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)

[PDF](http://docs.google.com/userguide/userguide.pdf)

* [User Manual Home](http://docs.google.com/userguide/userguide.html)
* [Release Notes](http://docs.google.com/release-notes.html)
* [Installing Gradle](http://docs.google.com/userguide/installation.html)
* [Tutorials](https://guides.gradle.org/)

### Reference

* [Groovy DSL Reference](http://docs.google.com/dsl/)
* [Gradle API Javadoc](http://docs.google.com/javadoc/)
* [Core Plugins](http://docs.google.com/userguide/plugin_reference.html)
* [Gradle & Third-party Tools](http://docs.google.com/userguide/third_party_integration.html)

### Getting Started

* [Creating New Gradle Builds](https://guides.gradle.org/creating-new-gradle-builds/)
* [Creating Build Scans](https://guides.gradle.org/creating-build-scans/)
* [Migrating From Maven](https://guides.gradle.org/migrating-from-maven/)

### Running Gradle Builds

* [Command-Line Interface](http://docs.google.com/userguide/command_line_interface.html)
* [Customizing Execution](#gjdgxs)
  + [Configuring the Build Environment](http://docs.google.com/userguide/build_environment.html)
  + [Configuring the Gradle Daemon](http://docs.google.com/userguide/gradle_daemon.html)
  + [Initialization Scripts](http://docs.google.com/userguide/init_scripts.html)
* [Directory Layout](http://docs.google.com/userguide/directory_layout.html)
* [Executing Multi-Project Builds](http://docs.google.com/userguide/intro_multi_project_builds.html)
* [Gradle Wrapper](http://docs.google.com/userguide/gradle_wrapper.html)
* [Troubleshooting](http://docs.google.com/userguide/troubleshooting.html)
* [Using Build Scans](https://docs.gradle.com/build-scan-plugin)
* [Enabling and Configuring the Build Cache](http://docs.google.com/userguide/build_cache.html)
* [Integrating Separate Gradle Builds (Composite Builds)](http://docs.google.com/userguide/composite_builds.html)

### Authoring Gradle Builds

* [Fundamentals](#30j0zll)
  + [Introducing the Basics of Build Scripts](http://docs.google.com/userguide/tutorial_using_tasks.html)
  + [Working with Tasks](http://docs.google.com/userguide/more_about_tasks.html)
  + [Learning More About Build Scripts](http://docs.google.com/userguide/writing_build_scripts.html)
  + [Working with Files](http://docs.google.com/userguide/working_with_files.html)
  + [Creating Custom Task Types](http://docs.google.com/userguide/custom_tasks.html)
  + [Using Gradle Plugins](http://docs.google.com/userguide/plugins.html)
  + [The Standard Gradle Plugins](http://docs.google.com/userguide/standard_plugins.html)
  + [Understanding the Build Lifecycle](http://docs.google.com/userguide/build_lifecycle.html)
  + [Working with Logging](http://docs.google.com/userguide/logging.html)
  + [Configuring Multi-Project Builds](http://docs.google.com/userguide/multi_project_builds.html)
* [Best Practices](#1fob9te)
  + [Authoring Maintainable Build Scripts](http://docs.google.com/userguide/authoring_maintainable_build_scripts.html)
  + [Organizing Gradle Projects](http://docs.google.com/userguide/organizing_gradle_projects.html)
  + [Optimizing Build Performance](https://guides.gradle.org/performance/)
  + [Using the Build Cache](https://guides.gradle.org/using-build-cache/)
* [Dependency Management](#3znysh7)
  + [Introduction to Dependency Management](http://docs.google.com/userguide/introduction_dependency_management.html)
  + [Dependency Management Terminology](http://docs.google.com/userguide/dependency_management_terminology.html)
  + [Dependency Types](http://docs.google.com/userguide/dependency_types.html)
  + [Repository Types](http://docs.google.com/userguide/repository_types.html)
  + [Declaring Dependencies](http://docs.google.com/userguide/declaring_dependencies.html)
  + [Declaring Repositories](http://docs.google.com/userguide/declaring_repositories.html)
  + [Inspecting Dependencies](http://docs.google.com/userguide/inspecting_dependencies.html)
  + [Managing Dependency Configurations](http://docs.google.com/userguide/managing_dependency_configurations.html)
  + [Managing Transitive Dependencies](http://docs.google.com/userguide/managing_transitive_dependencies.html)
  + [Dependency Locking](http://docs.google.com/userguide/dependency_locking.html)
  + [Troubleshooting Dependency Resolution](http://docs.google.com/userguide/troubleshooting_dependency_resolution.html)
  + [Customizing Dependency Resolution Behavior](http://docs.google.com/userguide/customizing_dependency_resolution_behavior.html)
  + [Dependency Cache Internals](http://docs.google.com/userguide/dependency_cache.html)
  + [Working with Dependencies](http://docs.google.com/userguide/working_with_dependencies.html)
* [Publishing Artifacts](http://docs.google.com/userguide/artifact_management.html)
* [C++ Projects](#2et92p0)
  + [Building Native Software](http://docs.google.com/userguide/native_software.html)
  + [Software Model Concepts](http://docs.google.com/userguide/software_model_concepts.html)
  + [Rule-based Model Configuration](http://docs.google.com/userguide/software_model.html)
  + [Implementing Model Rules in a Plugin](http://docs.google.com/userguide/rule_source.html)
  + [Extending the Software Model](http://docs.google.com/userguide/software_model_extend.html)
* [Java Projects](#tyjcwt)
  + [Building Java & JVM projects](http://docs.google.com/userguide/building_java_projects.html)
  + [Testing Java & JVM projects](http://docs.google.com/userguide/java_testing.html)
* [Advanced Techniques](#3dy6vkm)
  + [Configuring Tasks Lazily](http://docs.google.com/userguide/lazy_configuration.html)
  + [Developing Parallel Tasks](https://guides.gradle.org/using-the-worker-api/)
  + [Testing Your Build with TestKit](http://docs.google.com/userguide/test_kit.html)
  + [Using Ant from Gradle](http://docs.google.com/userguide/ant.html)
* [Sample Gradle builds](#1t3h5sf)
  + [Groovy DSL Samples](https://github.com/gradle/gradle/tree/master/subprojects/docs/src/samples)
  + [Kotlin DSL Samples](https://github.com/gradle/kotlin-dsl/tree/master/samples)

### Extending Gradle

* [Writing Custom Plugins](http://docs.google.com/userguide/custom_plugins.html)
* [Plugin Development Guides](https://gradle.org/guides/?q=Plugin+Development)

[Edit this page](https://github.com/gradle/gradle/edit/master/subprojects/docs/src/docs/userguide/)

# The Application Plugin

Contents

[Usage](#4d34og8)

[Tasks](#2s8eyo1)

[Convention properties](#17dp8vu)

The Application plugin facilitates creating an executable JVM application. It makes it easy to start the application locally during development, and to package the application as a TAR and/or ZIP including operating system specific start scripts.

Applying the Application plugin also implicitly applies the [Java plugin](http://docs.google.com/java_plugin.html#java_plugin). The main source set is effectively the “application”.

Applying the Application plugin also implicitly applies the [Distribution plugin](http://docs.google.com/distribution_plugin.html#distribution_plugin). A main distribution is created that packages up the application, including code dependencies and generated start scripts.

[Usage](#4d34og8)

To use the application plugin, include the following in your build script:

[Example: Using the application plugin](#3rdcrjn)

**build.gradle**

apply plugin: 'application'

The only mandatory configuration for the plugin is the specification of the main class (i.e. entry point) of the application.

[Example: Configure the application main class](#26in1rg)

**build.gradle**

mainClassName = "org.gradle.sample.Main"

You can run the application by executing the run task (type: [JavaExec](http://docs.google.com/javadoc/org.gradle.api.tasks.JavaExec.html)). This will compile the main source set, and launch a new JVM with its classes (along with all runtime dependencies) as the classpath and using the specified main class. You can launch the application in debug mode with gradle run --debug-jvm (see [JavaExec.setDebug(boolean)](http://docs.google.com/javadoc/org/gradle/api/tasks/JavaExec.html#setDebug-boolean-)).

Since Gradle 4.9, the command line arguments can be passed with --args. For example, if you want to launch the application with command line arguments foo --bar, you can use gradle run --args="foo --bar" (see [JavaExec.setArgsString(java.lang.String)](http://docs.google.com/javadoc/org/gradle/api/tasks/JavaExec.html#setArgsString-java.lang.String-).

If your application requires a specific set of JVM settings or system properties, you can configure the applicationDefaultJvmArgs property. These JVM arguments are applied to the run task and also considered in the generated start scripts of your distribution.

[Example: Configure default JVM settings](#lnxbz9)

**build.gradle**

applicationDefaultJvmArgs = ["-Dgreeting.language=en"]

If your application’s start scripts should be in a different directory than bin, you can configure the executableDir property.

[Example: Configure custom directory for start scripts](#35nkun2)

**build.gradle**

executableDir = "custom\_bin\_dir"

[The distribution](#1ksv4uv)

A distribution of the application can be created, by way of the [Distribution plugin](http://docs.google.com/distribution_plugin.html#distribution_plugin) (which is automatically applied). A main distribution is created with the following content:

Table 1. Distribution content

| **Location** | **Content** |
| --- | --- |
| (root dir) | src/dist |
| lib | All runtime dependencies and main source set class files. |
| bin | Start scripts (generated by createStartScripts task). |

Static files to be added to the distribution can be simply added to src/dist. More advanced customization can be done by configuring the [CopySpec](http://docs.google.com/javadoc/org.gradle.api.file.file.CopySpec.html) exposed by the main distribution.

[Example: Include output from other tasks in the application distribution](#44sinio)

**build.gradle**

task createDocs {  
 def docs = file("$buildDir/docs")  
 outputs.dir docs  
 doLast {  
 docs.mkdirs()  
 new File(docs, "readme.txt").write("Read me!")  
 }  
}  
  
distributions {  
 main {  
 contents {  
 from(createDocs) {  
 into "docs"  
 }  
 }  
 }  
}

By specifying that the distribution should include the task’s output files (see [more about tasks](http://docs.google.com/more_about_tasks.html#sec:task_inputs_outputs)), Gradle knows that the task that produces the files must be invoked before the distribution can be assembled and will take care of this for you.

[Example: Automatically creating files for distribution](#2jxsxqh)

**Output of gradle distZip**

> gradle distZip  
> Task :createDocs  
> Task :compileJava  
> Task :processResources NO-SOURCE  
> Task :classes  
> Task :jar  
> Task :startScripts  
> Task :distZip  
  
BUILD SUCCESSFUL in 0s  
5 actionable tasks: 5 executed

You can run gradle installDist to create an image of the application in build/install/*projectName*. You can run gradle distZip to create a ZIP containing the distribution, gradle distTar to create an application TAR or gradle assemble to build both.

[Customizing start script generation](#z337ya)

The application plugin can generate Unix (suitable for Linux, macOS etc.) and Windows start scripts out of the box. The start scripts launch a JVM with the specified settings defined as part of the original build and runtime environment (e.g. JAVA\_OPTS env var). The default script templates are based on the same scripts used to launch Gradle itself, that ship as part of a Gradle distribution.

The start scripts are completely customizable. Please refer to the documentation of [CreateStartScripts](http://docs.google.com/dsl/org.gradle.jvm.application.tasks.CreateStartScripts.html) for more details and customization examples.

[Tasks](#2s8eyo1)

The Application plugin adds the following tasks to the project.

run — [JavaExec](http://docs.google.com/dsl/org.gradle.api.tasks.JavaExec.html)

*Depends on*: classes

Starts the application.

startScripts — [CreateStartScripts](http://docs.google.com/dsl/org.gradle.api.tasks.application.tasks.application.CreateStartScripts.html)

*Depends on*: jar

Creates OS specific scripts to run the project as a JVM application.

installDist — [Sync](http://docs.google.com/dsl/org.gradle.api.tasks.Sync.html)

*Depends on*: jar, startScripts

Installs the application into a specified directory.

distZip — [Zip](http://docs.google.com/dsl/org.gradle.api.tasks.bundling.tasks.bundling.Zip.html)

*Depends on*: jar, startScripts

Creates a full distribution ZIP archive including runtime libraries and OS specific scripts.

distTar — [Tar](http://docs.google.com/dsl/org.gradle.api.tasks.bundling.tasks.bundling.Tar.html)

*Depends on*: jar, startScripts

Creates a full distribution TAR archive including runtime libraries and OS specific scripts.

[Convention properties](#17dp8vu)

The application plugin adds some properties to the project, which you can use to configure its behaviour. See the [Project](http://docs.google.com/dsl/org.gradle.api.Project.html) class in the API documentation.

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/)