Javadoc

Albert Weichselbraun

Javadoc

What’s Javadoc?

A tool which automatically generates a Java-style HTML-documentation from your annotated source code.

Why Javadoc?

- over the lifetime of a project documentation and implementation tend to diverge
- Javadoc offers a *single point* of control
- well documented source code is a requirement for any collaboration (source code is much more often read than written)

Javadoc Basics

- place Javadoc comments between `/**` and `*/`
  - immediately after the class/method name
  - package level comments are available as well
- the first sentence is used as summary (package index, top page)
- write comments in HTML
  - `<code>` for return values, `<ul>` for enumerations, ...)
  - use HTML character entities (`&lt; &gt; &amp;`)
  - avoid `<h1>, <h2>`

- include standard tags to further describe your code
  - `@param, @author, @return, @see`
  - each tag begins a line, followed by a text description
- use `@link` to reference other parts of the documentation
  - classes: `{@link Item}`
  - methods: `{@link Class#Method Description}`
  - rename links: `{@link Manager#freeze freeze}`
- document packages in the file `package-info.java`

Style and Key Issues

- the first sentence should summarize your methods, classes, etc.
- comments *must* provide *additional* insight
- never check in code containing boiler-plate comments (automatically generated Javadoc)
- do not comment bug fixes (SVN comments are a better place for this)
- use the HTML tag `<code>` to indicate return values, Java keywords, package names, class names, etc.

Generate Javadoc

- generate the Javadoc in Eclipse using `Project -> generate Javadoc`
- always check the generated documentation
- please visit the "Oracle Documentation on How to Write Doc Comments for the Javadoc Tool[1]" for more detailed information

[1] `http://www.oracle.com/technetwork/java/javase/documentation/index-137868.html`