

## **1-Que es Git?**

Git es un framework de versiones distribuido y descentralizado que facilita a los programadores mantener una cuenta completa de las evoluciones en un proyecto informático, cooperar con otros desarrolladores y supervisar distintas iteraciones de un proyecto.

## **2-Para que funciona el comando Git init?**

El comando "git init" sirve para establecer las bases de un repositorio Git en un directorio nuevo o existente. Esto genera un subdirectorio oculto "git" dentro del directorio de trabajo que almacena toda la información necesaria para el control de versiones de los archivos, tal como el registro de modificaciones, las ramas y la configuración.

En otras palabras, "git init" constituye el fundamento para comenzar a utilizar Git y a realizar un seguimiento de los cambios en los archivos de un proyecto. Convierte un simple directorio en un repositorio Git listo para la sincronización de versiones.

## **3-Que es una rama?**

Una rama en Git es una línea de desarrollo separada que permite a los desarrolladores trabajar en diferentes características o cambios de manera aislada sin afectar la rama principal. Las ramas también se pueden fusionar para integrar cambios en la rama principal.

## **4-Como saber es que rama estoy?**

Para determinar en qué rama de trabajo estás ubicado actualmente, ejecuta el comando "git branch". La rama activa se destacará con un asterisco.

De manera alternativa, puedes utilizar el comando "git status" para ver el estado actual del repositorio, incluida la rama vigente.

Ambas opciones le permitirán saber en qué línea de desarrollo o tributario se encuentra el proyecto en el presente momento.

## **5-Quien creo git?**

Linus Torvalds, el creador del kernel de Linux, desarrolló Git en 2005 como un sistema de gestión de versiones para facilitar el trabajo colaborativo en el desarrollo del kernel Linux.

## **6-Cuales son los comandos más esenciales de Git?**

Algunos de los comandos principales de Git son:

- "git clone": Copia un repositorio existente en un directorio local.
- "git add": Añade cambios en archivos al área de pre-integración (staging).
- "git commit": Confirma los cambios en el área de pre-integración y crea un nuevo commit.
- "git push": Envía los commits confirmados a un repositorio remoto.
- "git pull": Recupera los cambios del repositorio remoto y los combina en la rama local actual.
- "git branch": Crear, listar o eliminar ramas.
- "git merge": Fusiona los cambios de una rama en otra.
- "git checkout": Cambia entre diferentes ramas o versiones de archivos.

## **7-Que es git Flow?**

Git Flow es un enfoque para desarrollar software que se basa en una estructura de ramificación bien definida para mejorar la colaboración entre equipos y la administración de versiones. Este método sugiere un esquema de ramificación de dos niveles que consta de una rama principal conocida como "master" y una rama de desarrollo llamada "develop", así como varias ramas adicionales para características específicas, versiones y lanzamientos.

## **8-Que es trunk based development?**

Trunk Based Development (TBD) es un enfoque para desarrollar software que se concentra en tener una sola rama principal o tronco en lugar de crear múltiples ramas. Con TBD, se fomenta a los desarrolladores a integrar sus cambios en la rama principal varias veces al día para garantizar que todos los cambios se integren y se prueben continuamente. Esta práctica se considera más ágil y rápida que crear múltiples ramas y luego integrar los cambios más adelante.