# JUnit 5 Release Notes

# Table of Contents

This document contains the *change log* for all JUnit 5 releases since 5.5 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.

# 5.6.0-M1

**Date of Release:**

**Scope:**

For a complete list of all *closed* issues and pull requests for this release, consult the 5.6 M1 milestone page in the JUnit repository on GitHub.

# JUnit Platform

## Bug Fixes

- Module `org.junit.platform.launcher` now reads `java.logging` due to usage of types in package `java.util.logging`.

## Deprecations and Breaking Changes

- In the `EngineTestKit` API, the `all()`, `containers()`, and `tests()` methods in `EngineExecutionResults`

have been deprecated in favor of the new `allEvents()`, `containerEvents()`, and `testEvents()` methods, respectively. The deprecated methods will be removed in JUnit Platform 1.7.0.

## New Features and Improvements

- New `printFailuresTo(PrintWriter, int)` method in `TestExecutionSummary` that allows one to specify the maximum number of lines to print for exception stack traces.

- The `junit-platform-commons` module no longer has a dependency on the `java.compiler` module (in terms of the Java Module System). Specifically, a new internal utility has been introduced in `PackageUtils` that implements functionality equivalent to `javax.lang.model.SourceVersion.isName(CharSequence)` from the `java.compiler` module.

- Exceptions thrown by test engines during discovery and execution are now reported to `TestExecutionListeners`.

- Module `org.junit.platform.console` now provides a `java.util.spi.ToolProvider` implementation that can be acquired by `ToolProvider.findFirst("junit")` when running on Java 9 or above.

- New methods in `DiscoverySelectors` to select and execute individual tests in inherited nested classes, via specific selectors (`NestedClassSelector` and `NestedMethodSelector`).

# JUnit Jupiter

## Bug Fixes

-

## Deprecations and Breaking Changes

- `@EnabledIf` and `@DisabledIf` have been removed from Jupiter's API. Script-based condition APIs and their supporting implementations were deprecated in JUnit Jupiter 5.5 with the intent to remove them in JUnit Jupiter 5.6. Users must now rely on a combination of other built-in conditions or create and use a custom implementation of `ExecutionCondition` to evaluate the same conditions.

## New Features and Improvements

- Support for multi-character delimiters in `@CsvSource` and `@CsvFileSource`.

- Support for custom `null` values in `@CsvSource` and `@CsvFileSource`.

- Documented support for comments in CSV files loaded via `@CsvFileSource`.

- Auto-detection of enum type from method signature for `@EnumSource`.

- New `TypeBasedParameterResolver<T>` abstract base class that serves as a generic adapter for the `ParameterResolver` API and simplifies the implementation of a custom resolver that supports parameters of a specific type.

- New configuration parameter `junit.jupiter.execution.timeout.mode` to control whether timeouts are applied to tests or not. Supported values are: `enabled`, `disabled`, and `disabled_on_debug`.

- New `@EnabledForJreRange` and `@DisabledForJreRange` annotations for easier test execution handling on multiple versions of JRE.

# JUnit Vintage

## Bug Fixes

- JUnit 3 suites with duplicate test names are now reported correctly.

## Deprecations and Breaking Changes

- 

## New Features and Improvements

- Improve performance for projects with a large number of tests.
- Improve performance for test classes with a large number of methods.

# 5.5.2

**Date of Release:** September 8, 2019

**Scope:** Bug fixes since 5.5.1

For a complete list of all *closed* issues and pull requests for this release, consult the 5.5.2 milestone page in the JUnit repository on GitHub.

# Overall Improvements

- Published artifacts have been fixed regarding module descriptors.
  - Binary JAR files contain `module-info.class`.
  - Source JAR files contain `module-info.java`.
  - Javadoc JAR files contain neither `module-info.class` nor `module-info.java`.

# JUnit Platform

No changes.

# JUnit Jupiter

## Bug Fixes

- The `JupiterTestEngine` no longer crashes without executing any tests if JUnit 4 is on the classpath but Hamcrest is not. Specifically, initialization of the `OpenTest4JAndJUnit4AwareThrowableCollector` class no longer fails if the

`org.junit.internal.AssumptionViolatedException` class cannot be loaded from the classpath due to a missing Hamcrest dependency.

## JUnit Vintage

No changes.

# 5.5.1

**Date of Release:** July 20, 2019

**Scope:** Bug fixes since 5.5.0

For a complete list of all *closed* issues and pull requests for this release, consult the 5.5.1 milestone page in the JUnit repository on GitHub.

## JUnit Platform

No changes.

## JUnit Jupiter

### Bug Fixes

- Fix test discovery and execution of inherited `@Nested` classes.

## JUnit Vintage

No changes.

# 5.5.0

**Date of Release:** June 30, 2019

**Scope:**

- Declarative `@Timeout` support
- New `InvocationInterceptor` extension API
- New `LifecycleMethodExecutionExceptionHandler` extension API
- Deprecation of script-based conditions (`@EnabledIf` and `@DisabledIf`)
- Configurable test discovery implementation for `TestEngine` authors
- Explicit Java module descriptors
- Various minor improvements and bug fixes

For complete details consult the 5.5.0 Release Notes online.