

TUTORIAL GIT

Para empezar a trabajar con Git, se puede descargar de la página <https://git-scm.com/>, una vez instalado se procede a abrir la consola como se muestra a continuación.

```
MINGW64:/c/Users/Asus
Asus@DESKTOP-JMMFTCG MINGW64 ~ (master)
$ |
```

Lo primero que se va a hacer es dirigirse a la carpeta donde se va a crear el repositorio local.

```
MINGW64:/d/edwin/git
Asus@DESKTOP-JMMFTCG MINGW64 ~ (master)
$ cd ..

Asus@DESKTOP-JMMFTCG MINGW64 /c/Users
$ cd ..

Asus@DESKTOP-JMMFTCG MINGW64 /c
$ cd..
bash: cd..: command not found

Asus@DESKTOP-JMMFTCG MINGW64 /c
$ cd ..

Asus@DESKTOP-JMMFTCG MINGW64 /
$ cd d

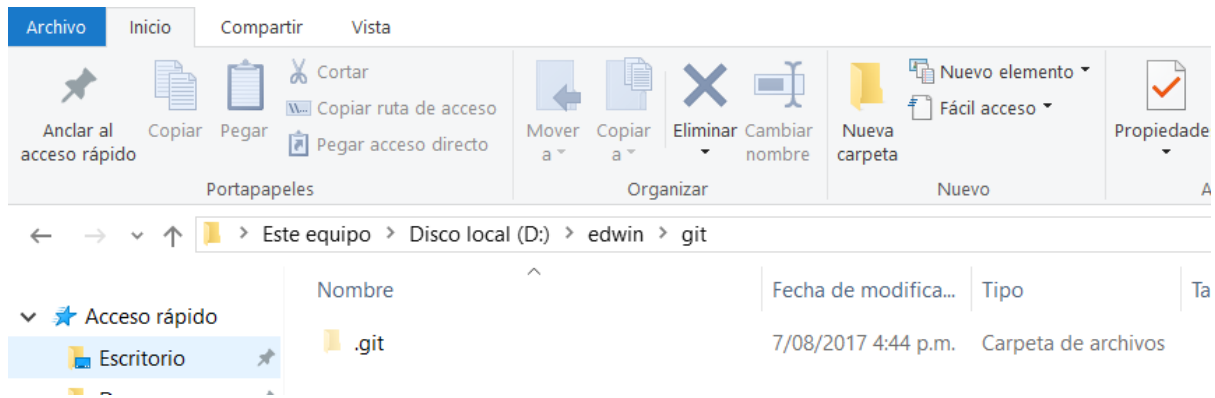
Asus@DESKTOP-JMMFTCG MINGW64 /d
$ cd edwin/

Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin
$ cd git/

Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git
$ |
```

Una vez situados en el repositorio local, con el comando git init, que se encarga de crear un subdirectorio .git, que contiene todos los archivos necesarios para trabajar con Git.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git
$ git init
Initialized empty Git repository in D:/edwin/git/.git/
```



Luego de haber usado el comando `init`, se puede comprobar el estado del repositorio mediante el comando `git status`, el cual nos debe indicar que no hay información de commits, ni archivos por subir.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

En la carpeta donde se creó el repositorio, se va a crear un archivo `txt` que se llamará `prueba.txt`. Al usar de nuevo el comando `status` podemos ver que no está indicando que hay un archivo pendiente por subir.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git status
On branch master

No commits yet

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

  prueba.txt

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

El comando usado para subir el archivo al repositorio local, se puede ver en la imagen siguiente.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git add prueba.txt
```

NOTA: Si se necesitaran adicionar varios archivos de un mismo tipo, por ejemplo varios txt, se puede usar el siguiente comando: `git add '*.txt'`.

Si se usa de nuevo el comando `status`, después de haber adicionado `prueba.txt`, se puede ver que indica que hay un archivo nuevo en el repositorio local, que está pendiente de hacer `commit`.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git status
On branch master

No commits yet

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

        new file:   prueba.txt
```

Mientras con el comando `add`, lo que hicimos fue indicar que hay un archivo listo para subir, el comando encargado de confirmar y subir el archivo es `git commit`.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git commit -m 'Se sube prueba'
[master (root-commit) a3e04c0] Se sube prueba
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 prueba.txt
```

Si se desea ver el historial de lo realizado en el repositorio, se debe usar el comando visto en la siguiente imagen:

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git log
commit a3e04c048e36e594eae840bb89f16beeab1221fc (HEAD -> master)
Author: Edwin Hernandez <ehernandez@j4sysol.com>
Date:   Mon Aug 7 16:49:38 2017 -0500


    Se sube prueba
```

Posteriormente, en GitHub se va a crear un repositorio, siguiendo los pasos que se muestran a continuación.

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

 edhernandez ▾

Repository name

pruebaSoftware1 ✓

Great repository names are short and memorable. Need inspiration? How about **fictional-octo-memory**.

Description (optional)



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.



Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None ▾

Add a license: None ▾



Create repository

edhernandez / pruebaSoftware1

Unwatch ▾ 1

★ Star 0

Fork 0

Code

Issues 0

Pull requests 0

Projects 0

Wiki

Settings

Insights ▾

Quick setup — if you've done this kind of thing before

Set up in Desktop

or

HTTPS

SSH

https://github.com/edhernandez/pruebaSoftware1.git



We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# pruebaSoftware1" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/edhernandez/pruebaSoftware1.git
git push -u origin master
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/edhernandez/pruebaSoftware1.git
git push -u origin master
```



La url que se tiene después de crear el proyecto en GitHub, se va a usar para enviar nuestro repositorio local a la nube, mediante el siguiente comando.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git remote add origin https://github.com/edhernandez/pruebaSoftware1.git
```

Mediante el comando push se le indica a Git dónde poner nuestros commits cuando estén listos. Así que se va a enviar los cambios locales al repositorio en GitHub.

En el siguiente comando podemos ver que se usa el nombre dado en el paso anterior, origin, y el nombre de la rama local predeterminada es master. El -u le dice a Git que recuerde los parámetros, de modo que la próxima vez se pueda simplemente ejecutar git push y Git sabrá qué hacer.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git push -u origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 223 bytes | 223.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/edhernandez/pruebaSoftware1.git
* [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

Si se observa en GitHub ya se puede ver el documento que se ha adicionado.

The screenshot shows the GitHub interface for the repository 'edhernandez / pruebaSoftware1'. At the top, there's a navigation bar with links for 'This repository', 'Search', 'Pull requests', 'Issues', 'Marketplace', and 'Gist'. Below this, the repository name is displayed along with statistics: 1 commit, 1 branch, 0 releases, and 0 contributors. A 'Clone or download' button is visible. The main content area shows a commit by 'Edwin Hernandez' titled 'Se sube prueba' with a file named 'prueba.txt' added. A message at the bottom encourages adding a README file.

Cuando los desarrolladores están trabajando en una característica o un error que a menudo crean una copia (rama) de su código, lo que les permite trabajar por separado. Luego, cuando terminen, pueden fusionar esta rama en su rama principal. Para crear una nueva rama se usa el comando git branch más el nombre de la nueva rama, como se ve a continuación.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git branch rama
```

Ahora, si escribe git branch verá dos ramas locales: una rama principal llamada master y su nueva rama llamada rama.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git branch
* master
  rama
```

Para cambiar de rama se usa el comando git checkout **nombreRama**.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git checkout rama
Switched to branch 'rama'

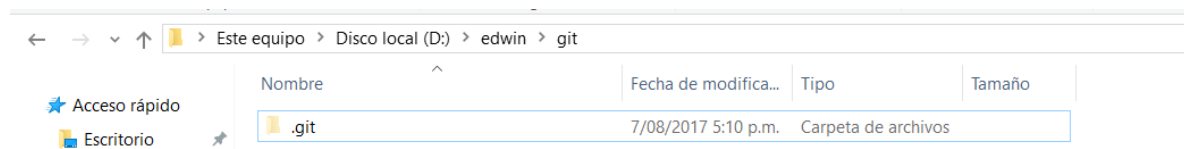
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (rama)
$ git branch
  master
* rama
```

De la rama que se acabó de crear, se va a eliminar el archivo subido anteriormente, para lo cual se usa el siguiente comando:

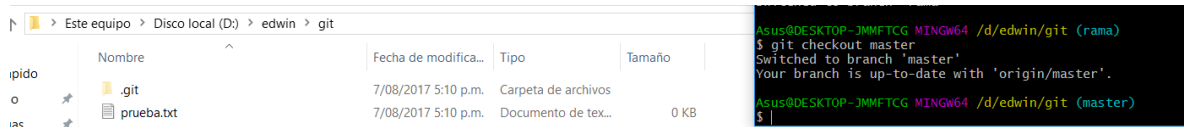
```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (rama)
$ git rm 'prueba.txt'
rm 'prueba.txt'

Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (rama)
$ git commit -m 'se elimina archivo'
[rama 8e82703] se elimina archivo
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 prueba.txt
```

Si se observa en la carpeta donde manejamos el repositorio se ve que se ha eliminado el archivo.



Pero debido a que sólo se eliminó de rama, si se hace checkout de master, podemos ver que vuelve a aparecer el archivo.



Si se desea borrar también de la rama master, lo que se va a hacer es fusionar las ramas mediante el comando `git merge nombreRama`, como se observa en la siguiente imagen, lo que hacemos luego es eliminar la rama con `git branch -d nombreRama` y por último se confirman los cambios usando `push`. Ya si se verifica en el GitHub no debe haber ningún archivo, y si se usa `git branch` sólo debe aparecer la rama master.

```
Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git merge rama
Updating a3e04c0..8e82703
Fast-forward
 prueba.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 prueba.txt

Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git branch -d rama
Deleted branch rama (was 8e82703).

Asus@DESKTOP-JMMFTCG MINGW64 /d/edwin/git (master)
$ git push
Counting objects: 2, done.
Writing objects: 100% (2/2), 207 bytes | 3.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0)
To https://github.com/edhernandez/pruebaSoftware1.git
 a3e04c0..8e82703  master -> master
```

EJERCICIO

1. Para el ejercicio 1, se debe crear un proyecto java que permita sumar y restar dos números, y subirlo a git, se va a trabajar en grupos de a tres personas, la primera persona se encargará de subir el proyecto a un repositorio GitHub, sin incluir funcionalidad, la segunda y tercera persona bajarán el proyecto usando el comando `clone`, y cada uno creará una rama, cada uno en su rama subirá la parte de sumar números y el otro hará la resta, la idea es que después fusionen las ramas y eliminen las ramas creadas, y dejen todo en la rama master. Adicionalmente deben agregarme en GitHub para que pueda ver el proyecto, mi usuario es edhernandez.
2. El segundo ejercicio consiste en crear otro proyecto en GitHub donde se encargarán de subir los requerimientos del proyecto que se va a trabajar durante el semestre, al igual que en el ejercicio anterior me deben agregar en el proyecto.