4.10.1

* Docs  
  [User Manual](http://docs.google.com/userguide/userguide.html)  
  [Guides and Tutorials](https://guides.gradle.org)  
  [DSL Reference](http://docs.google.com/dsl/)  
  [Javadoc](http://docs.google.com/javadoc/)  
  [Release Notes](http://docs.google.com/release-notes.html)
* [Forums](https://discuss.gradle.org/)
* [Training](https://gradle.org/training/)
* [Try Gradle Enterprise](https://gradle.com/enterprise)
* [User Manual Home](http://docs.google.com/userguide/userguide.html)
* [DSL Reference Home](http://docs.google.com/dsl/)
* [Release Notes](http://docs.google.com/release-notes.html)
  + [Properties](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N27763)
    - [jacoco plugin](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N2787A)
  + [Methods](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N2789E)
  + [Script blocks](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N27A95)
    - [jacoco plugin](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N27AB9)

### Build script blocks

* [allprojects { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:allprojects(groovy.lang.Closure))
* [artifacts { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:artifacts(groovy.lang.Closure))
* [buildscript { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:buildscript(groovy.lang.Closure))
* [configurations { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:configurations(groovy.lang.Closure))
* [dependencies { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:dependencies(groovy.lang.Closure))
* [repositories { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:repositories(groovy.lang.Closure))
* [sourceSets { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:sourceSets(groovy.lang.Closure))
* [subprojects { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:subprojects(groovy.lang.Closure))
* [publishing { }](http://docs.google.com/org.gradle.api.Project.html#org.gradle.api.Project:publishing(groovy.lang.Closure))

### Core types

* [Project](http://docs.google.com/org.gradle.api.Project.html)
* [Task](http://docs.google.com/org.gradle.api.Task.html)
* [Gradle](http://docs.google.com/org.gradle.api.invocation.Gradle.html)
* [Settings](http://docs.google.com/org.gradle.api.initialization.Settings.html)
* [IncludedBuild](http://docs.google.com/org.gradle.api.initialization.IncludedBuild.html)
* [Script](http://docs.google.com/org.gradle.api.Script.html)
* [JavaToolChain](http://docs.google.com/org.gradle.jvm.toolchain.JavaToolChain.html)
* [SourceSet](http://docs.google.com/org.gradle.api.tasks.SourceSet.html)
* [SourceSetOutput](http://docs.google.com/org.gradle.api.tasks.SourceSetOutput.html)
* [SourceDirectorySet](http://docs.google.com/org.gradle.api.file.SourceDirectorySet.html)
* [IncrementalTaskInputs](http://docs.google.com/org.gradle.api.tasks.incremental.IncrementalTaskInputs.html)
* [Configuration](http://docs.google.com/org.gradle.api.artifacts.Configuration.html)
* [ResolutionStrategy](http://docs.google.com/org.gradle.api.artifacts.ResolutionStrategy.html)
* [ArtifactResolutionQuery](http://docs.google.com/org.gradle.api.artifacts.query.ArtifactResolutionQuery.html)
* [ComponentSelection](http://docs.google.com/org.gradle.api.artifacts.ComponentSelection.html)
* [ComponentSelectionRules](http://docs.google.com/org.gradle.api.artifacts.ComponentSelectionRules.html)
* [ExtensionAware](http://docs.google.com/org.gradle.api.plugins.ExtensionAware.html)
* [ExtraPropertiesExtension](http://docs.google.com/org.gradle.api.plugins.ExtraPropertiesExtension.html)
* [PluginDependenciesSpec](http://docs.google.com/org.gradle.plugin.use.PluginDependenciesSpec.html)
* [PluginDependencySpec](http://docs.google.com/org.gradle.plugin.use.PluginDependencySpec.html)
* [PluginManagementSpec](http://docs.google.com/org.gradle.plugin.management.PluginManagementSpec.html)
* [ResourceHandler](http://docs.google.com/org.gradle.api.resources.ResourceHandler.html)
* [TextResourceFactory](http://docs.google.com/org.gradle.api.resources.TextResourceFactory.html)

### Publishing types

* [PublishingExtension](http://docs.google.com/org.gradle.api.publish.PublishingExtension.html)
* [IvyPublication](http://docs.google.com/org.gradle.api.publish.ivy.IvyPublication.html)
* [IvyArtifact](http://docs.google.com/org.gradle.api.publish.ivy.IvyArtifact.html)
* [IvyArtifactSet](http://docs.google.com/org.gradle.api.publish.ivy.IvyArtifactSet.html)
* [IvyModuleDescriptorSpec](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorSpec.html)
* [IvyModuleDescriptorAuthor](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorAuthor.html)
* [IvyModuleDescriptorLicense](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorLicense.html)
* [IvyModuleDescriptorDescription](http://docs.google.com/org.gradle.api.publish.ivy.IvyModuleDescriptorDescription.html)
* [MavenPublication](http://docs.google.com/org.gradle.api.publish.maven.MavenPublication.html)
* [MavenArtifact](http://docs.google.com/org.gradle.api.publish.maven.MavenArtifact.html)
* [MavenArtifactSet](http://docs.google.com/org.gradle.api.publish.maven.MavenArtifactSet.html)
* [MavenPom](http://docs.google.com/org.gradle.api.publish.maven.MavenPom.html)
* [MavenPomCiManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomCiManagement.html)
* [MavenPomContributor](http://docs.google.com/org.gradle.api.publish.maven.MavenPomContributor.html)
* [MavenPomContributorSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomContributorSpec.html)
* [MavenPomDeveloper](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDeveloper.html)
* [MavenPomDeveloperSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDeveloperSpec.html)
* [MavenPomDistributionManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomDistributionManagement.html)
* [MavenPomIssueManagement](http://docs.google.com/org.gradle.api.publish.maven.MavenPomIssueManagement.html)
* [MavenPomLicense](http://docs.google.com/org.gradle.api.publish.maven.MavenPomLicense.html)
* [MavenPomLicenseSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomLicenseSpec.html)
* [MavenPomMailingList](http://docs.google.com/org.gradle.api.publish.maven.MavenPomMailingList.html)
* [MavenPomMailingListSpec](http://docs.google.com/org.gradle.api.publish.maven.MavenPomMailingListSpec.html)
* [MavenPomOrganization](http://docs.google.com/org.gradle.api.publish.maven.MavenPomOrganization.html)
* [MavenPomRelocation](http://docs.google.com/org.gradle.api.publish.maven.MavenPomRelocation.html)
* [MavenPomScm](http://docs.google.com/org.gradle.api.publish.maven.MavenPomScm.html)

### Container types

* [TaskContainer](http://docs.google.com/org.gradle.api.tasks.TaskContainer.html)
* [ConfigurationContainer](http://docs.google.com/org.gradle.api.artifacts.ConfigurationContainer.html)
* [RepositoryHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.RepositoryHandler.html)
* [DependencyHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.DependencyHandler.html)
* [ArtifactHandler](http://docs.google.com/org.gradle.api.artifacts.dsl.ArtifactHandler.html)

### Build Cache types

* [BuildCacheConfiguration](http://docs.google.com/org.gradle.caching.configuration.BuildCacheConfiguration.html)
* [DirectoryBuildCache](http://docs.google.com/org.gradle.caching.local.DirectoryBuildCache.html)
* [HttpBuildCache](http://docs.google.com/org.gradle.caching.http.HttpBuildCache.html)

### Input Normalization types

* [InputNormalizationHandler](http://docs.google.com/org.gradle.normalization.InputNormalizationHandler.html)
* [InputNormalization](http://docs.google.com/org.gradle.normalization.InputNormalization.html)
* [RuntimeClasspathNormalization](http://docs.google.com/org.gradle.normalization.RuntimeClasspathNormalization.html)

### Help Task types

* [TaskReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.TaskReportTask.html)
* [ProjectReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.ProjectReportTask.html)
* [DependencyReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.DependencyReportTask.html)
* [DependencyInsightReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.DependencyInsightReportTask.html)
* [PropertyReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.PropertyReportTask.html)
* [ComponentReport](http://docs.google.com/org.gradle.api.reporting.components.ComponentReport.html)
* [DependentComponentsReport](http://docs.google.com/org.gradle.api.reporting.dependents.DependentComponentsReport.html)
* [ModelReport](http://docs.google.com/org.gradle.api.reporting.model.ModelReport.html)

### Task types

* [AntlrTask](http://docs.google.com/org.gradle.api.plugins.antlr.AntlrTask.html)
* [BuildEnvironmentReportTask](http://docs.google.com/org.gradle.api.tasks.diagnostics.BuildEnvironmentReportTask.html)
* [Checkstyle](http://docs.google.com/org.gradle.api.plugins.quality.Checkstyle.html)
* [CodeNarc](http://docs.google.com/org.gradle.api.plugins.quality.CodeNarc.html)
* [CompareGradleBuilds](http://docs.google.com/org.gradle.api.plugins.buildcomparison.gradle.CompareGradleBuilds.html)
* [Copy](http://docs.google.com/org.gradle.api.tasks.Copy.html)
* [CreateStartScripts](http://docs.google.com/org.gradle.jvm.application.tasks.CreateStartScripts.html)
* [Delete](http://docs.google.com/org.gradle.api.tasks.Delete.html)
* [Ear](http://docs.google.com/org.gradle.plugins.ear.Ear.html)
* [Exec](http://docs.google.com/org.gradle.api.tasks.Exec.html)
* [FindBugs](http://docs.google.com/org.gradle.api.plugins.quality.FindBugs.html)
* [GenerateIvyDescriptor](http://docs.google.com/org.gradle.api.publish.ivy.tasks.GenerateIvyDescriptor.html)
* [GenerateMavenPom](http://docs.google.com/org.gradle.api.publish.maven.tasks.GenerateMavenPom.html)
* [GenerateBuildDashboard](http://docs.google.com/org.gradle.api.reporting.GenerateBuildDashboard.html)
* [GradleBuild](http://docs.google.com/org.gradle.api.tasks.GradleBuild.html)
* [GroovyCompile](http://docs.google.com/org.gradle.api.tasks.compile.GroovyCompile.html)
* [Groovydoc](http://docs.google.com/org.gradle.api.tasks.javadoc.Groovydoc.html)
* [HtmlDependencyReportTask](http://docs.google.com/org.gradle.api.reporting.dependencies.HtmlDependencyReportTask.html)
* [JacocoReport](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoReport.html)
* [JacocoMerge](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoMerge.html)
* [JacocoCoverageVerification](http://docs.google.com/org.gradle.testing.jacoco.tasks.JacocoCoverageVerification.html)
* [Jar](http://docs.google.com/org.gradle.api.tasks.bundling.Jar.html)
* [JavaCompile](http://docs.google.com/org.gradle.api.tasks.compile.JavaCompile.html)
* [Javadoc](http://docs.google.com/org.gradle.api.tasks.javadoc.Javadoc.html)
* [JavaExec](http://docs.google.com/org.gradle.api.tasks.JavaExec.html)
* [JDepend](http://docs.google.com/org.gradle.api.plugins.quality.JDepend.html)
* [Pmd](http://docs.google.com/org.gradle.api.plugins.quality.Pmd.html)
* [PublishToIvyRepository](http://docs.google.com/org.gradle.api.publish.ivy.tasks.PublishToIvyRepository.html)
* [PublishToMavenRepository](http://docs.google.com/org.gradle.api.publish.maven.tasks.PublishToMavenRepository.html)
* [ScalaCompile](http://docs.google.com/org.gradle.api.tasks.scala.ScalaCompile.html)
* [ScalaDoc](http://docs.google.com/org.gradle.api.tasks.scala.ScalaDoc.html)
* [InitBuild](http://docs.google.com/org.gradle.buildinit.tasks.InitBuild.html)
* [Sign](http://docs.google.com/org.gradle.plugins.signing.Sign.html)
* [Sync](http://docs.google.com/org.gradle.api.tasks.Sync.html)
* [Tar](http://docs.google.com/org.gradle.api.tasks.bundling.Tar.html)
* [AbstractTestTask](http://docs.google.com/org.gradle.api.tasks.testing.AbstractTestTask.html)
* [Test](http://docs.google.com/org.gradle.api.tasks.testing.Test.html)
* [TestReport](http://docs.google.com/org.gradle.api.tasks.testing.TestReport.html)
* [Upload](http://docs.google.com/org.gradle.api.tasks.Upload.html)
* [War](http://docs.google.com/org.gradle.api.tasks.bundling.War.html)
* [Wrapper](http://docs.google.com/org.gradle.api.tasks.wrapper.Wrapper.html)
* [WriteProperties](http://docs.google.com/org.gradle.api.tasks.WriteProperties.html)
* [Zip](http://docs.google.com/org.gradle.api.tasks.bundling.Zip.html)

### Reporting types

* [CustomizableHtmlReport](http://docs.google.com/org.gradle.api.reporting.CustomizableHtmlReport.html)
* [SingleFileReport](http://docs.google.com/org.gradle.api.reporting.SingleFileReport.html)
* [DirectoryReport](http://docs.google.com/org.gradle.api.reporting.DirectoryReport.html)
* [FindBugsXmlReport](http://docs.google.com/org.gradle.api.plugins.quality.FindBugsXmlReport.html)
* [Report](http://docs.google.com/org.gradle.api.reporting.Report.html)
* [Reporting](http://docs.google.com/org.gradle.api.reporting.Reporting.html)
* [ReportContainer](http://docs.google.com/org.gradle.api.reporting.ReportContainer.html)
* [ReportingExtension](http://docs.google.com/org.gradle.api.reporting.ReportingExtension.html)

### Eclipse/IDEA model types

* [EclipseModel](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseModel.html)
* [EclipseProject](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseProject.html)
* [EclipseClasspath](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseClasspath.html)
* [EclipseJdt](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseJdt.html)
* [EclipseWtp](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtp.html)
* [EclipseWtpComponent](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtpComponent.html)
* [EclipseWtpFacet](http://docs.google.com/org.gradle.plugins.ide.eclipse.model.EclipseWtpFacet.html)
* [IdeaModel](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaModel.html)
* [IdeaProject](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaProject.html)
* [IdeaModule](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaModule.html)
* [IdeaWorkspace](http://docs.google.com/org.gradle.plugins.ide.idea.model.IdeaWorkspace.html)
* [XmlFileContentMerger](http://docs.google.com/org.gradle.plugins.ide.api.XmlFileContentMerger.html)
* [FileContentMerger](http://docs.google.com/org.gradle.plugins.ide.api.FileContentMerger.html)

### Eclipse/IDEA task types

* [GenerateEclipseProject](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseProject.html)
* [GenerateEclipseClasspath](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseClasspath.html)
* [GenerateEclipseJdt](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseJdt.html)
* [GenerateEclipseWtpComponent](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseWtpComponent.html)
* [GenerateEclipseWtpFacet](http://docs.google.com/org.gradle.plugins.ide.eclipse.GenerateEclipseWtpFacet.html)
* [GenerateIdeaModule](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaModule.html)
* [GenerateIdeaProject](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaProject.html)
* [GenerateIdeaWorkspace](http://docs.google.com/org.gradle.plugins.ide.idea.GenerateIdeaWorkspace.html)

### Native software types

* [PrebuiltLibrary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltLibrary.html)
* [PrebuiltSharedLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltSharedLibraryBinary.html)
* [PrebuiltStaticLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.PrebuiltStaticLibraryBinary.html)
* [NativeComponentSpec](http://docs.google.com/org.gradle.nativeplatform.NativeComponentSpec.html)
* [NativeExecutableSpec](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableSpec.html)
* [NativeLibrarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeLibrarySpec.html)
* [NativeTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.NativeTestSuiteSpec.html)
* [CUnitTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.cunit.CUnitTestSuiteSpec.html)
* [GoogleTestTestSuiteSpec](http://docs.google.com/org.gradle.nativeplatform.test.googletest.GoogleTestTestSuiteSpec.html)
* [NativeBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeBinarySpec.html)
* [NativeExecutableBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableBinarySpec.html)
* [NativeLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.NativeLibraryBinarySpec.html)
* [SharedLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.SharedLibraryBinarySpec.html)
* [StaticLibraryBinarySpec](http://docs.google.com/org.gradle.nativeplatform.StaticLibraryBinarySpec.html)
* [NativeTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.NativeTestSuiteBinarySpec.html)
* [CUnitTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.cunit.CUnitTestSuiteBinarySpec.html)
* [GoogleTestTestSuiteBinarySpec](http://docs.google.com/org.gradle.nativeplatform.test.googletest.GoogleTestTestSuiteBinarySpec.html)
* [NativePlatform](http://docs.google.com/org.gradle.nativeplatform.platform.NativePlatform.html)
* [BuildType](http://docs.google.com/org.gradle.nativeplatform.BuildType.html)
* [Flavor](http://docs.google.com/org.gradle.nativeplatform.Flavor.html)
* [Gcc](http://docs.google.com/org.gradle.nativeplatform.toolchain.Gcc.html)
* [Clang](http://docs.google.com/org.gradle.nativeplatform.toolchain.Clang.html)
* [VisualCpp](http://docs.google.com/org.gradle.nativeplatform.toolchain.VisualCpp.html)
* [AssemblerSourceSet](http://docs.google.com/org.gradle.language.assembler.AssemblerSourceSet.html)
* [CSourceSet](http://docs.google.com/org.gradle.language.c.CSourceSet.html)
* [CppSourceSet](http://docs.google.com/org.gradle.language.cpp.CppSourceSet.html)
* [ObjectiveCSourceSet](http://docs.google.com/org.gradle.language.objectivec.ObjectiveCSourceSet.html)
* [ObjectiveCppSourceSet](http://docs.google.com/org.gradle.language.objectivecpp.ObjectiveCppSourceSet.html)
* [WindowsResourceSet](http://docs.google.com/org.gradle.language.rc.WindowsResourceSet.html)
* [VisualStudioProject](http://docs.google.com/org.gradle.ide.visualstudio.VisualStudioProject.html)
* [VisualStudioSolution](http://docs.google.com/org.gradle.ide.visualstudio.VisualStudioSolution.html)
* [NativeExecutable](http://docs.google.com/org.gradle.nativeplatform.NativeExecutable.html)
* [NativeLibrary](http://docs.google.com/org.gradle.nativeplatform.NativeLibrary.html)
* [NativeBinary](http://docs.google.com/org.gradle.nativeplatform.NativeBinary.html)
* [NativeExecutableBinary](http://docs.google.com/org.gradle.nativeplatform.NativeExecutableBinary.html)
* [SharedLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.SharedLibraryBinary.html)
* [StaticLibraryBinary](http://docs.google.com/org.gradle.nativeplatform.StaticLibraryBinary.html)

### Native component task types

* [CppCompile](http://docs.google.com/org.gradle.language.cpp.tasks.CppCompile.html)
* [CCompile](http://docs.google.com/org.gradle.language.c.tasks.CCompile.html)
* [Assemble](http://docs.google.com/org.gradle.language.assembler.tasks.Assemble.html)
* [ObjectiveCCompile](http://docs.google.com/org.gradle.language.objectivec.tasks.ObjectiveCCompile.html)
* [ObjectiveCppCompile](http://docs.google.com/org.gradle.language.objectivecpp.tasks.ObjectiveCppCompile.html)
* [WindowsResourceCompile](http://docs.google.com/org.gradle.language.rc.tasks.WindowsResourceCompile.html)
* [LinkExecutable](http://docs.google.com/org.gradle.nativeplatform.tasks.LinkExecutable.html)
* [LinkSharedLibrary](http://docs.google.com/org.gradle.nativeplatform.tasks.LinkSharedLibrary.html)
* [CreateStaticLibrary](http://docs.google.com/org.gradle.nativeplatform.tasks.CreateStaticLibrary.html)
* [InstallExecutable](http://docs.google.com/org.gradle.nativeplatform.tasks.InstallExecutable.html)
* [RunTestExecutable](http://docs.google.com/org.gradle.nativeplatform.test.tasks.RunTestExecutable.html)

Test

**Table of Contents**

[Properties](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N27763)[Methods](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N2789E)[Script blocks](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N27A95)[Property details](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N27ADD)[Method details](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N27D73)[Script block details](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#N2811C)

| API Documentation: | [Test](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/Test.html) |
| --- | --- |

Executes JUnit (3.8.x, 4.x or 5.x) or TestNG tests. Test are always run in (one or more) separate JVMs. The sample below shows various configuration options.

apply plugin: 'java' // adds 'test' task  
  
test {  
 // enable TestNG support (default is JUnit)  
 useTestNG()  
 // enable JUnit Platform (a.k.a. JUnit 5) support  
 useJUnitPlatform()  
  
 // set a system property for the test JVM(s)  
 systemProperty 'some.prop', 'value'  
  
 // explicitly include or exclude tests  
 include 'org/foo/\*\*'  
 exclude 'org/boo/\*\*'  
  
 // show standard out and standard error of the test JVM(s) on the console  
 testLogging.showStandardStreams = true  
  
 // set heap size for the test JVM(s)  
 minHeapSize = "128m"  
 maxHeapSize = "512m"  
  
 // set JVM arguments for the test JVM(s)  
 jvmArgs '-XX:MaxPermSize=256m'  
  
 // listen to events in the test execution lifecycle  
 beforeTest { descriptor ->  
 logger.lifecycle("Running test: " + descriptor)  
 }  
  
 // Fail the 'test' task on the first test failure  
 failFast = true  
  
 // listen to standard out and standard error of the test JVM(s)  
 onOutput { descriptor, event ->  
 logger.lifecycle("Test: " + descriptor + " produced standard out/err: " + event.message )  
 }  
}

The test process can be started in debug mode (see [Test.getDebug()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:debug)) in an ad-hoc manner by supplying the `--debug-jvm` switch when invoking the build.

gradle someTestTask --debug-jvm

Properties

| Property | Description |
| --- | --- |
| [allJvmArgs](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:allJvmArgs) | The full set of arguments to use to launch the JVM for the process. This includes arguments to define system properties, the minimum/maximum heap size, and the bootstrap classpath. |
| [binResultsDir](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:binResultsDir) | Incubating  The root folder for the test results in internal binary format. |
| [bootstrapClasspath](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:bootstrapClasspath) | The bootstrap classpath to use for the process. The default bootstrap classpath for the JVM is used when this classpath is empty. |
| [classpath](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:classpath) | The classpath to use to execute the tests. |
| [debug](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:debug) | Determines whether debugging is enabled for the test process. When enabled — debug = true — the process is started in a suspended state, listening on port 5005. You should disable parallel test execution when debugging and you will need to reattach the debugger occasionally if you use a non-zero value for [Test.getForkEvery()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:forkEvery). |
| [enableAssertions](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:enableAssertions) | Returns true if assertions are enabled for the process. |
| [environment](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:environment) | The environment variables to use for the process. Defaults to the environment of this process. |
| [excludes](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:excludes) | The exclude patterns for test execution. |
| [executable](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:executable) | The name of the executable to use. |
| [failFast](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:failFast) | Indicates if this task will fail on the first failed test |
| [forkEvery](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:forkEvery) | The maximum number of test classes to execute in a forked test process. The forked test process will be restarted when this limit is reached. The default value is 0 (no maximum). |
| [ignoreFailures](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:ignoreFailures) | Specifies whether the build should break when the verifications performed by this task fail. |
| [includes](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:includes) | The include patterns for test execution. |
| [jvmArgs](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:jvmArgs) | The extra arguments to use to launch the JVM for the process. Does not include system properties and the minimum/maximum heap size. |
| [jvmArgumentProviders](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:jvmArgumentProviders) | Command line argument providers for the java process to fork. |
| [maxHeapSize](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:maxHeapSize) | The maximum heap size for the process, if any. |
| [maxParallelForks](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:maxParallelForks) | The maximum number of forked test processes to execute in parallel. The default value is 1 (no parallel test execution). It cannot exceed the value of max-workers for the current build. |
| [minHeapSize](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:minHeapSize) | The minimum heap size for the process, if any. |
| [options](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:options) | Returns test framework specific options. Make sure to call [Test.useJUnit()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit()), [Test.useJUnitPlatform()](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/Test.html#useJUnitPlatform--) or [Test.useTestNG()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG()) before using this method. |
| [reports](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:reports) | The reports that this task potentially produces. |
| [scanForTestClasses](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:scanForTestClasses) | Specifies whether test classes should be detected. When true the classes which match the include and exclude patterns are scanned for test classes, and any found are executed. When false the classes which match the include and exclude patterns are executed. |
| [systemProperties](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:systemProperties) | The system properties which will be used for the process. |
| [testClassesDir](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:testClassesDir) | Deprecated  The root folder for the compiled test sources. |
| [testClassesDirs](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:testClassesDirs) | The directories for the compiled test sources. |
| [testLogging](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:testLogging) | Allows to set options related to which test events are logged to the console, and on which detail level. For example, to show more information about exceptions use: |
| [workingDir](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:workingDir) | The working directory for the process. Defaults to the project directory. |

Properties added by the jacoco plugin

| Property | Description |
| --- | --- |
| [jacoco](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:jacoco) | The [JacocoTaskExtension](http://docs.google.com/dsl/org.gradle.testing.jacoco.plugins.JacocoTaskExtension.html) added by the jacoco plugin. |

Methods

| Method | Description |
| --- | --- |
| [addTestListener](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:addTestListener(org.gradle.api.tasks.testing.TestListener))(listener) | Registers a test listener with this task. Consider also the following handy methods for quicker hooking into test execution: [AbstractTestTask.beforeTest(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:beforeTest(groovy.lang.Closure)), [AbstractTestTask.afterTest(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:afterTest(groovy.lang.Closure)), [AbstractTestTask.beforeSuite(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:beforeSuite(groovy.lang.Closure)), [AbstractTestTask.afterSuite(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:afterSuite(groovy.lang.Closure)) |
| [addTestOutputListener](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:addTestOutputListener(org.gradle.api.tasks.testing.TestOutputListener))(listener) | Registers a output listener with this task. Quicker way of hooking into output events is using the [AbstractTestTask.onOutput(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:onOutput(groovy.lang.Closure)) method. |
| [afterSuite](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:afterSuite(groovy.lang.Closure))(closure) | Adds a closure to be notified after a test suite has executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) and [TestResult](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestResult.html) instance are passed to the closure as a parameter. |
| [afterTest](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:afterTest(groovy.lang.Closure))(closure) | Adds a closure to be notified after a test has executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) and [TestResult](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestResult.html) instance are passed to the closure as a parameter. |
| [beforeSuite](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:beforeSuite(groovy.lang.Closure))(closure) | Adds a closure to be notified before a test suite is executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) instance is passed to the closure as a parameter. |
| [beforeTest](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:beforeTest(groovy.lang.Closure))(closure) | Adds a closure to be notified before a test is executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) instance is passed to the closure as a parameter. |
| [bootstrapClasspath](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:bootstrapClasspath(java.lang.Object%5B%5D))(classpath) | Adds the given values to the end of the bootstrap classpath for the process. |
| [copyTo](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:copyTo(org.gradle.process.JavaForkOptions))(target) | Copies these options to the given options. |
| [copyTo](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:copyTo(org.gradle.process.ProcessForkOptions))(target) | Copies these options to the given target options. |
| [environment](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:environment(java.lang.String,%20java.lang.Object))(name, value) | Adds an environment variable to the environment for this process. |
| [environment](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:environment(java.util.Map))(environmentVariables) | Adds some environment variables to the environment for this process. |
| [exclude](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:exclude(groovy.lang.Closure))(excludeSpec) | Adds an exclude spec. This method may be called multiple times to append new specs.The given closure is passed a [FileTreeElement](http://docs.google.com/javadoc/org/gradle/api/file/FileTreeElement.html) as its parameter. The closure should return true or false. Example: |
| [exclude](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:exclude(java.lang.Iterable))(excludes) | Adds exclude patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')). |
| [exclude](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:exclude(java.lang.String%5B%5D))(excludes) | Adds exclude patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')). |
| [exclude](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:exclude(org.gradle.api.specs.Spec))(excludeSpec) | Adds an exclude spec. This method may be called multiple times to append new specs. If excludes are not provided, then no files will be excluded. If excludes are provided, then files must not match any exclude pattern to be processed. |
| [executable](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:executable(java.lang.Object))(executable) | Sets the name of the executable to use. |
| [include](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:include(groovy.lang.Closure))(includeSpec) | Adds an include spec. This method may be called multiple times to append new specs. The given closure is passed a [FileTreeElement](http://docs.google.com/javadoc/org/gradle/api/file/FileTreeElement.html) as its parameter. If includes are not provided, then all files in this container will be included. If includes are provided, then a file must match at least one of the include patterns or specs to be included. |
| [include](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:include(java.lang.Iterable))(includes) | Adds include patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')). |
| [include](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:include(java.lang.String%5B%5D))(includes) | Adds include patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')). |
| [include](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:include(org.gradle.api.specs.Spec))(includeSpec) | Adds an include spec. This method may be called multiple times to append new specs. If includes are not provided, then all files in this container will be included. If includes are provided, then a file must match at least one of the include patterns or specs to be included. |
| [jvmArgs](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:jvmArgs(java.lang.Iterable))(arguments) | Adds some arguments to use to launch the JVM for the process. |
| [jvmArgs](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:jvmArgs(java.lang.Object%5B%5D))(arguments) | Adds some arguments to use to launch the JVM for the process. |
| [onOutput](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:onOutput(groovy.lang.Closure))(closure) | Adds a closure to be notified when output from the test received. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) and [TestOutputEvent](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestOutputEvent.html) instance are passed to the closure as a parameter. |
| [options](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:options(org.gradle.api.Action))(testFrameworkConfigure) | Configures test framework specific options. Make sure to call [Test.useJUnit()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit()), [Test.useJUnitPlatform()](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/Test.html#useJUnitPlatform--) or [Test.useTestNG()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG()) before using this method. |
| [removeTestListener](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:removeTestListener(org.gradle.api.tasks.testing.TestListener))(listener) | Unregisters a test listener with this task. This method will only remove listeners that were added by calling [AbstractTestTask.addTestListener(org.gradle.api.tasks.testing.TestListener)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:addTestListener(org.gradle.api.tasks.testing.TestListener)) on this task. If the listener was registered with Gradle using [Gradle.addListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:addListener(java.lang.Object)) this method will not do anything. Instead, use [Gradle.removeListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:removeListener(java.lang.Object)). |
| [removeTestOutputListener](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:removeTestOutputListener(org.gradle.api.tasks.testing.TestOutputListener))(listener) | Unregisters a test output listener with this task. This method will only remove listeners that were added by calling [AbstractTestTask.addTestOutputListener(org.gradle.api.tasks.testing.TestOutputListener)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:addTestOutputListener(org.gradle.api.tasks.testing.TestOutputListener)) on this task. If the listener was registered with Gradle using [Gradle.addListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:addListener(java.lang.Object)) this method will not do anything. Instead, use [Gradle.removeListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:removeListener(java.lang.Object)). |
| [reports](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:reports(org.gradle.api.Action))(configureAction) | Configures the reports that this task potentially produces. |
| [setTestNameIncludePatterns](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:setTestNameIncludePatterns(java.util.List))(testNamePattern) | Incubating  Sets the test name patterns to be included in execution. Classes or method names are supported, wildcard '\*' is supported. For more information see the user guide chapter on testing. For more information on supported patterns see [TestFilter](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestFilter.html) |
| [systemProperties](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:systemProperties(java.util.Map))(properties) | Adds some system properties to use for the process. |
| [useJUnit](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit())() | Specifies that JUnit should be used to execute the tests. |
| [useJUnit](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit(groovy.lang.Closure))(testFrameworkConfigure) | Specifies that JUnit should be used to execute the tests, configuring JUnit specific options. |
| [useJUnit](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit(org.gradle.api.Action))(testFrameworkConfigure) | Specifies that JUnit should be used to execute the tests, configuring JUnit specific options. |
| [useTestNG](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG())() | Specifies that TestNG should be used to execute the tests. |
| [useTestNG](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG(groovy.lang.Closure))(testFrameworkConfigure) | Specifies that TestNG should be used to execute the tests, configuring TestNG specific options. |
| [useTestNG](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG(org.gradle.api.Action))(testFrameworkConfigure) | Specifies that TestNG should be used to execute the tests, configuring TestNG specific options. |
| [workingDir](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:workingDir(java.lang.Object))(dir) | Sets the working directory for the process. The supplied argument is evaluated as per [Project.file(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.Project.html#org.gradle.api.Project:file(java.lang.Object)). |

Script blocks

| Block | Description |
| --- | --- |
| [options](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:options(groovy.lang.Closure)) | Configures test framework specific options. Make sure to call [Test.useJUnit()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit()), [Test.useJUnitPlatform()](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/Test.html#useJUnitPlatform--) or [Test.useTestNG()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG()) before using this method. |

Script blocks added by the jacoco plugin

| Block | Description |
| --- | --- |
| [jacoco](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:jacoco(groovy.lang.Closure)) | Configures the [JacocoTaskExtension](http://docs.google.com/dsl/org.gradle.testing.jacoco.plugins.JacocoTaskExtension.html) added by the jacoco plugin. |

Property details

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> allJvmArgs

The full set of arguments to use to launch the JVM for the process. This includes arguments to define system properties, the minimum/maximum heap size, and the bootstrap classpath.

[**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html) binResultsDir

Note: This property is [incubating](http://docs.google.com/userguide/feature_lifecycle.html) and may change in a future version of Gradle.

The root folder for the test results in internal binary format.

Default:*project.testResultsDir*/binary/*task.name*

[**FileCollection**](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) bootstrapClasspath

The bootstrap classpath to use for the process. The default bootstrap classpath for the JVM is used when this classpath is empty.

Default with java plugin:[]

[**FileCollection**](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) classpath

The classpath to use to execute the tests.

Default with java plugin:project.sourceSets.test.runtimeClasspath

boolean debug

Determines whether debugging is enabled for the test process. When enabled — debug = true — the process is started in a suspended state, listening on port 5005. You should disable parallel test execution when debugging and you will need to reattach the debugger occasionally if you use a non-zero value for [Test.getForkEvery()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:forkEvery).

Default with java plugin:false

boolean enableAssertions

Returns true if assertions are enabled for the process.

Default with java plugin:true

[**Map**](http://download.oracle.com/javase/8/docs/api/java/util/Map.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html), [**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html)> environment

The environment variables to use for the process. Defaults to the environment of this process.

Default with java plugin:environment of the current process

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> excludes

The exclude patterns for test execution.

Default with java plugin:[]

[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) executable

The name of the executable to use.

Default with java plugin:**java** command for the current JVM.

boolean failFast

Indicates if this task will fail on the first failed test

Default with java plugin:false

long forkEvery

The maximum number of test classes to execute in a forked test process. The forked test process will be restarted when this limit is reached. The default value is 0 (no maximum).

Default with java plugin:0

boolean ignoreFailures

Specifies whether the build should break when the verifications performed by this task fail.

[**Set**](http://download.oracle.com/javase/8/docs/api/java/util/Set.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> includes

The include patterns for test execution.

Default with java plugin:[]

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> jvmArgs

The extra arguments to use to launch the JVM for the process. Does not include system properties and the minimum/maximum heap size.

Default with java plugin:[]

[**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**CommandLineArgumentProvider**](http://docs.google.com/javadoc/org/gradle/process/CommandLineArgumentProvider.html)> jvmArgumentProviders (read-only)

Command line argument providers for the java process to fork.

Default with java plugin:[]

[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) maxHeapSize

The maximum heap size for the process, if any.

Default with java plugin:null

int maxParallelForks

The maximum number of forked test processes to execute in parallel. The default value is 1 (no parallel test execution). It cannot exceed the value of max-workers for the current build.

Default with java plugin:1

[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) minHeapSize

The minimum heap size for the process, if any.

Default with java plugin:null

[**TestFrameworkOptions**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestFrameworkOptions.html) options (read-only)

Returns test framework specific options. Make sure to call [Test.useJUnit()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit()), [Test.useJUnitPlatform()](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/Test.html#useJUnitPlatform--) or [Test.useTestNG()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG()) before using this method.

[**TestTaskReports**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestTaskReports.html) reports (read-only)

The reports that this task potentially produces.

boolean scanForTestClasses

Specifies whether test classes should be detected. When true the classes which match the include and exclude patterns are scanned for test classes, and any found are executed. When false the classes which match the include and exclude patterns are executed.

Default with java plugin:true

[**Map**](http://download.oracle.com/javase/8/docs/api/java/util/Map.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html), [**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html)> systemProperties

The system properties which will be used for the process.

Default with java plugin:[:]

[**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html) testClassesDir

Note: This property is [deprecated](http://docs.google.com/userguide/feature_lifecycle.html) and will be removed in the next major version of Gradle.

The root folder for the compiled test sources.

[**FileCollection**](http://docs.google.com/javadoc/org/gradle/api/file/FileCollection.html) testClassesDirs

The directories for the compiled test sources.

Default with java plugin:project.sourceSets.test.output.classesDirs

[**TestLoggingContainer**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.logging.TestLoggingContainer.html) testLogging (read-only)

Allows to set options related to which test events are logged to the console, and on which detail level. For example, to show more information about exceptions use:

apply plugin: 'java'  
  
test.testLogging {  
 exceptionFormat "full"  
}

For further information see [TestLoggingContainer](http://docs.google.com/dsl/org.gradle.api.tasks.testing.logging.TestLoggingContainer.html).

[**File**](http://download.oracle.com/javase/8/docs/api/java/io/File.html) workingDir

The working directory for the process. Defaults to the project directory.

Default with java plugin:project.projectDir

[**JacocoTaskExtension**](http://docs.google.com/dsl/org.gradle.testing.jacoco.plugins.JacocoTaskExtension.html) jacoco (read-only)

The [JacocoTaskExtension](http://docs.google.com/dsl/org.gradle.testing.jacoco.plugins.JacocoTaskExtension.html) added by the jacoco plugin.

Method details

void addTestListener([**TestListener**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestListener.html) listener)

Registers a test listener with this task. Consider also the following handy methods for quicker hooking into test execution: [AbstractTestTask.beforeTest(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:beforeTest(groovy.lang.Closure)), [AbstractTestTask.afterTest(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:afterTest(groovy.lang.Closure)), [AbstractTestTask.beforeSuite(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:beforeSuite(groovy.lang.Closure)), [AbstractTestTask.afterSuite(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:afterSuite(groovy.lang.Closure))

This listener will NOT be notified of tests executed by other tasks. To get that behavior, use [Gradle.addListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:addListener(java.lang.Object)).

void addTestOutputListener([**TestOutputListener**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestOutputListener.html) listener)

Registers a output listener with this task. Quicker way of hooking into output events is using the [AbstractTestTask.onOutput(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:onOutput(groovy.lang.Closure)) method.

void afterSuite([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) closure)

Adds a closure to be notified after a test suite has executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) and [TestResult](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestResult.html) instance are passed to the closure as a parameter.

This method is also called after all test suites are executed. The provided descriptor will have a null parent suite.

void afterTest([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) closure)

Adds a closure to be notified after a test has executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) and [TestResult](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestResult.html) instance are passed to the closure as a parameter.

void beforeSuite([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) closure)

Adds a closure to be notified before a test suite is executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) instance is passed to the closure as a parameter.

This method is also called before any test suites are executed. The provided descriptor will have a null parent suite.

void beforeTest([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) closure)

Adds a closure to be notified before a test is executed. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) instance is passed to the closure as a parameter.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) bootstrapClasspath([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html)... classpath)

Adds the given values to the end of the bootstrap classpath for the process.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) copyTo([**JavaForkOptions**](http://docs.google.com/javadoc/org/gradle/process/JavaForkOptions.html) target)

Copies these options to the given options.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) copyTo([**ProcessForkOptions**](http://docs.google.com/javadoc/org/gradle/process/ProcessForkOptions.html) target)

Copies these options to the given target options.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) environment([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html) name, [**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) value)

Adds an environment variable to the environment for this process.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) environment([**Map**](http://download.oracle.com/javase/8/docs/api/java/util/Map.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html), ?> environmentVariables)

Adds some environment variables to the environment for this process.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) exclude([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) excludeSpec)

Adds an exclude spec. This method may be called multiple times to append new specs.The given closure is passed a [FileTreeElement](http://docs.google.com/javadoc/org/gradle/api/file/FileTreeElement.html) as its parameter. The closure should return true or false. Example:

copySpec {  
 from 'source'  
 into 'destination'  
 //an example of excluding files from certain configuration:  
 exclude { it.file in configurations.someConf.files }  
}

If excludes are not provided, then no files will be excluded. If excludes are provided, then files must not match any exclude pattern to be processed.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) exclude([**Iterable**](http://download.oracle.com/javase/8/docs/api/java/lang/Iterable.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> excludes)

Adds exclude patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')).

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) exclude([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)... excludes)

Adds exclude patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')).

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) exclude([**Spec**](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html)<[**FileTreeElement**](http://docs.google.com/javadoc/org/gradle/api/file/FileTreeElement.html)> excludeSpec)

Adds an exclude spec. This method may be called multiple times to append new specs. If excludes are not provided, then no files will be excluded. If excludes are provided, then files must not match any exclude pattern to be processed.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) executable([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) executable)

Sets the name of the executable to use.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) include([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) includeSpec)

Adds an include spec. This method may be called multiple times to append new specs. The given closure is passed a [FileTreeElement](http://docs.google.com/javadoc/org/gradle/api/file/FileTreeElement.html) as its parameter. If includes are not provided, then all files in this container will be included. If includes are provided, then a file must match at least one of the include patterns or specs to be included.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) include([**Iterable**](http://download.oracle.com/javase/8/docs/api/java/lang/Iterable.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> includes)

Adds include patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')).

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) include([**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)... includes)

Adds include patterns for the files in the test classes directory (e.g. '\*\*/\*Test.class')).

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) include([**Spec**](http://docs.google.com/javadoc/org/gradle/api/specs/Spec.html)<[**FileTreeElement**](http://docs.google.com/javadoc/org/gradle/api/file/FileTreeElement.html)> includeSpec)

Adds an include spec. This method may be called multiple times to append new specs. If includes are not provided, then all files in this container will be included. If includes are provided, then a file must match at least one of the include patterns or specs to be included.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) jvmArgs([**Iterable**](http://download.oracle.com/javase/8/docs/api/java/lang/Iterable.html)<?> arguments)

Adds some arguments to use to launch the JVM for the process.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) jvmArgs([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html)... arguments)

Adds some arguments to use to launch the JVM for the process.

void onOutput([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) closure)

Adds a closure to be notified when output from the test received. A [TestDescriptor](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestDescriptor.html) and [TestOutputEvent](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestOutputEvent.html) instance are passed to the closure as a parameter.

apply plugin: 'java'  
  
test {  
 onOutput { descriptor, event ->  
 if (event.destination == TestOutputEvent.Destination.StdErr) {  
 logger.error("Test: " + descriptor + ", error: " + event.message)  
 }  
 }  
}

[**TestFrameworkOptions**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestFrameworkOptions.html) options([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**TestFrameworkOptions**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestFrameworkOptions.html)> testFrameworkConfigure)

Configures test framework specific options. Make sure to call [Test.useJUnit()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit()), [Test.useJUnitPlatform()](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/Test.html#useJUnitPlatform--) or [Test.useTestNG()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG()) before using this method.

void removeTestListener([**TestListener**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestListener.html) listener)

Unregisters a test listener with this task. This method will only remove listeners that were added by calling [AbstractTestTask.addTestListener(org.gradle.api.tasks.testing.TestListener)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:addTestListener(org.gradle.api.tasks.testing.TestListener)) on this task. If the listener was registered with Gradle using [Gradle.addListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:addListener(java.lang.Object)) this method will not do anything. Instead, use [Gradle.removeListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:removeListener(java.lang.Object)).

void removeTestOutputListener([**TestOutputListener**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestOutputListener.html) listener)

Unregisters a test output listener with this task. This method will only remove listeners that were added by calling [AbstractTestTask.addTestOutputListener(org.gradle.api.tasks.testing.TestOutputListener)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html#org.gradle.api.tasks.testing.AbstractTestTask:addTestOutputListener(org.gradle.api.tasks.testing.TestOutputListener)) on this task. If the listener was registered with Gradle using [Gradle.addListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:addListener(java.lang.Object)) this method will not do anything. Instead, use [Gradle.removeListener(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.invocation.Gradle.html#org.gradle.api.invocation.Gradle:removeListener(java.lang.Object)).

[**TestTaskReports**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestTaskReports.html) reports([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**TestTaskReports**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestTaskReports.html)> configureAction)

Configures the reports that this task potentially produces.

[**AbstractTestTask**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.AbstractTestTask.html) setTestNameIncludePatterns([**List**](http://download.oracle.com/javase/8/docs/api/java/util/List.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html)> testNamePattern)

Note: This method is [incubating](http://docs.google.com/userguide/feature_lifecycle.html) and may change in a future version of Gradle.

Sets the test name patterns to be included in execution. Classes or method names are supported, wildcard '\*' is supported. For more information see the user guide chapter on testing. For more information on supported patterns see [TestFilter](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestFilter.html)

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) systemProperties([**Map**](http://download.oracle.com/javase/8/docs/api/java/util/Map.html)<[**String**](http://download.oracle.com/javase/8/docs/api/java/lang/String.html), ?> properties)

Adds some system properties to use for the process.

void useJUnit()

Specifies that JUnit should be used to execute the tests.

To configure JUnit specific options, see [Test.useJUnit(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit(groovy.lang.Closure)).

void useJUnit([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) testFrameworkConfigure)

Specifies that JUnit should be used to execute the tests, configuring JUnit specific options.

The supplied closure configures an instance of [JUnitOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/junit/JUnitOptions.html), which can be used to configure how JUnit runs.

void useJUnit([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**JUnitOptions**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/junit/JUnitOptions.html)> testFrameworkConfigure)

Specifies that JUnit should be used to execute the tests, configuring JUnit specific options.

The supplied action configures an instance of [JUnitOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/junit/JUnitOptions.html), which can be used to configure how JUnit runs.

void useTestNG()

Specifies that TestNG should be used to execute the tests.

To configure TestNG specific options, see [Test.useTestNG(groovy.lang.Closure)](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG(groovy.lang.Closure)).

void useTestNG([**Closure**](http://docs.groovy-lang.org/2.4.15/html/gapi/groovy/lang/Closure.html) testFrameworkConfigure)

Specifies that TestNG should be used to execute the tests, configuring TestNG specific options.

The supplied closure configures an instance of [TestNGOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/testng/TestNGOptions.html), which can be used to configure how TestNG runs.

void useTestNG([**Action**](http://docs.google.com/javadoc/org/gradle/api/Action.html)<? super [**TestFrameworkOptions**](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestFrameworkOptions.html)> testFrameworkConfigure)

Specifies that TestNG should be used to execute the tests, configuring TestNG specific options.

The supplied action configures an instance of [TestNGOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/testng/TestNGOptions.html), which can be used to configure how TestNG runs.

[**Test**](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html) workingDir([**Object**](http://download.oracle.com/javase/8/docs/api/java/lang/Object.html) dir)

Sets the working directory for the process. The supplied argument is evaluated as per [Project.file(java.lang.Object)](http://docs.google.com/dsl/org.gradle.api.Project.html#org.gradle.api.Project:file(java.lang.Object)).

Script block details

options { }

Configures test framework specific options. Make sure to call [Test.useJUnit()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useJUnit()), [Test.useJUnitPlatform()](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/Test.html#useJUnitPlatform--) or [Test.useTestNG()](http://docs.google.com/dsl/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:useTestNG()) before using this method.

Delegates to:[TestFrameworkOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/testing/TestFrameworkOptions.html) from [options](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:options)

jacoco { }

Configures the [JacocoTaskExtension](http://docs.google.com/dsl/org.gradle.testing.jacoco.plugins.JacocoTaskExtension.html) added by the jacoco plugin.

Delegates to:[JacocoTaskExtension](http://docs.google.com/dsl/org.gradle.testing.jacoco.plugins.JacocoTaskExtension.html) from [jacoco](http://docs.google.com/org.gradle.api.tasks.testing.Test.html#org.gradle.api.tasks.testing.Test:jacoco)

**Docs**

* [User Manual](http://docs.google.com/userguide/userguide.html)
* [DSL Reference](http://docs.google.com/dsl/)
* [Release Notes](http://docs.google.com/release-notes.html)
* [Javadoc](http://docs.google.com/javadoc/)

**News**

* [Blog](https://blog.gradle.org/)
* [Newsletter](https://newsletter.gradle.com/)
* [Twitter](https://twitter.com/gradle)

**Products**

* [Build Scans](https://gradle.com/build-scans)
* [Build Cache](https://gradle.com/build-cache)
* [Enterprise Docs](https://gradle.com/enterprise/resources)

**Get Help**

* [Forums](https://discuss.gradle.org/c/help-discuss)
* [GitHub](https://github.com/gradle/)
* [Training](https://gradle.org/training/)
* [Services](https://gradle.org/services/)

Subscribe for important Gradle updates and news

Subscribe

By entering your email, you agree to our [Terms](https://gradle.org/terms/) and [Privacy Policy](https://gradle.org/privacy/), including receipt of emails. You can unsubscribe at any time.

© [Gradle Inc.](https://gradle.com)2018 All rights reserved.

[Careers](https://gradle.com/careers) | [Privacy](https://gradle.org/privacy) | [Terms of Service](https://gradle.org/terms) | [Contact](https://gradle.org/contact/)