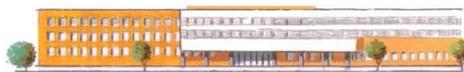


CHEAT SHEETS

Git & Maven



IES Ramón del Valle-Inclán

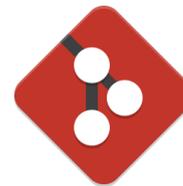


Co-funded by the
Erasmus+ Programme
of the European Union



The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Git cheat sheets



<code>git init</code>	Crea un nuevo repositorio de git.
<code>git clone <url_repository></code>	Clonar un repositorio.
<code>git add <filename></code>	Registrar los cambios de un determinado archivo.
<code>git add .</code>	Registrar todos los cambios.
<code>git checkout -- <filename></code>	Deshacer los cambios de un determinado archivo.
<code>git checkout -- .</code>	Deshacer todos los cambios.
<code>git commit -m "Commit message"</code>	Almacenar las modificaciones en el repositorio local.
<code>git push origin <branch></code>	Almacenar las modificaciones en el repositorio remoto.
<code>git remote add origin <server></code>	En caso de no haber clonado un repositorio, ejecutar para enlazar repositorio local con repositorio remoto.
<code>git checkout -b <branch></code>	Crear una nueva rama.
<code>git checkout <branch></code>	Mover a una determinada rama.
<code>git branch -d <branch></code>	Borrar una determinada rama.
<code>git pull</code>	Descargar y fusionar los cambios remotos.
<code>git merge <branch></code>	Fusionar una determinada rama en la rama activa.
<code>git diff <source_branch> <target_branch></code>	Revisar diferencias entre ramas.
<code>git tag <version> <id_commit></code>	Crear nueva etiqueta para un determinado commit.
<code>git log</code>	Consultar el log de la rama activa
<code>git fetch</code>	Descargar la última versión del servidor remoto.

Maven Cheat Sheets



<code>mvn clean</code>	Limpia el target del proyecto.
<code>mvn validate</code>	Valida si el proyecto está construido correctamente.
<code>mvn compile</code>	Compila el código fuente que se crea en <i>target/classes</i> .
<code>mvn test</code>	Ejecuta los tests.
<code>mvn package</code>	Empaqueta el código compilado en un formato distribuible (<i>jar, war, etc</i>).
<code>mvn verify</code>	Verifica que el código empaquetado es válido y cumple los criterios de calidad.
<code>mvn install</code>	Instala en tu repositorio local el empaquetado.
<code>mvn deploy</code>	Despliega el empaquetado en un repositorio remoto.

Plugins y comandos útiles

Dependency plugin	Manipular e inspeccionar las dependencias.
<code>mvn dependency:analyze</code>	Analiza las dependencias (no usadas, desactualizadas, etc).
<code>mvn dependency:tree</code>	Pinta un árbol de dependencias
Compiler plugin	Compilar el código.
Version plugin	Manejar las versiones del proyecto
Spring Boot plugin	Compilar, empaquetar y arrancar una aplicación SpringBoot
<code>mvn spring-boot:run</code>	Levanta una aplicación Spring Boot.