



PROGRAMACION 3

# TAREA 3



BRIAN PERALTA, 2021-2214  
PROFESOR: KELLY TEJADA

---

## *¿Qué es Git?*

---

Es un sistema de control de versiones compartidas de código abierto que funciona para rastrear cambios en el código fuente durante el desarrollo de software. Brinda un historial completo de cambios, hace más sencillo la colaboración entre desarrolladores y permite la gestión eficiente de ramas de desarrollo. Varias características clave de Git incluyen ramificación y fusión flexibles, velocidad y rendimientos y la capacidad de trabajar de forma descentralizada. Este sistema fue creado e implementado por **Linus Torvalds** en 2005 y ha logrado ser una herramienta fundamental para el desarrollo de software moderno.

---

## *¿Para que funciona el comando Git init?*

---

La función del comando “**Git init**” se utiliza para iniciar un repositorio Git en un directorio existente. Esto implementa un nuevo repositorio Git, con un subdirectorio `.git` que contiene todos los archivos necesarios para el control de versiones. Este comando es el primer paso para comenzar a utilizar Git en un proyecto existente o nuevo.

---

## *¿Qué es una rama?*

---

Es una línea independiente de desarrollo que permite a los desarrolladores trabajar en características, correcciones de errores o experimentos sin afectar directamente el código en la rama principal.

Son útiles para organizar el trabajo, colaborar en paralelo y mantener un flujo de trabajo ordenado.

---

### *¿Como saber es que rama estoy?*

---

Para entender en que rama nos ubicamos actualmente, podemos utilizar el comando “git Branch”. Este comando funciona para mostrar todas las ramas presentes en el repositorio y resalta la rama en la que se encuentra actualmente.

---

### *¿Quién creo git?*

---

El creado es git es Linus Torvalds, fue la misma persona que creo el kernel del sistema operativo Linux. Su desarrollo comenzó en 2005 para manejar el código fuente de Linux y facilitar la colaboración entre desarrolladores.

---

### *¿Cuáles son los comandos más esenciales de Git?*

---

**Git add:** Para agregar cambios al área de preparación.

**Git commit:** Para confirmar los cambios.

**Git push:** Para enviar los cambios a un repositorio remoto.

**Git pull:** para obtener cambios de un reposito remoto.

**Git Branch:** Para administrar ramas.

**Git merge:** Para fusionar ramas.

---

## *¿Qué es git Flow?*

---

Es un modelo de ramificación y fusión para git que define un conjunto estricto de reglas para la gestión de ramas. Brinda una base clara para el desarrollo de funciones, versiones y correcciones de errores, y se basa en ramas específicas para cada tipo de desarrollo.

---

## *¿Que es trunk-based development?*

---

Es una vista de Desarrollo de software que se centra en mantener una única rama principal (trunk) como la fuente de verdad para el código. Se centraliza en la integración continua y la entrega continua, lo que significa que los cambios se integran rápidamente en la rama principal y se despliegan en producción con frecuencia. Este acercamiento promueve la simplicidad, la velocidad y la estabilidad en el desarrollo de software.

**Link del proyecto de GitHub:** <https://github.com/Bragrullon/Prog-3>