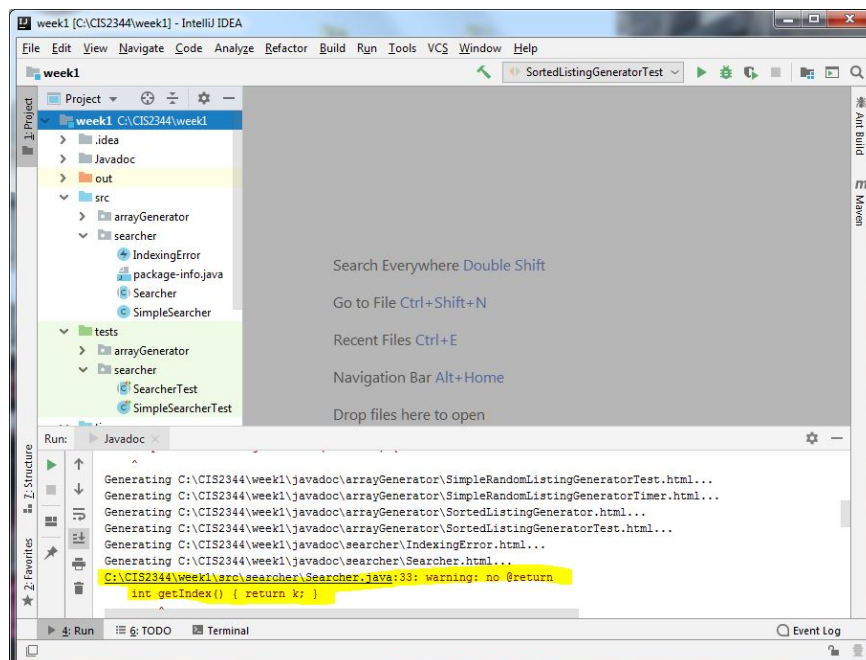
- * You can then click on the “OK” button, and IntelliJ will call Javadoc
- Upon completion, Javadoc should tell you how many errors and warnings it found



- In the Javadoc output you can find more details of the errors

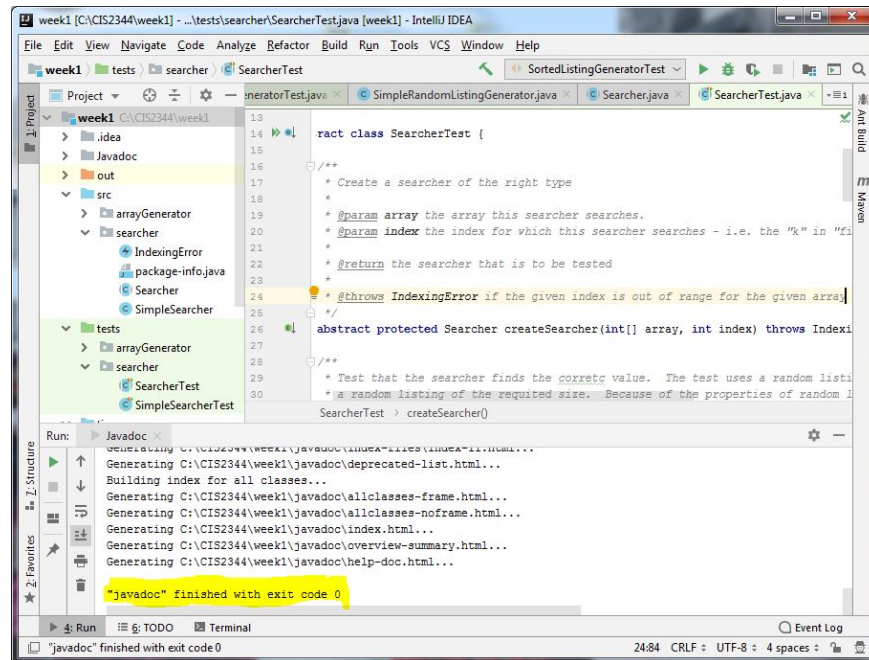


- and warnings



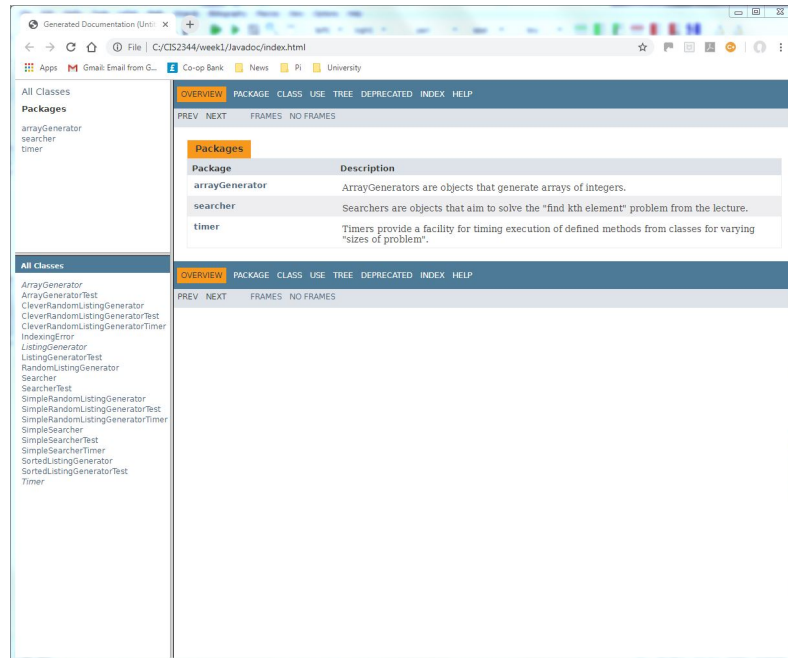
As you can see, there are also links to the relevant parts of your code.

- Once you have fixed the errors and warnings you hope to see



This doesn't necessarily mean that your documentation is good, but at least it is Javadoc compliant.

- If you ticked the “Open generated documentation...” box your JavaDoc documentation should now automatically open in a browser (though it may take a while)



It is a good idea to have a look at this documentation, and consider whether it looks as you would expect a good API to look. If it doesn't, you probably want to adjust your code's documentation, and try again.