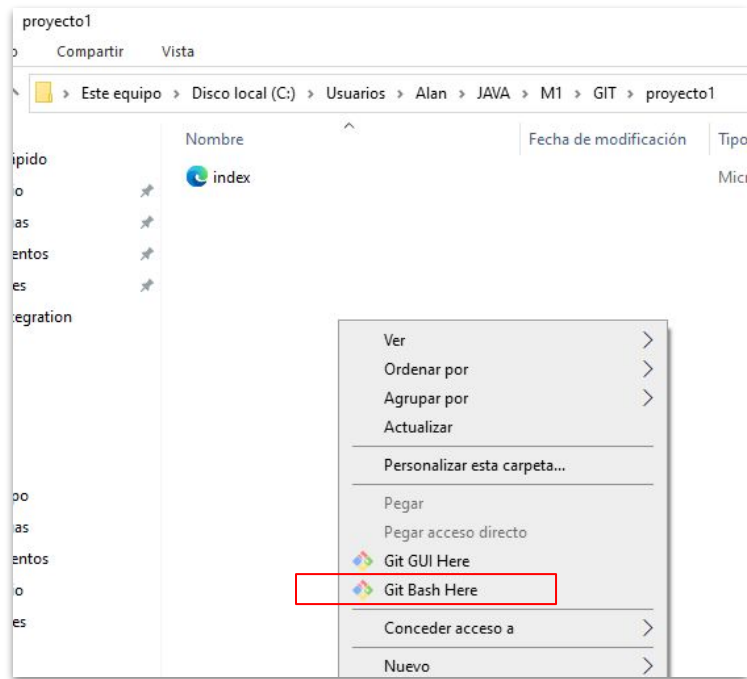


# Inicializar Git en un proyecto con Git Bash

Para que git funcione sobre un proyecto primero debe ser inicializado sobre el mismo, a partir de entonces git podrá rastrear nuevos cambios y versionarlos.

Para inicializar git haciendo uso de la herramienta Git Bash primero abrimos la carpeta raíz del proyecto sobre el que se quiere aplicar git.

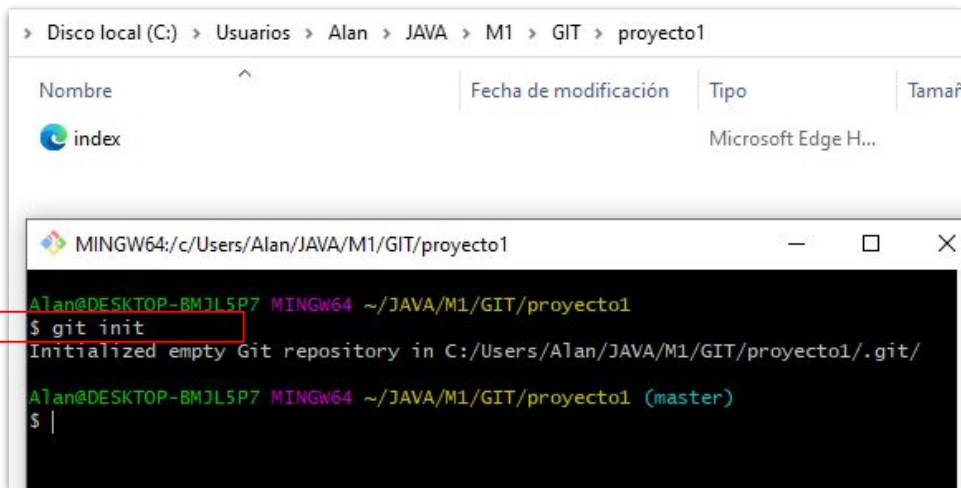


# Inicializar Git en un proyecto con Git Bash

Una vez abierta la herramienta Git Bash sobre el proyecto deseado ejecutaremos el comando:

**git init**

A partir de este momento la tecnología git rastreará los nuevos cambios que se produzcan en los archivos. Para registrar los cambios utilizaremos los comandos git add y git commit.



```
Disco local (C:) > Usuarios > Alan > JAVA > M1 > GIT > proyecto1

Nombre ^
Fecha de modificación Tipo Tamaño
index Microsoft Edge H...

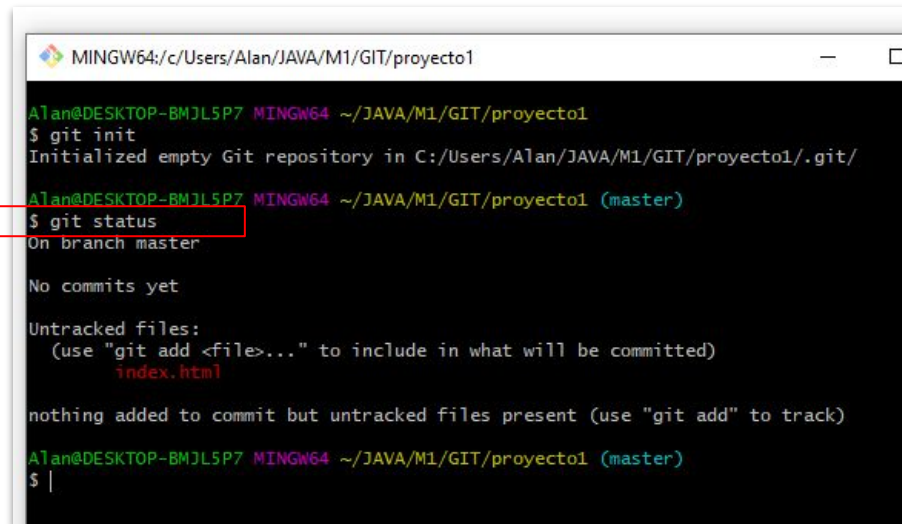
MINGW64: c:/Users/Alan/JAVA/M1/GIT/proyecto1
Alan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1
$ git init
Initialized empty Git repository in C:/Users/Alan/JAVA/M1/GIT/proyecto1/.git/
Alan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ |
```

# Registrar cambios en un proyecto con Git Bash

Sólo es necesario inicializar git en un proyecto la primera vez.

A partir de entonces registramos los cambios realizados cada vez que el proyecto alcance un punto que se desea guardar.

El comando **git status** nos permite observar el estado de los archivos. Inicialmente todos los archivos cuando se crean están en estado untracked, esto significa que git sabe que existe el archivo pero no existía en la versión anterior, para que lo incluya en la versión debemos registrarlo en el **área de ensayo** y el **área de repositorio local** con los comandos **git add** y **git commit**.



```
MINGW64:/c/Users/Alan/JAVA/M1/GIT/proyecto1

Alan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1
$ git init
Initialized empty Git repository in C:/Users/Alan/JAVA/M1/GIT/proyecto1/.git/

Alan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)

Alan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ |
```

# Registrar cambios en un proyecto con Git Bash

Indicamos a git que registre el nuevo archivo en el área de ensayo con el comando:

**git add index.html**

**Nota:** si queremos registrar todos los archivos de golpe podemos ejecutar el comando:

**git add .**

```
Alan@DESKTOP-BMJL5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git add index.html

Alan@DESKTOP-BMJL5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git status
On branch master

No commits yet

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

# Registrar cambios en un proyecto con Git Bash

Una vez en el **área de ensayo** si queremos registrarlo en el **área de repositorio local** para obtener un snapshot del archivo o archivos ejecutaremos el comando:

**git commit -m “descripción cambios”**

El mensaje debe ser un pequeño texto de menos de 50 caracteres que describa de manera resumida los cambios realizados. Normalmente se suele utilizar el id y título de la tarea que se ha realizado, los cuales proceden de la plataforma de gestión de proyectos utilizada para la planificación de los desarrollos.

```
Alan@DESKTOP-BMJL5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git commit -m "Estructura base"
[master (root-commit) df8469c] Estructura base
1 file changed, 11 insertions(+)
create mode 100644 index.html

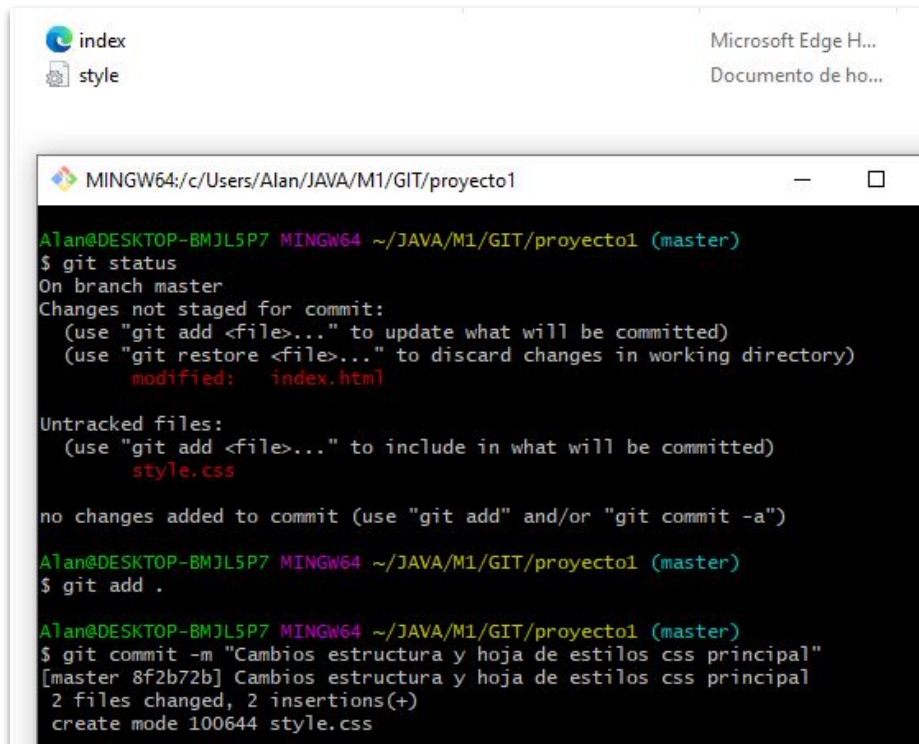
Alan@DESKTOP-BMJL5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git status
On branch master
nothing to commit, working tree clean
```

# Registrar cambios en un proyecto con Git Bash

Git detecta nuevos archivos y también los cambios sobre archivos ya registrados. Con **git status** podemos ver los cambios detectados.

Realizando el mismo proceso de **git add** y **git commit** podemos registrar los nuevos cambios.

**Nota:** para registrar directamente al área de repositorio local sin ejecutar git add se puede ejecutar el comando commit con el parámetro **-a**:  
**git commit -a -m 'Ejemplo mensaje'**



The screenshot shows a Windows File Explorer window with the address bar displaying 'index' and 'style'. Below it, a Git Bash terminal window is open, showing the output of the 'git status' command. The terminal output indicates that there are changes not staged for commit, specifically 'index.html' which has been modified. It also shows untracked files, including 'style.css'. The terminal then shows the execution of 'git add .' which stages all untracked files. Finally, the terminal shows the execution of 'git commit -m "Cambios estructura y hoja de estilos css principal"', which creates a new commit with the message 'Cambios estructura y hoja de estilos css principal' and includes the 2 files changed (2 insertions(+)).

```
ALan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (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

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

no changes added to commit (use "git add" and/or "git commit -a")

ALan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git add .

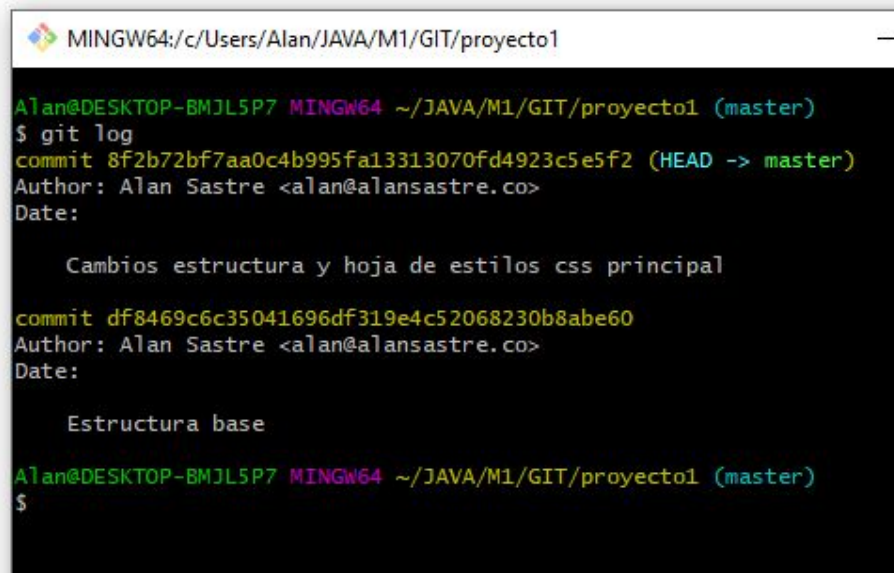
ALan@DESKTOP-BM3L5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git commit -m "Cambios estructura y hoja de estilos css principal"
[master 8f2b72b] Cambios estructura y hoja de estilos css principal
2 files changed, 2 insertions(+)
create mode 100644 style.css
```

# Registrar cambios en un proyecto con Git Bash

Con **git log** podemos ver el historial de cambios registrados.

**Nota:** el parámetro **pretty** nos permite mostrar todos los commit en una sola línea para una mejor visualización.

**git log --pretty=oneline**



```
MINGW64:/c:/Users/Alan/JAVA/M1/GIT/proyecto1

Alan@DESKTOP-BMJL5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$ git log
commit 8f2b72bf7aa0c4b995fa13313070fd4923c5e5f2 (HEAD -> master)
Author: Alan Sastre <alan@alansastre.co>
Date:

    Cambios estructura y hoja de estilos css principal

commit df8469c6c35041696df319e4c52068230b8abe60
Author: Alan Sastre <alan@alansastre.co>
Date:

    Estructura base

Alan@DESKTOP-BMJL5P7 MINGW64 ~/JAVA/M1/GIT/proyecto1 (master)
$
```