

# Create and Upload Test Framework as a Library

In this lesson, we will learn to create a framework jar and upload it to an artifactory.

## WE'LL COVER THE FOLLOWING



- Tasks to be added in `build.gradle`
- Generating the test Framework Jar
- Uploading the test Framework Jar

Below is a guide to create a distribution RELEASE version of the automation library.

For demonstration, we will be using **Gradle**.

Tasks to be added in `build.gradle` #

```
plugins {  
    id 'java'  
}  
  
group = 'com.educative.core-automation'  
version = '1.0.0-RELEASE'  
jar.baseName = 'ui-automation-lib'  
  
task javadocJar(type: Jar, dependsOn: javadoc) {  
    classifier = "javadoc"  
    from javadoc.destinationDir  
}  
  
task sourcesJar(type: Jar, dependsOn: classes) {  
    classifier = "sources"  
    from sourceSets.main.allSource  
}  
  
if (JavaVersion.current().isJava8Compatible()) {
```

```

    allprojects {
        tasks.withType(Javadoc) {

            options.addStringOption("Xdoclint:none", "-quiet")
        }
    }
}

artifacts {
    archives sourcesJar
    archives javadocJar
}

uploadArchives {
    repositories {
        mavenDeployer {
            repository(url: "<maven-url>") {
                authentication(userName: "<username>", password: "<password>")
            }
        }
    }
}

```

## Generating the test Framework Jar #

After running the below command:

```
./gradlew clean build -x test
```

The project will be cleaned, compiled and the distribution jar will be created along with Javadoc and sources at `${project.dir}/build/libs`

```

$ ls -a build/libs | cat
.
..
ui-automation-lib-1.0.0-RELEASE-javadoc.jar
ui-automation-lib-1.0.0-RELEASE-sources.jar
ui-automation-lib-1.0.0-RELEASE.jar

```

# Uploading the test Framework Jar #

Before running the command:

```
./gradlew uploadArchives
```

Make sure the `maven url`, `username` and `password` arguments are set with the right values in `uploadArchives` task found inside `build.gradle`

---

In the next lesson, we will learn how to share and distribute the framework jar for reuse across different teams and projects.