

Curso Práctico Git



Qué es?

Es un sistema de control distribuido

Se puede revertir alguna versión.

Conceptos claves de Git

Ayuda en la toma de desiciones sobre cuando hacer cambios, se guarda el proyecto y se puede revertir.

En git hay 3 estados:

1. *Working directory*: Donde se trabaja con todos los archivos.
2. *Staging area*: Donde se agregarán todos los archivos para el guardado. Listado de múltiples cambios, se pasa todos los archivos a staging area.
3. *Repository*: Cuando ya estemos seguros para guardar el cambio se lo pasa a un repositorio.

Comandos básicos de Git

- **git init**:
 - Se crea un proyecto nuevo o en uno que ya estemos trabajando, se empieza a usar git.
 - Inicializa un nuevo repositorio Git.
- **git add**:
 - Se pasan los archivos de working directory a staging area.
 - Agrega todos los archivos modificados al staging.
- **git status**:
 - Para ver en que estado estan los archivos (en el working directory o en el staging area)
 - Muestra los cambios en archivos (modificados, staged, untracked).
- **git commit -m "mensaje"**:
 - Pasarlo del staging area al repositorio.
 - Hacer commit con un mensaje.
- **git log**: Ver historial de commits.
- **git branch**: Ver ramas existentes.
- **git checkout <rama>**: Cambiar a otra rama.
- **git push**:
 - Para subirlo a un repositorio remoto (subirlo a un servidor para que mas personas puedan trabajarlo)
 - Subir commits a repositorio remoto.
- **git pull**:
 - Traer cambios de otros desarrolladores.
 - Traer cambios del repositorio remoto.
- **git clone**: Traer un código a tu ordenador para que puedas trabajarlo.

En un proyecto de vscode:

Comandos de navegación de carpetas:

- **pwd**: Muestra la ruta de la carpeta actual (print working directory).
- **ls**: Lista los archivos y carpetas dentro del directorio actual.
- **ls -l**: Lista archivos con detalles (permisos, tamaño, fecha).
- **ls -a**: Lista todos los archivos, incluidos los ocultos.
- **cd < ruta>**: Cambiar al directorio < ruta>. Ej: cd proyectodegit.
- **cd ..**: Subir un nivel en la carpeta.
- **cd ~**: Ir a tu carpeta de usuario.
- **mkdir < nombre>**: Crear una nueva carpeta.
- **rmdir < nombre>**: Borrar una carpeta vacía.
- **rm -r < nombre>**: Borrar una carpeta y todo su contenido.

Comandos de archivos:

- **touch < archivo>**: Crear un archivo vacío.
 - **cat < archivo>**: Mostrar el contenido de un archivo en la terminal.
 - **nano < archivo>**: Abrir un archivo en Nano, un editor de texto simple en la terminal.
 - **code < archivo>**: Abrir un archivo en VS Code desde la terminal.
 - **rm < archivo>**: Eliminar un archivo.
 - **cp < origen> < destino>**: Copiar archivo o carpeta.
 - **mv < origen> < destino>**: Mover o renombrar archivo o carpeta.
-

Comandos de ayuda y control:

- **man < comando>**: Muestra el manual de un comando. Ej: man ls.
 - **< comando> --help**: Muestra ayuda rápida sobre un comando. Ej: git commit --help.
 - **clear**: Limpia la pantalla de la terminal.
 - **exit**: Cierra la terminal.
-

```
• heidy > [ ] proyectodegit > ?master # [ ] ?2 > [ ] pwd
/c/Users/heidy/Desktop/proyectodegit
• heidy > [ ] proyectodegit > ?master # [ ] ?2 > [ ] ls
app.js index.html
• heidy > [ ] proyectodegit > ?master # [ ] ?2 > [ ] git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    app.js
    index.html

nothing added to commit but untracked files present (use "git add" to track)
• heidy > [ ] proyectodegit > ?master # [ ] ?2 > [ ] git add .
• heidy > [ ] proyectodegit > ?master # [ ] +2 > [ ] git status
```

```
• heidy > [ ] proyectodegit > ?master # [ ] +2 > [ ] git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   app.js
    new file:   index.html
```

Cuando se ha hecho un cambio:

```
heidy > proyectodegit %master # 0 +2 > git status

● heidy > proyectodegit %master # 0 +2 > git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   app.js
    new file:   index.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    style.css
```

Se guarda el cambio:

```
heidy > proyectodegit %master # 0 ?1 0 +2 > git add style.css
heidy > proyectodegit %master # 0 +3 > git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   app.js
    new file:   index.html
    new file:   style.css
```

git commit -m "mensaje"

```
● heidy > proyectodegit %master # 0 +3 > git commit
[master (root-commit) f459683] Primer commit: agregando archivos iniciales
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 app.js
create mode 100644 index.html
create mode 100644 style.css
```

-En la imagen dice que 3 archivos han sido cambiado y crea un id para cada archivo (primera versión).

Ahora con git log

```
● heidy > proyectodegit %master # > git log
commit f4596836489a329984e4e8566ac47b1759593110 (HEAD -> master)
Author: KarliBoo <karli_cruz29@hotmail.com>
Date:   Sun Oct 5 22:15:26 2025 -0600

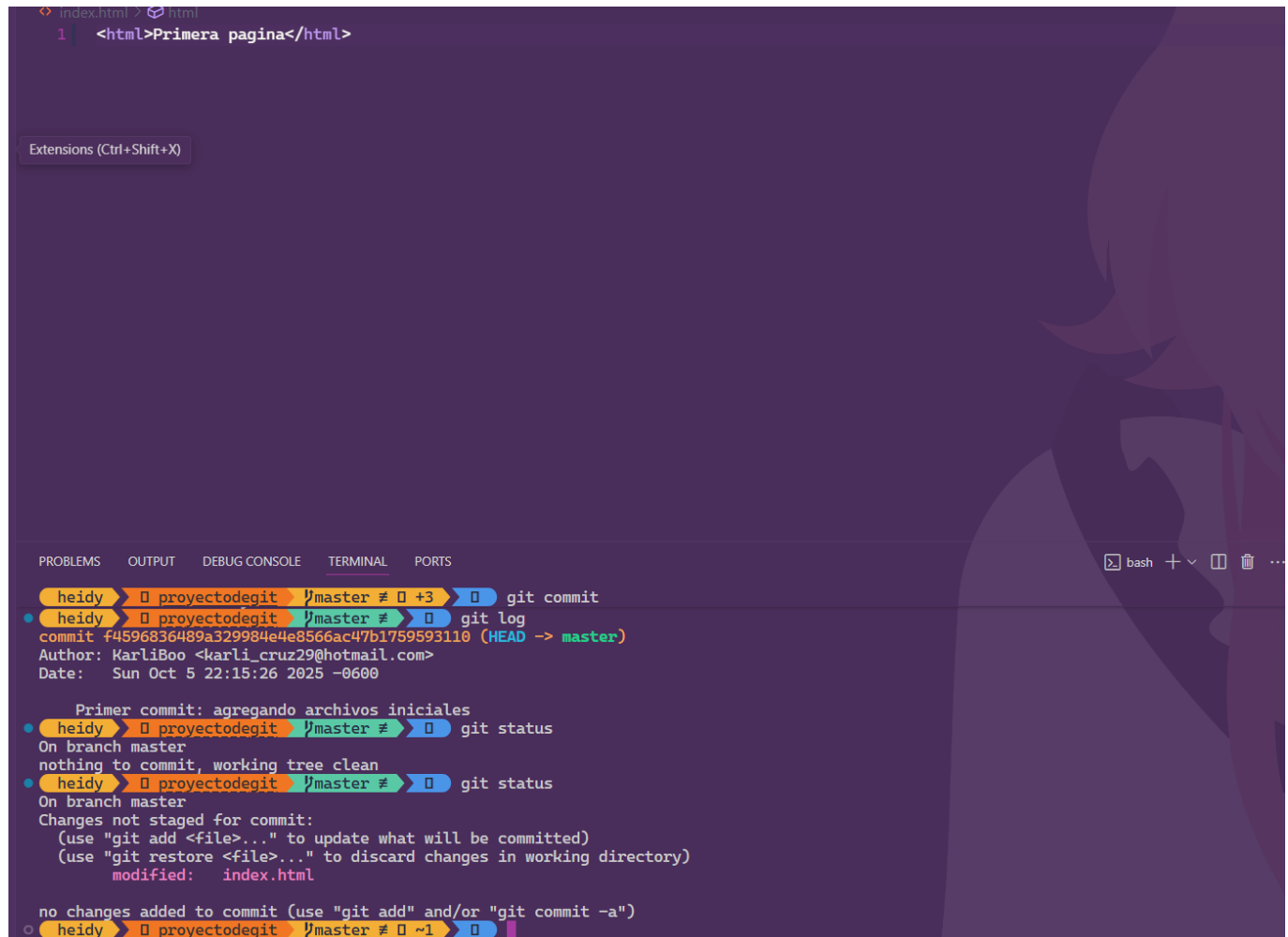
    Primer commit: agregando archivos iniciales
```

-Da una manera de diferenciar los commits que hemos creado, cada versión. Dice autor, usuario, e-mail y en que momento se hizo el cambio.

git status dice que todos los cambios fueron hechos y ya no hay nada por hacer:

```
heidy > cd proyectodegit > git status
On branch master
nothing to commit, working tree clean
```

Ahora se modificó el index.html, ctrl s



The screenshot shows the VS Code editor with a file named `index.html` open. The file content is `<html>Primera pagina</html>`. Below the editor, the terminal window is active, showing a series of git commands and their outputs. The commands include `git commit`, `git log`, and `git status`. The output of `git log` shows a commit by KarliBoo. The output of `git status` shows that the working tree is clean. The terminal also displays instructions for staging changes with `git add` and discarding changes with `git restore`.

```
heidy > cd proyectodegit > git commit
heidy > cd proyectodegit > git log
commit f4596836489a329984e4e8566ac47b1759593110 (HEAD -> master)
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 22:15:26 2025 -0600

Primer commit: agregando archivos iniciales
heidy > cd proyectodegit > git status
On branch master
nothing to commit, working tree clean
heidy > cd proyectodegit > git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
heidy > cd proyectodegit >
```

-git checkout -- nombre del archivo, descartó los cambios y vuelve a escribir lo que estaba anteriormente. No se puede regresar a esa versión porque no se guardó.

```
<> index.html X JS app.js # style.css
<> index.html
1 Generate code (Ctrl+I), or select a language (Ctrl+K M). Start typing to dismiss or don't show this again

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
heidy > [?] projectodegit [?] master # [?] git log
heidy > [?] projectodegit [?] master # [?] git status
On branch master
nothing to commit, working tree clean
heidy > [?] projectodegit [?] master # [?] git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
heidy > [?] projectodegit [?] master # [?] ~1 [?] git restore <index.html>
bash: syntax error near unexpected token `newline'
heidy > [?] projectodegit [?] master # [?] ~1 [?] git checkout -- index.html
heidy > [?] projectodegit [?] master # [?] git status
On branch master
nothing to commit, working tree clean
heidy > [?] projectodegit [?] master # [?] [?]
? master 0.0.0 Live Share
```

Se vuelve a cambiar el archivo:

```
<> index.html > html > head > title
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Primera pagina</title>
7  </head>
8  <body>
9
10 </body>
11 </html>
12
13
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
heidy > proyectodegit /master # git status
On branch master
nothing to commit, working tree clean
heidy > proyectodegit /master # git status
On branch master ...
modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
heidy > proyectodegit /master # ~1
```

git diff nombre del archivo ;para revisar que se cambi3

```
heidy > proyectodegit /master # ~1 git diff index.html
diff --git a/index.html b/index.html
index e69de29..973c57c 100644
--- a/index.html
+++ b/index.html
@@ -0,0 +1,12 @@
+<!DOCTYPE html>
+<html lang="en">
+<head>
+  <meta charset="UTF-8">
+  <meta name="viewport" content="width=device-width, initial-scale=1.0">
+  <title>Primera pagina</title>
+</head>
+<body>
+
+</body>
+</html>
+
```

Ahora está en el entorno de trabajo:

```

● heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ] git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
● heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ] git add index.html
● heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ] git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   index.html

○ heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ]

```

Para guardar:

```

● heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ] git commit -m "ahora el index.html tiene código html"
[master 077fac6] ahora el index.html tiene código html
1 file changed, 12 insertions(+)

```

con git log se ven los commits:

```

● heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ] git status
        modified:   index.html

● heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ] git commit -m "ahora el index.html tiene código html"
[master 077fac6] ahora el index.html tiene código html
1 file changed, 12 insertions(+)

● heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ] git log
commit 077fac61adc9d1f41e20a1145990025215be927b (HEAD -> master)
Author: KarliBoo <karli_cruz29@hotmail.com>
Date:   Sun Oct 5 23:08:14 2025 -0600

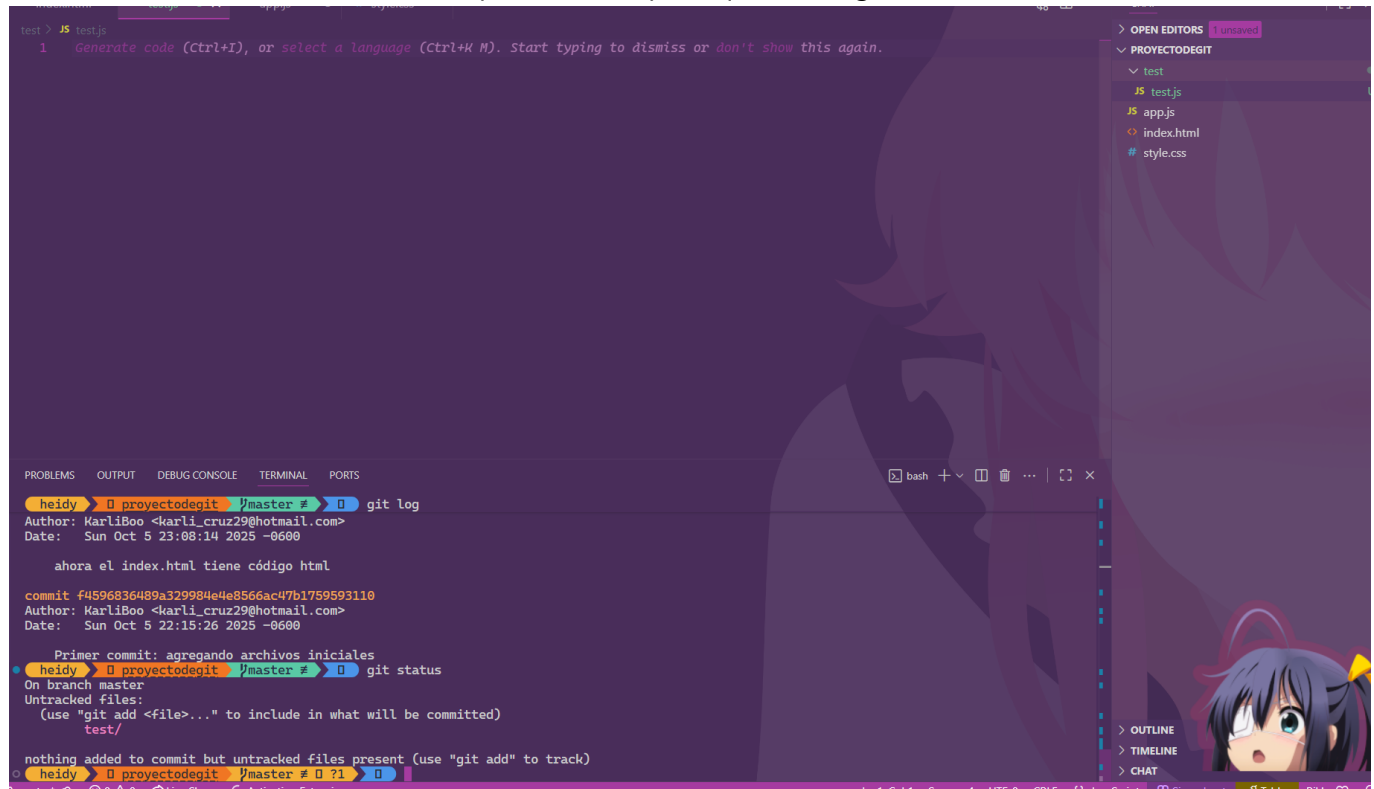
    ahora el index.html tiene código html

commit f4596836489a329984e4e8566ac47b1759593110
Author: KarliBoo <karli_cruz29@hotmail.com>
Date:   Sun Oct 5 22:15:26 2025 -0600

    Primer commit: agregando archivos iniciales
○ heidy > [ ] proyectodegit > { } master ≠ [ ] ~1 > [ ]

```


test termina con / nos damos cuenta que es una carpeta, queremos ignorarla:



The screenshot shows a VS Code editor interface. The main editor area displays a file named `test.js` with the following content:

```
test > JS test.js
1 Generate code (Ctrl+I), or select a language (Ctrl+K M). Start typing to dismiss or don't show this again.
```

The right sidebar shows the Explorer view with the following structure:

- PROYECTODEGIT
 - test
 - test.js
 - app.js
 - index.html
 - style.css

The bottom panel shows the TERMINAL view with the following output:

```
heidy > proyectodegit /master # git log
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 23:08:14 2025 -0600

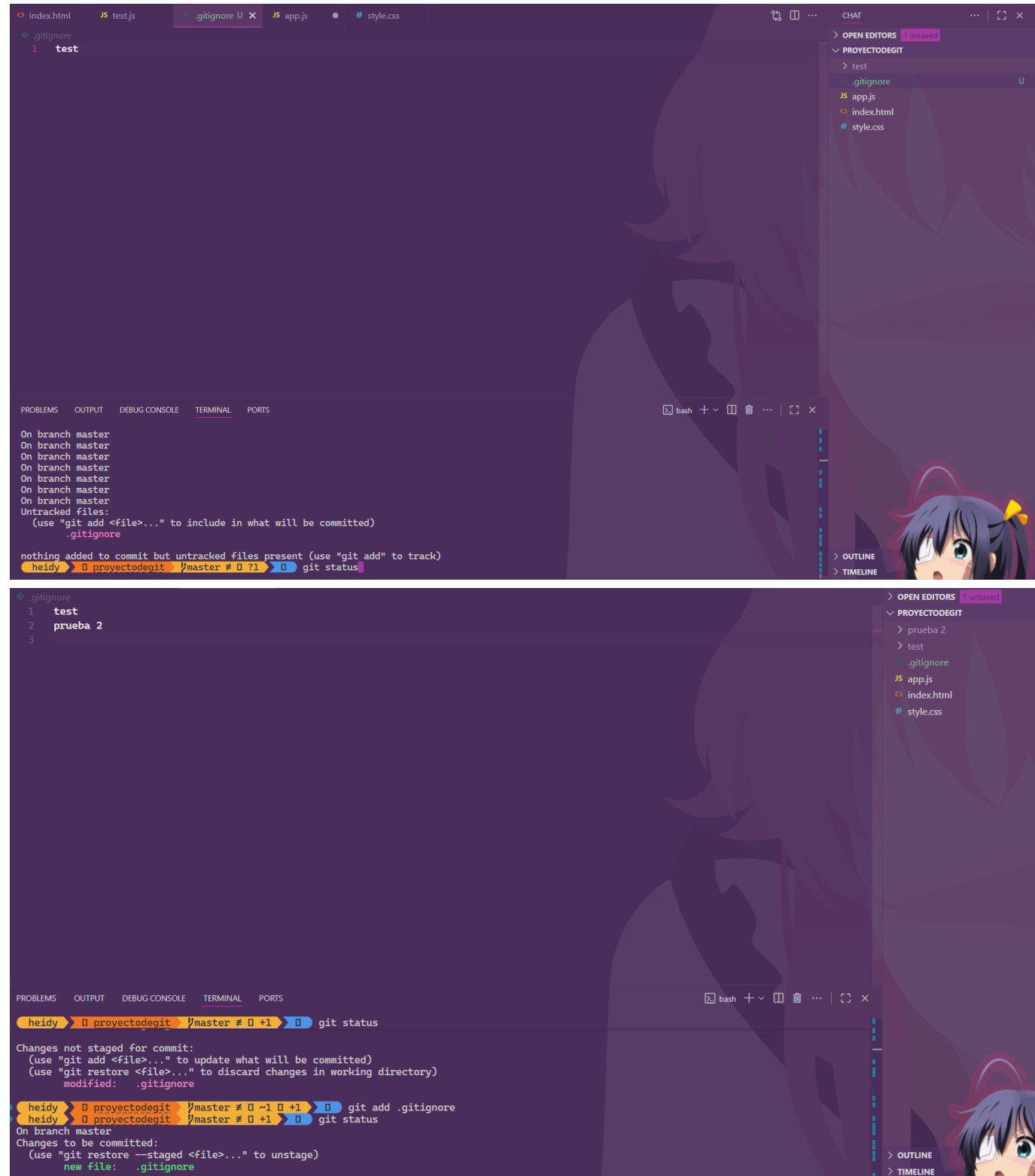
ahora el index.html tiene código html

commit f4596836489a329984e4e8566ac47b1759593110
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 22:15:26 2025 -0600

Primer commit: agregando archivos iniciales
heidy > proyectodegit /master # git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    test/

nothing added to commit but untracked files present (use "git add" to track)
heidy > proyectodegit /master # ?1
```

-Se crea una carpeta de `.gitignore` y ponemos dentro el nombre de que queremos ignorar, luego ponemos el `git status` y nos sale el `gitignore`:



```

heidy > cd proyectodegit > master # 0 +1 > git status
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   .gitignore

heidy > cd proyectodegit > master # 0 ~1 0 +1 > git add .gitignore
heidy > cd proyectodegit > master # 0 +1 > git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   .gitignore

heidy > cd proyectodegit > master # 0 +1 > git commit -m "Se ha agregado un gitignore"
[master 0e22015] Se ha agregado un gitignore
1 file changed, 2 insertions(+)
create mode 100644 .gitignore

```

Se crea una nueva version alternativa llamada login:

```

heidy > cd proyectodegit > master # > git branch
* master
heidy > cd proyectodegit > master # > git branch login
heidy > cd proyectodegit > master # >

```

Para moverse dentro de login:

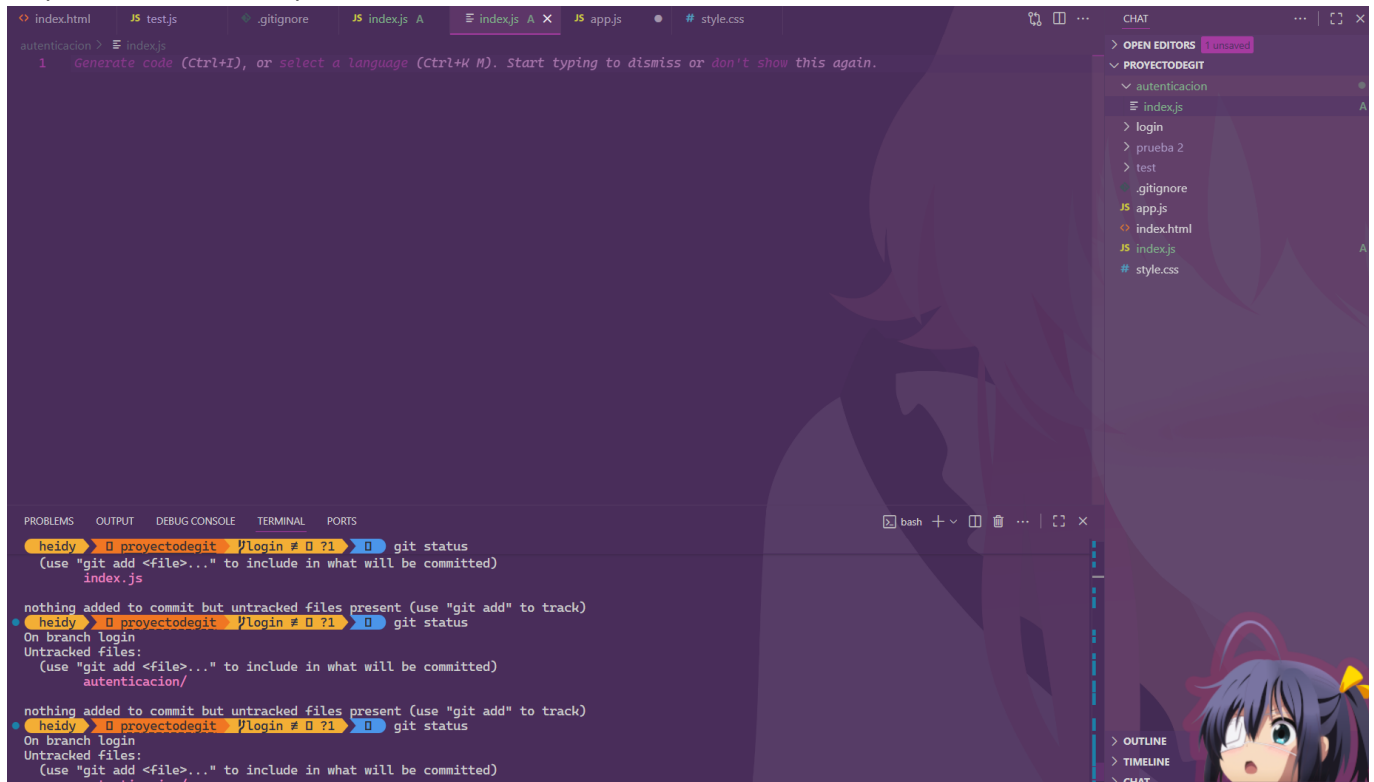
```

heidy > cd proyectodegit > master # > git branch login
heidy > cd proyectodegit > master # > git branch
login
* master
heidy > cd proyectodegit > master # > git checkout login
Switched to branch 'login'
heidy > cd proyectodegit > login # >

Switched to branch 'login'
heidy > cd proyectodegit > login # > git branch
* login
  master
heidy > cd proyectodegit > login # >

```

Se puede crear mas carpetas (se tomarán en cuenta si tienen un archivo adentro)



```

autenticacion? index.js
1 Generate code (Ctrl+I), or select a language (Ctrl+K M). Start typing to dismiss or don't show this again.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
heidy > cd proyectodegit /login # 0 ?1 git status
(use "git add <file>..." to include in what will be committed)
index.js

nothing added to commit but untracked files present (use "git add" to track)
heidy > cd proyectodegit /login # 0 ?1 git status
On branch login
Untracked files:
(use "git add <file>..." to include in what will be committed)
autenticacion/

nothing added to commit but untracked files present (use "git add" to track)
heidy > cd proyectodegit /login # 0 ?1 git status
On branch login
Untracked files:
(use "git add <file>..." to include in what will be committed)
autenticacion/
  
```

```

heidy > cd proyectodegit /login # 0 ?1 git status
On branch login
Untracked files:
(use "git add <file>..." to include in what will be committed)
autenticacion/
index.js

nothing added to commit but untracked files present (use "git add" to track)
heidy > cd proyectodegit /login # 0 ?2 git add .
heidy > cd proyectodegit /login # 0 +2 git status
On branch login
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
new file:   autenticacion/index.js
new file:   index.js

heidy > cd proyectodegit /login # 0 +2
  
```

```

heidy > cd proyectodegit /login # 0 ?1 git status

nothing added to commit but untracked files present (use "git add" to track)
heidy > cd proyectodegit /login # 0 ?2 git add .
heidy > cd proyectodegit /login # 0 +2 git status
On branch login
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
new file:   autenticacion/index.js
new file:   index.js

heidy > cd proyectodegit /login # 0 +2 git commit -m "versión alternativa con un login"
[login 4c6199c] versión alternativa con un login
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 autenticacion/index.js
create mode 100644 index.js

heidy > cd proyectodegit /login #
  
```

con git log se observa los cambios en login

```

heidy > projectodegit login # git log
commit 4c6199c8f63b178cbab3cc08cdad32882281ac06 (HEAD -> login)
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 23:32:12 2025 -0600

    versión alternativa con un login

commit 0e22015766eefe81f0c30ed79ba6c13edac84e47 (master)
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 23:21:17 2025 -0600

    Se ha agregado un gitignore

commit 077fac61adc9d1f41e20a1145990025215be927b
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 23:08:14 2025 -0600

```

se regresa a la otra rama

```

heidy > projectodegit login # git branch
* login
master

heidy > projectodegit login # git checkout master
Switched to branch 'master'

heidy > projectodegit master # git branch
login
* master

heidy > projectodegit master # git log
commit 0e22015766eefe81f0c30ed79ba6c13edac84e47 (HEAD -> master)
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 23:21:17 2025 -0600

    Se ha agregado un gitignore

commit 077fac61adc9d1f41e20a1145990025215be927b
Author: KarliBoo <karli_cruz29@hotmail.com>

```

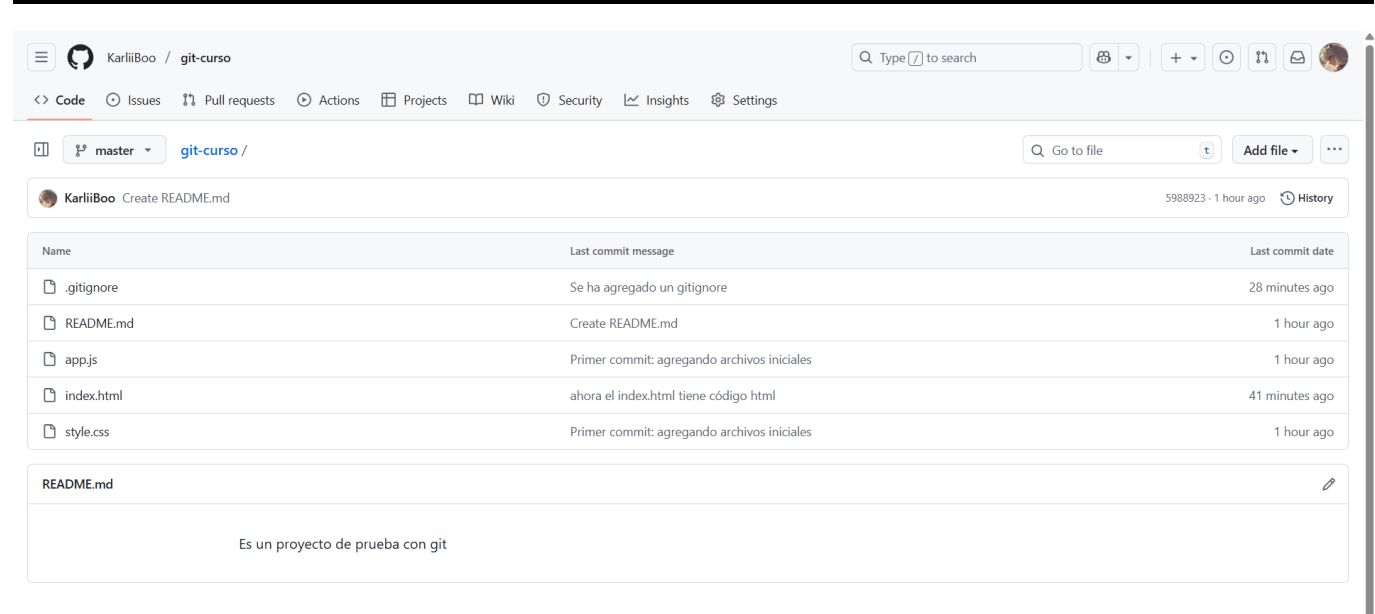
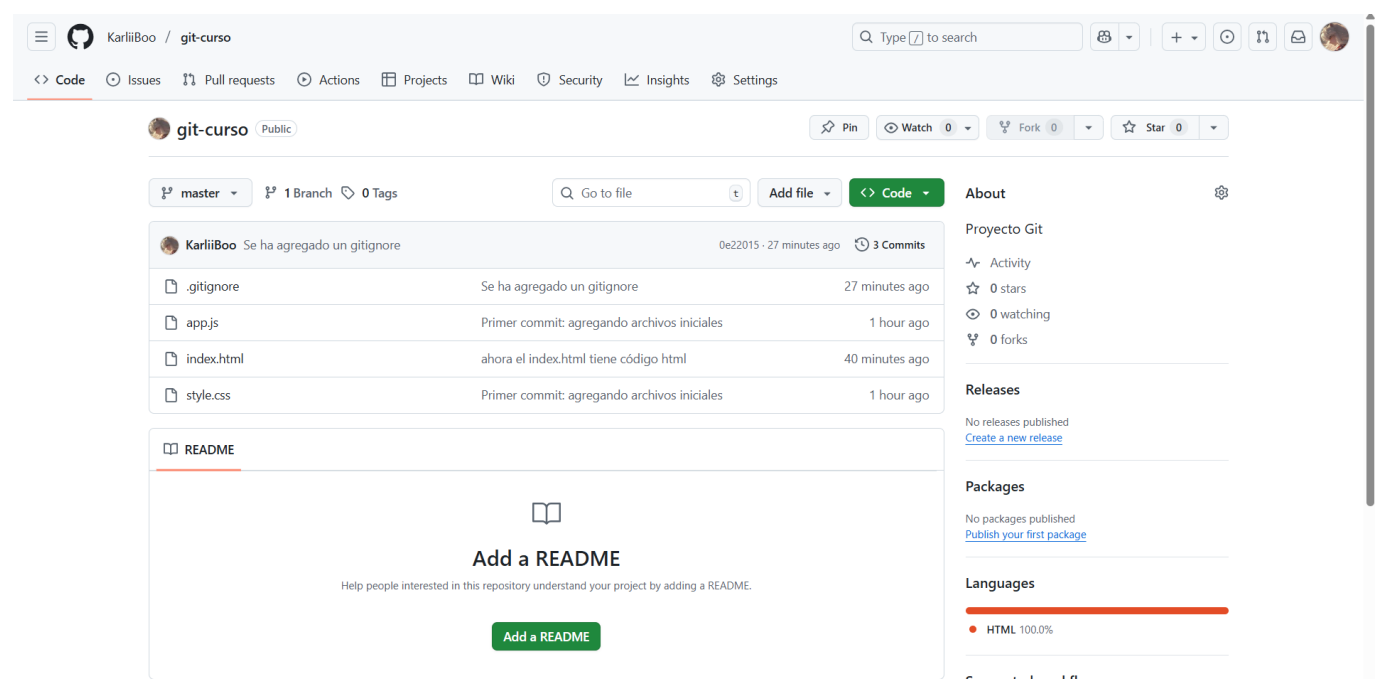
Para subir al repositorio de github

```

heidy > projectodegit master # git remote add origin https://github.com/KarliiBoo/git-curso.git
heidy > projectodegit master # git status
On branch master
nothing to commit, working tree clean

heidy > projectodegit master # git push -u origin master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 20 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (9/9), 890 bytes | 890.00 KiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/KarliiBoo/git-curso.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.

```



Si se borra debemos descargar de github y usar git clone

```
and the repository exists.
• heidy git clone https://github.com/KarliiBoo/git-curso.git
Cloning into 'git-curso'...
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 12 (delta 3), reused 8 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (12/12), done.
Resolving deltas: 100% (3/3), done.
• heidy
```

The screenshot shows a GitHub repository page for 'KarliiBoo' with the title 'Crear README.md'. The repository contains several files: `.gitignore`, `LÉAME.md`, `aplicación.js`, `índice.html`, and `estilo.css`. The README file is selected, showing the text 'Es un proyecto de prueba con git'. A dropdown menu is open over the repository, showing options for cloning the repository. The 'Local' tab is selected, and the 'Clon' option is chosen. The dropdown shows the URL `https://github.com/KarliiBoo/git-curso.git` and options to clone using HTTPS, SSH, or the GitHub CLI. Other options include 'Abrir con GitHub Desktop' and 'Descargar ZIP'.

Se entra en el proyecto y git log

```
heidy ~$ git clone https://github.com/KarliiBoo/git-curso.git
Cloning into 'git-curso'...
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 12 (delta 3), reused 8 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (12/12), done.
Resolving deltas: 100% (3/3), done.
heidy ~$ cd git
bash: cd: git: No such file or directory
heidy ~$ cd git-curso
heidy ~$ git log
commit 5988923513c0c0c5b03e75576e84f65d65affc39 (HEAD -> master, origin/master, origin/HEAD)
Author: KarliiBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 22:48:56 2025 -0600

    Create README.md

commit 0e22015766eefe81f0c30ed79ba6c13edac84e47
Author: KarliBoo <karli_cruz29@hotmail.com>
Date: Sun Oct 5 23:21:17 2025 -0600
```