Gradle: overview

Gradle is a modern automation tool that helps build and manage projects written in Java, Kotlin, Scala, and other JVM-based languages.
It describes project dependencies and determines how to build a project.

Gradle uses a well-designed plugin system, which is why it is a highly extensible tool.

You can use plugins for automatic versioning, automatic testing, reporting about the build, and so on.

One of the best things about Gradle is its Groovy-based domain-specific language (DSL) that gives developers a specific way to form custom build scripts.

The Kotlin developers are especially lucky since Gradle also started to support Kotlin for such scripts. So, there are two languages to write the Gradle build scripts (Groovy and Kotlin) and you can choose any of them.

Please, note, that DSL is a computer language specialized to a particular application domain (like build automation). This is in contrast to a general-purpose language (GPL), which is broadly applicable across domains.

Nowadays, Gradle is a de facto standard for building Android applications.

However, programmers use it for server-side and desktop development gradually replacing other build tools with it. Here is the full list of features, if you want to get more information on them.

The key concepts

The key features of Gradle are as follows:

- **Settings files.** Gradle uses several types of settings files to describe how to build a project.
- **Build-by-convention.** A programmer doesn't need to specify every building step that needs to be done. Instead, Gradle uses default settings and behavior. Although, every step of the default build process can be customized if necessary.
- **Dependency management.** Gradle automatically downloads specified external libraries and solves conflict cases with dependencies. You can declare as many dependencies as you need for a project.
- **Builds**. Gradle allows programmers to design well-structured and easily maintained comprehensible builds. It also supports complex cases such as multi-project or partial builds.
- **Ease of migration**. Gradle can easily adapt to any project structure

you have. Therefore, you can always develop your project exactly the way you want.

- **DSL** (based on Groovy and Kotlin) for writing scripts in settings files.

Downloading and installing Gradle

You can download Gradle from the official website and unpack the archive somewhere on your computer. We recommend choosing version 5.0 or higher. To install Gradle, follow the installation instructions for your operating system.

To verify that the installation has been completed correctly, run the following command:

```
gradle -v
```

The result should look similar to:

```
------------------------------------------------------------
Gradle 5.6.4
------------------------------------------------------------
```

Your version of Gradle may differ; the main thing is that the command should successfully work.