

JUnitDoclet: A Simple Open Source Tool to Generate JUnit Test Cases.

Jeffrey Peacock
jeffp@JeffreyPeacock.com



Unit Test Cases:

- ... are extremely useful for both initial functional verification and regression testing.
- ... are an evangelized “Best Practice”.
- ... take time to write and can be tedious.
- ... may need to be refactored when the original code is refactored.

JUnit is a defacto standard for Java.

Tools for Generating JUnit Test Cases

- Proprietary: extensive in features and scope, very high per seat cost (\$1000's)
 - Eclipse: limited test case generation functionality but “understands” JUnit.
 - JUnitDoclet: simple w/ limited features, but extensible and Open Source (\$0's).
-
-

JunitDoclet: Pros

- For each public class: Generates skeletons of `junit.framework.TestCase`
 - For each public method: Generates a test method
 - Combines setter and getter tests.
 - Uses customizable “template” mechanism.
 - Provides support for refactoring.
 - Core functionality utilizes Java Doclet API.
 - Small tool, easily extensible.
-
-

JunitDoclet: Cons

- Is not a panacea.
 - Is not full featured.
 - Active development seems to have “paused”.
 - Some “code wrangling” was required to get the “style” I wanted.
 - Template mechanism is a little cumbersome.
 - Preserving customizations is based on “embedded keywords” in Java comments.
-
-

Cost Analysis

- Did I mention it was free? ...
- ... but as with ALL tools, free or otherwise, there is the cost of learning the tool and adapting it into your development environment.
- <http://www.junitdoclet.org>

