

## Práctica: Uso de Git a través de comandos - Historial de cambios

Muestra el historial de cambios del repositorio. (Comando log)

```
$ cd libro
DAW@PC-Ruben MINGW64 ~/libro (master)
$ git log
commit b2fea81659b863fe61424aad7d966ca21cff8b46 (HEAD -> master)
Author: N3BUL054 <n3burub3n39@gmail.com>
Date: Fri Feb 2 13:41:42 2024 +0100

    Añadido capítulo 3 sobre gestion de ramas al indice

commit c6d563784e5026724e28c3255b438a486fc72a27
Author: N3BUL054 <n3burub3n39@gmail.com>
Date: Fri Feb 2 13:37:55 2024 +0100

    aaaaaa
```

Crea la carpeta capítulos y crea dentro de ella el fichero capítulo.txt con el siguiente texto: "Git es un sistema de control de versiones ideado por Linus Torvalds."

```
DAW@PC-Ruben MINGW64 ~/libro (master)
$ mkdir capitulos

DAW@PC-Ruben MINGW64 ~/libro (master)
$ cd capitulos

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ nano capitulo.txt

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$
```

Añade los cambios a la zona de intercambio temporal. (comando add .)  
Haz commit de los cambios y vuelve a mostrar el historial.

```
GNU nano 7.2 capitulo.txt
Git es un sistema de control de versiones ideado para linus torvalds

[ Read 1 line ]
^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
```

Crea el fichero `capitulo2.txt` en la carpeta `capitulos` con el siguiente texto:

```
GNU nano 7.2          capitulo2.txt
El flujo de trabajo básico con Git consiste en:
1- Hacer cambios en el repositorio.
2- Añadir los cambios a la zona de intercambio temporal.
3- Hacer un commit de los cambios.
```

El flujo de trabajo básico con Git consiste en:

- 1- Hacer cambios en el repositorio.
- 2- Añadir los cambios a la zona de intercambio temporal.
- 3- Hacer un commit de los cambios.

Añade los cambios a la zona de intercambio temporal.

```
DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   capitulo2.txt

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

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ |
```

Haz un commit de los cambios con el mensaje "Añadido capítulo 2."

```
DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git commit -m "añadido capitulo 2"
[master 9841bd0] añadido capitulo 2
1 file changed, 4 insertions(+)
create mode 100644 capitulos/capitulo2.txt

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$
```

Muestra las diferencias entre la última versión y dos versiones anteriores.  
(comando `diff HEAD~2..HEAD`)

Muestra también quién ha hecho los cambios

```
$ git diff HEAD~2..HEAD
diff --git a/capitulos/capitulo2.txt b/capitulos/capitulo2.txt
new file mode 100644
index 0000000..0070310
--- /dev/null
+++ b/capitulos/capitulo2.txt
@@ -0,0 +1,4 @@
+El flujo de trabajo básico con Git consiste en:
+1- Hacer cambios en el repositorio.
+2- Añadir los cambios a la zona de intercambio temporal.
+3- Hacer un commit de los cambios.
diff --git a/indice.txt b/indice.txt
index 8e62a31..1dd57ee 100644
--- a/indice.txt
+++ b/indice.txt
@@ -1,3 +1,5 @@
  Capítulo 1: Introducción a Git
  Capítulo 2: Flujo de trabajo básico
- Capítulo 3: Repositorios remotos
+ Capítulo 3: Gestión de ramas
+ Capítulo 4: Repositorios remotos
```

Añade al final del fichero indice.txt del ejercicio anterior una línea. Añade los cambios a la zona de intercambio temporal, haz commit y muestra quién ha hecho cambios sobre el fichero (comando annotate).

```
DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git add capitulo.txt

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git add capitulo2.txt

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git annotate capitulo2.txt
9841bd04      ( N3BUL054      2024-02-06 12:06:58 +0100      1)El flujo de tr
abajo básico con Git consiste en:
9841bd04      ( N3BUL054      2024-02-06 12:06:58 +0100      2)1- Hacer cambi
os en el repositorio.
9841bd04      ( N3BUL054      2024-02-06 12:06:58 +0100      3)2- Añadir los
cambios a la zona de intercambio temporal.
9841bd04      ( N3BUL054      2024-02-06 12:06:58 +0100      4)3- Hacer un co
mmit de los cambios.
```

Hecho esto, elimina esa última línea y comprueba el estado del repositorio. Vuelve a la versión anterior, deshaciendo los cambios (comando checkout)

```
DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   capitulo.txt

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:   capitulo2.txt

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git checkout -- capitulo2.txt
error: pathspec 'capitulo2.txt' did not match any file(s) known to git

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
$ git checkout
error: pathspec 'capitulo2.txt' did not match any file(s) known to git

DAW@PC-Ruben MINGW64 ~/libro/capitulos (master)
```