About UTA Actions

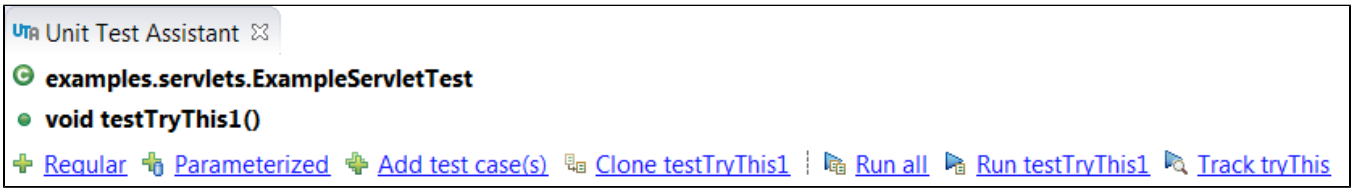
The following interfaces contain UTA action links that perform a range of functions, including creating and running tests, updating code, and navigating resources:

- Unit Test Assistant View
- Context Menu
- Quick Fix Dialog

In addition, actions and quick-fixes associated with factory methods and particular recommendations are displayed in the Factory Methods and Recommendations views (see Configuring Factory Methods and Working with Recommendations for details).

Unit Test Assistant View

The Unit Test Assistant view is the primary interface for creating, running, and interacting with your tests. It is context-sensitive and displays all actions that are available for the current selection in the editor.



Actions that create tests:

- **Regular** - creates a regular test (see Creating a Basic Unit Test).
- **Parameterized** - creates a parameterized test (see Creating a Parameterized Unit Test).
- **Add test case(s)** - creates multiple regular or parameterized tests (see Creating Multiple Unit Tests and Creating a Parameterized Unit Test).
- **Regular Spring** - creates a regular test for a Spring component method (see Creating a Spring Unit Test)
- **Parameterized Spring** -creates a parameterized test for a Spring component method (see Creating a Spring Unit Test)
- **Regular private** - creates a regular test for a private method (see Creating a Basic Unit Test).
- **Parameterized private** - creates a parameterized test for a private method (see Creating a Parameterized Unit Test).
- **Regular Spring private** - creates a regular test for a private Spring component method (see Creating a Spring Unit Test)
- **Parameterized Spring private** - creates a parameterized test for a private Spring component method (see Creating a Spring Unit Test)
- **Clone test[method name]** - duplicates an existing test (see Creating a Basic Unit Test)
- **Cover the line** - creates a test to cover a line selected in the editor (see Covering a Selected Line)

Actions that run tests:

- **Run test[method_name]** - runs the test for the selected method (see Executing Unit Tests with Unit Test Assistant).
- **Run all** - runs all tests in the method's class (see Executing Unit Tests with Unit Test Assistant).
- **Track [method_name**] - runs the test and displays information about the values that changed during execution (see Tracking Object Changes and Creating Assertions).

Actions that update the tests code:

- **Track and make assertions** - runs the test and automatically makes assertions (see Tracking Object Changes and Creating Assertions).
- **Mock** - creates mocks (see Creating Mocks).
- **Instantiate** - instantiates objects.
- **Instantiate using factory** - instantiates objects using factory methods that are tagged in the Javadoc (see Configuring Factory Methods)
- **Tag factory method** - adds the @jtest.factory tag to the Javadoc comment for the method (see Configuring Factory Methods)
- **Untag factory method** - removes the @jtest.factory tag from the Javadoc comment for the method (see Configuring Factory Methods)

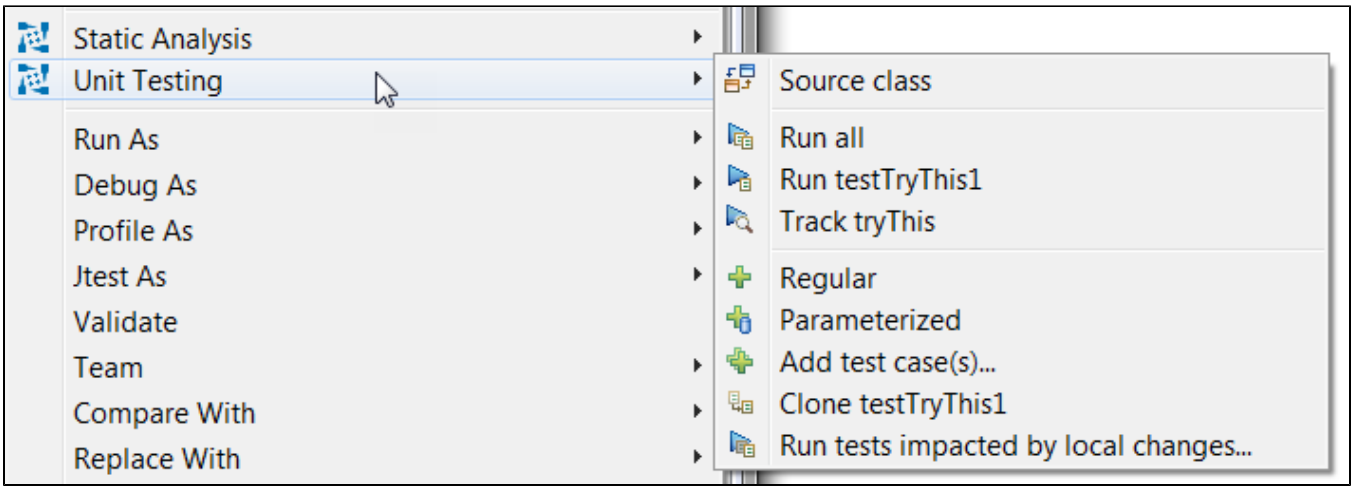
Actions that navigate between the test and the source:

- **Source class** - navigates to the source class.
- **Test class** - navigates to the test class.

Context Menu

The actions that are displayed in the Unit Test Assistant view are also available in the context menu. The context menu may also include an additional option for change-based testing (see Test Impact Analysis).

Right-click a class or method in the editor and choose **Unit Testing** to view the available actions.



To display options available for entire packages and projects, right-click a selected package or project in the Package Explorer or Project Explorer.

Quick Fix Dialog

You can select the options that help you run tests and modify your code in the context assist dialog, which opens with a keyboard shortcut.

Press **Ctrl + 1** (in Eclipse) or **Alt + Enter** (in IntelliJ) to see the options available for the code at the current cursor position: