Maven

# Syntaxe proměnné pro verze

<properties>

<maven.compiler.source>1.8</maven.compiler.source>

<maven.compiler.target>1.8</maven.compiler.target>

**<spring-version>5.2.17.RELEASE</spring-version>**

</properties>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

**<version>${spring-version}</version>**

</dependency>

# Příkazy

1. mvn clean – Smaže adresář target
2. mvn compile – Vytvoří adresář target a zkompiluje .java soubory do .class souborů
3. mvn package – Vytvoří archiv podle toho, co zadáme v hlavičce pom souboru do tagů <packaging>??</packaging>. Můžeme vytvořit jar, war, ejb atd.

Pro archivaci je potřeba nastavit pluginy.

* 1. Pro JAR soubor:

<build>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-jar-plugin</artifactId>

<version>2.4</version>

<configuration>

<archive>

<manifest>

<mainClass>Test</mainClass>

</manifest>

</archive>

</configuration>

</plugin>

</plugins>

</build>

* 1. Pro WAR soubor:

<build>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-war-plugin</artifactId>

<version>3.3.2</version>

<configuration>

<webappDirectory>

**/Users/NcFerrari/Documents/Programování/Java/tutorial/Java/Spring/TestProject/target**

</webappDirectory>

</configuration>

</plugin>

</plugins>

</build>

Ale u warka je důležité i nastavit správně web adresář v projektu. V adresáři src/main je potřeba vytvořit adresář **webapp** (takže bude na stejné úrovni, jako adresáře **java**, **resources** a **test**). A v adresáři **webapp** vytvořte adresář **WEB-INF** a v něm soubor **web.xml**. Do tohoto souboru následně zkopírujte následující text:

<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd"

version="3.1">

</web-app>

A nakonec do adresáře **webapp** (vedle nově vytvořeného adresáře **WEB-INF**) vytvořte soubor úvodní stránky (třeba **index.html**) no a do něj už snad víte, co napsat.

1. mvn install – Tento příkaz je zásadní a vykonává v sobě spoustu dalších subpříkazů:
   1. Stáhne knihovny ze všech dependencies do adresáře .m2
   2. Zkontroluje zdrojový kód aplikace
   3. Zkompiluje zdrojový kód
   4. Zkontroluje Testy aplikace (unitové, integrační i mock)
   5. Zkompiluje Testy
   6. Spustí Testy
   7. Zarchivuje aplikaci (pokud je nastavený plugin a package)
   8. Nasadí aplikaci na prostředí (pokud je nastavený plugin)

Testy se dají přeskočit. Buď má IDE na to tlačítko a nebo příkazem mvn –DskipTests=true install