

Fundamentos de Java

Maven

Maven

Proyecto open source mantenido por Apache

- Nace en 2002, desarrollado por Jason van Zyl
- En 2005 se lanzan Maven 2.
- <https://maven.apache.org/>

Maven

Herramienta que gestiona el ciclo de vida de un proyecto

- **Compilación**
- **Prueba**
- **Empaquetamiento**
- **Despliegue**

Maven

- Estructura de proyecto consistente
- Construye aplicaciones multi-modulo
- Orientado a plugins
- Genera información del Proyecto como un site HTML
- Gestión de dependencias de librerías
 - Repositorios de librerías
 - Gestión de versiones de las librerías
 - Dependencias transitivas
- Orientado a desarrollo ágil e integración continua
- Tests
- Conexión con herramientas de control de versiones

Competencia

- Ant (2000)
 - Granddaddy of Java Build Tools
 - Scripting in XML
 - Muy flexible
- Ant+Ivy (2004)
 - Ant pero con dependencias de Gestión
- Gradle (2008)
 - Intento de combinar la estructura de Maven structure con “Groovy Scripting”
 - Fácilmente extensible

Instalación

- Descarga maven desde sitio oficial
- Descomprimir archivo zip
- Crear las variables de entorno
 - M2
 - M2_HOME
 - MAVEN_OPTS
 - Actualizar PATH (apuntar a carpeta bin de maven)
 - **https://www.tutorialspoint.com/maven/maven_environment_setup.htm**

POM

- Acrónimo de **Project Object Model** describe un Proyecto:
- Nombre y versión
 - Tipo de Artefacto
 - Localización del código fuente
 - Dependencias
 - Plugins
 - Perfiles (Alternate build configurations)
- Uso de XML
 - Diferente a la forma en la que lo utiliza Ant

Proyecto

Para identificar un Proyecto Maven solo requiere:

- **groupId**: Arbitrary project grouping identifier
- **artifactId**: Arbitrary name of project
- **version**: Version of project
 - Formato {Major}.{Minor}.{Maintenance}
 - Agrega '-SNAPSHOT' para identificar en desarrollo















Proyecto saludos

Mi primer proyecto

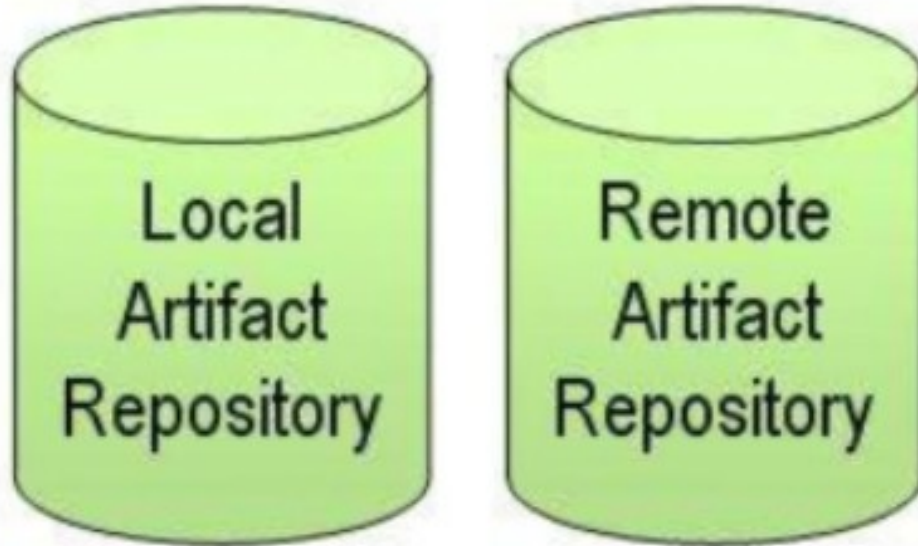
- `mvn -B archetype:generate -DgroupId=com.mrhc.saludos -DartifactId=hola -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4`
- eclipse

Estructura de directorios

- **src/main/java**: Código fuente
- **src/main/resources**: Archivos de configuración
- **src/main/webapp**: Estructura Web
- **src/test/java**: Código de pruebas
- **src/test/resources**: Archivos de configuración
- **target**: Paquetes generados y código compilado

-  src
 -  main
 -  config
 -  filters
 -  java
 -  resources
 -  webapp
 -  site
 -  test
 -  target
 -  classes
 -  surefire-reports
 -  test-classes
-  pom.xml

Gestión de dependencias



Ciclo de vida de un proyecto

Ciclo de vida básico:

- **generate-sources/generate-resources**
- **compile**
- **test**
- **package**
- **integration-test (pre and post)**
- **install**
- **deploy**
- **Además existe un “Clean lifecycle”**

Tipos de archivos

Tipos de ficheros

- **JAR: librería Java**
 - Clases Java (ficheros .class)
 - Ficheros de configuración
 - No contiene otras librerías
- **WAR: aplicación web**
 - Se despliega en un servidor web
 - Clases Java, ficheros de configuración
 - Librerías auxiliares (JAR)
 - Ficheros propios de la aplicación web (HTML, JPEG, etc.)
- **EAR: aplicación enterprise**
 - Se despliega en un servidor de aplicaciones
 - Clases Java, ficheros de configuración
 - Librerías auxiliares (JAR) y componentes EJB