Gradle

**Prerequisites:**

Gradle Quick Learn Course

<https://docs.gradle.org/current/userguide/tutorial_java_projects.html>

Supplementary references:

<https://www.tutorialspoint.com/gradle/index.htm>

<https://guides.gradle.org/creating-new-gradle-builds/?_ga=2.146906551.1673785928.1531137357-1093184657.1530003553>

## Lecture 2

### Lecture is divided into 3 parts:

* Practical part with Maven project (plugins, goals, phases configuration in pom - advanced topic, tomcat plugin as an example)
* Practical part with Gradle project (wrapper, convert Maven to Gradle)

**Gradle Content:**

1. Tell about the Maven and Gradle config file project structures: <https://www.tutorialspoint.com/gradle/gradle_deployment.htm>
2. Take a basic pom file as an example and generate gradle build file from it by executing the command:

gradle init --type pom

1. Generate and run SpringBoot app: <https://start.spring.io/>
2. Tell about the scopes: testCompile (for compilation of the classes for running tests), runtime (for the dependent libraries to be included into the build in the runtime environment), testCompile (for compilation of the source files to be included into the bundle)
3. compile group: ‘org.spring’, name: ‘spring-core’, version: ‘1.5.4-SNAPSHOT’
4. Diff between gradle and ./gradlew: <https://stackoverflow.com/questions/39627231/difference-between-using-gradlew-and-gradle>

The embedded version is better to execute since it performs independently from the installed gradle version on the developer’s machine. That makes work synchronized.

1. Tell about the repositories: Nexus and central Maven. Show the repository structure and how to find them in Intellij.
2. Tell about the gradle tasks and how to run task with ./gradlew task\_name
3. What is a wrapper\_gradle.zip and why it is so good for some projects.
4. Plugins adds a number of new tasks to the project:

apply plugin: 'java'

This is a major plugin to compile and build the Java project

Some other plugins: apply plugin: ‘scala’

apply plugin: groovy’

1. Best java base plugin description in the resource:

Build directory (clean task removes all inside),   
 <https://docs.gradle.org/current/userguide/tutorial_java_projects.html>

Clean, assemble, check - basic tasks to clean, build jar files, run tests independently

./gradlew dependencies - builds tree of dependent libraries in the dependencies

1. Publishing command will submit the file to the Maven repository:

apply plugin: 'maven'  
  
uploadArchives {  
 repositories {  
 mavenDeployer {  
 repository(url: "file://localhost/tmp/myRepo/")  
 }  
 }  
}

Maven uploadArchive task will

1. Build a bundle with all dependencies packed in one jar (add a fatJar task):

//create a single Jar with all dependencies

task fatJar(type: Jar) {

manifest {

attributes 'Implementation-Title': 'Gradle Jar File Example',

'Implementation-Version': version,

'Main-Class': 'com.epam.functions.apps.predicates.PredicateApplication'

}

baseName = project.name + '-all'

from { configurations.compile.collect { it.isDirectory() ? it : zipTree(it) } }

with jar

}

>./gradlew farJar - will produce the jar