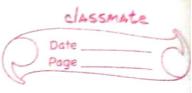# GRADLE

- Build tool which automates a lot of the compilation & running processes required for JAVA applications.

- Uses 'tasks' to achieve different objectives
  - gradlew <task-name>
  - gradlew tasks
    → lists all tasks

- Plugins —— tools that help add tasks to the gradle project ↗ core gradle plugin.

  plugins {
      id 'java'
  }

  - Configures the project in a certain way & adds additional tasks.

  compileJava → *.java to .class, in build/
  processResources → copies resources into build/
  jar → compiled classes + resources to a
  ~~test~~   jar archive'.
  test → compiles test, process resources,
           run tests & creates a test
           report

Define dependencies:

```
dependencies {
    testImplementation "org.junit.jupiter:junit-
                        jupiter-api:5.6.0"
}
```

group → jupiter   name → junit

version → 5.6.0

Gradle Java project layout:

The plugin 'java' expects java files to
be found in:

	src/main/java	⇒ build/classes/java/main
resources in	src/main/resources	⇒ build/resources/main
test.java in	src/test/java	⇒ build/classes/java/test
test resources in	src/test/resources	⇒ build/resources/test

- Configurable, but try to use default

Groovy

- Scripting language running in JVM
- extends Java, making dev. quicker
- build.gradle written in Groovy.
- Java can be written in Groovy.
- Brackets are optional if the function has
  one or more arguments

void dependencies (Closure var 1)

Task graph:

. Tasks can depend on other tasks

      Build

   Assemble   Check.

. 2 types of tasks

      ↳ perform an action

      ↳ aggregate tasks