



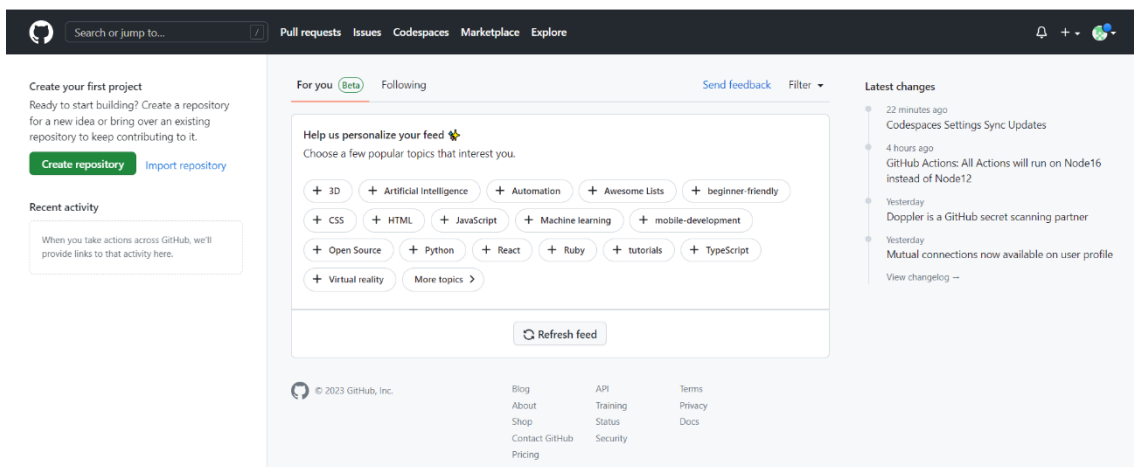
**Local:** Es nuestro ordenador donde tenemos nuestros proyectos y podemos trabajar solo nosotros.

**Intermediario:** Es el encargado de poder pasar nuestro proyecto a la nube (en este caso nuestra "nube" es GitHub)

**Repositorio en la nube:** Es donde almacenamos nuestros proyectos en la nube. Este repositorio es bueno para los desarrolladores (Programadores)

## Paso 1:

Crear una cuenta en Git Hub.

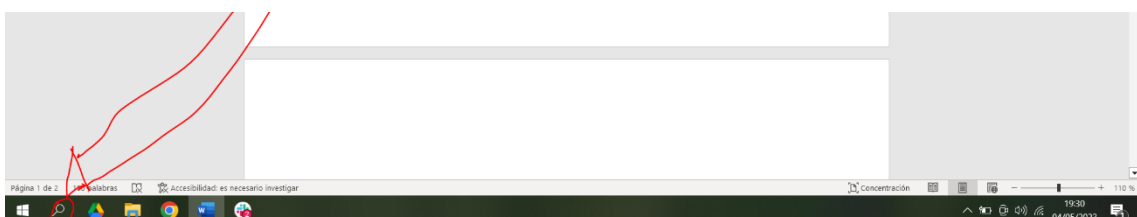


## Paso 2:

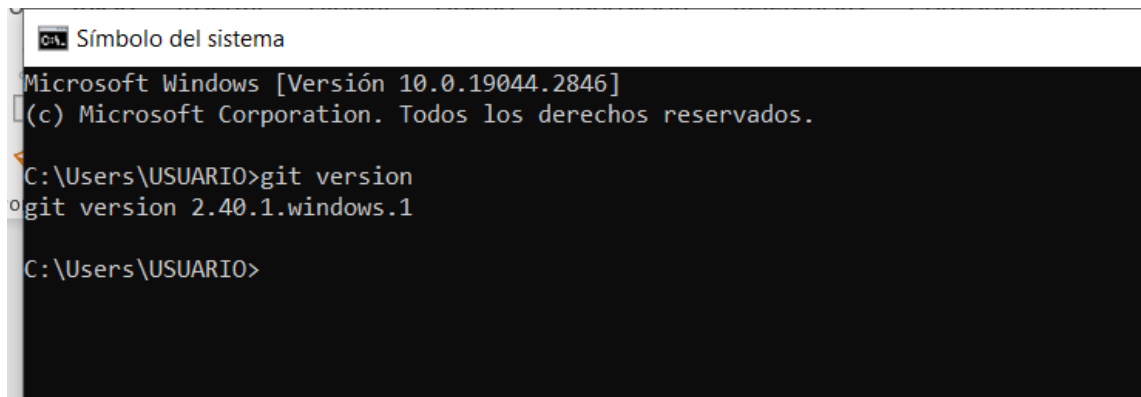
Descargar Git e instalarlo (Solo debes darle next a todo, no cambies nada).

Para verificar si está bien instalado en tu ordenador debes abrir el CMD.

Le das en la tecla Windows de tu teclado o en buscar a lado izquierdo de tu pantalla y poner cmd.



- Comando para verificar si está correctamente instalado.  
**git versión**



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19044.2846]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\USUARIO>git version
git version 2.40.1.windows.1

C:\Users\USUARIO>
```

Ese es el resultado que debe salir en mi caso es la versión 2.40.1

### Paso 3:

Ejecuta los siguientes comandos con tu información para establecer un nombre de usuario y un correo electrónico predeterminados para cuando vayas a salvar tu trabajo.

**git config --global user.name "(Escribes tu nombre)"**

**git config --global user.email "(Escribes tu email)"**

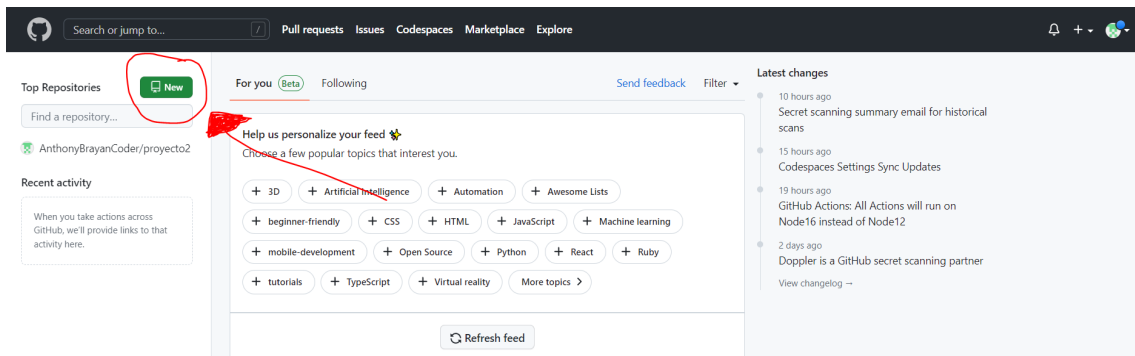


```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19044.2846]
(c) Microsoft Corporation. Todos los derechos reservados.

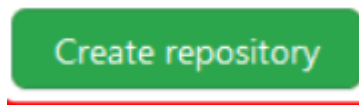
C:\Users\USUARIO>git config --global user.name "AnthonyBrayan"
C:\Users\USUARIO>git config --global user.email "brayan.farronan@gmail.com"
C:\Users\USUARIO>
```

### Paso 4:

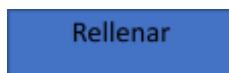
Crear un repositorio en GitHub para comenzar a crear el proyecto.



Yo ya tengo creado otros repositorios por eso me sale un NEW, si en caso es la primera vez que vas a crear un repositorio te va a salir esto:



Te va a salir este formulario, importante completa lo que está encerrado en círculo rojo.



### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \* AnthonyBrayanCoder Repository name \*

Great repository names are short and memorable. Need inspiration? How about [super-duper-pancake?](#)

Description (optional)

☒ Public Anyone on the internet can see this repository. You choose who can commit.

☐ Private You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file This is where you can write a long description for your project. [Learn more about READMEs.](#)

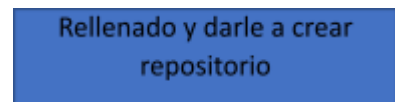
Add .gitignore .gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

🔔 You are creating a public repository in your personal account.



### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \* AnthonyBrayanCoder Repository name Example Example is available.

Great repository names are short and memorable. Need inspiration? How about [bug-free-doodle?](#)

Description (optional)

☒ Public Anyone on the internet can see this repository. You choose who can commit.

☐ Private You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore .gitignore template: None

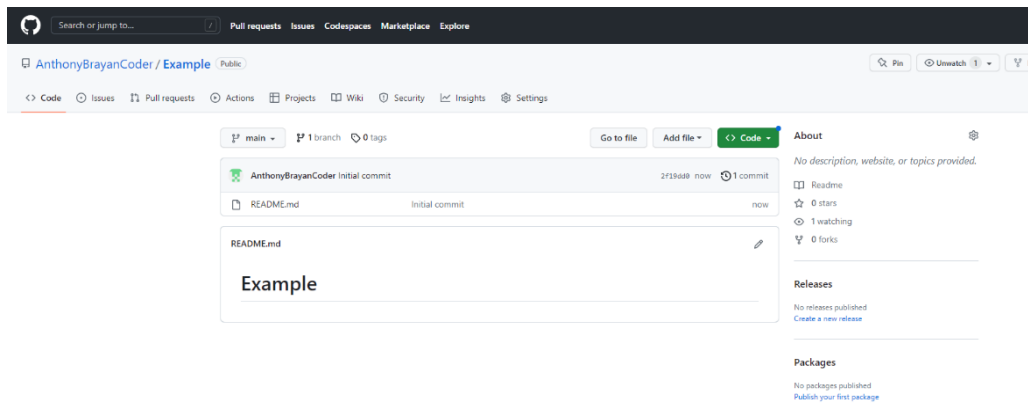
Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set `main` as the default branch. Change the default name in your settings.

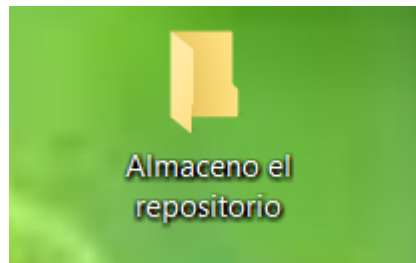
🔔 You are creating a public repository in your personal account.



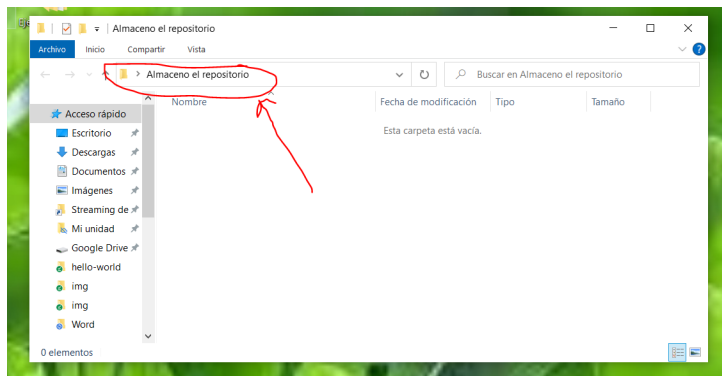
## Paso 5

Crear una carpeta en donde guardarás el repositorio de tu proyecto.

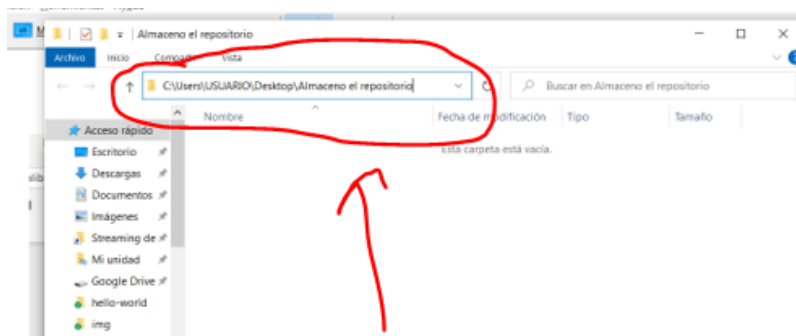
Un ejemplo crearé una carpeta en mi escritorio y allí se almacenará el repositorio que clone de GitHub.



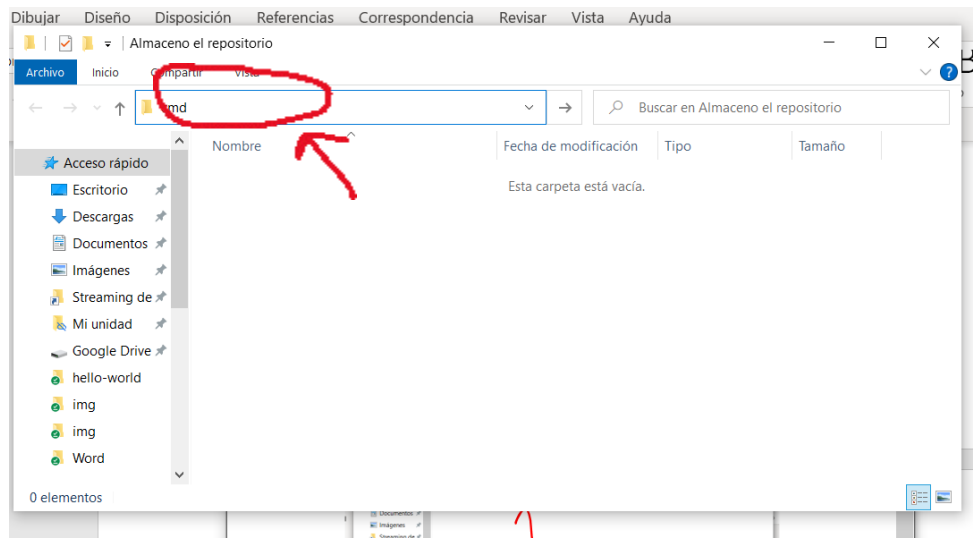
Es importante abrir la carpeta creada para abrir en CMD y darle click en ese espacio.



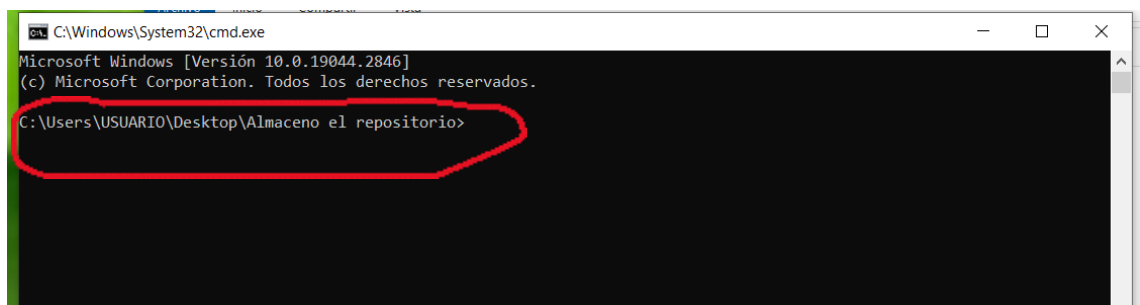
Para que de esa manera te muestre la ruta de la carpeta donde vas a almacenar el repositorio.



Cuando le des click en ese espacio borra la ruta, escribe cmd y le das enter.



De esa manera entrarás a cmd con la ruta de la carpeta donde se almacenará tu repositorio de gitHub.

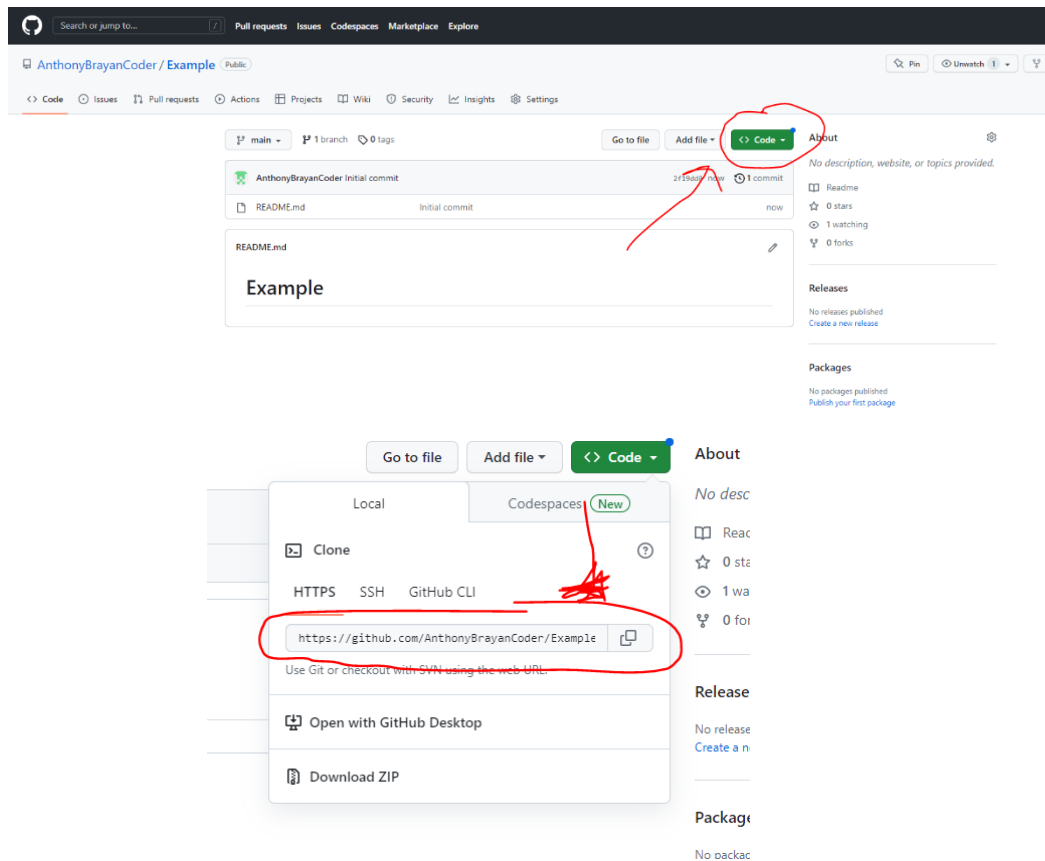


Importante observa que debes estar dentro de la carpeta donde se realizará el almacenamiento de tu repositorio de gitHub y allí ya podemos trabajar.

## Paso 6

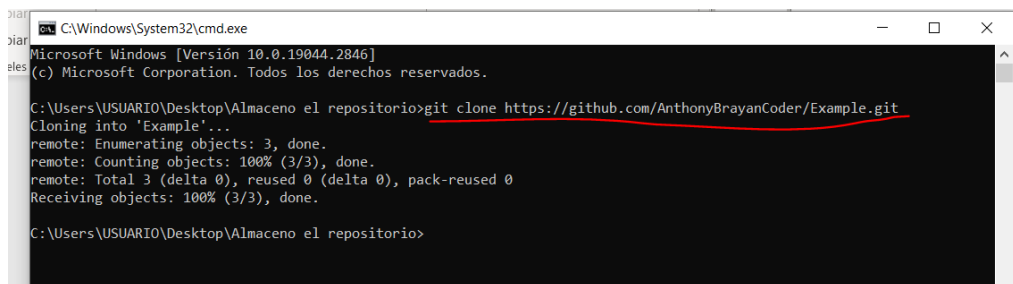
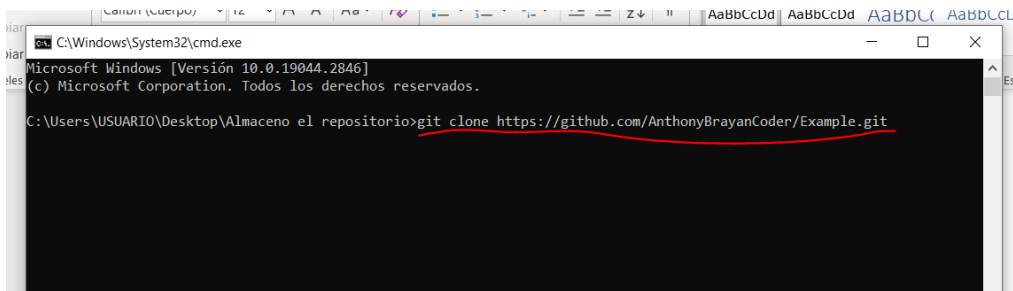
Clonar el repositorio para tenerlo en nuestro ordenador.

Es importante primero copiar el link de tu repositorio de GitHub, puedes hacerlo de la siguiente manera:

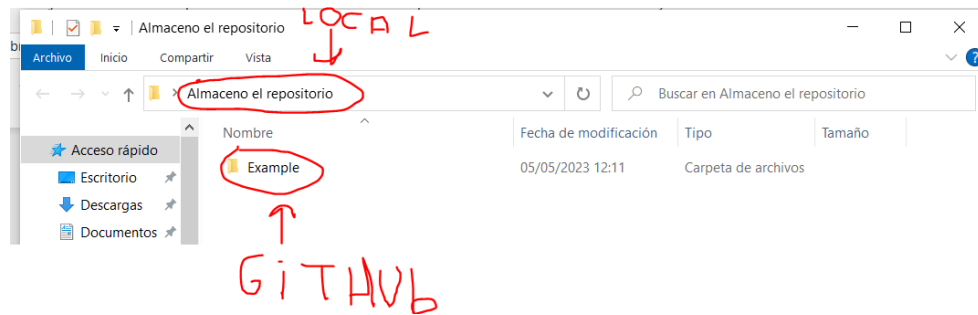


Ahora ya sabiendo y copiado el link de tu repositorio vamos a CMD.

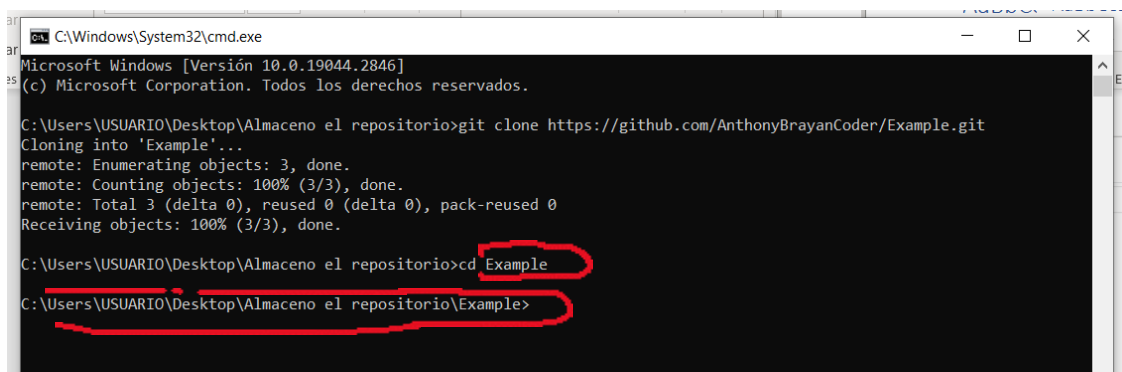
**git clone (link de tu repositorio)**



Con esto ya puedes verificar que ya se a podido clonar tu repositorio en tu carpeta que has creado en tu escritorio.

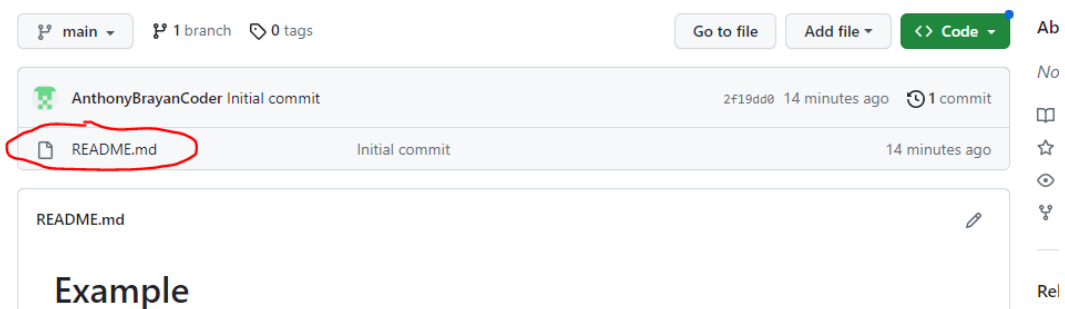


Importanten ahora en CMD debes entrar dentro de tu repositorio y puedes hacer asi:  
**cd (Nombre de tu repositorio que has almacenado en tu carpeta del escritotio)**

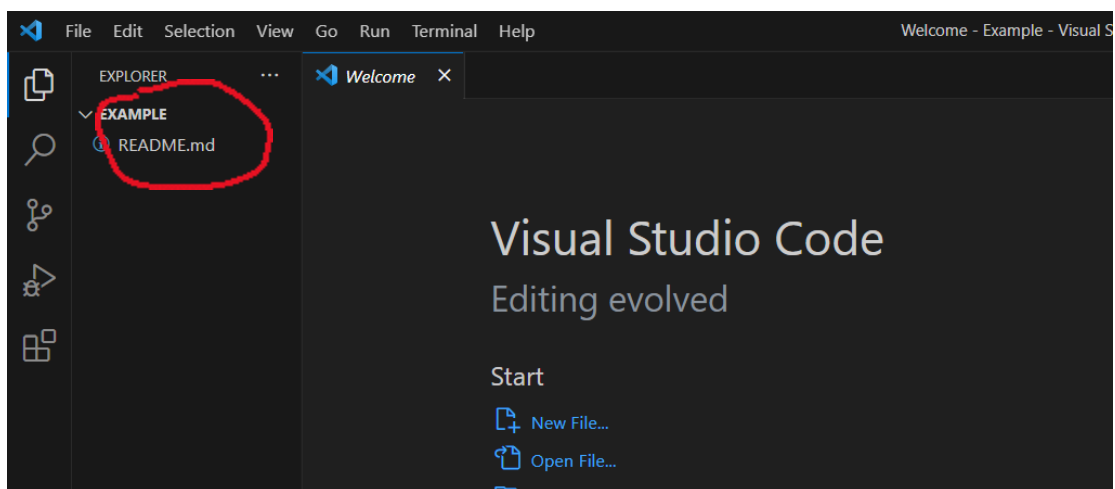
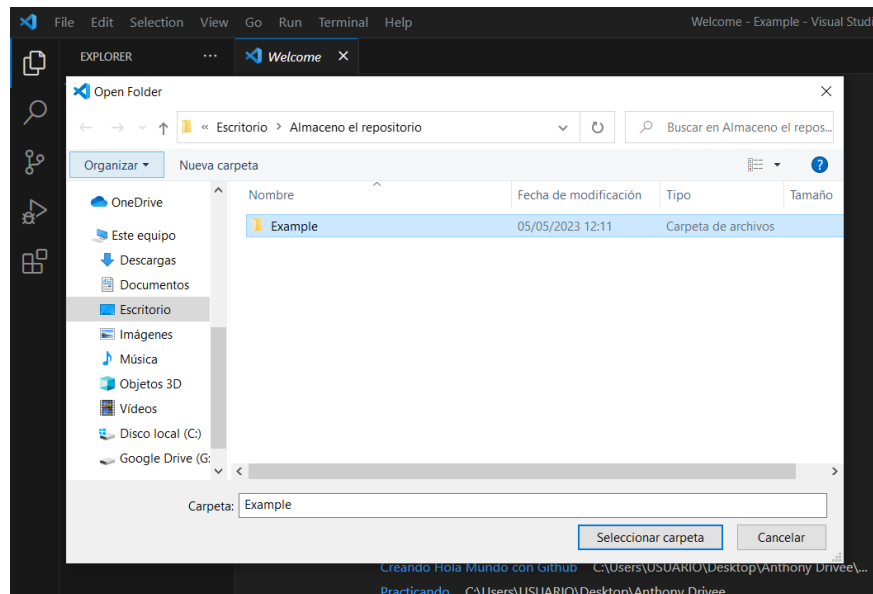


Ahora que ya estamos dentro de nuestro proyecto podemos trabajar. Abre tu Visual code y ponte a programar...

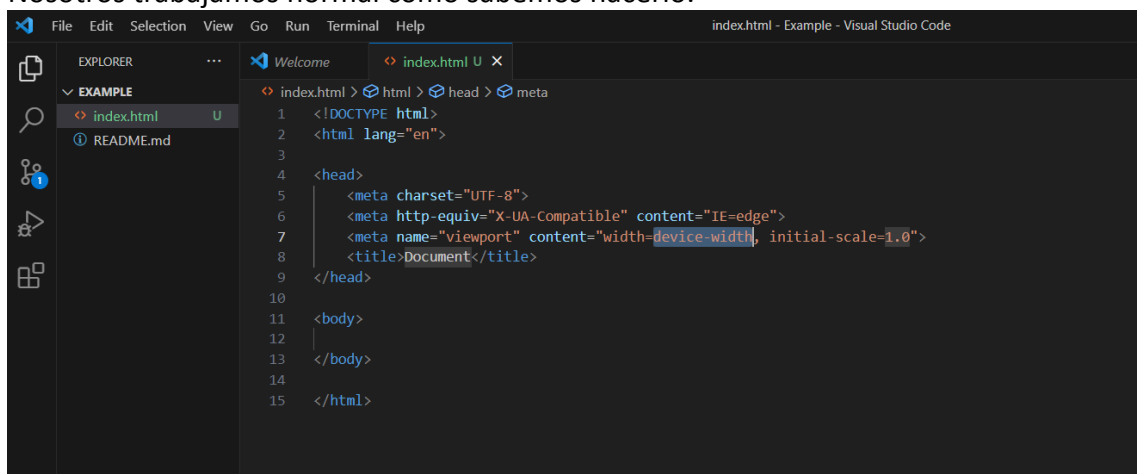
Actualmente en el repositorio de GitHub solo tenemos este documento.



Si abrimos en visual code desde la carpeta que hemos creado en nuestro escritorio se vería así.



Nosotros trabajamos normal como sabemos hacerlo.





Ahora para subir los cambios que hemos hecho, hacemos lo siguiente:

¡IMPORTANTE! Debemos estar en cmd en la ruta dentro de nuestro proyecto para que nos reconozca los comandos, ahora ponemos el siguiente comando.

**git status**

```
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git status
On branch main
Your branch is up to date with 'origin/main'.

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)
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>
```

**git add \***

```
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git status
On branch main
Your branch is up to date with 'origin/main'.

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)
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git add *
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>
```

**git commit -m "(Escribir una pequeña descripción dentro de las comillas)"**

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.19044.2846]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\USUARIO\Desktop\Almaceno el repositorio>git clone https://github.com/AnthonyBrayanCoder/Example.git
Cloning into 'Example'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

C:\Users\USUARIO\Desktop\Almaceno el repositorio>cd Example

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git status
On branch main
Your branch is up to date with 'origin/main'.

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)
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git add *

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git commit -m "Subiendo ejemplo"
[main 57d1e86] Subiendo ejemplo
1 file changed, 15 insertions(+)
create mode 100644 index.html
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>
```

Por buena práctica siempre poner este comando para actualizar el proyecto de la nube si en caso alguien a modificado algo del proyecto:

**git pull**

```
C:\Users\USUARIO\Desktop\Almaceno el repositorio>git clone https://github.com/AnthonyBraya
Cloning into 'Example'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

C:\Users\USUARIO\Desktop\Almaceno el repositorio>cd Example

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git status
On branch main
Your branch is up to date with 'origin/main'.

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)

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git add *

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git commit -m "Subiendo ejemplo"
[main 57d1e86] Subiendo ejemplo
 1 file changed, 15 insertions(+)
 create mode 100644 index.html

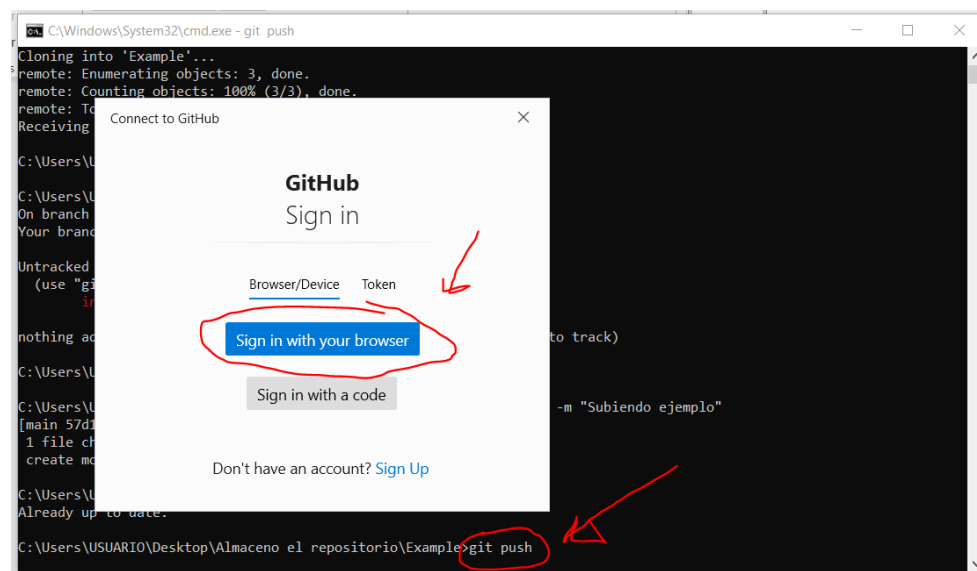
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git pull
Already up to date.

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>
```

En este caso nadie a hecho cambios en el repositorio.

Ahora seguimos con el último comando para subir nuestro código a la nube:

**git push**



Cuando puse el git push y es la primera vez que haces esto tienes que iniciar sesion solo por única vez, después ya no será necesario y lo subirá directo.

```
C:\Windows\System32\cmd.exe
Your branch is up to date with 'origin/main'.

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)

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git add *

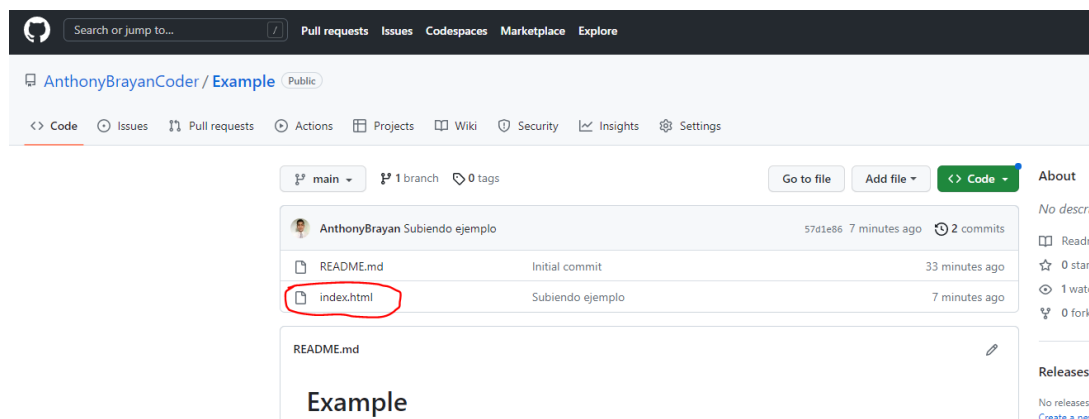
C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git commit -m "Subiendo ejemplo"
[main 57d1e86] Subiendo ejemplo
 1 file changed, 15 insertions(+)
 create mode 100644 index.html

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git pull
Already up to date.

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>git push
info: please complete authentication in your browser...
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 468 bytes | 468.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AnthonyBrayanCoder/Example.git
 2f19dd0..57d1e86  main -> main

C:\Users\USUARIO\Desktop\Almaceno el repositorio\Example>
```

Despues de confirmar y autorizar todo ya puedes verificar que se actualizó en el repositorio (Actualiza la página de Git Hub para ver los cambios).



Observación, ahora en adelante cada cambio que realices en tu ciusal code y quieres subirlo a la nube (GitHub) vas aplicar estos 5 pasos que anteriormente le expliqué

1. git status
2. git add \*
3. git commit -m "(Descripción corta dentro de las comillas)"
4. git pull
5. git push