

Directory structure

- Production source code---> `src/main/java`
- Test source code ---> `src/test/java`
- JAR files as resources --> `src/main/resources`
- Classpath for test run --> `src/test/resource`
- Output files ---> `build/libs`

Building the project

- `gradle build` ---> compile and test the code, and create a JAR file containing main classes and resources.
- `clean` ---> Deletes the build directory.
- `assemble` ---> Compiles and jars code. `artifacts` If uses the WAR plugin, it will build the WAR file
- `check` ---> compiles and test the code.

External dependency

Java project have some dependency on external jar files, to reference these jar files, need to tell the gradle where to find these files. In gradle artifacts such as jar files are located in `repository`.

`repository` can be used for fetching the dependencies of project, or publishing the artifacts of the project.