



CLASE 3

Ing. SILVESTRE ALEJANDRO
INFORMÁTICA III
IUA - 2023



GIT



¿Qué es GIT?

Git es un **sistema de control de versiones** distribuido que se utiliza ampliamente en el desarrollo de software y la gestión de proyectos. Proporciona una manera eficiente y colaborativa de rastrear cambios en el código fuente y otros archivos a lo largo del tiempo.

En esencia, Git es como una especie de "historial" de cambios para tu proyecto, donde puedes ver cómo ha evolucionado el código a lo largo del tiempo, quién ha realizado qué cambios y cuándo.



Beneficios de Git

- Seguimiento de cambios preciso.
- Trabajo en equipo sin conflicto.
- Ramificación y fusión sencillas.
- Historial de versiones completo.



Conceptos Fundamentales

Repositorio: Un repositorio Git es un lugar donde se almacenan todos los archivos y su historial de cambios. Puede ser local (en tu máquina) o remoto (en un servidor, por ejemplo, GitHub).

Commit: Un commit es una instantánea de los cambios que has realizado en tu repositorio. Cada commit tiene un mensaje asociado que describe lo que se ha modificado.

Rama: Las ramas son líneas independientes de desarrollo en un repositorio. Puedes crear ramas para trabajar en nuevas características o solucionar problemas sin afectar la rama principal.



Comandos Básicos de Git

- `git init`: Iniciar un repositorio.
- `git add`: Añadir cambios al área de preparación.
- `git commit`: Registrar cambios en el repositorio.
- `git status`: Ver el estado de los archivos.
- `git log`: Visualizar el historial de commits.



Comandos relacionados con ramificación

- `git branch`: Listar ramas.
- `git checkout`: Cambiar de rama.
- `git merge`: Fusionar ramas.



Flujo de Trabajo Típico

1. Clonar un repositorio.
2. Crear una rama para una nueva característica.
3. Realizar cambios y commits en la rama.
4. Fusionar la rama en la rama principal (master/main).
5. Subir los cambios al servidor remoto.



GUI Gráficas

- GitKraken Client
- SourceTree
- TortoiseGit
- GitHub Desktop



RECURSIVIDAD



¿Qué es la recursividad?

Una función que se llama a sí misma para resolver un problema.





Condiciones de la recursividad

1. Se tiene que llamar así mismo (sino no es recursiva).
2. Tiene que tener una sentencia de corte.
3. Tiene que recibir al menos un parámetro



MANOS A LA OBRA



Actividad

1. Crear repositorio de GIT y compartir a alejandro.j.silvestre@gmail.com.
 - a. Crear una estructura de carpetas con los nombres de los prácticos.
2. Realizar el práctico de [Recursividad](#) y subirlo al repositorio de GIT