

# JUnit Sampler Tutorial

This tutorial attempts to explain the basic design, functionality and usage of the new JUnit Sampler for Jmeter. The sampler is currently in Jmeter's CVS and is not part of Jmeter 2.1 release candidate.

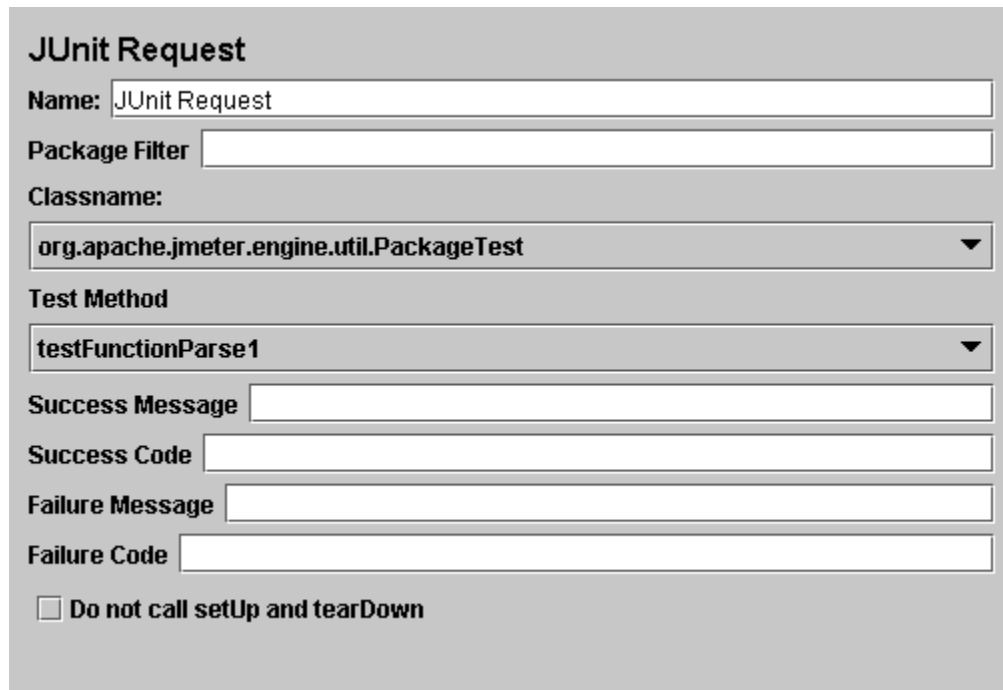
## Design

The current implementation only supports standard JUnit convention and does not support other extensions, like `oneTimeSetUp`. The feature can be added with a few hours of work. The sampler works like the `JavaSampler` with some differences.

1. rather than use Jmeter's test interface, it scans the jar files for classes extending junit's `TestCase` class.
2. JUnit test jar files are copied to `jmeter/lib/junit` instead of `jmeter/lib`
3. JUnit sampler does not use name/value pairs for configuration. The sampler assumes `setUp` and `tearDown` will configure the test correctly.
4. The sampler measures the elapsed time only for the test method and does not include `setUp` and `tearDown`.
5. Each time the test method is called, Jmeter will pass the result to the listeners.

## Functionality

Here is a description of the functionality.



The image shows a configuration window titled "JUnit Request". It contains several input fields and dropdown menus for configuring a JUnit test sampler. The fields are: "Name" (set to "JUnit Request"), "Package Filter" (empty), "Classname" (dropdown menu showing "org.apache.jmeter.engine.util.PackageTest"), "Test Method" (dropdown menu showing "testFunctionParse1"), "Success Message" (empty), "Success Code" (empty), "Failure Message" (empty), and "Failure Code" (empty). At the bottom, there is a checkbox labeled "Do not call setUp and tearDown" which is currently unchecked.

JUnit Request	
Name:	JUnit Request
Package Filter	
Classname:	org.apache.jmeter.engine.util.PackageTest
Test Method	testFunctionParse1
Success Message	
Success Code	
Failure Message	
Failure Code	
<input type="checkbox"/> Do not call setUp and tearDown	

**Name** – name for the sample. This is the same as all jmeter samplers.

**Package Filter** – provides a way to filter the classes by package name.

**Classname** – the name of the class to test. The sampler will scan the jar files in jmeter/lib/ext and jmeter/lib/junit for classes extending junit's TestCase.

**Test Method** – the name of the method to test in the sampler.

**Success message** – a descriptive message indicating what success means.

**Success code** – an unique code indicating the test was successful.

**Failure message** – a descriptive message indicating what failure means.

**Failure code** – an unique code indicating the test failed

**Do not call setUp and tearDown** – set the sampler not to call setUp and tearDown. By default, setUp and tearDown should be called. Not calling those methods could affect the test and make it inaccurate. This option should be used with caution.

## Usage

Here is a short step-by-step.

1. write your junit test and jar the classes
2. copy and paste the jar files into jmeter/lib/junit directory
3. start jmeter
4. select “test plan”
5. right click add -> thread group
6. select “thread group”
7. right click add -> sampler -> junit request
8. enter “my unit test” in the name
9. enter the package of your junit test
10. select the class you want to test
11. select a method to test
12. enter “test successful” in success message
13. enter “1000” in success code
14. enter “test failed” in failure message
15. enter “0001” in failure code
16. select the thread group
17. right click add -> listener -> view results tree