**[GRADLE](https://docs.gradle.org/current/userguide/plugin_reference.html" \l "jvm_languages_and_frameworks) Plugins**

[JVM languages and frameworks](https://docs.gradle.org/current/userguide/plugin_reference.html#jvm_languages_and_frameworks)

[**Java**](https://docs.gradle.org/current/userguide/java_plugin.html)

Provides support for building any type of Java project.

[**Java Library**](https://docs.gradle.org/current/userguide/java_library_plugin.html)

Provides support for building a Java library.

[**Java Platform**](https://docs.gradle.org/current/userguide/java_platform_plugin.html)

Provides support for building a Java platform.

[**Groovy**](https://docs.gradle.org/current/userguide/groovy_plugin.html)

Provides support for building any type of [Groovy](https://groovy-lang.org/) project.

[**Scala**](https://docs.gradle.org/current/userguide/scala_plugin.html)

Provides support for building any type of [Scala](https://www.scala-lang.org/) project.

[**ANTLR**](https://docs.gradle.org/current/userguide/antlr_plugin.html)

Provides support for generating parsers using [ANTLR](http://www.antlr.org/).

[Native languages](https://docs.gradle.org/current/userguide/plugin_reference.html#native_languages)

[**C++ Application**](https://docs.gradle.org/current/userguide/cpp_application_plugin.html)

Provides support for building C++ applications on Windows, Linux, and macOS.

[**C++ Library**](https://docs.gradle.org/current/userguide/cpp_library_plugin.html)

Provides support for building C++ libraries on Windows, Linux, and macOS.

[**C++ Unit Test**](https://docs.gradle.org/current/userguide/cpp_unit_test_plugin.html)

Provides support for building and running C++ executable-based tests on Windows, Linux, and macOS.

[**Swift Application**](https://docs.gradle.org/current/userguide/swift_application_plugin.html)

Provides support for building Swift applications on Linux and macOS.

[**Swift Library**](https://docs.gradle.org/current/userguide/swift_library_plugin.html)

Provides support for building Swift libraries on Linux and macOS.

[**XCTest**](https://docs.gradle.org/current/userguide/xctest_plugin.html)

Provides support for building and running XCTest-based tests on Linux and macOS.

[Packaging and distribution](https://docs.gradle.org/current/userguide/plugin_reference.html#packaging_and_distribution)

[**Application**](https://docs.gradle.org/current/userguide/application_plugin.html)

Provides support for building JVM-based, runnable applications.

[**WAR**](https://docs.gradle.org/current/userguide/war_plugin.html)

Provides support for building and packaging WAR-based Java web applications.

[**EAR**](https://docs.gradle.org/current/userguide/ear_plugin.html)

Provides support for building and packaging Java EE applications.

[**Maven Publish**](https://docs.gradle.org/current/userguide/publishing_maven.html)

Provides support for [publishing artifacts](https://docs.gradle.org/current/userguide/publishing_setup.html) to Maven-compatible repositories.

[**Ivy Publish**](https://docs.gradle.org/current/userguide/publishing_ivy.html)

Provides support for [publishing artifacts](https://docs.gradle.org/current/userguide/publishing_setup.html) to Ivy-compatible repositories.

[**Legacy Maven Plugin**](https://docs.gradle.org/current/userguide/maven_plugin.html)

Provides support for publishing artifacts using the [legacy mechanism](https://docs.gradle.org/current/userguide/artifact_management.html) to Maven-compatible repositories.

[**Distribution**](https://docs.gradle.org/current/userguide/distribution_plugin.html)

Makes it easy to create ZIP and tarball distributions of your project.

[**Java Library Distribution**](https://docs.gradle.org/current/userguide/java_library_distribution_plugin.html)

Provides support for creating a ZIP distribution of a Java library project that includes its runtime dependencies.

[Code analysis](https://docs.gradle.org/current/userguide/plugin_reference.html#code_analysis)

[**Checkstyle**](https://docs.gradle.org/current/userguide/checkstyle_plugin.html)

Performs quality checks on your project’s Java source files using [Checkstyle](https://checkstyle.org/index.html) and generates associated reports.

[**PMD**](https://docs.gradle.org/current/userguide/pmd_plugin.html)

Performs quality checks on your project’s Java source files using [PMD](http://pmd.github.io/) and generates associated reports.

[**JaCoCo**](https://docs.gradle.org/current/userguide/jacoco_plugin.html)

Provides code coverage metrics for your Java project using [JaCoCo](http://www.eclemma.org/jacoco/).

[**CodeNarc**](https://docs.gradle.org/current/userguide/codenarc_plugin.html)

Performs quality checks on your Groovy source files using [CodeNarc](http://codenarc.sourceforge.net/index.html) and generates associated reports.

[IDE integration](https://docs.gradle.org/current/userguide/plugin_reference.html#ide_integration)

[**Eclipse**](https://docs.gradle.org/current/userguide/eclipse_plugin.html)

Generates Eclipse project files for the build that can be opened by the IDE. This set of plugins can also be used to fine tune [Buildship’s](http://projects.eclipse.org/projects/tools.buildship) import process for Gradle builds.

[**IntelliJ IDEA**](https://docs.gradle.org/current/userguide/idea_plugin.html)

Generates IDEA project files for the build that can be opened by the IDE. It can also be used to fine tune IDEA’s import process for Gradle builds.

[**Visual Studio**](https://docs.gradle.org/current/userguide/visual_studio_plugin.html)

Generates Visual Studio solution and project files for build that can be opened by the IDE.

[**Xcode**](https://docs.gradle.org/current/userguide/xcode_plugin.html)

Generates Xcode workspace and project files for the build that can be opened by the IDE.

[Utility](https://docs.gradle.org/current/userguide/plugin_reference.html#utility)

[**Base**](https://docs.gradle.org/current/userguide/base_plugin.html)

Provides common lifecycle tasks, such as clean, and other features common to mostbuilds.

[**Build Init**](https://docs.gradle.org/current/userguide/build_init_plugin.html)

Generates a new Gradle build of a specified type, such as a Java library. It can also generate a build script from a Maven POM — see [*Migrating from Maven to Gradle*](https://guides.gradle.org/migrating-from-maven/) for more details.

[**Signing**](https://docs.gradle.org/current/userguide/signing_plugin.html)

Provides support for digitally signing generated files and artifacts.

[**Plugin Development**](https://docs.gradle.org/current/userguide/java_gradle_plugin.html)

Makes it easier to develop and publish a Gradle plugin.

[**Project Report Plugin**](https://docs.gradle.org/current/userguide/project_report_plugin.html)

Helps to generate reports containing useful information about your build.