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.compile.CompileOptions.html#N235AD)
  + [Methods](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#N236A7)

### 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)

CompileOptions

**Table of Contents**

[Properties](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#N235AD)[Methods](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#N236A7)[Script blocks](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#N236DD)[Property details](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#N236E2)[Method details](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#N238E3)

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

Main options for Java compilation.

Properties

| Property | Description |
| --- | --- |
| [annotationProcessorGeneratedSourcesDirectory](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:annotationProcessorGeneratedSourcesDirectory) | Incubating  The directory to place source files generated by annotation processors. |
| [annotationProcessorPath](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:annotationProcessorPath) | The classpath to use to load annotation processors. This path is also used for annotation processor discovery. If set to null, it means use the compile classpath. |
| [bootClasspath](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:bootClasspath) | Deprecated  The bootstrap classpath to be used for the compiler process. Defaults to null. |
| [bootstrapClasspath](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:bootstrapClasspath) | The bootstrap classpath to be used for the compiler process. Defaults to null. |
| [compilerArgs](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:compilerArgs) | Returns any additional arguments to be passed to the compiler. Defaults to the empty list. Compiler arguments not supported by the DSL can be added here. For example, it is possible to pass the --release option of JDK 9: |
| [compilerArgumentProviders](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:compilerArgumentProviders) | Incubating  Compiler argument providers. |
| [debug](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:debug) | Tells whether to include debugging information in the generated class files. Defaults to true. See [DebugOptions.getDebugLevel()](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/DebugOptions.html#getDebugLevel--) for which debugging information will be generated. |
| [debugOptions](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:debugOptions) | Returns options for generating debugging information. |
| [deprecation](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:deprecation) | Tells whether to log details of usage of deprecated members or classes. Defaults to false. |
| [encoding](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:encoding) | The character encoding to be used when reading source files. Defaults to null, in which case the platform default encoding will be used. |
| [extensionDirs](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:extensionDirs) | The extension dirs to be used for the compiler process. Defaults to null. |
| [failOnError](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:failOnError) | Tells whether to fail the build when compilation fails. Defaults to true. |
| [fork](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:fork) | Tells whether to run the compiler in its own process. Note that this does not necessarily mean that a new process will be created for each compile task. Defaults to false. |
| [forkOptions](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:forkOptions) | Returns options for running the compiler in a child process. |
| [headerOutputDirectory](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:headerOutputDirectory) | Incubating  If this option is set to a non-null directory, it will be passed to the Java compiler's `-h` option, prompting it to generate native headers to that directory. |
| [incremental](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:incremental) | informs whether to use incremental compilation feature. See [CompileOptions.setIncremental(boolean)](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/CompileOptions.html#setIncremental-boolean-) |
| [listFiles](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:listFiles) | Tells whether to log the files to be compiled. Defaults to false. |
| [sourcepath](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:sourcepath) | Incubating  The source path to use for the compilation. |
| [verbose](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:verbose) | Tells whether to produce verbose output. Defaults to false. |
| [warnings](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:warnings) | Tells whether to log warning messages. The default is true. |

Methods

| Method | Description |
| --- | --- |
| [debug](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:debug(java.util.Map))(debugArgs) | Convenience method to set [DebugOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/DebugOptions.html) with named parameter syntax. Calling this method will set debug to true. |
| [fork](http://docs.google.com/org.gradle.api.tasks.compile.CompileOptions.html#org.gradle.api.tasks.compile.CompileOptions:fork(java.util.Map))(forkArgs) | Convenience method to set [ForkOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/ForkOptions.html) with named parameter syntax. Calling this method will set fork to true. |

Script blocks

No script blocks

Property details

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

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

The directory to place source files generated by annotation processors.

Default:null

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

The classpath to use to load annotation processors. This path is also used for annotation processor discovery. If set to null, it means use the compile classpath.

Default:null

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

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 bootstrap classpath to be used for the compiler process. Defaults to null.

Default:null

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

The bootstrap classpath to be used for the compiler process. Defaults to null.

Default:null

[**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)> compilerArgs

Returns any additional arguments to be passed to the compiler. Defaults to the empty list. Compiler arguments not supported by the DSL can be added here. For example, it is possible to pass the --release option of JDK 9:

compilerArgs.addAll(['--release', '7'])

Note that if --release is added then -target and -source are ignored.

Default:[]

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

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

Compiler argument providers.

Default:[]

boolean debug

Tells whether to include debugging information in the generated class files. Defaults to true. See [DebugOptions.getDebugLevel()](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/DebugOptions.html#getDebugLevel--) for which debugging information will be generated.

Default:true

[**DebugOptions**](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/DebugOptions.html) debugOptions

Returns options for generating debugging information.

boolean deprecation

Tells whether to log details of usage of deprecated members or classes. Defaults to false.

Default:false

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

The character encoding to be used when reading source files. Defaults to null, in which case the platform default encoding will be used.

Default:null

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

The extension dirs to be used for the compiler process. Defaults to null.

Default:null

boolean failOnError

Tells whether to fail the build when compilation fails. Defaults to true.

Default:true

boolean fork

Tells whether to run the compiler in its own process. Note that this does not necessarily mean that a new process will be created for each compile task. Defaults to false.

Default:false

[**ForkOptions**](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/ForkOptions.html) forkOptions

Returns options for running the compiler in a child process.

[**DirectoryProperty**](http://docs.google.com/javadoc/org/gradle/api/file/DirectoryProperty.html) headerOutputDirectory (read-only)

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

If this option is set to a non-null directory, it will be passed to the Java compiler's `-h` option, prompting it to generate native headers to that directory.

Default:null

boolean incremental

informs whether to use incremental compilation feature. See [CompileOptions.setIncremental(boolean)](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/CompileOptions.html#setIncremental-boolean-)

Default:false

boolean listFiles

Tells whether to log the files to be compiled. Defaults to false.

Default:false

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

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

The source path to use for the compilation.

The source path indicates the location of source files that *may* be compiled if necessary. It is effectively a complement to the class path, where the classes to be compiled against are in source form. It does *not* indicate the actual primary source being compiled.

The source path feature of the Java compiler is rarely needed for modern builds that use dependency management.

The default value for the source path is null, which indicates an *empty* source path. Note that this is different to the default value for the -sourcepath option for javac, which is to use the value specified by -classpath. If you wish to use any source path, it must be explicitly set.

Default:null

boolean verbose

Tells whether to produce verbose output. Defaults to false.

Default:false

boolean warnings

Tells whether to log warning messages. The default is true.

Default:true

Method details

[**CompileOptions**](http://docs.google.com/dsl/org.gradle.api.tasks.compile.CompileOptions.html) debug([**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)> debugArgs)

Convenience method to set [DebugOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/DebugOptions.html) with named parameter syntax. Calling this method will set debug to true.

[**CompileOptions**](http://docs.google.com/dsl/org.gradle.api.tasks.compile.CompileOptions.html) fork([**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)> forkArgs)

Convenience method to set [ForkOptions](http://docs.google.com/javadoc/org/gradle/api/tasks/compile/ForkOptions.html) with named parameter syntax. Calling this method will set fork to true.

**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/)