

Comandos más utilizados de Maven (mvn) por desarrolladores

Introducción

Maven es una herramienta que nos permite gestionar proyectos Java de forma sencilla y organizada. Hace posible compilar, empaquetar, probar y desplegar nuestras aplicaciones, todo siguiendo un flujo estándar y predecible.

A lo largo de nuestro trabajo diario como desarrolladores, es común encontrarnos con comandos que utilizamos una y otra vez para compilar, instalar dependencias o generar artefactos.

En este documento veremos de forma clara y directa cuáles son esos comandos, su función y en qué casos conviene utilizarlos.

Desarrollo

A continuación, presentamos los comandos más utilizados en Maven y su función.

- **mvn clean**
Elimina la carpeta `target` y todo lo que se haya generado en compilaciones anteriores.
Esto es útil cuando queremos asegurarnos de que el nuevo build sea completamente limpio.
- **mvn compile**
Compila el código fuente del proyecto y lo deja listo en `target/classes`.
Puede ir acompañado de `clean` para iniciar desde cero.
- **mvn test**
Ejecuta las pruebas unitarias configuradas en el proyecto.
Si lo usamos junto con `clean`, primero limpia y después prueba.
- **mvn package**
Compila, ejecuta las pruebas y genera un artefacto `.jar` o `.war` según esté definido en el `pom.xml`.
- **mvn install**
Realiza todo el proceso (compilación, pruebas, empaquetado) y además instala el artefacto en nuestro repositorio local (`~/.m2/repository`) para poder ser utilizado por otros proyectos.
- **mvn deploy**
Igual que `install`, pero además sube el artefacto a un repositorio remoto como Nexus o Artifactory.
- **mvn verify**
Ejecuta verificaciones y validaciones para asegurar que el proyecto está listo antes de ser desplegado.
- **mvn dependency:tree**
Muestra la jerarquía de dependencias del proyecto, lo cual ayuda a identificar conflictos de versiones.

- **mvn help:effective-pom**

Muestra el POM resultante después de aplicar herencias y valores por defecto.

- **mvn site**

Genera un sitio web con documentación y reportes del proyecto.

- **mvn help:describe**

Proporciona información detallada sobre un comando específico.

Ejemplo de flujo común

En el desarrollo diario es frecuente ejecutar:

```
mvn clean install
```

Esto limpia el proyecto, lo compila, corre las pruebas y deja el artefacto listo para ser usado localmente.

Conclusión

Conocer estos comandos nos permite trabajar de manera más eficiente con Maven.

Al tener claro qué hace cada uno, podemos aplicarlos en el momento adecuado y optimizar nuestro flujo de desarrollo.