

< Teach
Me
Skills />

Maven. Основные возможности инструмента



Что это?

Инструмент для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM (англ. Project Object Model), являющемся подмножеством XML

Официальный веб-сайт: <https://maven.apache.org/>

- **Внутри IntelliJ Idea есть предустановленный Maven**
- [Скачать здесь](#)
- [Гайд по установке на Windows](#)
- [Гайд по установке на Mac/Unix \(English\)](#)



Аналоги



1

Gradle

2

Ant

А что умеет?



1

Зависимости (dependencies)

2

Параметры (properties)

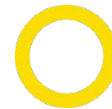
3

Управление жизненным циклом проекта (LifeCycle)

4

Запуск тестов из командной строки (CLI)

Зависимости



Библиотека библиотек:

- <https://mvnrepository.com/>

```
<dependencies>
```

```
<!-- https://mvnrepository.com/artifact/org.testng/testng -->
```

```
<dependency>
```

```
<groupId>org.testng</groupId>
```

```
<artifactId>testng</artifactId>
```

```
<version>7.0.0</version>
```

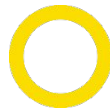
```
<scope>test</scope>
```

```
</dependency>
```

```
</dependencies>
```

Обновление версий

- ***mvn versions:display-dependency-updates***
 - команда для просмотра доступных обновлений библиотек
- ***mvn versions:use-latest-versions***
 - команда для автоматического обновления всех версий



Параметры

Внутри pom.xml можно использовать переменные для упрощения поддержки версии/других целей

```
<properties>  
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
  <java.faker.version>0.18</java.faker.version>  
  <selenide.version>5.2.3</selenide.version>  
</properties>
```

```
<!-- https://mvnrepository.com/artifact/com.codeborne/selenide -->  
<dependency>  
  <groupId>com.codeborne</groupId>  
  <artifactId>selenide</artifactId>  
  <version>${selenide.version}</version>  
  <scope>test</scope>  
</dependency>
```


Scope

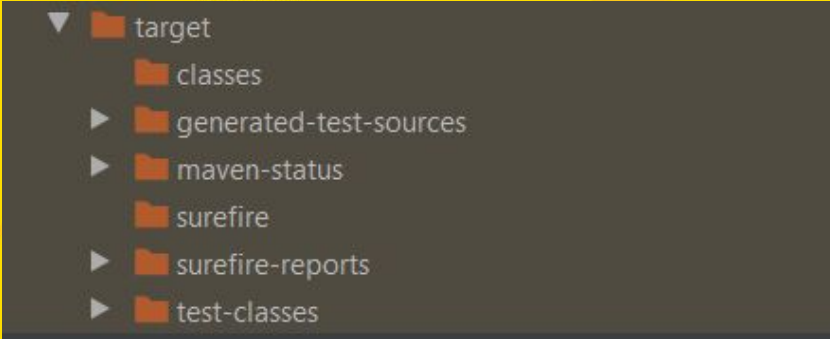
Изначально, без указания тега `<scope>` все зависимости доступны по всему проекту, т. е. вы можете сделать импорт библиотеку из класса, находящегося в `main.java` и в `test.java`

Если какая-то библиотека используется только в `test.java` - рекомендуется указать `<scope>test</scope>`

```
<!-- https://mvnrepository.com/artifact/com.codeborne/selenide -->
<dependency>
  <groupId>com.codeborne</groupId>
  <artifactId>selenide</artifactId>
  <version>${selenide.version}</version>
  <scope>test</scope>
</dependency>
```


Target






Директория, содержащая все ресурсы необходимые для сборки проекта. Из интересного: содержит автогенерируемые репорты TestNG и Junit



```
▼ target
  classes
  ► generated-test-sources
  ► maven-status
  surefire
  ► surefire-reports
  ► test-classes
```



Lifecycle






clean
test
compile
test-compile
install

Запуск тестов

[Официальная документация](#)



```
mvn -Dtest=TestCircle test
mvn -Dtest=TestCircle#mytest test
mvn
-Dtest=TestCircle#testOne+testTwo
test
mvn "-Dtest=???Test, !Unstable*,
pkg/**/Ci*leTest.java,
*Test#test*One+testTwo?????,
#fast*+slowTest" test
```

Запуск testng.xml файла

- `mvn clean test -DsuiteXmlFile=src/test/resources/SmokeTest.xml`

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-surefire-plugin</artifactId>
  <version>3.0.0-M4</version>
  <configuration>
    <suiteXmlFiles>
      <suiteXmlFile>${suiteXmlFile}</suiteXmlFile>
    </suiteXmlFiles>
  </configuration>
</plugin>
```

CLI options

[Официальная документация](#)



mvn -v

mvn -h

mvn -f {path-to-another/pom.xml}

Parameters

`mvn test -DtestProp=test123`

```
@Test
public void runTest() {
    System.out.println(System.getProperty("testProp"));
}
```