

Crear un proyecto

Visual Studio 2019

Abrir recientes

Esta semana

 UrielAdonay.sln 10/06/2023 11:38 a. m.
C:\Users\MONZON\source\repos\UrielAdonay

Este mes

 SinalGreen.sln 03/06/2023 11:22 a. m.
C:\Users\MONZON\source\repos\SinalGreen

 ConsoleApp.sln 27/05/2023 11:15 a. m.
C:\Users\MONZON\source\repos\ConsoleApp

 Menu.sln 27/05/2023 10:47 a. m.


 menuu.sln 27/05/2023 10:41 a. m.
C:\Users\MONZON\source\repos\menuu

 PrimerPain 20/05/2023 10:21 a. m.
C:\Users\MONZON\source\repos\PrimerP

Anterior

 ProyectoEjemploFramework.sln 25/04/2021 09:16 a. m.
C:\Users\MONZON\source\repos\ProyectoEjemploFramework

Tareas iniciales

 **Clonar un repositorio**
Obtiene código desde un repositorio en línea, como GitHub o Azure DevOps.

 **Abrir un proyecto o una solución**
Abre un archivo .sln o proyecto de Visual Studio local.

 **Abrir una carpeta local**
Navegar y editar el código en cualquier carpeta

 **Crear un proyecto**
Elija una plantilla de proyecto con la técnica scaffolding de código para comenzar.

[Continuar sin código](#) →

En la siguiente pantalla seleccionare una aplicación Windows Forms

Crear un proyecto

Plantillas de proyecto recientes

 Aplicación de consola (.NET Core) C#


 Aplicación web ASP.NET Core C#

Buscar plantillas (Alt+S)

[Borrar todo](#)


C# Todas las plataformas Todos los tipos de proy...


C# Linux macOS Windows Biblioteca Web

 **Servicio Worker**
Una plantilla de proyecto para crear un servicio Worker con .NET Core.
C# Linux macOS Windows Nube Servicio

 **Proyecto de prueba de MSTest (.NET Core)**
Proyecto que contiene pruebas unitarias de MSTest que se pueden ejecutar en .NET Core en Windows, Linux y MacOS.
C# Linux macOS Windows Prueba

 **Proyecto de prueba NUnit (.NET Core)**
Un proyecto que contiene tests de NUnit que pueden ejecutarse en .NET Core en Windows, Linux y MacOS.
C# Linux macOS Windows Escritorio Prueba Web

 **Aplicación de Windows Forms (.NET Framework)**
Proyecto para crear una aplicación con una interfaz de usuario de Windows Forms (WinForms)
C# Windows Escritorio

 **Aplicación de WPF (.NET Framework)**
Aplicación cliente de Windows Presentation Foundation.
C# XAML Windows Escritorio

 **WPF App (.NET)**
Aplicación cliente de Windows Presentation Foundation.
C# XAML Windows Escritorio

 **WPF Custom Control Library (.NET)**
Biblioteca de controles personalizados de Windows Presentation Foundation.

[Atrás](#)

[Siguiente](#)

Le pongo nombre

Configure su nuevo proyecto

Aplicación de Windows Forms (.NET Framework) C# Windows Escritorio

Nombre del proyecto

WindowsFormProgramacion1

Ubicación

C:\Users\MONZON\source\repos --

Nombre de la solución ⓘ

WindowsFormProgramacion1

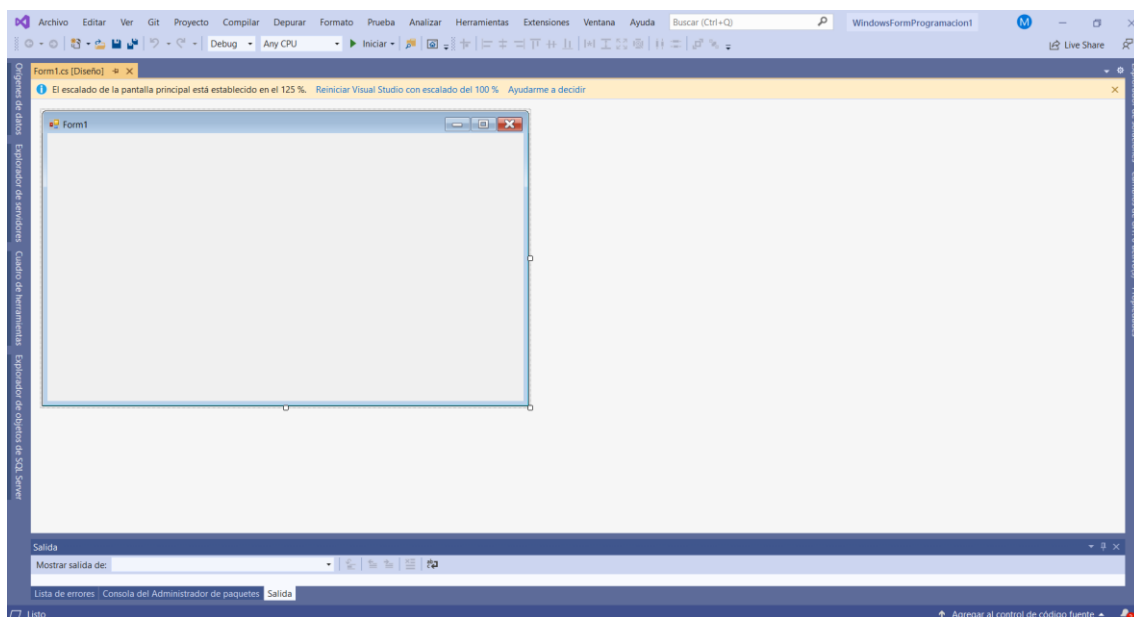
☐ Colocar la solución y el proyecto en el mismo directorio

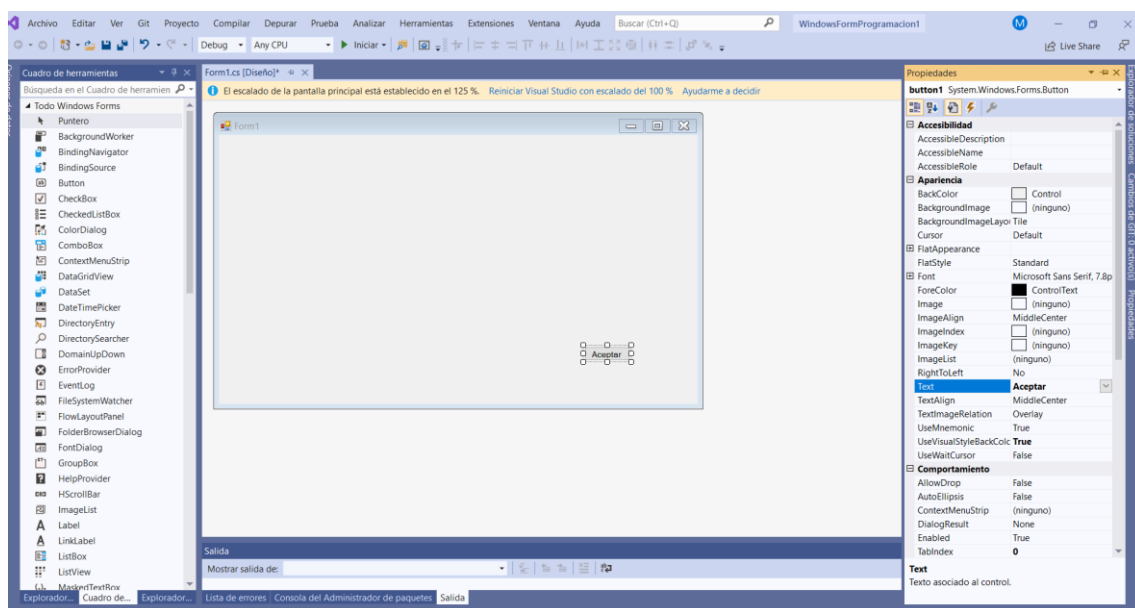
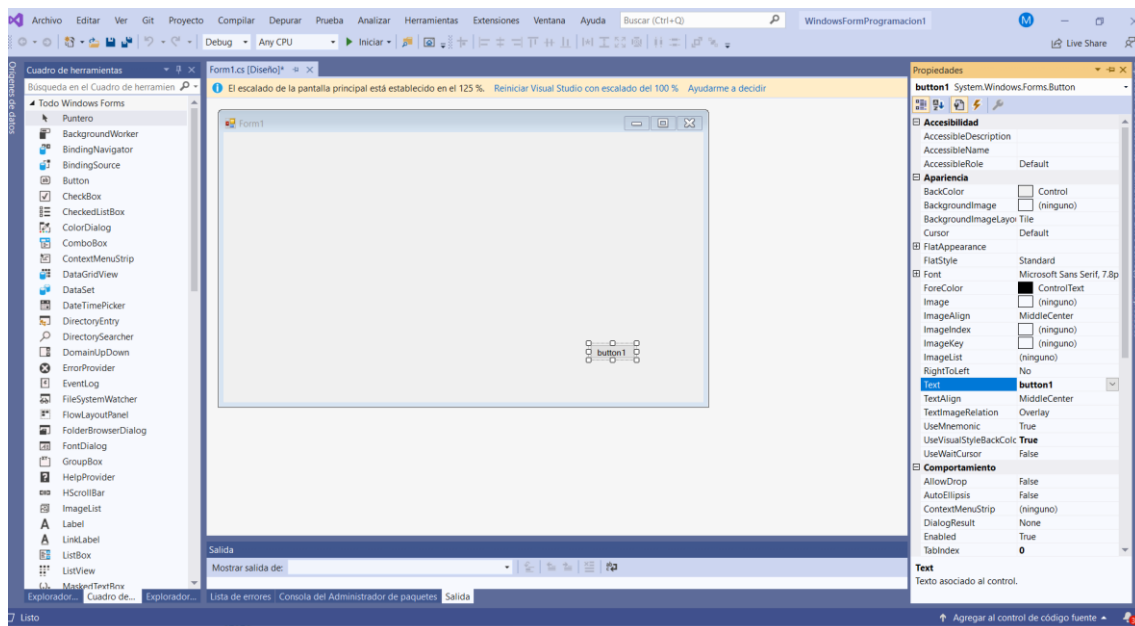
Framework

.NET Framework 4.7.2

Atrás Crear

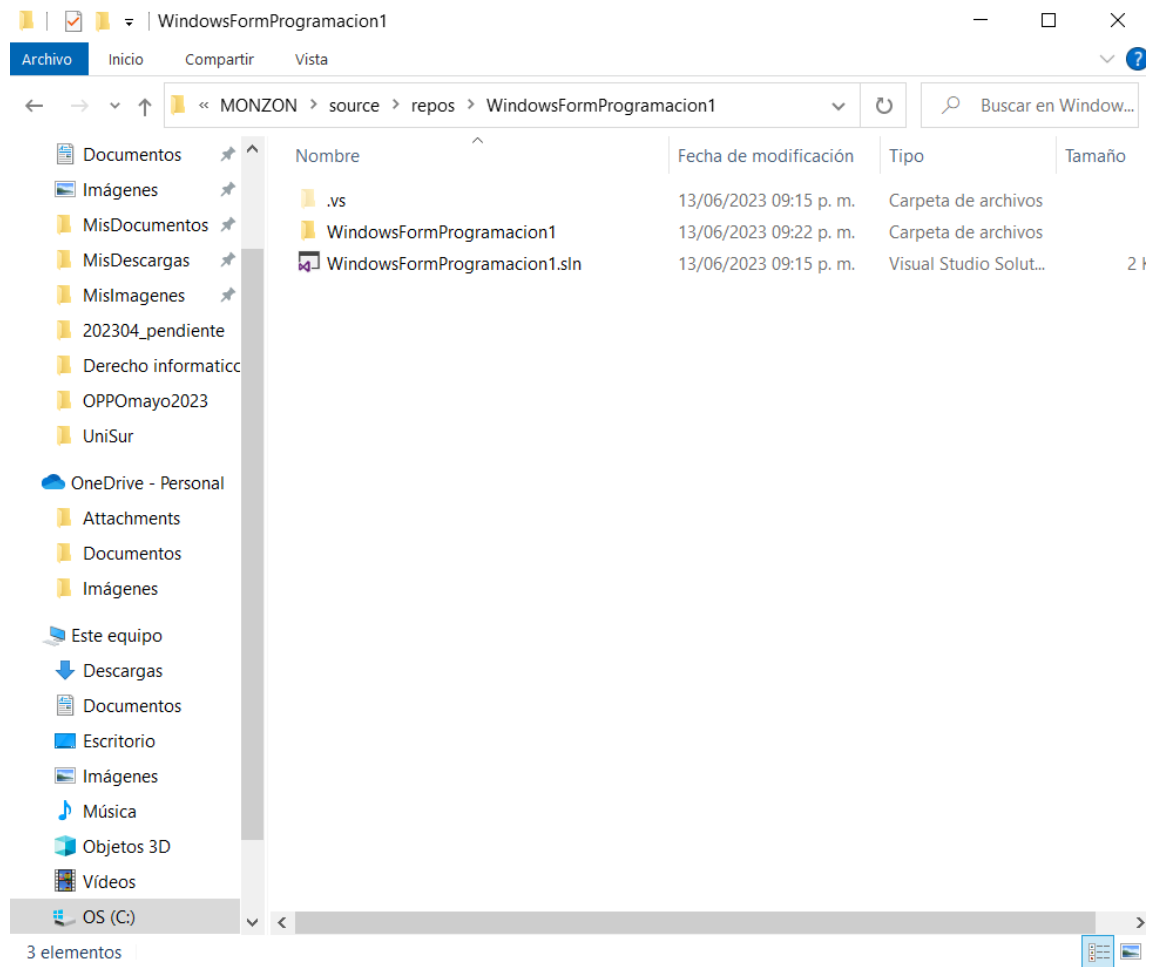
Una vez que mi proyecto esta creado le agrego algo un botón, por ejemplo





Pero ahora lo subiré a GitHub

Para empezar, debo saber como se llama mi proyecto




Para que en Github crear un repositorio idéntico en nombre


Mail - Monzon Mejia, Eduardo D... x | (18) 7 Personajes Desperdiciados


→ ↻ 🔒 https://github.com


Gmail YouTube Maps jsonschema2pojo Home - Cur





Top Repositories New


 Diaser/UrielAdonay


 Diaser/Primos

 Diaser/futbolStarter

 Diaser/MoviesApp

 Diaser/MasterDesarrolloAndroidJavaYKotlin

 Diaser/TataMonzon


 Diaser/ENVACACIONES


Show more

Recent activity

For you B

Welcome
We're up
some pec
[Send feed](#)

 Trend

 **Anto**
Specify w
Python

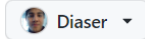
Por lo pronto solo le agrego nombre y descripción y le doy crear

Create a new repository

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

Required fields are marked with an asterisk (*).

Owner *



Repository name *

WindowsFormProgramaci

✓ WindowsFormProgramacion1 is available.

Great repository names are short and memorable. Need inspiration? How about [urban-system](#)?

Description (optional)

Programa que mostrará como hacer formularios

☒ 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.](#)

Description (optional)

Programa que mostrará como hacer formularios

☒ 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

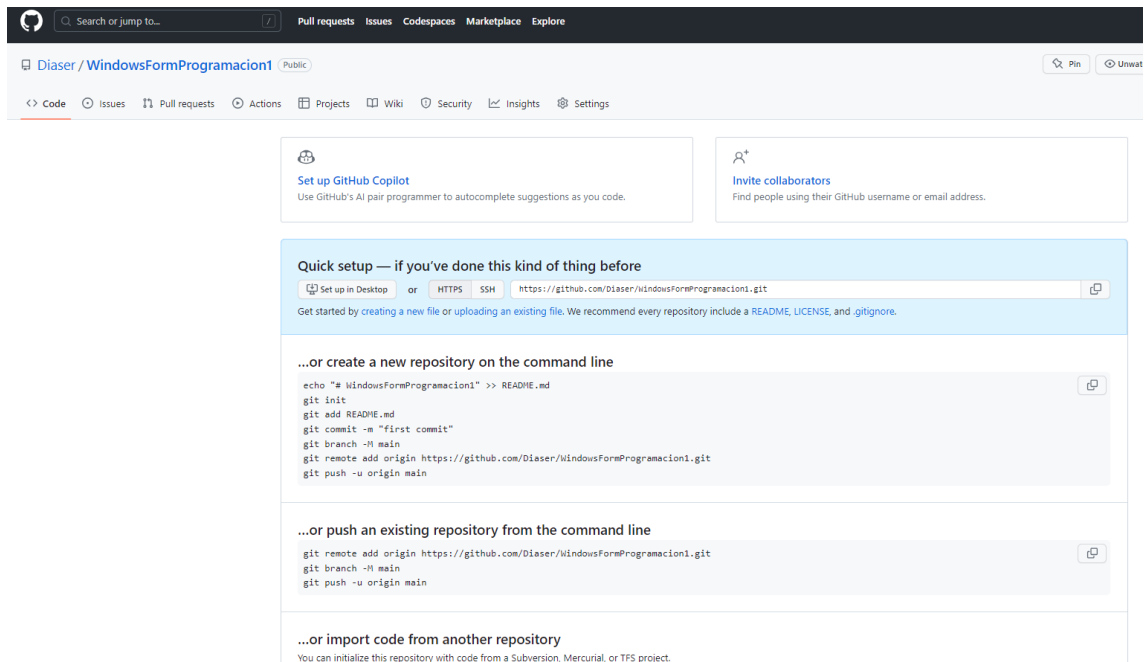
License: None

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 repository

Y listo el repositorio esta creado



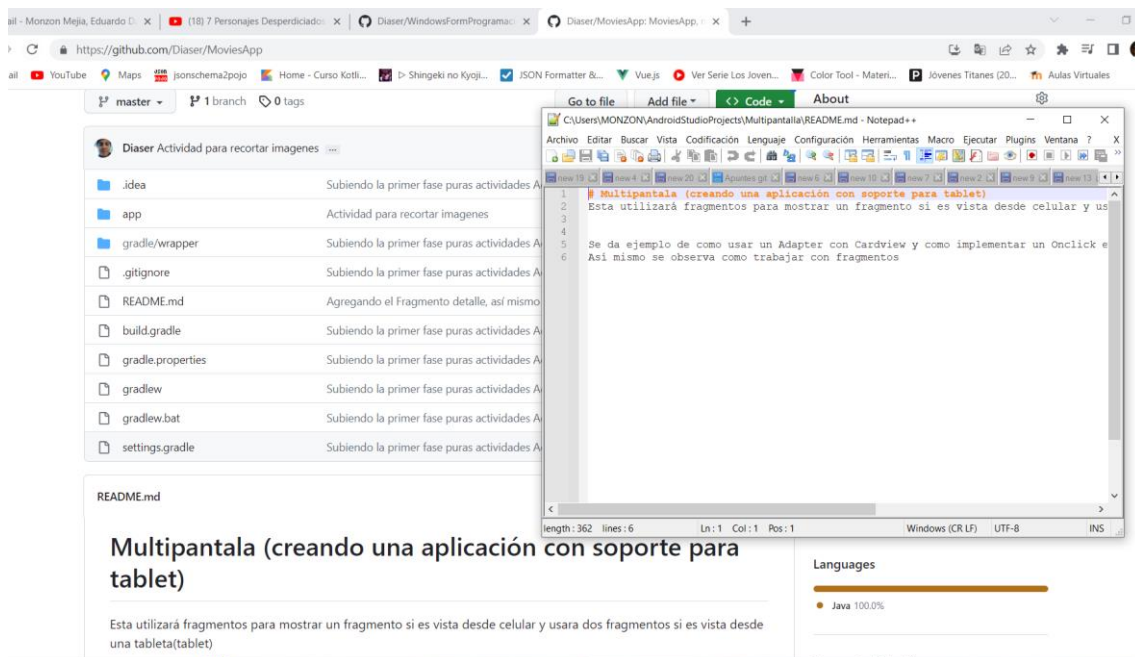
Listo ya esta listo, pero hace falta subirlo y aunque ahí explica como, que si lo pueden hacer así, en mi caso no lo subiré así, explicare porque..

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

```
echo "# WindowsFormProgramacion1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Diaser/WindowsFormProgramacion1.git
git push -u origin main
```

git add README.md

Esta línea solo agregará el archivo Readme y para empezar no lo tienen, lo pueden hacer, pues ayuda a que su repositorio tenga presentación por ejemplo miren mi archivo readme de otro proyecto y cómo se presenta..



Pero yo no he explicado cómo hacerlo y utilizando otro comando como git add . me añade todo incluido el archivo Readme

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

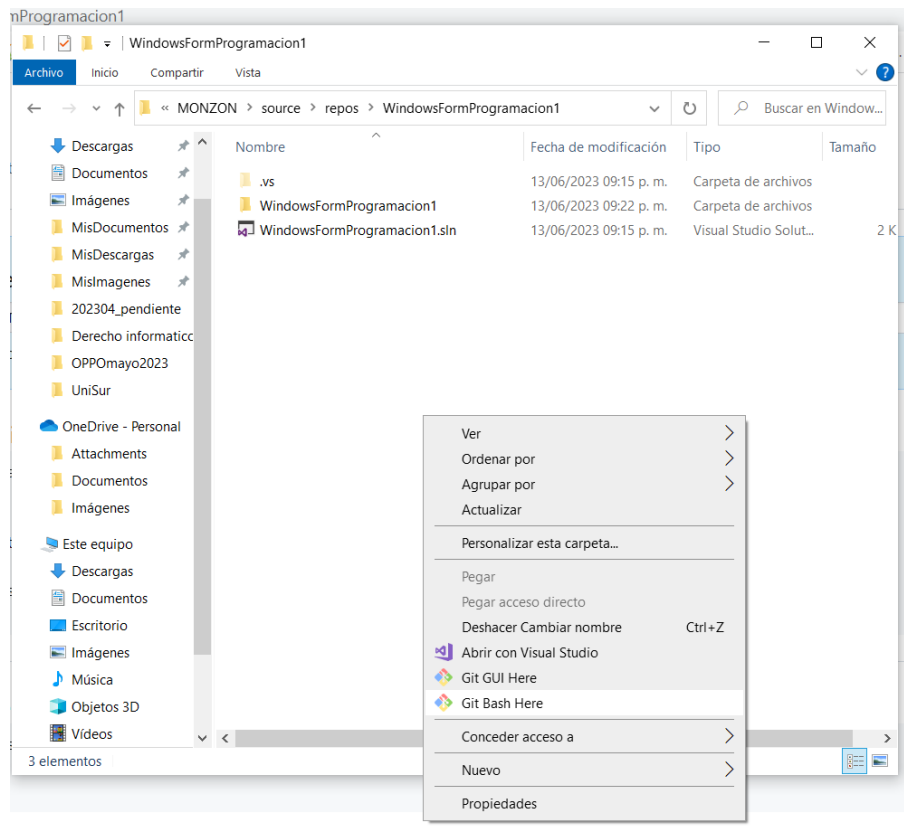
```
echo "# WindowsFormProgramacion1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Diaser/WindowsFormProgramacion1.git
git push -u origin main
```

git Branch -M main es un comando que creará una rama llamada main y aún no les he enseñado a usar ramas, entonces no lo veo necesario aún

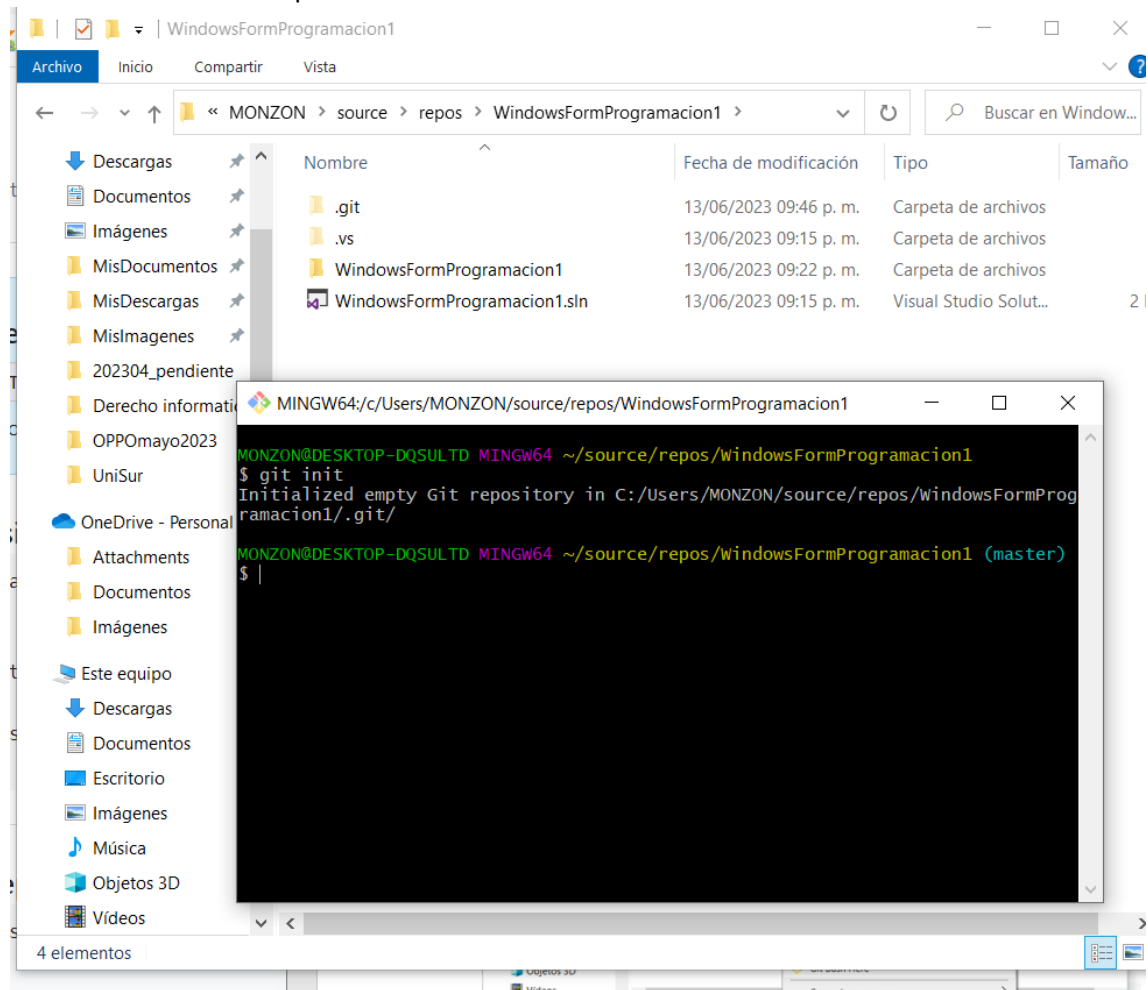
Por lo mismo el push no será enviado a la rama main pues esta no existirá en este ejemplo

A partir de ahora si empezaremos a subir el código

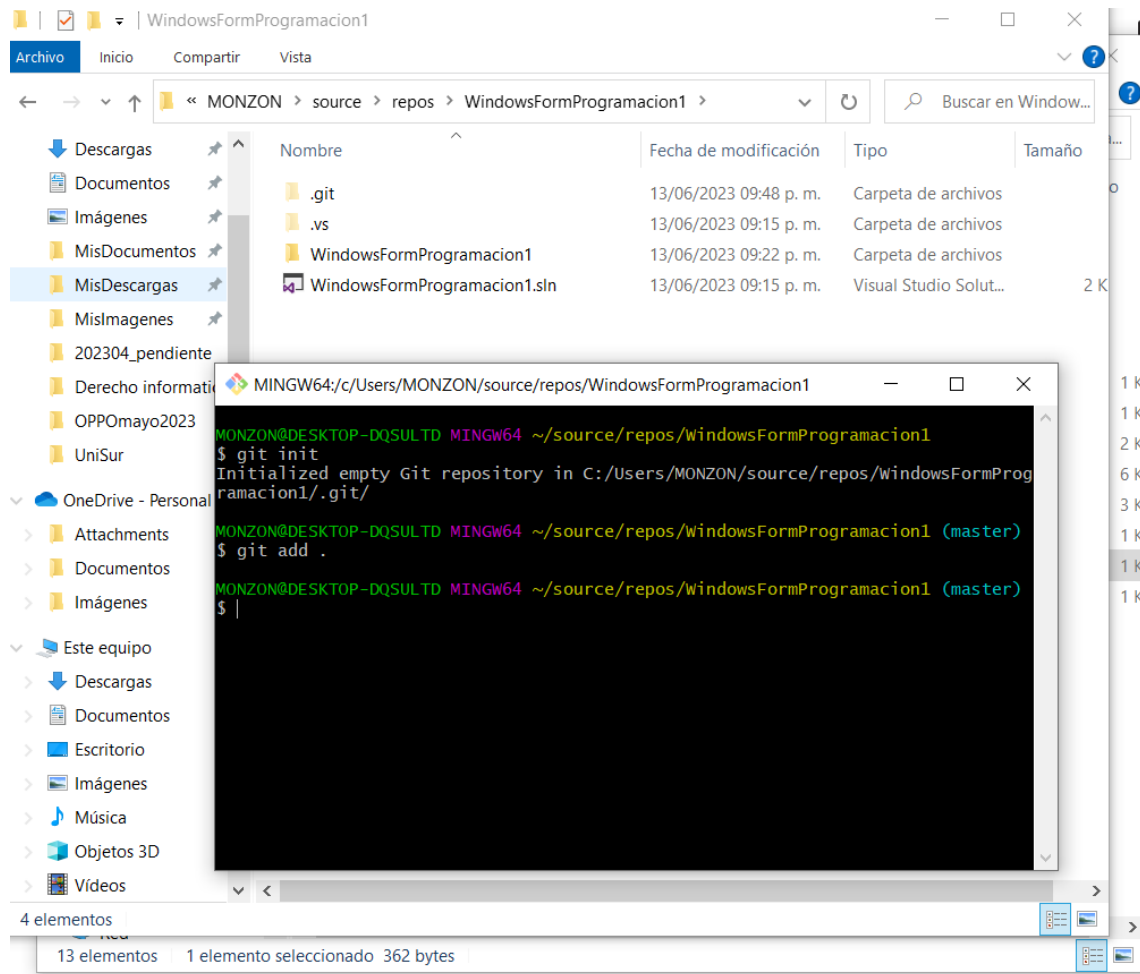
Nos dirigiremos a donde se encuentra nuestro proyecto y abriremos una consola gitbash



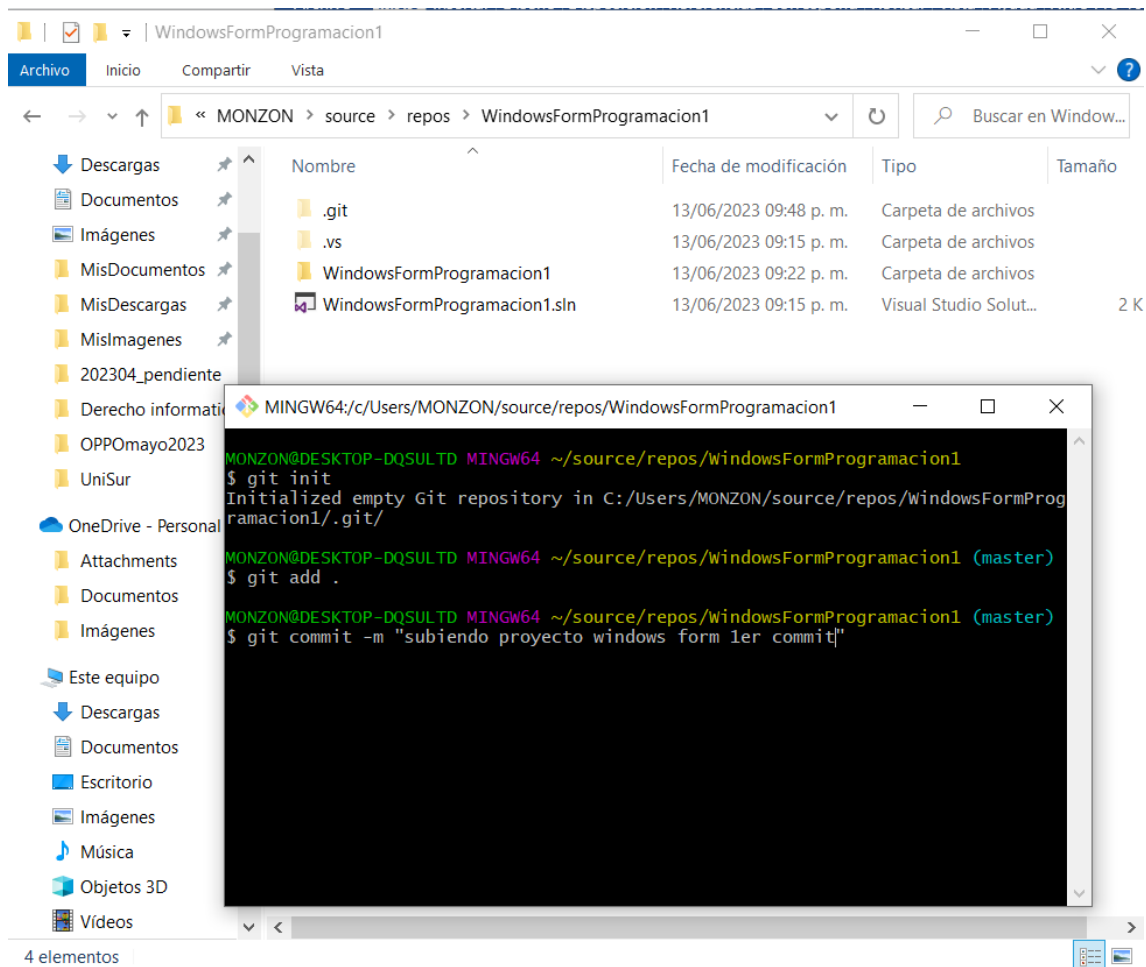
En la consola escribimos nuestro git init, el cual nos dirá que se a inicializado un repositorio vacío en esta misma carpeta.



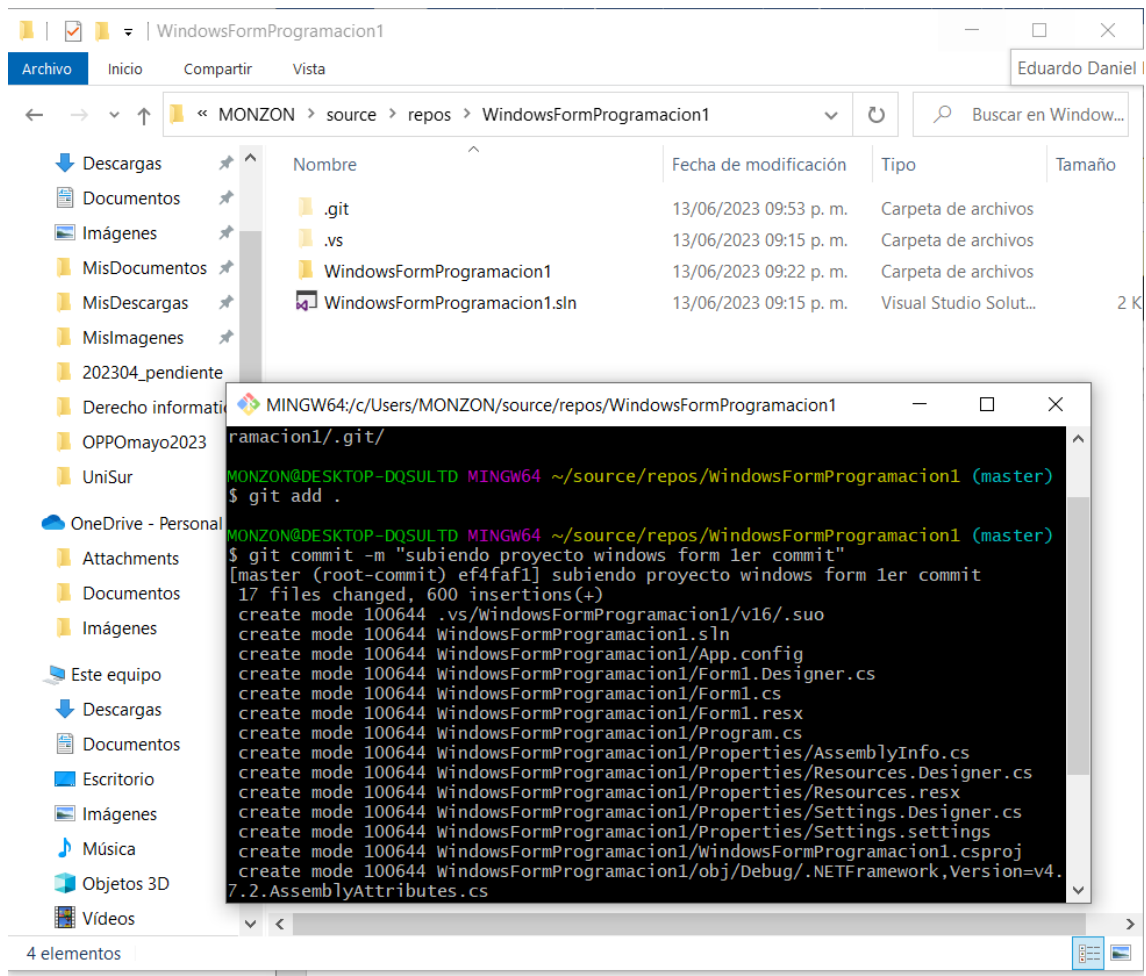
Después añadiremos todos los archivos a un staged con git add . recordemos que hay un espacio entre el add y el punto, esto hace que se agreguen todos los archivos al staged(Sala de espera para ser subido a un repositorio)



Después haremos nuestro commit que es hacer los cambios en nuestro repositorio local, los commit's nos pedirán un mensaje lo añadimos con -m



Una vez hecho esto nos dirá todos los archivos creados en el repositorio local



Ahora haremos match de este repositorio local con el repositorio que creamos en GitHub, pero recordemos que aquí usaremos el que nos da la página

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

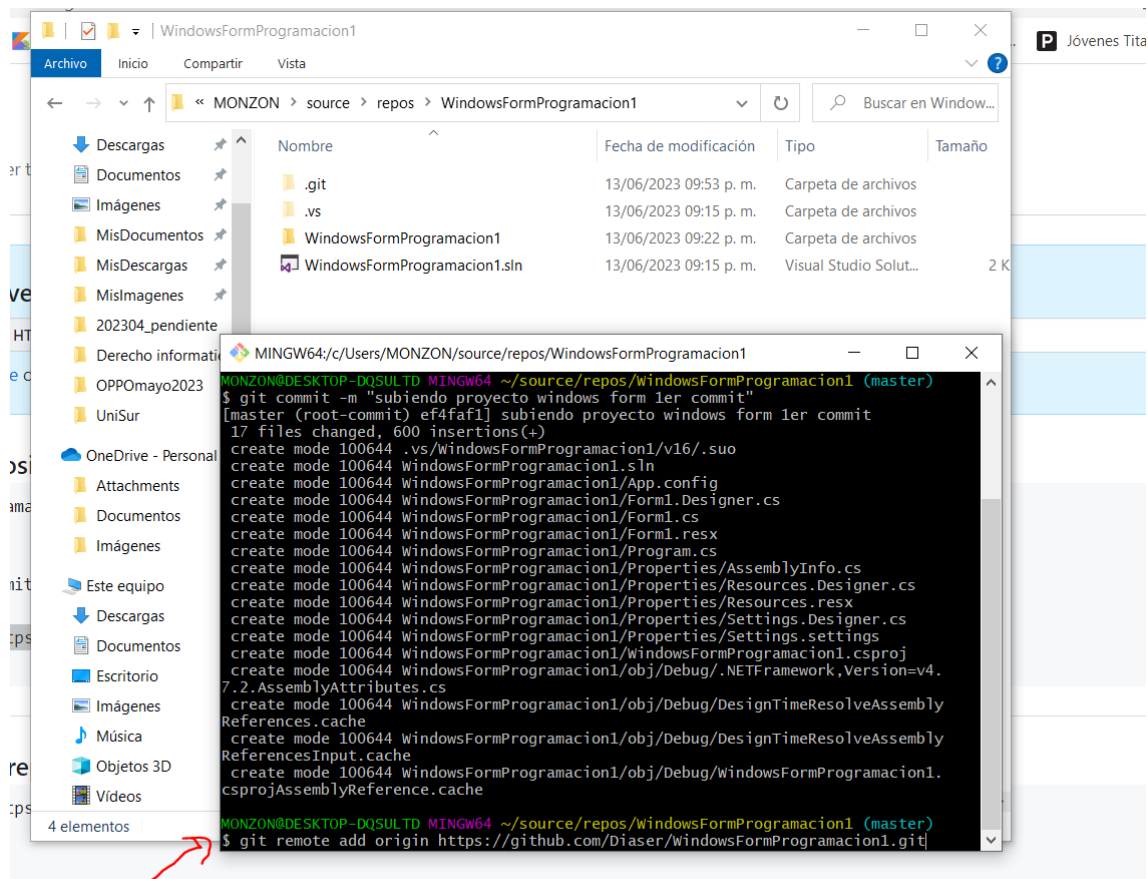
[Set up in Desktop](#) or [HTTPS](#) [SSH](#) <https://github.com/Diaser/WindowsFormProgramacion1.git>

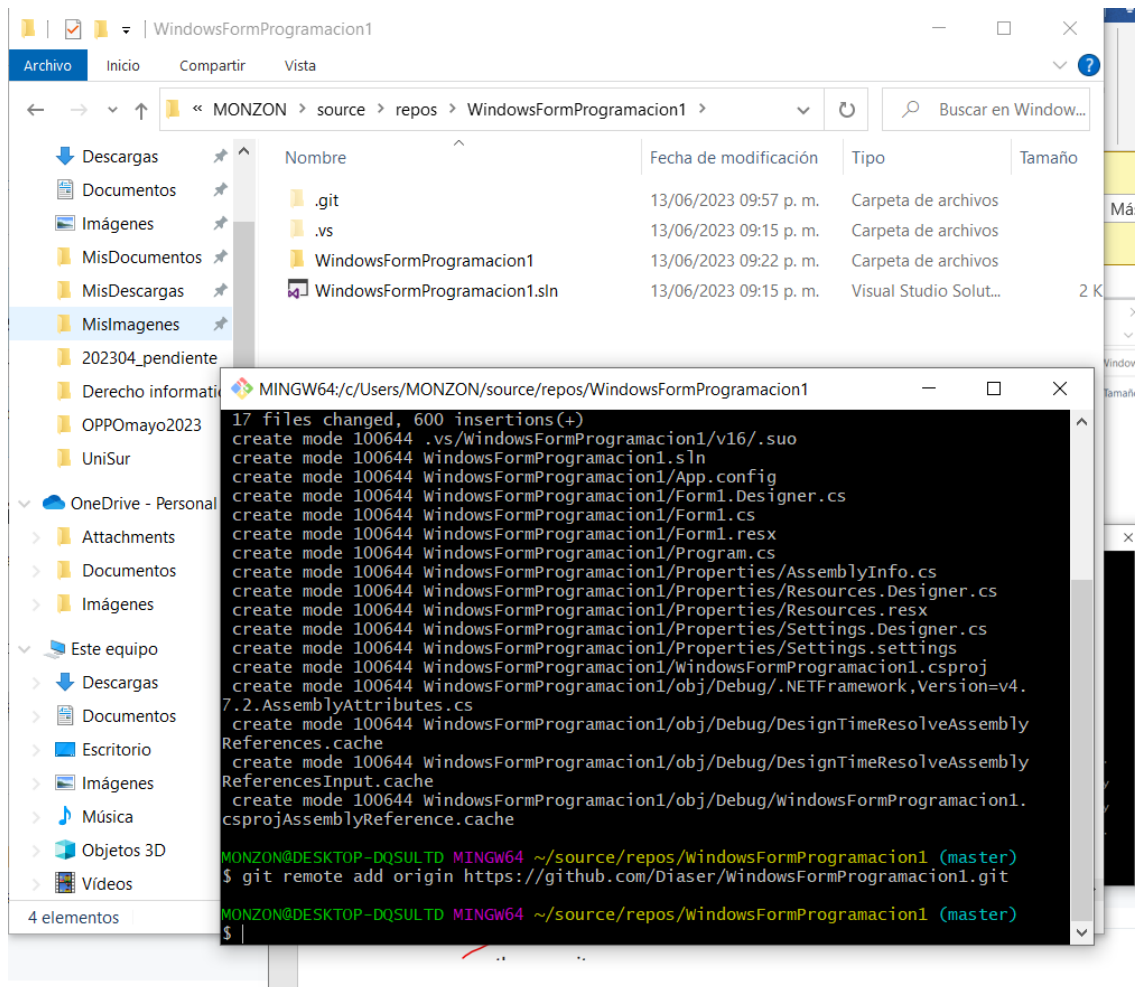
Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

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

```
echo "# WindowsFormProgramacion1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Diaser/WindowsFormProgramacion1.git
git push -u origin main
```

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



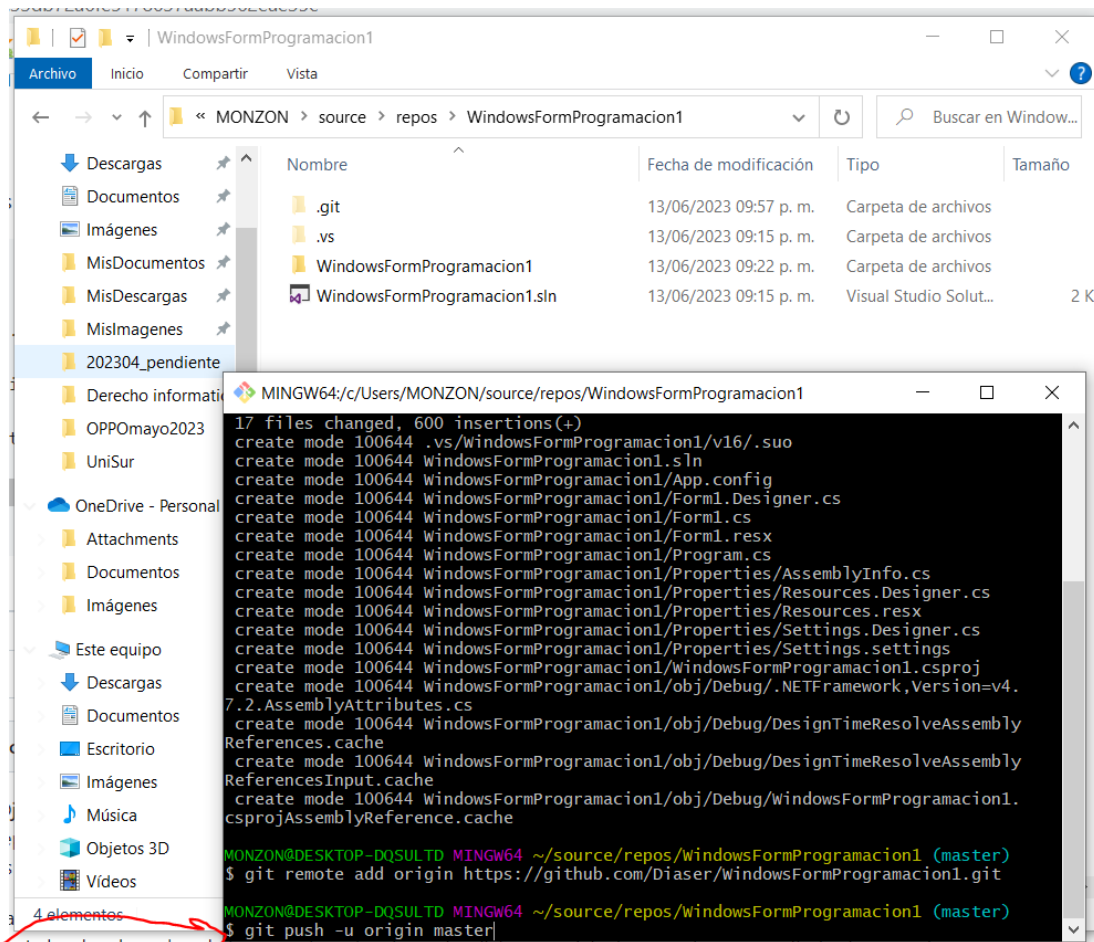


Al final haremos un push para empujar todo nuestro repositorio local que ya no está vacío hacia el repositorio con el que hicimos match..

Usare el siguiente comando, aguas no usare el Main porque nunca hice una rama Main

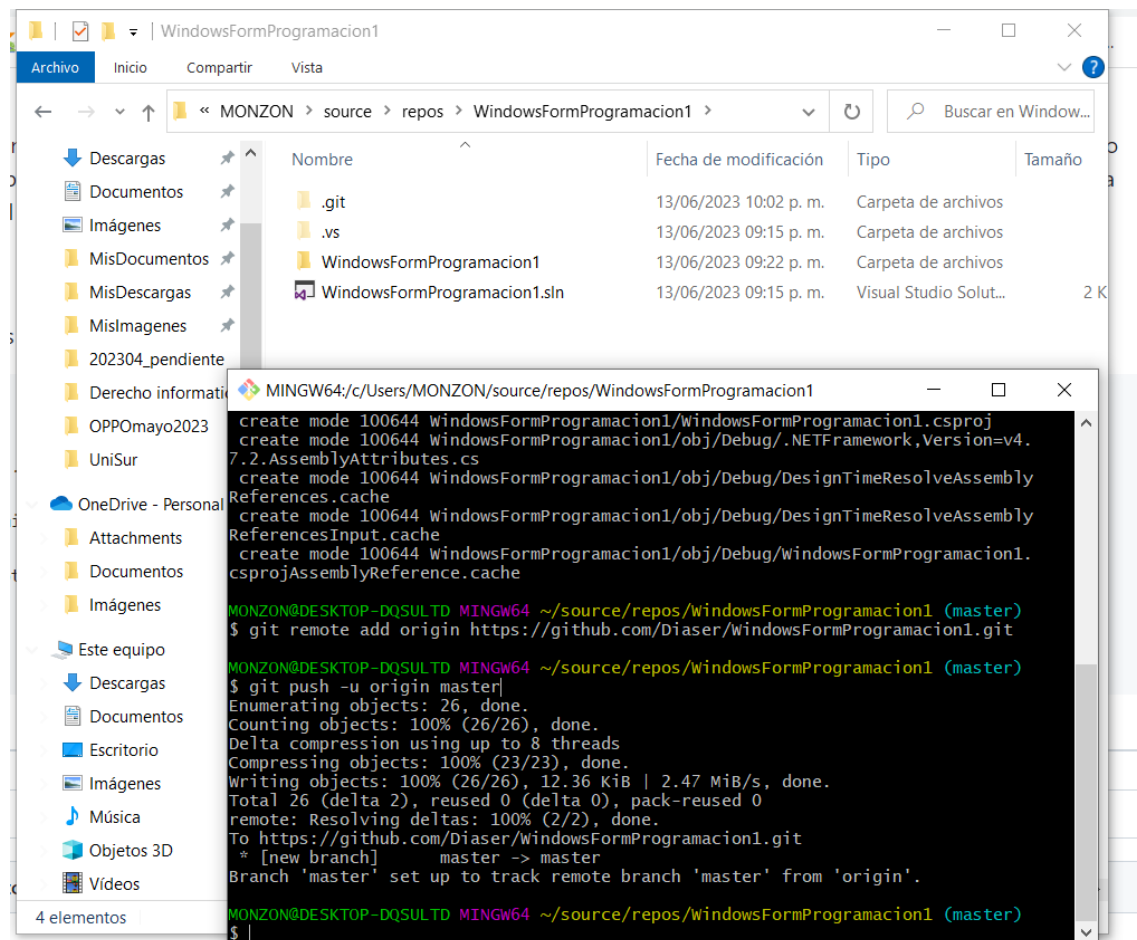
```
git push -u origin master
```

La master es la rama principal, entonces como nunca hice otra rama solo la envío a la principal



orio local o el nombre de tu proyecto, así no te confundirás más adelante o que tengas resultados inesperados.

Entonces me enumera los archivos que se subieron



johnduarte48 Principalmente recomiendo que cuando quieras cargar/subir un repositorio local a tu repositorio remoto no añadas el archivo README.md, el archivo README.md no es obligatorio y lo puedes incluir posteriormente, por que si lo incluyes desde el inicio

Y e realizado mi primer commit con os siguientes comandos

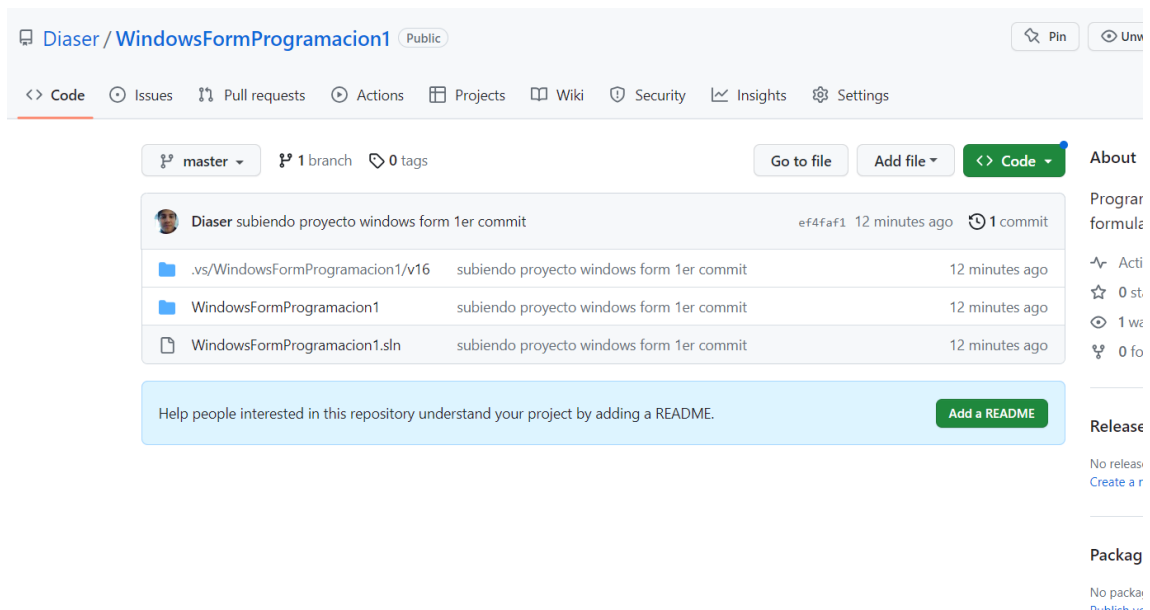
```
git init

git add .

git commit -m "first commit"

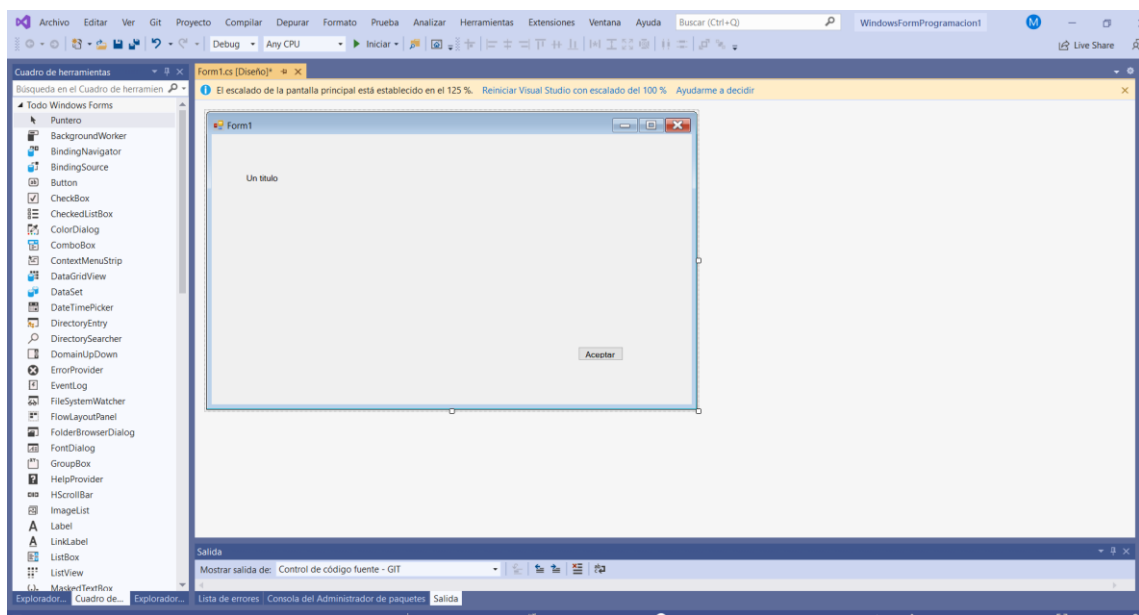
git remote add origin https://github.com/NOMBRE_USUARIO/NOMBRE_PROYECTO.git

git push -u origin master
```



Realizando cambios al proyecto

Añadiré un pequeño label y le daré en guardar



Limpiare consola con clear

```
MINGW64/c/Users/MONZON/source/repos/WindowsFormProgramacion1
create mode 100644 WindowsFormProgramacion1/WindowsFormProgramacion1.csproj
create mode 100644 WindowsFormProgramacion1/obj/Debug/.NETFramework,Version=v4.7.2.AssemblyAttributes.cs
create mode 100644 WindowsFormProgramacion1/obj/Debug/DesignTimeResolveAssemblyReferences.cache
create mode 100644 WindowsFormProgramacion1/obj/Debug/DesignTimeResolveAssemblyReferencesInput.cache
create mode 100644 WindowsFormProgramacion1/obj/Debug/WindowsFormProgramacion1.csproj.AssemblyReference.cache
MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git remote add origin https://github.com/Diaser/WindowsFormProgramacion1.git
MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git push -u origin master
Enumerating objects: 26, done.
Counting objects: 100% (26/26), done.
Delta compression using up to 8 threads
Compressing objects: 100% (23/23), done.
Writing objects: 100% (26/26), 12.36 KiB | 2.47 MiB/s, done.
Total 26 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/Diaser/WindowsFormProgramacion1.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ clear
```

Y después preguntare el estatus del proyecto con git status

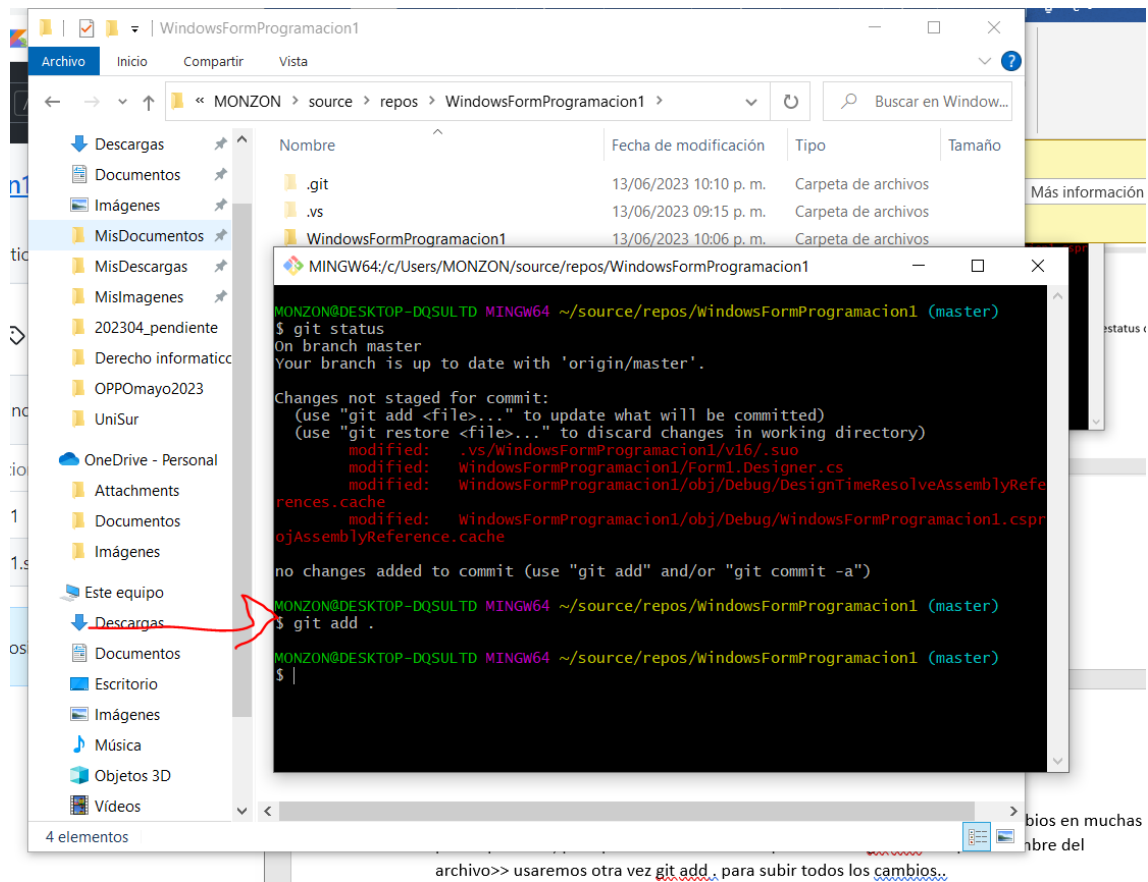
```
master 1 branch 0 tags
Diaser
.vs/WindowsFormProgramacion1
WindowsFormProgramacion1/Form1.Designer.cs
WindowsFormProgramacion1/obj/Debug/DesignTimeResolveAssemblyReferences.cache
WindowsFormProgramacion1/obj/Debug/WindowsFormProgramacion1.csproj.AssemblyReference.cache
Help people

MINGW64/c/Users/MONZON/source/repos/WindowsFormProgramacion1
MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

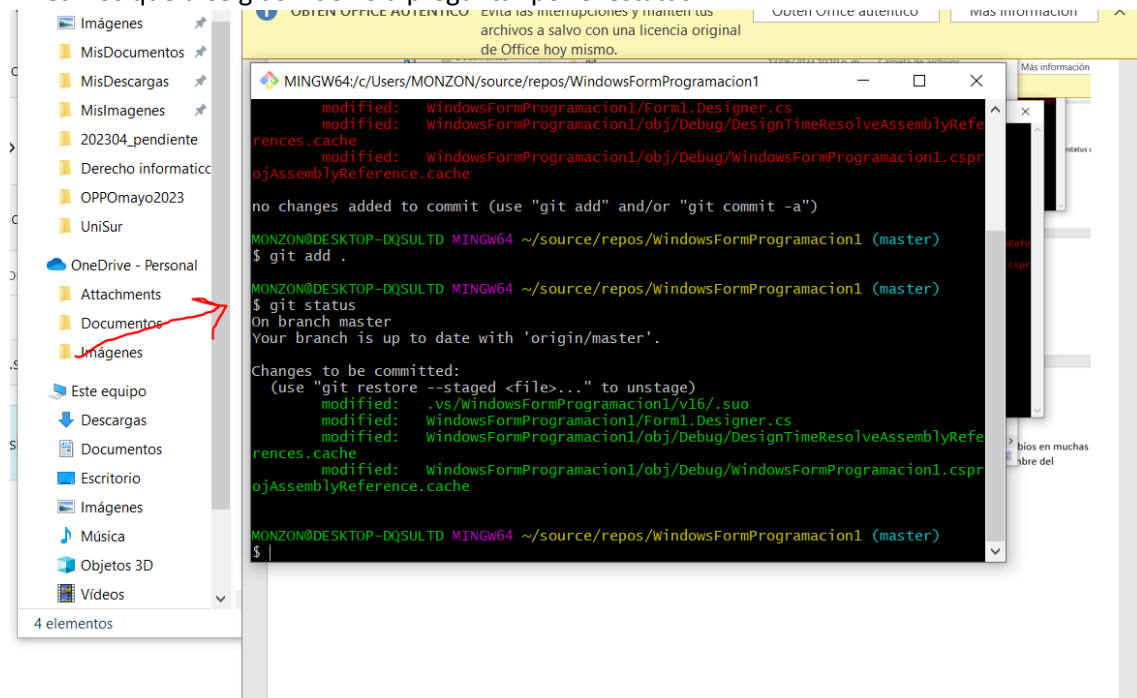
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   .vs/WindowsFormProgramacion1/v16/.suo
        modified:   WindowsFormProgramacion1/Form1.Designer.cs
        modified:   WindowsFormProgramacion1/obj/Debug/DesignTimeResolveAssemblyReferences.cache
        modified:   WindowsFormProgramacion1/obj/Debug/WindowsFormProgramacion1.csproj.AssemblyReference.cache

no changes added to commit (use "git add" and/or "git commit -a")
MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ |
```

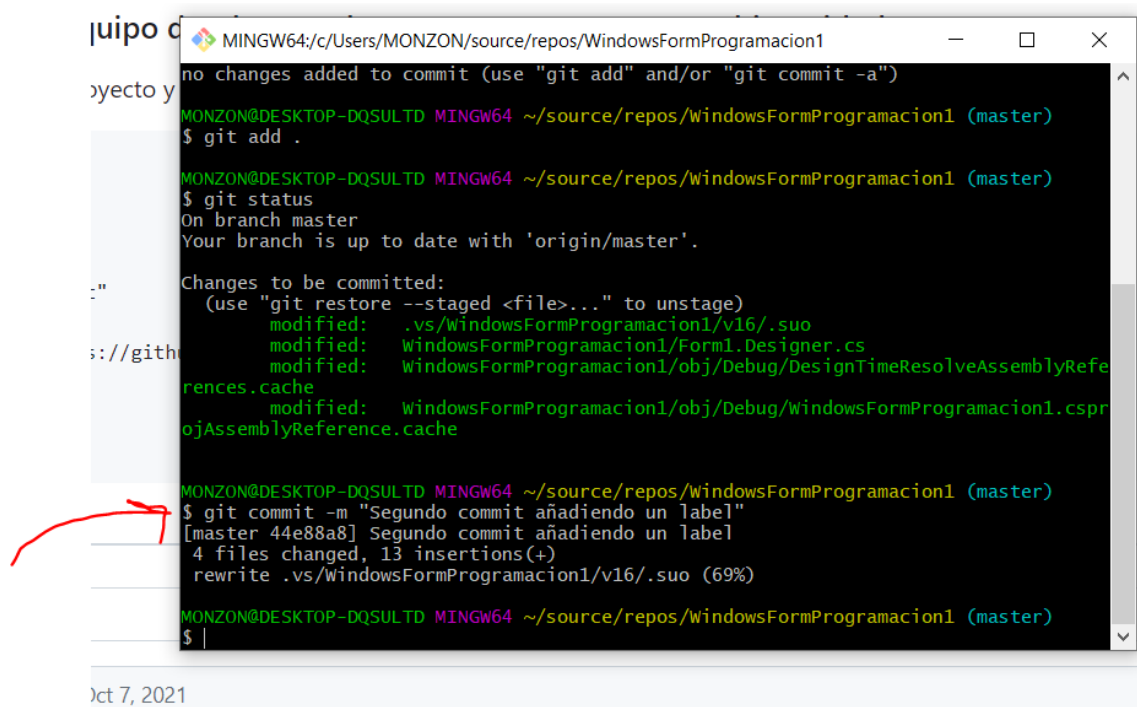
Me dice que hay 3 archivos modificados, uff es que los formularios hacen cambios en muchas partes por todo, pero para no añadirlos uno por uno con git add <<aquí el nombre del archivo>> usaremos otra vez git add . para subir todos los cambios..



Y veamos que dice git si vuelvo a preguntar por el estatus



Ahora me dice que en el staged, los archivos en rojo ahora están en verde listos para que le hagamos commit, entonces démosle commit y agreguemos el mensaje



```
MINGW64/c/Users/MONZON/source/repos/WindowsFormProgramacion1
no changes added to commit (use "git add" and/or "git commit -a")

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git add .

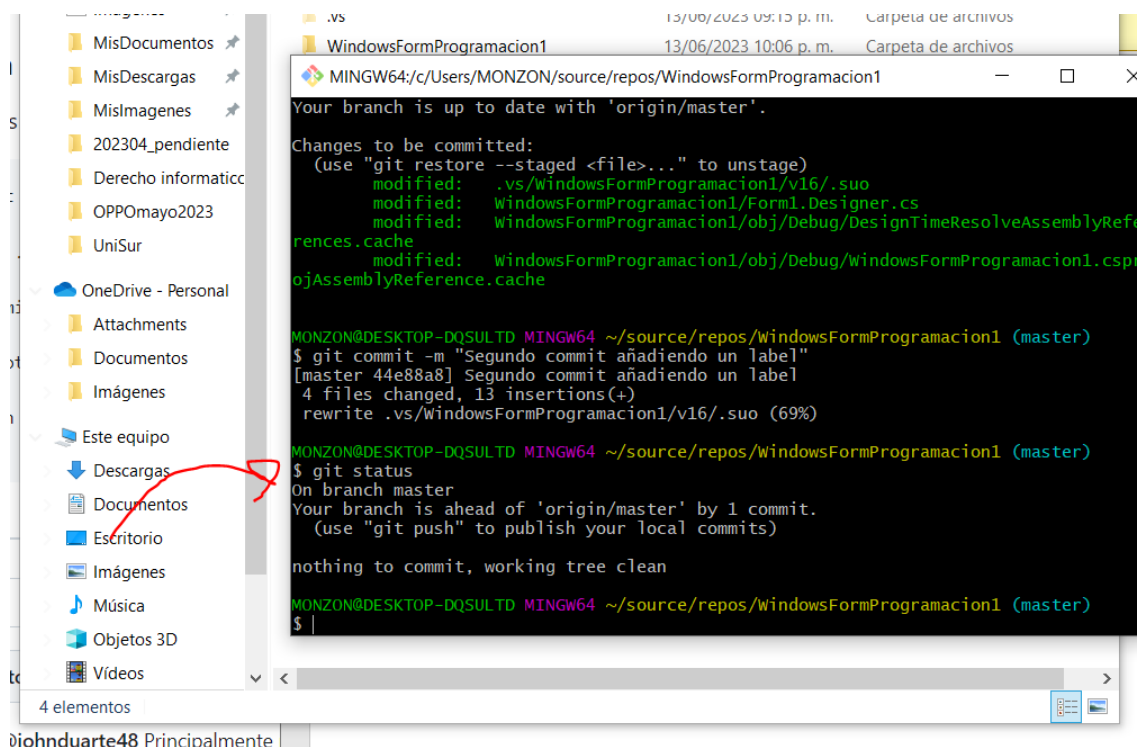
MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   .vs/WindowsFormProgramacion1/v16/.suo
        modified:   WindowsFormProgramacion1/Form1.Designer.cs
        modified:   WindowsFormProgramacion1/obj/Debug/DesignTimeResolveAssemblyReferences.cache
        modified:   WindowsFormProgramacion1/obj/Debug/WindowsFormProgramacion1.csproj
        modified:   WindowsFormProgramacion1/obj/Debug/WindowsFormProgramacion1.csproj.AssemblyReference.cache

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git commit -m "Segundo commit añadiendo un label"
[master 44e88a8] Segundo commit añadiendo un label
 4 files changed, 13 insertions(+)
 rewrite .vs/WindowsFormProgramacion1/v16/.suo (69%)

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$
```

Veamos de nuevo el status



```
MINGW64/c/Users/MONZON/source/repos/WindowsFormProgramacion1
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   .vs/WindowsFormProgramacion1/v16/.suo
        modified:   WindowsFormProgramacion1/Form1.Designer.cs
        modified:   WindowsFormProgramacion1/obj/Debug/DesignTimeResolveAssemblyReferences.cache
        modified:   WindowsFormProgramacion1/obj/Debug/WindowsFormProgramacion1.csproj
        modified:   WindowsFormProgramacion1/obj/Debug/WindowsFormProgramacion1.csproj.AssemblyReference.cache

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git commit -m "Segundo commit añadiendo un label"
[master 44e88a8] Segundo commit añadiendo un label
 4 files changed, 13 insertions(+)
 rewrite .vs/WindowsFormProgramacion1/v16/.suo (69%)

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$
```

Nos dice que la rama master tiene commit y que puedo publicar los cambios con un push, pues hagámolo

```
MINGW64: c:/Users/MONZON/source/repos/WindowsFormProgramacion1
$ git commit -m "Segundo commit añadiendo un label"
[master 44e88a8] Segundo commit añadiendo un label
4 files changed, 13 insertions(+)
rewrite .vs/WindowsFormProgramacion1/v16/.suo (69%)

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

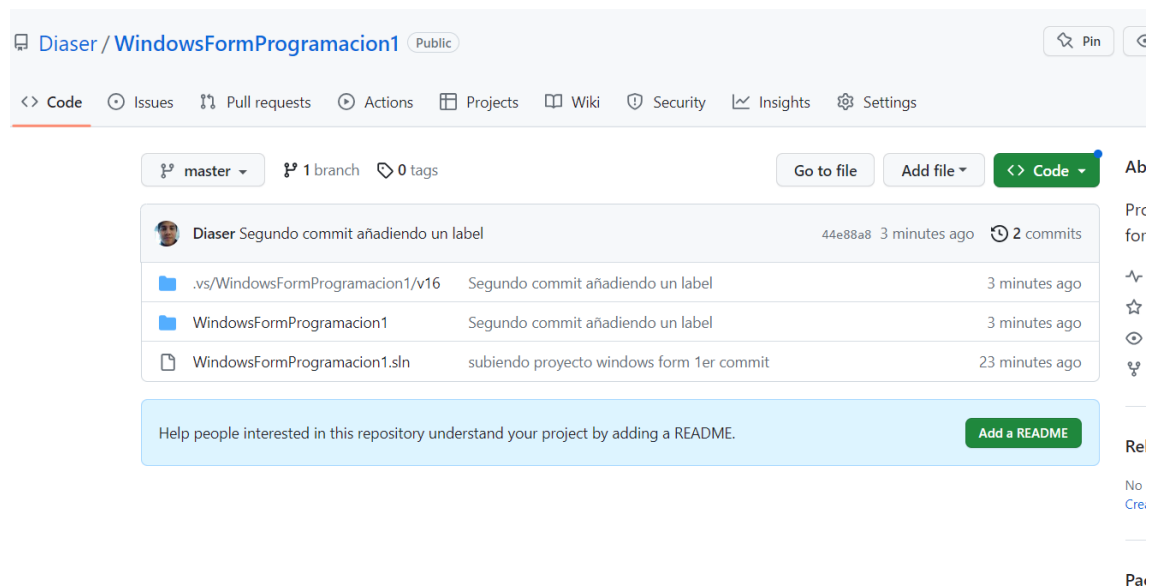
nothing to commit, working tree clean

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$ git push -u origin master
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 8 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (12/12), 6.17 KiB | 2.06 MiB/s, done.
Total 12 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/Diaser/WindowsFormProgramacion1.git
ef4faf1..44e88a8 master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

MONZON@DESKTOP-DQSULTD MINGW64 ~/source/repos/WindowsFormProgramacion1 (master)
$
```

mente recomiendo que cuando quieras cargar/subir un repositorio local a tu repositorio remoto no añada

Y Listo



Después arreglo este archivo para su mayor entendimiento... Saludos!