

□ 1.1 – Crea un directorio llamado repo01 en local (desde tu máquina) e ejecuta el comando pertinente para que dicho directorio para que se transforme el repositorio en local ¿Cómo podemos identificar que el repositorio se ha inicializado?

Para empezar he ejecutado el comando pwd para saber el directorio en el que estoy:

```
MINGW64:/c/Users/borja

borja@borja-PC MINGW64 ~
$ pwd
/c/Users/borja

borja@borja-PC MINGW64 ~
$
```

Después con los comando cd y ls he dirigido al directorio de la siguiente forma:

```
MINGW64:/c/Users/borja
borja@borja-PC MINGW64 ~
$ cd ..
borja@borja-PC MINGW64 /c/Users
$ ls
All Users* Default* 'Default User's* Public* borja* desktop.ini postgres/
$ cd ..
borja@borja-PC MINGW64 /c
$ ls
[Security.Bin*] *cmdr* BOOTSECT.BAK 'Documents and Settings*' Intel* Microsoft* Perflgo* 'Program Files (x86)*' Recovery* 'System Volume Information*' Users* bin_projects* found.00* hiberfil.sys pagefile.sys tmp/ usr/
SECURITY~1* 'Archivos de programa*' boot/ 'Norton Tablet*' MSDCache* 'PentablistDriver*' 'Program Files*' ProgramData* 'System Volume Information*'
borja@borja-PC MINGW64 /c
$ cd users
borja@borja-PC MINGW64 /c/users
$ ls
All Users* Default* 'Default User's* Public* borja* desktop.ini postgres/
borja@borja-PC MINGW64 /c/users
$ cd borja
borja@borja-PC MINGW64 /c/users/borja
$ ls
Audata* IntelGraphicsProfiles/ Playlistas          stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
Cisco Packet Tracer 7.2.2*/ Links* Resonance          stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
'Configuración local*' 'Menu Inicio*' 'Saved Games*'      stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
Contexts*           'Win documents*'   'Search*'           stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
Cookies*            Music*             stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
'Creative Cloud Files*' NTUSER.DAT[018848bc-60cf-11e6-804b-408d5c818f07].tbl.0.regtrans-ms stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
'Archivos de programa*' NTUSER.DAT[018848bc-60cf-11e6-804b-408d5c818f07].tbl.1.regtrans-ms stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
Desktop*            NTUSER.DAT[018848bc-60cf-11e6-804b-408d5c818f07].tbl.1.regtrans-ms stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
Documents*          NTUSER.DAT[018848bc-60cf-11e6-804b-408d5c818f07].tbl.bif stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
Downloads*          NTUSER.DAT[018848bc-60cf-11e6-804b-408d5c818f07].tbl.bif stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
'Historia de red*'   NTUSER.DAT[018848bc-60cf-11e6-804b-408d5c818f07].tbl.bif stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
Internet*            NTUSER.DAT[018848bc-60cf-11e6-804b-408d5c818f07].tbl.bif stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
'Imágenes*'         Pictures*          'InternetExpress*'  stuser.dat[2d7c8809-51fa-11e6-804b-408d5c818f07].tm.bif
$ cd music
borja@borja-PC MINGW64 /c/users/borja/music
$ ls
Thumbs.db desktop.ini
borja@borja-PC MINGW64 /c/users/borja/music
$
```

Para crear el repositorio he ejecutado el comando mkdir con el nombre del repositorio:

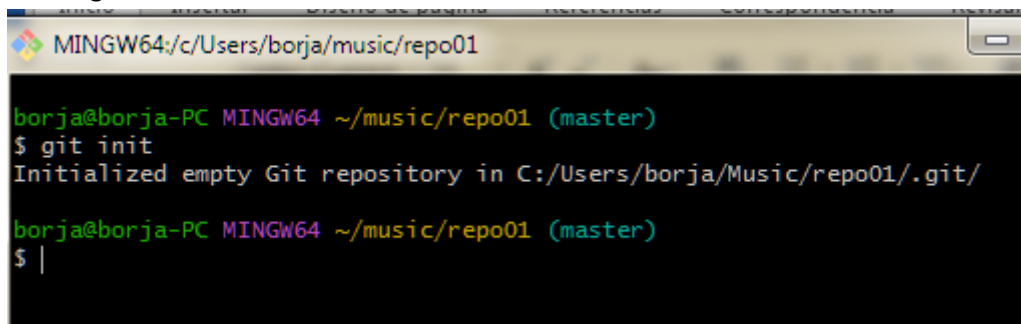
```
MINGW64:/c/users/borja/music

borja@borja-PC MINGW64 /c/users/borja/music
$ mkdir repo01

borja@borja-PC MINGW64 /c/users/borja/music
$ ls
Thumbs.db desktop.ini repo01/

borja@borja-PC MINGW64 /c/users/borja/music
$
```

Para poder inicializar el repositorio se entra en el repositorio y se ejecuta el comando git init de la siguiente forma:

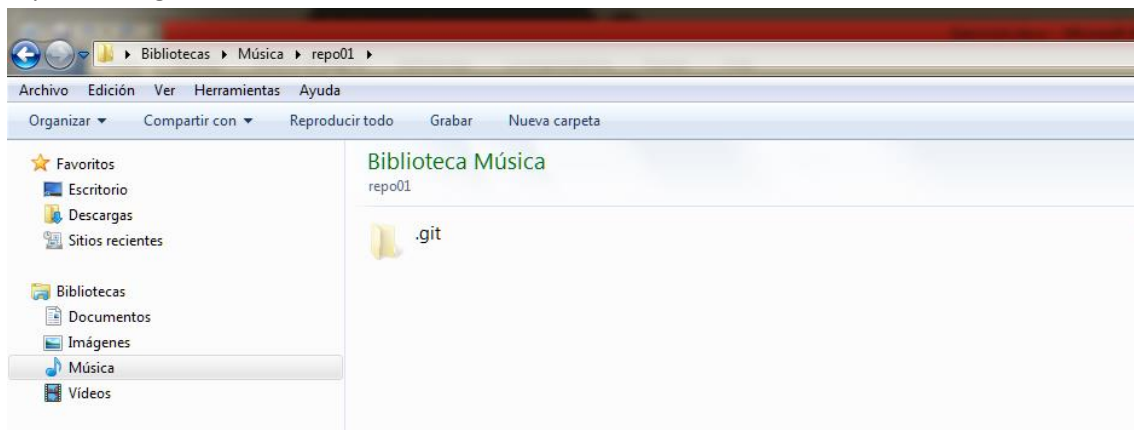


```
MINGW64:/c/Users/borja/music/repo01

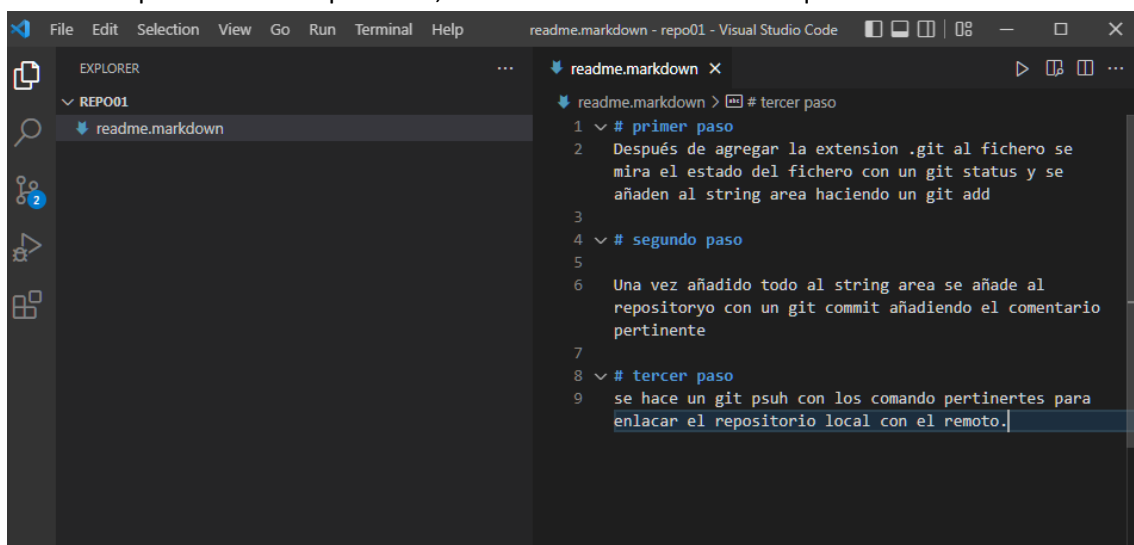
borja@borja-PC MINGW64 ~/music/repo01 (master)
$ git init
Initialized empty Git repository in C:/Users/borja/Music/repo01/.git/

borja@borja-PC MINGW64 ~/music/repo01 (master)
$ |
```

Se sabe que esta inicializado porque se marca con la etiqueta master y a parte se a creado un repositorio .git:



□ 1.2 – Añade un documento llamado readme.md dentro del repositorio (recuerda que MD es la extensión de los ficheros Markdown) y documenta en su interior todos los pasos que vas realizando para crear un repositorio, etc. Puedes añadir fotos o lo que creas conveniente.



```
borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git status
On branch master

No commits yet

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

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

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git add readme.markdown
```

```
borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git status
On branch master

No commits yet

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

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$
```

El archivo a pasado del working directory al string área.

- 1.3 – Añade el fichero que acabamos de añadir al repositorio al staging area, visualiza el estado del repositorio (con git status) y haz un snapshot (commit) del fichero hacía nuestro repositorio local. ¿En que “file status lifecycle” se encuentra el fichero?

```

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git status
On branch master

No commits yet

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

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git commit -m "Se añade el fichero al repositorio"
[master (root-commit) 79f4bbd] Se añade el fichero al repositorio
1 file changed, 9 insertions(+)
create mode 100644 readme.markdown

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git status
On branch master
nothing to commit, working tree clean

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$

```

Se a añadido el readme.markdown al repository

□ 1.4 – Intenta subir los ficheros al repositorio remoto mediante al comando git push ¿Se te ocurre que está pasando? (si no lo sabes aún no te preocupes)

```

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git push
fatal: No configured push destination.
Either specify the URL from the command-line or configure a remote repository using

    git remote add <name> <url>

and then push using the remote name

    git push <name>

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$

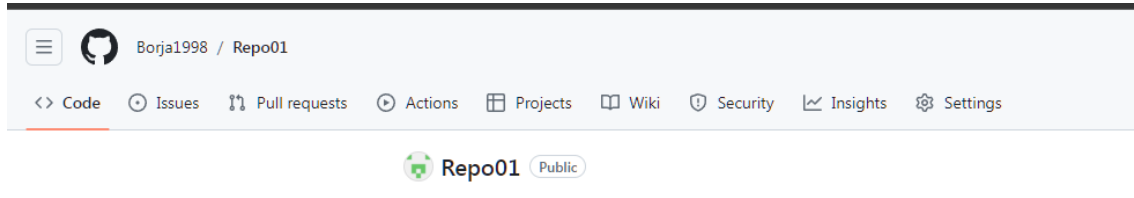
```

En este casi no se ha asociado el repositorio remoto al repositorio local.

- 1.5 – Ejecuta el comando `git remote -v` e investiga porque no nos aparece nada

En este caso n sale nada debido a que no hay ninguna conexión ni se a creado ninguna ni tampoco se ha eliminado ninguna rama

- 1.6 – Crea un repositorio remoto llamado repo01, asócialo a tu repositorio local



- 1.7 – Vuelve a ejecutar el comando `git remote -v` nuevamente y explica el porque ahora si que aparece.

En este caso aparece debido a que se ha creado una rama y se puede visualizar.

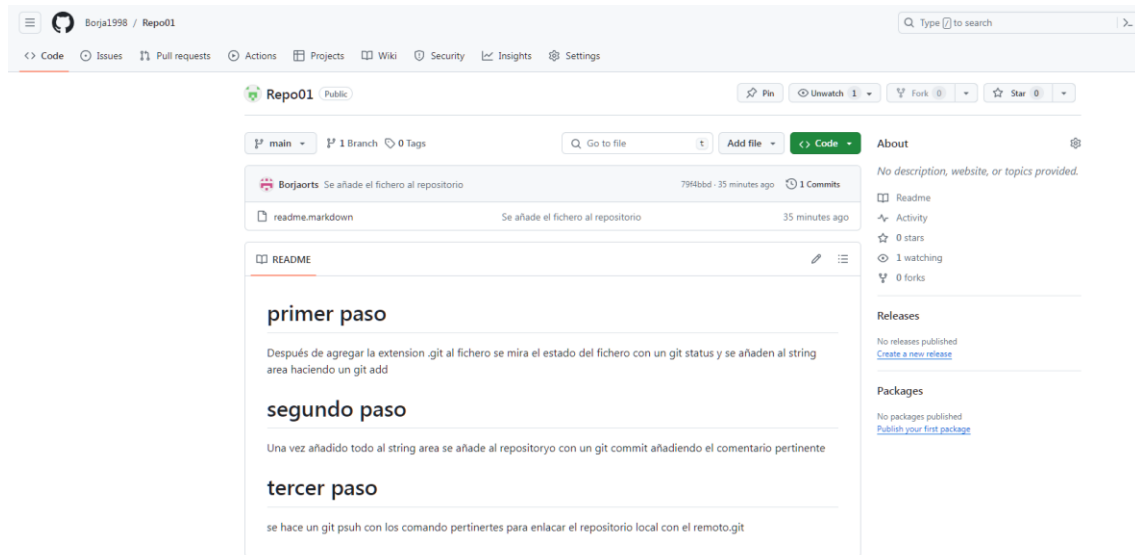
- 1.8 – Sube los cambios que hemos subido al snapshot local (`commit`) hacía al repositorio remoto.

```
borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git remote add origin https://github.com/Borja1998/Repo01.git

borja@borja-PC MINGW64 ~/Music/repo01 (master)
$ git branch -M main

borja@borja-PC MINGW64 ~/Music/repo01 (main)
$ git push -u origin main
info: please complete authentication in your browser...
fatal: Acceso denegado
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 478 bytes | 478.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Borja1998/Repo01.git
 * [new branch]      main -> main
error: could not lock config file .git/config: Permission denied
error: unable to write upstream branch configuration
hint:
```

□ 1.9 – Ves al repositorio remoto (en este caso GitHub) y comprueba que se haya realizado el commit correctamente y observa que pasa en el repositorio ¿Observas algo peculiar?



Si en este caso se puede ver el archivo markdown de forma directa.