

# Git Basico



2025-01



10 min.

# Que es Git?

Git es un sistema de control de versiones.

# GitHub, GitLab, BitBucket

Paginas web que proveen hosting de repositorios de Git.

# ★ Concepto 1: Repositorio

Un repositorio (de software) o "repo", es un almacenamiento para un software.

Usualmente, un repo equivale a un proyecto.

Un repo tiene un historial de cambios, donde cada entrada es un commit.

## Concepto 2: Commit

Un commit es una entrada en el historial de cambios de un repositorio.

"Un commit es un cambio."

## ★ Concepto 3: .gitignore

Archivo con reglas para excluir archivos o carpetas de git.

Si un archivo o carpeta ya tiene historial en git, el .gitignore es ignorado.

## ★ Concepto 3: .gitignore

Ejemplo:

```
.env  
.env.local  
.env.development
```

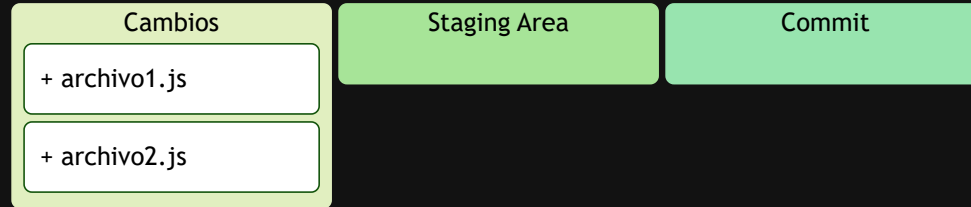
## ★ Concepto 4: Stage y Unstage

Los cambios realizados en un software o proyecto deben ser agregados a lo que se llama "staging área".



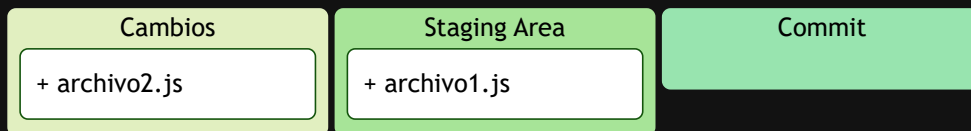
# ★ Concepto 4: Stage y Unstage

Creé dos archivos: archivo1.js y archivo2.js



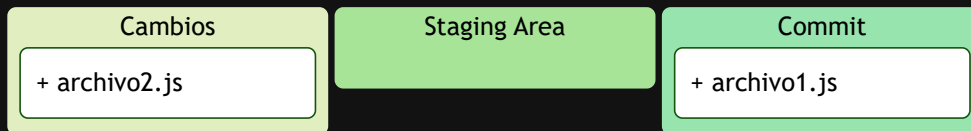
# ★ Concepto 4: Stage y Unstage

Hice **stage** de la creacion de archivo1.js



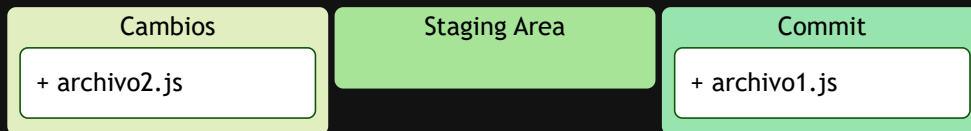
## ★ Concepto 4: Stage y Unstage

Hago commit de los cambios en la staging area.



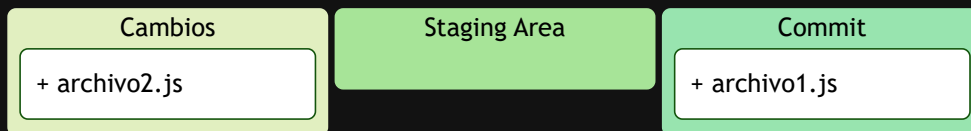
## ★ Concepto 4: Stage y Unstage

Que pasa si hago un commit ahora mismo?



## ★ Concepto 4: Stage y Unstage

Nada, porque el staging area está vacío.



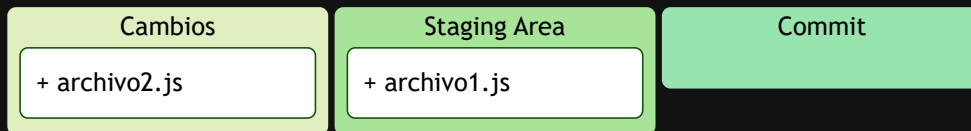
## ★ Concepto 4: Stage y Unstage

Los cambios realizados en un software o proyecto deben ser agregados a lo que se llama "staging area".

Al hacer un commit, **solo lo que esté en el staging area** será parte del commit.

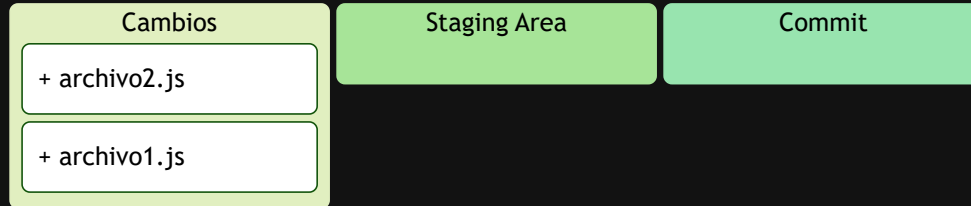
## ★ Concepto 4: Stage y Unstage

Unstage es cuando un cambio en el staging area es removido del staging area.



## ★ Concepto 4: Stage y Unstage

Unstage es cuando un cambio en el staging area es removido del staging area.





## ★ Concepto 5: Remote

Remote es el repositorio (y branches) que están en la nube.

Como pueden estar las branches en la nube?

## ★ Concepto 6: Push + Pull

**Push:** Enviar cambios de una branch local a la versión remota de la branch. *Si la branch remota no existe, **push** la crearía en el repo remote.*

**Pull:** Bajar los cambios de una branch remota a la versión local de la branch.

# Ahora mismo

1. Creen una cuenta de Github (si no la tienen).
2. Creen un repositorio de Github nuevo, con un nombre similar a "talleres-dllo-backend".
  - Hay un botón verde en la pagina principal de Github que dice **New**, o **Nuevo**.
  - Lo unico requerido es que le pongan nombre.

# Suban un archivo

1. Creen un archivo en su computador llamado "taller1.js".
2. Suban el archivo al repositorio.
  - En la pagina del repositorio debe haber un archivo que diga **Add file**, o **Agregar archivo**
  - Alternativamente, pueden arrastar el archivo sobre el repositorio.
  - Finalmente, usen el botón verde de **Commit**.



**Felicitades!**