

Crear un repositorio local

Comando GIT INIT

Para iniciar el control de versiones con Git, ejecute el comando `git init` en el directorio de trabajo.

En esta serie, el siguiente comando crea un directorio de trabajo llamado `git`.

```
$ mkdir -p ~/workspace/secundario/text/git
```

La opción `-p` del comando `mkdir` tiene el significado de `parents` (padres).

La opción `-p` se puede utilizar para crear varios directorios con relaciones padre-hijo.

A continuación, vayamos al directorio `git` que creamos:

```
$ cd ~/workspace/secundario/text/git
```

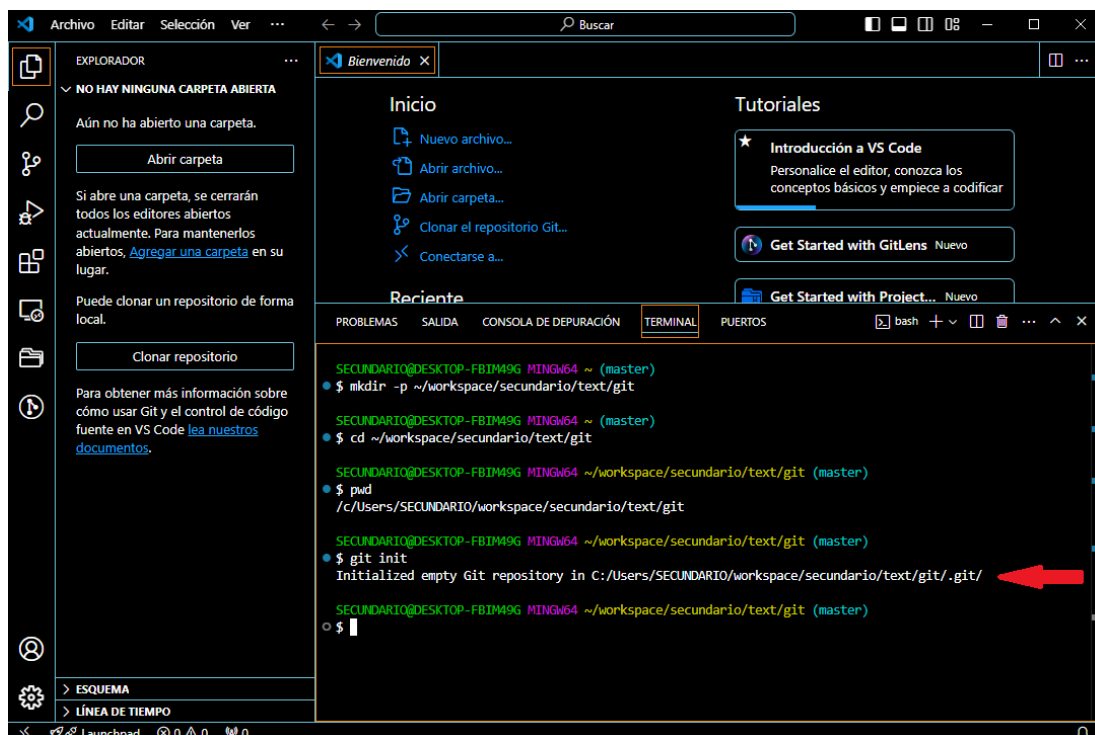
Comprueba que te has movido al directorio `git` ejecutando el comando `pwd`.

```
$ pwd
```

No hay problema si se muestra `/workspace/secundario/text/git` creado anteriormente.

A continuación, ejecuta el comando `git init`.

```
$ git init
```



The screenshot shows the Visual Studio Code interface with a terminal window open. The terminal output shows the following commands and their results:

```
SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~ (master)
$ mkdir -p ~/workspace/secundario/text/git

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~ (master)
$ cd ~/workspace/secundario/text/git

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
$ pwd
/c/Users/SECUNDARIO/workspace/secundario/text/git

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
$ git init
Initialized empty Git repository in C:/Users/SECUNDARIO/workspace/secundario/text/git/.git/

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
$
```

A red arrow points to the output of the `git init` command, which is "Initialized empty Git repository in C:/Users/SECUNDARIO/workspace/secundario/text/git/.git/".

Cuando ejecutes el comando `git init`, verás el mensaje

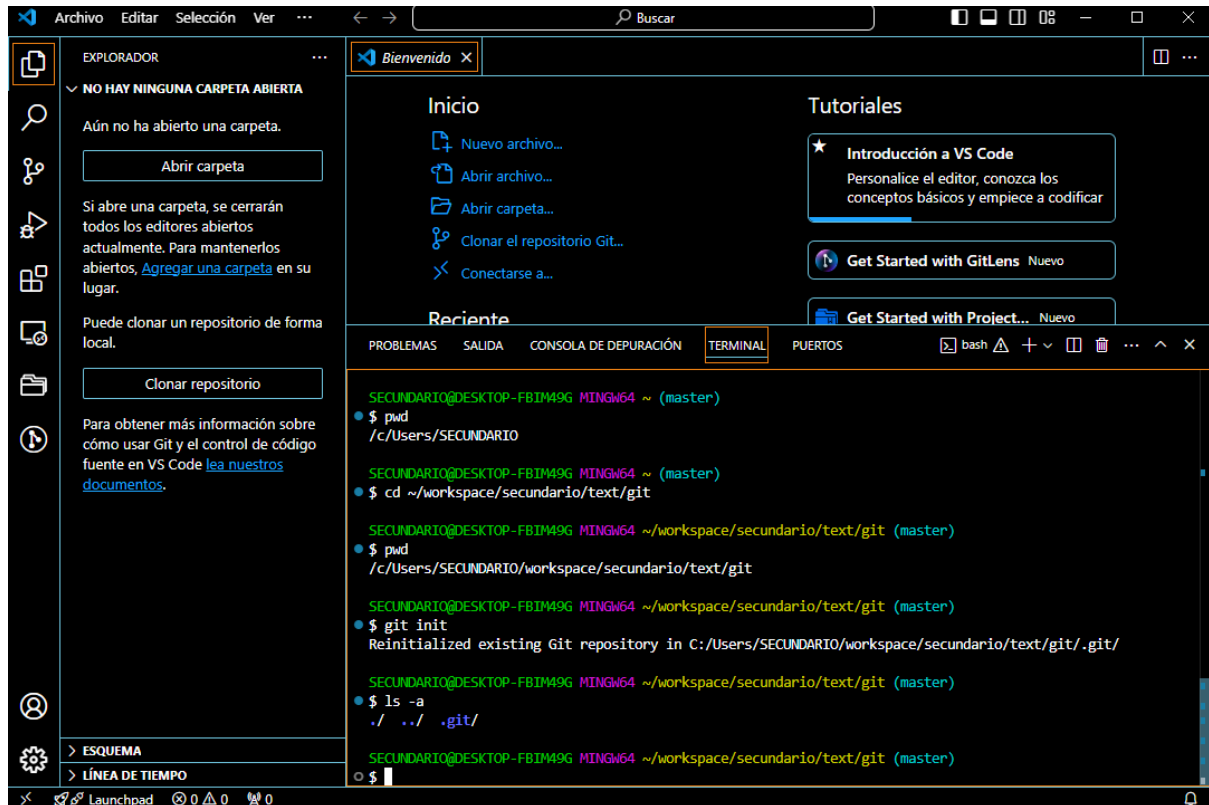
Repositorio Git vacío inicializado en `~/.`git.

Esto significa que has inicializado un repositorio vacío en un directorio llamado `.git` lo que significa que el repositorio ha sido creado.

Comprobar repositorio propio de git

Comprobar en el directorio actual con el siguiente comando:

`$ ls -a`



The screenshot shows the Visual Studio Code interface. On the left is the Explorer sidebar with a message 'NO HAY NINGUNA CARPETA ABIERTA'. The main area displays the 'Inicio' (Home) view with options like 'Nuevo archivo...', 'Abrir archivo...', 'Abrir carpeta...', 'Clonar el repositorio Git...', and 'Conectarse a...'. Below this is the 'Reciente' (Recent) section. At the bottom, the 'TERMINAL' tab is active, showing the following command history and output:

```
SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~ (master)
• $ pwd
/c/Users/SECUNDARIO

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~ (master)
• $ cd ~/workspace/secundario/text/git

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
• $ pwd
/c/Users/SECUNDARIO/workspace/secundario/text/git

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
• $ git init
Reinitialized existing Git repository in C:/Users/SECUNDARIO/workspace/secundario/text/git/.git/

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
• $ ls -a
./ ../ .git/

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
o $
```

Resultado: `.git`

Había un directorio llamado `.git` .

Este directorio se creó como resultado de ejecutar el comando `git init`.

Este directorio contiene los archivos de configuración de Git.

Compruebe qué archivos se han creado en el directorio `.git` con el siguiente comando.

`$ ls -la ~/workspace/secundario/text/git/.git`

```
• $ ls -la ~/workspace/secundario/text/git/.git
total 11
drwxr-xr-x 1 SECUNDARIO 197121  0 sep.  2 22:16 ./
drwxr-xr-x 1 SECUNDARIO 197121  0 sep.  2 22:07 ../
-rw-r--r-- 1 SECUNDARIO 197121 130 sep.  2 22:16 config
-rw-r--r-- 1 SECUNDARIO 197121  73 sep.  2 22:07 description
-rw-r--r-- 1 SECUNDARIO 197121  23 sep.  2 22:07 HEAD
drwxr-xr-x 1 SECUNDARIO 197121  0 sep.  2 22:07 hooks/
drwxr-xr-x 1 SECUNDARIO 197121  0 sep.  2 22:07 info/
drwxr-xr-x 1 SECUNDARIO 197121  0 sep.  2 22:07 objects/
drwxr-xr-x 1 SECUNDARIO 197121  0 sep.  2 22:07 refs/

SECUNDARIO@DESKTOP-FBIM49G MINGW64 ~/workspace/secundario/text/git (master)
$
```

Explicaré brevemente los archivos y directorios en el marco rojo de arriba. No es necesario recordarlo.

HEAD

- Este archivo contiene información sobre dónde se encuentra la rama de trabajo actual. Aprenderás sobre las ramas en otros textos.

Config

- Este archivo contiene varias configuraciones que se aplican al ejecutar comandos Git.

Description (descripción)

- Este archivo se utiliza para un servicio llamado GitWeb, como descripción del repositorio. Hoy en día, este archivo se utiliza muy poco, ya que es más común utilizar los servicios GitHub y GitLab.

Hooks

- Este directorio gestiona el proceso de scripting que se ejecuta por separado cuando se ejecuta cada comando Git.

Info

- En un archivo llamado `exclude` en este directorio, puedes listar los archivos que quieres tratar como fuera del alcance de la gestión de Git. Sin embargo, lo más habitual es establecerlos como no gestionados en un archivo llamado `.gitignore`.

Objects

- El directorio mantiene información sobre cada confirmación de rama. Los dos primeros caracteres del valor hash de cada commit se registran como directorio y el resto de IDs como ficheros.

Refs

- El directorio donde se gestiona la rama de trabajo. Genera un fichero con el mismo nombre de la rama creada.

Resumen

Se puede crear un repositorio local ejecutando el comando `git init`.

Se crea un archivo de configuración Git en directorio `.git`.