



## PRÁCTICA 1:

PAT 3ºB GITT

Diego Luis de Mazarredo Martín

## OBJETIVOS

21/01/25

### 1.1 Objetivo

La **Práctica 1** tiene como objetivo principal **preparar el entorno de trabajo del alumno** para el desarrollo de futuras actividades, asegurando el conocimiento básico sobre el uso de **Git** y **GitHub**. Debemos aprender a manejar comandos esenciales de Git, como **clone**, **status**, **add**, **commit**, **push** y **checkout**, y documentar su propósito y los resultados observados al ejecutarlos.

Además, se busca configurar el entorno de desarrollo para Java instalando herramientas clave como **Java 17**, **Maven** y un editor de código (**VSCode** o **IntelliJ**), proporcionando evidencias de su correcta instalación y funcionamiento.

### 1.2 GitHub

**git clone <url>**: Descargo un proyecto o repositorio de GitHub a mi ordenador. Básicamente, me traigo una copia para trabajar en ella en local.

**git status**: Veo qué cambios he hecho en mi proyecto. Me dice si hay archivos nuevos, modificados o listos para guardar.

**git add <archivo>**: Preparo un archivo para guardarlo. Es básicamente decirle que ese archivo lo quiero guardar en el próximo paso.

**git commit -m "mensaje"**: Guardo los cambios con un mensaje que explica qué hice. Es como tomar una foto de mi proyecto en ese momento.

**git push**: Envío mis cambios al repositorio en GitHub para que queden guardados online.

**git checkout -b <rama>**: Creo una nueva rama para trabajar en algo diferente sin afectar lo que ya tengo. Es como hacer una copia paralela para probar cosas.

### 1.3 Java

Para la instalación de **Java 17**, he descargado el JDK desde la página oficial de Adoptium y procedido a instalarlo en mi sistema operativo Windows. Posteriormente, he configurado la variable de entorno **JAVA\_HOME** para que apunte a la carpeta de instalación del JDK **C:\Program Files\Java\jdk-17** y he añadido **%JAVA\_HOME%\bin** al **PATH**. Finalmente, he verificado la instalación mediante el comando **java --version** en la terminal, confirmando que la versión instalada es la 17.

```
PS C:\Users\diegu> java --version
java 17.0.12 2024-07-16 LTS
Java(TM) SE Runtime Environment (build 17.0.12+8-LTS-286)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.12+8-LTS-286, mixed mode, sharing)
PS C:\Users\diegu> |
```

*Evidencia 1.3: java\_evidencia.png*

### 1.4 Maven

En cuanto a **Maven**, he descargado la versión más reciente desde la página oficial de Apache Maven y descomprimido los archivos en una carpeta local. Luego, he configurado la variable de entorno **MAVEN\_HOME** para señalar la ubicación del directorio de Maven y he añadido **%MAVEN\_HOME%\bin** al **PATH** para permitir el acceso global desde la terminal.

Para validar la instalación, he ejecutado el comando **mvn --version**, comprobando que Maven reconoce correctamente el JDK previamente instalado y que está listo para gestionar proyectos Java.

```
C:\Users\diegu>mvn --version
Apache Maven 3.9.9 (8e8579a9e76f7d015ee5ec7bfc9c97d260186937)
Maven home: C:\Program Files\Apache\Maven
Java version: 17.0.12, vendor: Oracle Corporation, runtime: C:\Program Files\java\jdk-17
Default locale: es_ES, platform encoding: Cp1252
OS name: "windows 11", version: "10.0", arch: "amd64", family: "windows"
```

*Evidencia 1.4: mvn\_evidencia.png*

### 1.5 Conclusión

Con esta práctica he aprendido los fundamentos de **Git**, **GitHub** y he configurado correctamente mi entorno de desarrollo. Esta base me permitirá realizar futuras prácticas de manera más eficiente y organizada. Especialmente adaptar los conocimientos de clase para realmente saber su aplicación práctica y la instalación en local para que sea más cómodo realizar todo en un futuro como se me indicó en clase.