

# Шпаргалка по POM.XML

Программирование

## Шпаргалка по `pom.xml`

### 1. Основные теги POM-файла

- `<modelVersion>` — версия POM-файла.

XML

```
<modelVersion>4.0.0</modelVersion>
```

- `<groupId>` — уникальный идентификатор группы (например, организация или домен).

XML

```
<groupId>org.example</groupId>
```

- `<artifactId>` — уникальное имя артефакта (проекта).

XML

```
<artifactId>lab_5</artifactId>
```

- `<version>` — версия проекта (можно использовать `SNAPSHOT` для нестабильных версий).

XML

```
<version>1.0-SNAPSHOT</version>
```

- `<packaging>` — тип пакета (например, `jar`, `war`).

```
<packaging>jar</packaging>
```

---

## 2. <properties>

- Указывает свойства, которые используются в проекте (например, версия компилятора).

XML

```
<properties>  
  <maven.compiler.source>17</maven.compiler.source>  
  <maven.compiler.target>17</maven.compiler.target>  
</properties>
```

---

## 3. <dependencies>

- Зависимости, необходимые для проекта (библиотеки, которые скачиваются).

XML

```
<dependencies>  
  <dependency>  
    <groupId>org.slf4j</groupId>  
    <artifactId>slf4j-api</artifactId>  
    <version>1.7.30</version>  
  </dependency>  
</dependencies>
```

---

## 4. <build>

- Содержит информацию о процессе сборки, включая плагины.

### <plugins>

- Секция для конфигурации плагинов, например, для создания JAR.

XML

```
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-jar-plugin</artifactId>
      <version>3.1.1</version>
      <configuration>
        <archive>
          <manifest>
            <mainClass>org.example.Main</mainClass>
          </manifest>
        </archive>
      </configuration>
    </plugin>
  </plugins>
</build>
```

---

## 5. <repositories>

- Указывает дополнительные репозитории для скачивания зависимостей.

```
<repositories>
  <repository>
    <id>some-repository</id>
    <url>https://some.repository.url</url>
  </repository>
</repositories>
```

---

## 6. <parent>

- Указывает родительский проект, если проект является дочерним.

```
<parent>
  <groupId>org.example</groupId>
  <artifactId>parent-project</artifactId>
  <version>1.0</version>
</parent>
```

---

## 7. <profiles>

- Настройки для различных сценариев (например, для разработки или продакшн).

```
<profiles>
  <profile>
    <id>dev</id>
    <properties>
      <maven.compiler.source>1.8</maven.compiler.source>
      <maven.compiler.target>1.8</maven.compiler.target>
    </properties>
  </profile>
</profiles>
```

---

## Типичный пример pom.xml

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <groupId>org.example</groupId>
    <artifactId>lab_5</artifactId>
    <version>1.0-SNAPSHOT</version>
    <packaging>jar</packaging>

    <properties>
        <maven.compiler.source>17</maven.compiler.source>
        <maven.compiler.target>17</maven.compiler.target>
    </properties>

    <dependencies>
        <dependency>
            <groupId>org.slf4j</groupId>
            <artifactId>slf4j-api</artifactId>
            <version>1.7.30</version>
        </dependency>
    </dependencies>

    <build>
        <plugins>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-jar-plugin</artifactId>
                <version>3.1.1</version>
                <configuration>
                    <archive>
                        <manifest>
                            <mainClass>org.example.Main</mainClass>
                        </manifest>
                    </archive>
                </configuration>
            </plugin>
        </plugins>
    </build>
</project>
```

```
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>
```

---

## Пояснение к элементам:

1. `<modelVersion>` — версия формата POM-файла.
  2. `<groupId>`, `<artifactId>`, `<version>` — уникальная идентификация проекта.
  3. `<packaging>` — тип пакета, например, JAR.
  4. `<dependencies>` — список зависимостей, которые Maven должен скачать.
  5. `<build>` — настройки для сборки проекта (включая плагины).
  6. `<plugins>` — плагины для выполнения различных задач.
  7. `<repositories>` — дополнительные репозитории для скачивания зависимостей.
  8. `<parent>` — если проект наследует настройки другого проекта.
  9. `<profiles>` — различные конфигурации для разных сред.
- 

## Основные команды Maven:

- **Сборка проекта (создание JAR)**

```
mvn clean package
```

SHELL

- **Запуск проекта (JAR)**

```
java -jar target/your-artifact-id.jar
```

SHELL

