Gradle tutorial

1. Testing gradle installation

A screenshot of a computer

Description automatically generated

1. Now, we will create a new directory called tutorial and cd into it

C:\tutorial>

1. Now this next step is impt

A screenshot of a computer program

Description automatically generated

1. The folder now looks like this

A screenshot of a computer

Description automatically generated

1. Now , we see the gradlew.bat present in the above folder

So, we want to invoke it

A screenshot of a computer program

Description automatically generated

1. What happened?

A build directory was generated

A screenshot of a computer

Description automatically generated

1. All the files generated by the build process go into the build directory unless otherwise specified.

Standard gradle’s project structure looks as below

A screenshot of a computer

Description automatically generated

1. There is a settings.gradle right at the top
2. There is a root project
3. One or more subprojects – and each subproject will have its own build.gradle file
4. Comparing the above with our own generated project

A screenshot of a computer

Description automatically generated

1. There is a settings.gradle
2. The root project is tutorial
3. App is the subproject
4. So now – what is a build ?

A build is a bundle of related software that you can build, test and release together.

1. So now – what is a project ?

Represents a single piece of architecture – eg. App, a library etc

1. Now , we open the project in intellij

A screenshot of a computer

Description automatically generated

1. Now, we open the settings.gradle file and the build.gradle file side by side

A screenshot of a computer program

Description automatically generated

The presence of the word “include” turns the app subdirectory into a subproject!

1. Now, we understand the build script

A screenshot of a computer program

Description automatically generated

1. Now, we look at how to run gradle tasks.

What is a task? – a task is a basic unit of work that can be done by gradle as part of the build!

Is there a way to find out all the available tasks in the project? Absolutely

A screenshot of a computer

Description automatically generated

You can also list the tasks only available in the app subproject by running ./gradlew :app:tasks.

A screenshot of a computer program

Description automatically generated

1. Is there a way to create own tasks ?

Absolutely

A screenshot of a computer program

Description automatically generated

1. Now, lets say we want to run only this task.

A screenshot of a computer

Description automatically generated

1. Now lets say we have another task and we want that task to run only after the above task has run

A screenshot of a computer program

Description automatically generated

1. Now, lets run the task helloNish2

A screenshot of a computer

Description automatically generated

1. Now, lets say we want the build task , which is out of the box, to depend on my helloNish task
2. The below is how to do that

A screenshot of a computer program

Description automatically generated

1. Now, lets run the gradlew build task

A black screen with white text

Description automatically generated

1. Is there a way to view tasks in the IDE? Why not !

A screenshot of a computer

Description automatically generated

1. You can run tasks via intellij by double clicking any of the above shown tasks.
2. Now, I ran build from the ide

A screen shot of a computer

Description automatically generated

As we can see from the above, the build task DEPENDS ON the jar task

The jar task creates an executable jar file of the app

Since it is a task, we can run it by itself , as per below

1. We can see the jar being created

A screenshot of a computer

Description automatically generated

1. Now, we invoke the run task

A screen shot of a computer

Description automatically generated