

Doubler Home

Contributed by Jay Flowers
Tuesday, 02 May 2006
Last Updated Monday, 30 October 2006

Description

Doubler is a code generator that makes unit testing easier. It is especially useful when working with legacy code. It is a Reflector add-in, a tool already woven into your workflow. It will help you cleave dependencies apart, create test doubles, and write unit tests with little effort and less coding.

Features

Doubler is a Reflector add-in that helps with creating unit tests. It offers four code generators: Recording Generator – Use against an abstract type. It will create a Recording Test Stub. Recorders generated have the following features for each method on the subject type: Property named Called of type Boolean that records if the method was called. Properties for each parameter passed named in the following format Passed of the same type as the parameter. Property named ReturnValue of the same type as return type of the subject method. If subject method is void this property is not created. When the method is called the this value is returned. Property named ExceptionToThrow of type Exception. When the method is called if this property is not null the property value is thrown. Wrapper/Interface Generator – Use against a concrete type. It will create a mirror interface of the subject type and an implementation of the interface that passes all calls to a private instance of the subject type. This can be used to help control dependencies, allowing unit tests to insert a test double in place of the subject (run the Recording Generator on the mirror interface). Fake Generator - Use against an abstract type. It will create a Fake Object. Test Generator – Use against a concrete type. It will generate a unit test fixture and test methods for each public method on the test subject. The test methods will contain a start on the unit test.

[Subscribe to Doubler.User](#)

Email:

[Browse Archives at groups.google.com](#)