

Actividad 1 – Comandos Linux

Comprueba la ruta en la que te encuentras

Comando pwd

```
roberto@roberto MINGW64 ~/Desktop
$ pwd
/c/Users/roberto/Desktop
```

Muévete por los distintos directorios de tu ordenador, comprobando que se encuentra en su interior

Utilizar el comando “cd” con sus distintas opciones para movernos y el comando “ls” para comprobar lo que hay en cada directorio. Por ej.

```
roberto@roberto MINGW64 ~
$ cd /D

roberto@roberto MINGW64 /D
$ ls
'$RECYCLE.BIN'/' 'Maquina virtual'/' Roberto/
Descargas/' 'Program Files'/' 'System Volume I
MSOCache/' Programas/' Windows10Upgrad

roberto@roberto MINGW64 /D
$ cd Roberto

roberto@roberto MINGW64 /D/Roberto
$ cd ..
```

Crear directorios

Utilizar el comando “mkdir [nombreDirectorio]”.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED
$ mkdir Biblioteca
```

Crear archivos

Podemos hacerlo tanto utilizando el comando “touch” o utilizando el comando “nano”, que sirve para editar archivos, y luego guardando el fichero.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ touch Prueba.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ nano Prueba1.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ ls
Prueba.txt Prueba1.txt
```

Mover o copiar archivos a otro directorio

Para copiar archivos utilizamos el comando “cp” y para moverlos el comando “mv”.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ cp Prueba.txt /D/Roberto/Clases/EED

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ ls
Prueba.txt  Prueba1.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ cd ..

roberto@roberto MINGW64 /D/Roberto/Clases/EED
$ ls
Biblioteca/  EntornosDesarrollo/  Prueba.txt  TemarioRobe
roberto@roberto MINGW64 /D/Roberto/Clases/EED
```

El comando “mv” nos puede servir también para cambiar el nombre de un archivo

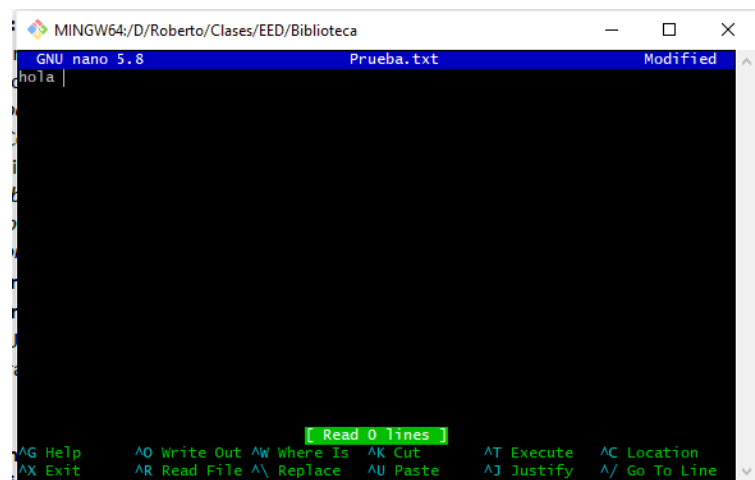
```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ mv Prueba1.txt PruebaMv.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ ls
Prueba.txt  PruebaMv.txt
```

Editar archivos

Con el comando nano abrimos el editor y podremos modificar nuestros archivos.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ nano Prueba.txt
```



Para guardar pulsar “Ctrl + X” y decir que sí “Y”.

Eliminar archivos o directorios

Utilizando el comando “rm” eliminaremos documentos y directorios.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ rm PruebaMv.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ ls
Prueba.txt
```

Para eliminar una carpeta utilizaremos la opción -R “rm -R [nombre_carpeta]”

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED
$ rm -R Practica1/
```

Actividad 2 – Primeros pasos con GIT

Crea un nuevo repositorio llamado Biblioteca y, una vez creado, muestra su contenido

Creemos una carpeta que se llame Biblioteca y utilizamos el comando “git init”.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca
$ git init
Initialized empty Git repository in D:/Roberto/Clases/EED/Biblioteca/.git/

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ |
```

Comprueba el estado del repositorio

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git status
On branch master

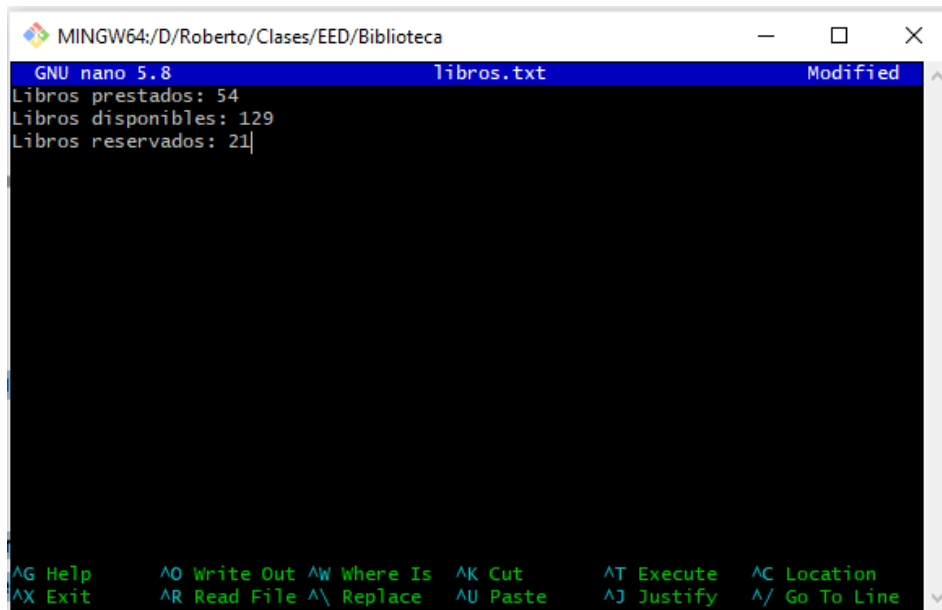
No commits yet

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

Crea un fichero libros.txt con el siguiente contenido:

- Libros prestados: 54
- Libros disponibles: 129
- Libros reservados: 21

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ nano libros.txt
```



A screenshot of a Windows terminal window titled "MINGW64;D:/Roberto/Clases/EED/Biblioteca". The window shows the GNU nano 5.8 text editor editing a file named "libros.txt". The file's content is: "Libros prestados: 54", "Libros disponibles: 129", and "Libros reservados: 21". The bottom status bar of the nano editor displays various keyboard shortcuts in green text, such as "AG Help", "AO Write Out", "AW Where Is", "AK Cut", "AT Execute", "AC Location", "AX Exit", "AR Read File", "A\ Replace", "AU Paste", "AJ Justify", and "A/ Go To Line".

Comprueba de nuevo el estado del repositorio

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git status
On branch master

No commits yet

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

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

Añade el fichero al índice, volviendo a comprobar el estado del repositorio.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git add libros.txt
warning: LF will be replaced by CRLF in libros.txt.
The file will have its original line endings in your working directory

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git status
On branch master

No commits yet

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

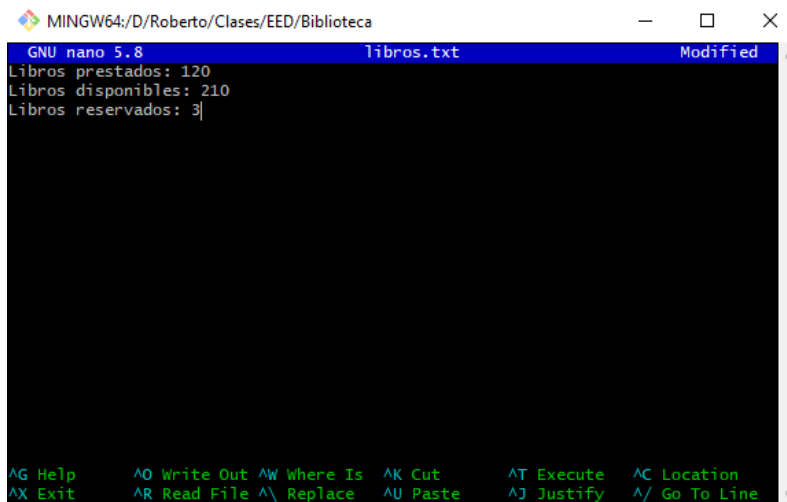
Realiza commit de los últimos cambios con el mensaje "Inventario libros 05/10/2021" y mira de nuevo el estado del repositorio.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git commit -m "Inventario libros 05/10/2021"
[master (root-commit) 117d5fb] Inventario libros 05/10/2021
1 file changed, 3 insertions(+)
create mode 100644 libros.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Cambia el fichero libros.txt para que contenga la siguiente información:

- Libros prestados: 120
- Libros disponibles: 210
- Libros reservados: 3



The screenshot shows a window titled "MINGW64/D/Roberto/Clases/EED/Biblioteca" with a nano 5.8 editor open to the file "libros.txt". The file contains the following text:

```
Libros prestados: 120
Libros disponibles: 210
Libros reservados: 3|
```

The bottom of the window shows the nano editor's command shortcuts.

Mostrar los cambios con respecto a la última versión guardada en el repositorio

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (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:   libros.txt

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

Hacer un commit de los cambios con el mensaje “Inventario actualizado a 07/10/2021”
En este caso, para ver esta opción también, utilizaremos la opción `–a` del commit que nos ahorrará el comando `“git add”`.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git commit -am "Inventario actualizado 07/10/2021"
warning: LF will be replaced by CRLF in libros.txt.
The file will have its original line endings in your working directory
[master 725060f] Inventario actualizado 07/10/2021
1 file changed, 3 insertions(+), 3 deletions(-)
```

Actividad 3 – Creando ramas

Crea una rama de desarrollo a partir de la rama master

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git branch desarrollo

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git branch
desarrollo
* master
```

En esta rama de desarrollo Crea un segundo fichero llamados revistas.txt. Su contenido será:

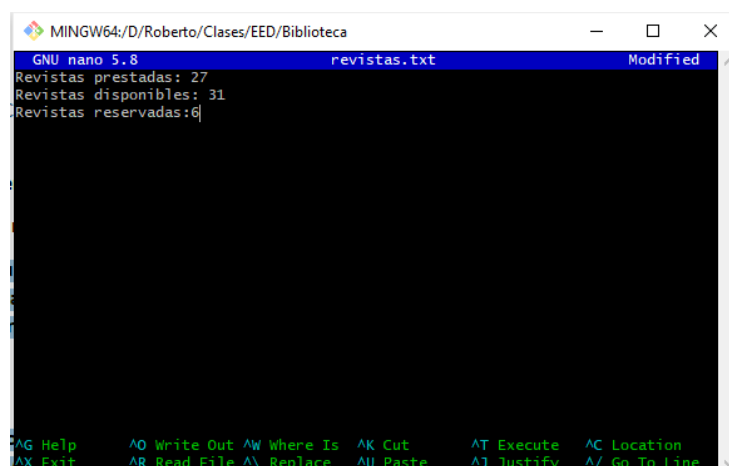
- **Revistas prestadas: 27**
- **Revistas disponibles: 31**
- **Revistas reservadas: 6**

Para crearlo en la rama desarrollo primero tendremos que movernos a esta rama.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git checkout desarrollo
Switched to branch 'desarrollo'

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ |
```

Una vez aquí ya creamos el fichero



Súbelo al repositorio de desarrollo con el comentario “Desarrollo inventario revistas 08/10/2020”

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git add revistas.txt
warning: LF will be replaced by CRLF in revistas.txt.
The file will have its original line endings in your working directory

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git commit -m "Desarrollo inventario revistas 08/10/2020"
[desarrollo 412d34b] Desarrollo inventario revistas 08/10/2020
1 file changed, 3 insertions(+)
create mode 100644 revistas.txt
```

Mira el historial y el historial resumido

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git log
commit 412d34b7b1c99e20010ff04f43fe3fd9feb73d3a (HEAD -> desarrollo)
Author: Roberto <roberto@roberto>
Date:   Wed Oct 13 19:28:38 2021 +0200

    Desarrollo inventario revistas 08/10/2020

commit 725060fe0ad94dcc105b31ecfd71c3beea830d7f (master)
Author: Roberto <roberto@roberto>
Date:   Wed Oct 13 19:19:05 2021 +0200

    Inventario actualizado 07/10/2021

commit 117d5fb309f9a911fbd202442b7255d0efdafbb1
Author: Roberto <roberto@roberto>
Date:   Wed Oct 13 19:04:56 2021 +0200

    Inventario libros 05/10/2021

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git log --oneline
412d34b (HEAD -> desarrollo) Desarrollo inventario revistas 08/10/2020
725060f (master) Inventario actualizado 07/10/2021
117d5fb Inventario libros 05/10/2021
```

Revisa las diferencias entre la rama de desarrollo y la master

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git diff master
diff --git a/revistas.txt b/revistas.txt
new file mode 100644
index 0000000..2a10372
--- /dev/null
+++ b/revistas.txt
@@ -0,0 +1,3 @@
+Revistas prestadas: 27
+Revistas disponibles: 31
+Revistas reservadas:6
```

Haz una fusión de la línea de desarrollo y la línea master con el comentario “Incurción inventario revistas 08/10/2020 en producción”

Como queremos incluir los cambios de desarrollo en master, primeros nos tenemos que mover a la línea master.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git checkout master
Switched to branch 'master'

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git merge desarrollo -m "Incurción inventario revistas 08/10/2020
n"
Updating 725060f..412d34b
Fast-forward (no commit created; -m option ignored)
 revistas.txt | 3 +++
 1 file changed, 3 insertions(+)
 create mode 100644 revistas.txt
```

Pon una etiqueta en la línea master que indique que es la versión 1.0 de nuestro proyecto.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git tag v.1.0
```

Visualiza el estado del repositorio y el log del mismo.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git status
On branch master
nothing to commit, working tree clean

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git log
commit 412d34b7b1c99e20010ff04f43fe3fd9feb73d3a (HEAD -> master, tag: v.1.0, desarrollo)
Author: Roberto <roberto@roberto>
Date:   Wed Oct 13 19:28:38 2021 +0200

    Desarrollo inventario revistas 08/10/2020

commit 725060fe0ad94dcc105b31ecfd71c3beea830d7f
Author: Roberto <roberto@roberto>
Date:   Wed Oct 13 19:19:05 2021 +0200

    Inventario actualizado 07/10/2021

commit 117d5fb309f9a911fbd202442b7255d0efdafbb1
Author: Roberto <roberto@roberto>
Date:   Wed Oct 13 19:04:56 2021 +0200

    Inventario libros 05/10/2021
```


Actividad 4 - Borrando archivos

Vuelve a la rama de desarrollo e incluye un fichero con el nombre películas.txt

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git checkout desarrollo
Switched to branch 'desarrollo'

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ touch peliculas.txt
```

Añade y guarda este fichero en esta rama.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git add peliculas.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git commit -m "añadimos peliculas a desarrollo"
[desarrollo 6c7249f] añadimos peliculas a desarrollo
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 peliculas.txt
```

Parece que el desarrollo de las películas no tiene sentido, por lo que nos piden eliminar este fichero, pero como sabemos que desde la dirección de la biblioteca las normas cambian muy a menudo, no hacemos commit de este cambio.

Borramos sin hacer commit

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git rm peliculas.txt
rm 'peliculas.txt'

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git status
On branch desarrollo
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    deleted:    peliculas.txt
```

Efectivamente, ahora la dirección quiere continuar con este desarrollo y nos piden que añadamos la siguiente información.

- Revistas prestadas: 0
- Revistas disponibles: 10
- Revistas reservadas: 0

En primer lugar recuperamos el archivo.

```

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git restore --staged peliculas.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git status
On branch desarrollo
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    peliculas.txt

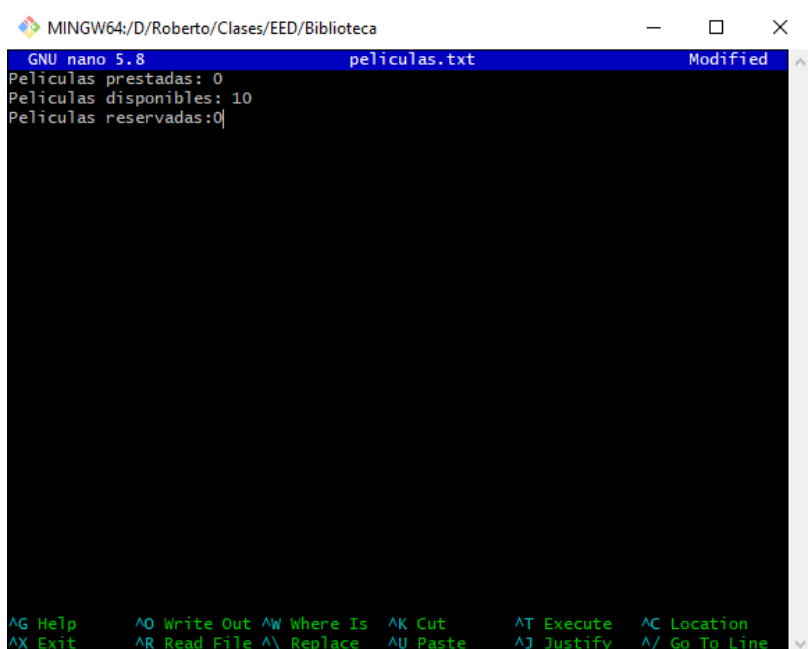
no changes added to commit (use "git add" and/or "git commit -a")

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git restore peliculas.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ ls
libros.txt  peliculas.txt  revistas.txt

```

Y después lo editamos.



```

MINGW64:/D/Roberto/Clases/EED/Biblioteca
GNU nano 5.8 peliculas.txt Modified
Películas prestadas: 0
Películas disponibles: 10
Películas reservadas:0
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line

```

Guardamos los cambios en el repositorio

```

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git add peliculas.txt
warning: LF will be replaced by CRLF in peliculas.txt.
The file will have its original line endings in your working directory

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git commit -m "añadimos peliculas"
[desarrollo 271f9f5] añadimos peliculas
1 file changed, 4 insertions(+)

```

Días después nos dicen que definitivamente no se va a continuar este desarrollo puesto que el número de películas es muy escaso y no tendrá éxito. En este momento eliminamos el fichero y hacemos commit

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git rm peliculas.txt
rm 'peliculas.txt'

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git commit -m "borramos peliculas"
[desarrollo 775661f] borramos peliculas
1 file changed, 4 deletions(-)
delete mode 100644 peliculas.txt
```

De repente un donante anónimo decide donar su colección de películas y la biblioteca comienza a promocionar esta iniciativa y la gente comienza a reservar. Nos dicen que añadamos la siguiente información:

- Revistas prestadas: 0
- Revistas disponibles: 172
- Revistas reservadas: 25

Es decir, tendremos que recuperar el fichero previo y añadir los cambios

En primer lugar miramos el id de la última versión con el fichero, 271f9f5 en este caso.

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git log --oneline
775661f (HEAD -> desarrollo) borramos peliculas
271f9f5 añadimos peliculas
6c7249f añadimos peliculas a desarrollo
412d34b (tag: v.1.0, master) Desarrollo inventario revistas 08/10/2020
725060f Inventario actualizado 07/10/2021
117d5fb Inventario libros 05/10/2021
```

Ahora tendremos que recuperarlo. Nos movemos a esta versión, 271f9f5, y decimos el fichero que deseamos

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git checkout 271f9f5 peliculas.txt
Updated 1 path from 9015e68

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ ls
libros.txt  peliculas.txt  revistas.txt
```

Una vez lo tenemos, lo restauramos y guardamos los cambios en git

```
roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git status
On branch desarrollo
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   peliculas.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git restore --staged peliculas.txt
```

```

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git add peliculas.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git commit -m "fichero recuperado"
[desarrollo 88a6589] fichero recuperado
1 file changed, 4 insertions(+)
create mode 100644 peliculas.txt

```

Para que la gente pueda reservar, habrá que pasar este desarrollo a producción. Una vez ya hemos recuperado es el momento de fusionar este desarrollo con la rama master, y además añadir la etiqueta de versión 1.1

```

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (desarrollo)
$ git checkout master
Switched to branch 'master'

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git merge desarrollo
Updating 412d34b..88a6589
Fast-forward
 peliculas.txt | 4 ++++
 1 file changed, 4 insertions(+)
 create mode 100644 peliculas.txt

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git tag v.1.1

roberto@roberto MINGW64 /D/Roberto/Clases/EED/Biblioteca (master)
$ git log --oneline
88a6589 (HEAD -> master, tag: v.1.1, desarrollo) fichero recuperado
775661f borramos peliculas
271f9f5 añadimos peliculas
6c7249f añadimos peliculas a desarrollo
412d34b (tag: v.1.0) Desarrollo inventario revistas 08/10/2020
725060f Inventario actualizado 07/10/2021
117d5fb Inventario libros 05/10/2021

```