JAVAWORLD

~~reloper, JavaWorld~~
JAN 30, 2020 9:13 AM PST

About 🔊

Everything you need to know about Java programming tools and APIs, with code and examples.

FEATURE

What is Maven? Build and dependency management for Java

Get started with Maven, the most popular build and dependency management tool for Java

As you develop or maintain software, you will need to add new features to an existing project from time to time. Fortunately, Maven makes adding new features very easy. Let's say you're working on the project defined in Listing 1, and realize you need some utilities from the Apache Commons Language library. You can easily insert that dependency right into the original POM file after JUnit, as shown in Listing 3.

Listing 3. Adding a dependency

```
<dependency>
    <groupId>org.apache.commons</groupId>
    <artifactId>commons-lang3</artifactId>
    <version>3.0</version>
</dependency>
```

Finding dependencies is also easy; for this one, you could just type "maven stringutils" into a web browser, and it would turn up. Or you could search the Central Maven Repository directly. You might also find a link to a dependency you need when you're reading about it on a project homepage or other resource. Just cut and paste the URL and be sure to note the correct version number, and you're good to go.

[Do more with Maven: Learn how Maven works hand-in-hand with Jenkins, JUnit, and Docker for a classic and capable CI/CD framework.]

Conclusion

Maven is so commonly used because it works well and is mature enough to handle virtually anything you throw at it. Understanding Maven and using it as a base for your projects unlocks the vast amount of capability available to you in the Java ecosystem. To get just a taste of what's possible, consider checking out any of these projects, which are all accessible via the Maven core plugins.

Maven core plugins

maven-surefire-plugin	Runs unit tests in a dedicated class loader.
spring-boot-maven-plugin	Used for running a spring app via spring-boot:run.
maven-deploy-plugin	Used for publishing a package to a remote repository.
maven-assembly-plugin	Used for building fat or uber JARs, meaning JAR files that contain all the dependencies and assets in one file, capable of being run as-is.
maven-shade-plugin	Used for building a fat JAR that enables removing unused files or renaming clashing ones.