Object-Oriented Programming in Java

Guillaume Muller

based on work from:

Ch. Gravier, F. Laforest, J. Subercaze

Télécom Saint-Étienne

{pénom.nom}@univ-st-etienne.fr

14 September 2020
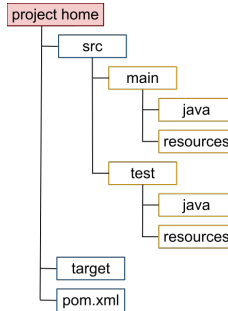
- Goal: Manage compilation chain & dependencies

Maven[1]

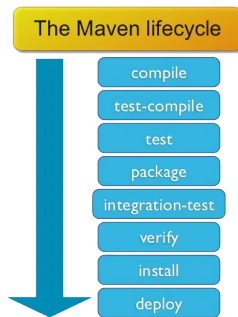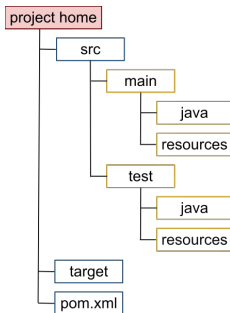- Goal: Manage compilation chain & dependencies

- Uses conventions

[1]https://dzone.com/refcardz/apache-maven-2

- Goal: Manage compilation chain & dependencies

- Uses conventions

  - Project architecture


```
project home
  └─ src
       └─ main
            └─ java
            └─ resources
       └─ test
            └─ java
            └─ resources
  └─ target
  └─ pom.xml
```

---

[1] https://dzone.com/refcardz/apache-maven-2

Maven[1]

- Goal: Manage compilation chain & dependencies

- Uses conventions
  - Project architecture
  - Pre-configured rules: clean, compile...

The Maven lifecycle

compile
test-compile
test
package
integration-test
verify
install
deploy

project home
  src
    main
      java
      resources
    test
      java
      resources
  target
  pom.xml

---

[1] https://dzone.com/refcardz/apache-maven-2

Maven[1]

- Goal: Manage compilation chain & dependencies

- Uses conventions

- Configuration via `pom.xml` file

---
[1]

Maven[1]

- Goal: Manage compilation chain & dependencies

- Uses conventions

- Configuration via `pom.xml` file

- Example usage:
  `mvn clean compile`

---

[1]

Maven[1]

- Goal: Manage compilation chain & dependencies

- Uses conventions

- Configuration via `pom.xml` file

- Example usage:
  `mvn clean compile`

- We will let Eclipse do (most of) that for us!!

---

[1]https://dzone.com/refcardz/apache-maven-2

Maven – Example `pom.xml` file

```xml
 1    <project>
 2      <!-- model version is always 4.0.0 for Maven 2.x POMs -->
 3      <modelVersion>4.0.0</modelVersion>
 4      <!-- project coordinates, i.e. a group of values which uniquely identify this project -->
 5      <groupId>com.mycompany.app</groupId>
 6      <artifactId>my-app</artifactId>
 7      <version>1.0</version>
 8      <!-- library dependencies -->
 9      <dependencies>
10        <dependency>
11          <!-- coordinates of the required library -->
12          <groupId>junit</groupId>
13          <artifactId>junit</artifactId>
14          <version>3.8.1</version>
15          <!-- this dependency is only used for running and compiling tests -->
16          <scope>test</scope>
17        </dependency>
18      </dependencies>
19    </project>
```

- archetype for simple JAva project = `maven-archetype-quickstart`

- `groupId` ≈ package

- `artifactId` ≈ main class