

1) Look up java plugin documentation. Make changes in manifest to make it executable with correct class. When run using `java -jar JAR_NAME_HERE` the output should be text "Hello World" on the console.

```
ERROR: Could not find or load main class Application
ttn@ttn:libs (master)$ java -jar GradleAssignment-1.0.jar
Hello World!
ttn@ttn:libs (master)$
```

2) look up idea plugin. make changes in build.gradle so that the sources of `src/main/java` as well as `src/main/java2` are taken as sources. Ensure that when you make JAR file class files in both are added to the JAR. This will teach you how projects with non-conventional structure can be used with gradle.

```
Java 2 class method
ttn@ttn:libs (master)$ java -jar GradleAssignment-1.0.jar
Hello World!
Java 1 class method
Java 2 class method
```

3) add 2 files `file1.xml` and `file1.txt` in `src/main/resources` manually. make changes so that when creating jar only `file1.xml` is added to the jar.

Output as per 'before excluding' and 'after excluding':

```
ttn@ttn:libs (master)$ jar tf GradleAssignment-1.0.jar
META-INF/
META-INF/MANIFEST.MF
Java1.class
Application.class
Java2.class
File1.xml
File1.txt
ttn@ttn:libs (master)$ jar tf GradleAssignment-1.0.jar
META-INF/
META-INF/MANIFEST.MF
Java1.class
Application.class
Java2.class
File1.xml
ttn@ttn:libs (master)$
```

4) find how to what is an uberjar. Make changes so you can use commons lang3 `StringUtil` in your jar. Make this uber jar executable. The output should be text but that should be using the `StringUtils` class of commons lang3

```
ttn@ttn:libs (master)$ java -jar GradleAssignment-1.0.jar
Hello World!
Java 1 class method
Java 2 class method
olleH
ttn@ttn:libs (master)$
```

5) Find a maven repository and add it as a repository. You can use bintray, jcenter

Done in build.gradle.

6) Write a task in file "mytasks.gradle" and use it in your build.gradle

```
I actionable task: 1 executed
ttn@ttn:GradleAssignment (master)$ gradle myTask

> Configure project :
I am a task from mytasks.gradle file

Deprecated Gradle features were used in this build, making it incompatible with
Gradle 5.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/4.10.2/userguide/command_line_interface.html#sec:com
mand_line_warnings

BUILD SUCCESSFUL in 0s
```

Resources:

<https://www.baeldung.com/gradle-fat-jar>