JUnit 5 Release Notes

Stefan Bechtold, Sam Brannen, Johannes Link, Matthias Merdes, Marc Philipp, Christian Stein

Version 5.4.0-SNAPSHOT

Table of Contents

This document contains the *change log* for all JUnit 5 releases since 5.3 GA.

Please refer to the User Guide for comprehensive reference documentation for programmers writing tests, extension authors, and engine authors as well as build tool and IDE vendors.

5.4.0-M1

Date of Release:

Scope:

For a complete list of all *closed* issues and pull requests for this release, consult the 5.4.0-M1 milestone page in the JUnit repository on GitHub.

JUnit Platform

Bug Fixes

-

Deprecations and Breaking Changes

-

New Features and Improvements

- JUnit 4's `AssumptionViolatedException` is now supported in JUnit Jupiter for aborting a test mid-flight due to a failed assumption — for example, via JUnit 4's `org.junit.Assume` utility class.
- In addition to returning streams, `@TestFactory`-annotated methods may return a single `DynamicNode`, i.e. a `DynamicTest` or a `DynamicContainer`, instance.
- New `DisplayNameGenerator` interface and `@DisplayNameGeneration` annotation that allow declarative configuration of a pre-defined or custom display name generator.

JUnit Jupiter

Bug Fixes

- `Assertions.assertAll()` is now thread-safe, i.e. it can be used with *parallel* `Streams`.

Deprecations and Breaking Changes

-

New Features and Improvements

-

JUnit Vintage

Bug Fixes

-

Deprecations and Breaking Changes

-

New Features and Improvements

-

5.3.1

Date of Release: September 11, 2018

Scope: Bug fixes since 5.3.0

For a complete list of all *closed* issues and pull requests for this release, consult the 5.3.1 milestone page in the JUnit repository on GitHub.

JUnit Platform

Bug Fixes

- An `OutOfMemoryError` regression introduced in JUnit 5.3.0 has been fixed.
    - Specifically, the `NodeTestTask` used by implementations of `HierarchicalTestEngine` (such as the Jupiter and Vintage test engines) no longer retains references to contextual state after a node has completed execution. This allows state such as instances of test classes to be properly garbage collected by the JVM.
    - Previously, a `NodeTestTask` instance was created for each `TestDescriptor` before starting execution. Now they are created on the fly and can be garbage collected by the JVM after the enclosing container has finished.
- The OpenTest4J dependency has been updated to 1.1.1 to fix a serialization incompatibility between 1.0.0 and 1.1.0 that caused failure messages to be discarded when used from Gradle and potentially other tools and IDEs.

JUnit Jupiter

Bug Fixes

- Invocations of `assertThrows()` that are passed a method reference for an overloaded method with a `void` return type once again compile.

    - For example, given an instance of `java.lang.Object` stored in a variable named `object`, `assertThrows(Exception.class, object::wait)` compiled against JUnit 5.2.0, failed to compile against JUnit 5.3.0, but now compiles against JUnit 5.3.1.

Breaking Changes

- In order to revert the aforementioned breaking change, variants of `assertThrows()` introduced in JUnit 5.3.0 that accept `ThrowingSupplier` arguments have been removed.

JUnit Vintage

No changes

5.3.0

Date of Release: September 3, 2018

Scope: Parallel test execution, output capture for `System.out` and `System.err`, new `TestInstanceFactory` extension API, custom test sources for dynamic tests, promotion of the dynamic test API from *experimental* to *maintained* status, discontinuation of the `junit-platform-gradle-plugin`, deprecation of the `junit-platform-surefire-provider`, as well as various minor improvements and bug fixes.

For complete details consult the 5.3.0 Release Notes online.