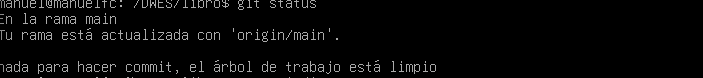
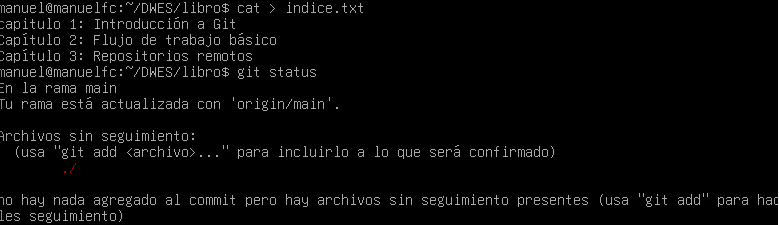
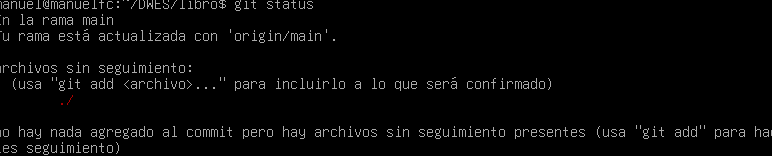
CREACIÓN Y ACTUALIZACIÓN DE REPOSITORIOS

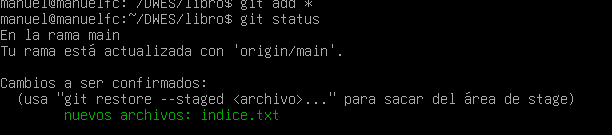
EJERCICIO 3 y 4

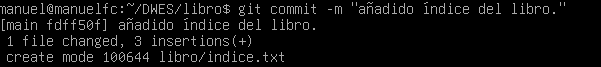
1.- Al hacer un git status, tras crear el repositorio libro, y estar vacío muestra eso:

2.- Se crea el documento de texto mediante el comando señalado (cat > índice.txt) y en la consola se refleja, tras pulsar ctrl+D y hacer de nuevo una comprobación lo siguiente:

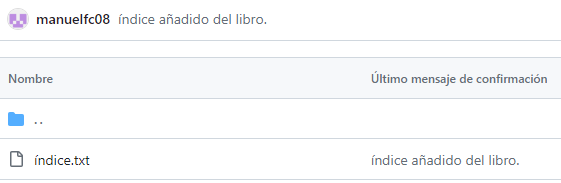
3.- Vemos el estado del repositorio tras crear el fichero de texto y además nos indica que debemos hacer un git add para añadir el fichero al repositorio libro.

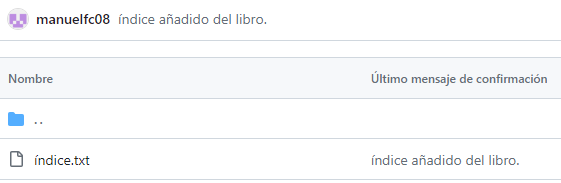
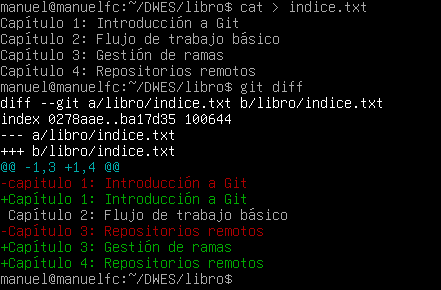


4.- Se añade temporalmente el fichero, ya que hasta que no se confirma con commit, no se puede subir el fichero al repositorio.

5.- Al hacer de nuevo la comprobación tras el commit y el git push, confirma que se ha añadido el nuevo archivo a GitHub.

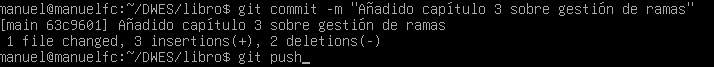
Aquí efectivamente queda reflejado que se subió el archivo índice al repositorio.



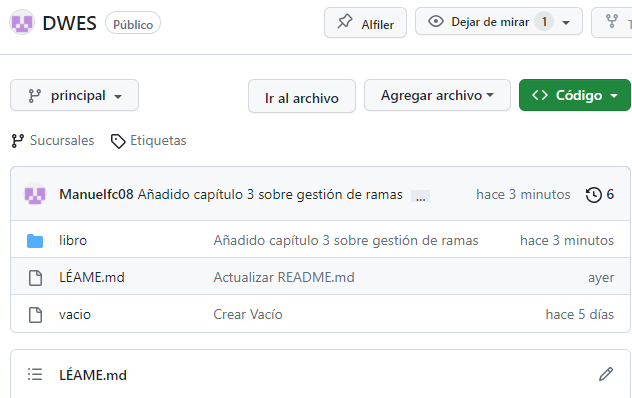
EJERCICIO 5

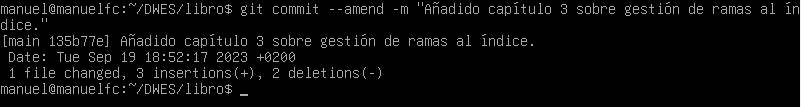
Primero se añaden los datos nuevos al fichero, y al comprobar con git diff, se aprecia lo que había antes, con respecto a lo que se añade como nuevo.

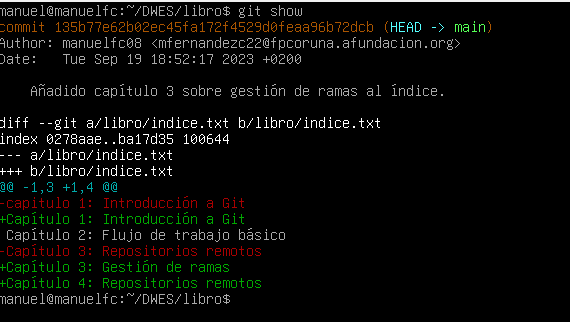
Finalmente, al hacer el commit con el mensaje nuevo se cambia el anterior por el que añadimos posteriormente: "Añadido capítulo 3 sobre gestión de ramas"



EJERCICIO 6

Tras subir los cambios al repositorio, vemos como quedan aplicados en la siguiente captura:

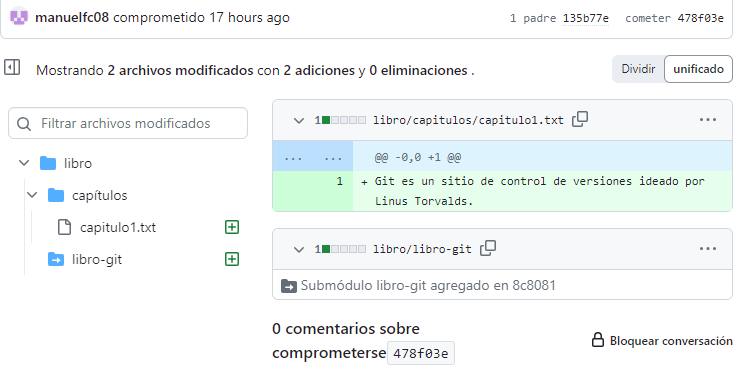
En esta captura se refleja la operación de modificación de mensaje, el anterior por el nuevo.

En esta otra con el comando de mostrar, se observa el mensaje cambiado.

HISTORIAL DE CAMBIOS

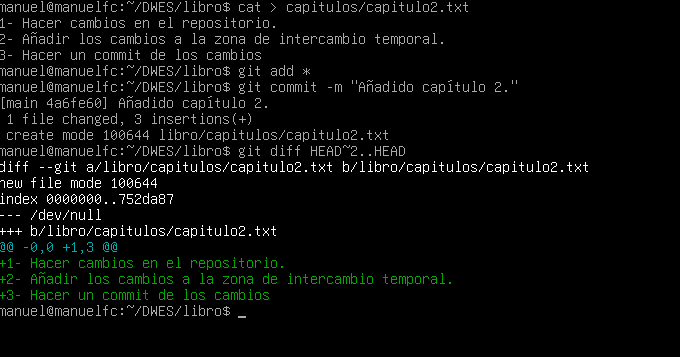
EJERCICIO 1´

No puedo presentar captura de la consola, porque tuve un error en el ejercicio.

Pero como se aprecia en repositorio remoto, si tengo creado capítulo y capítulo1.

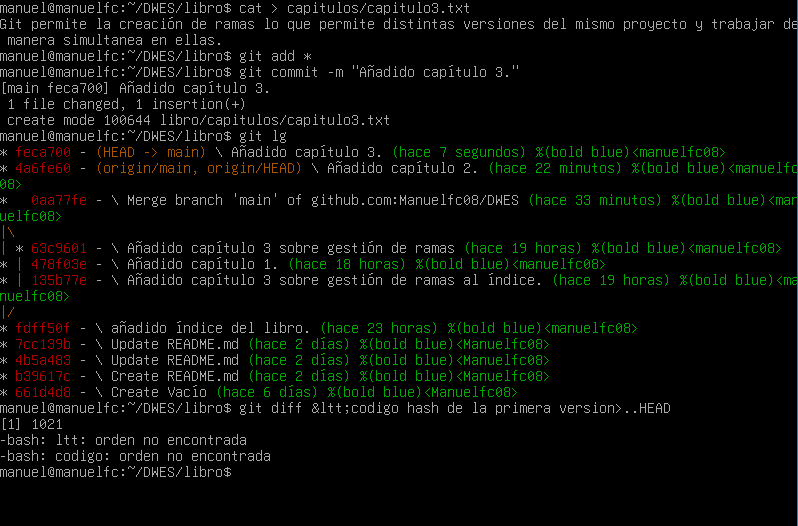
EJERCICIO 2

Añade el siguiente capítulo al repositorio libro



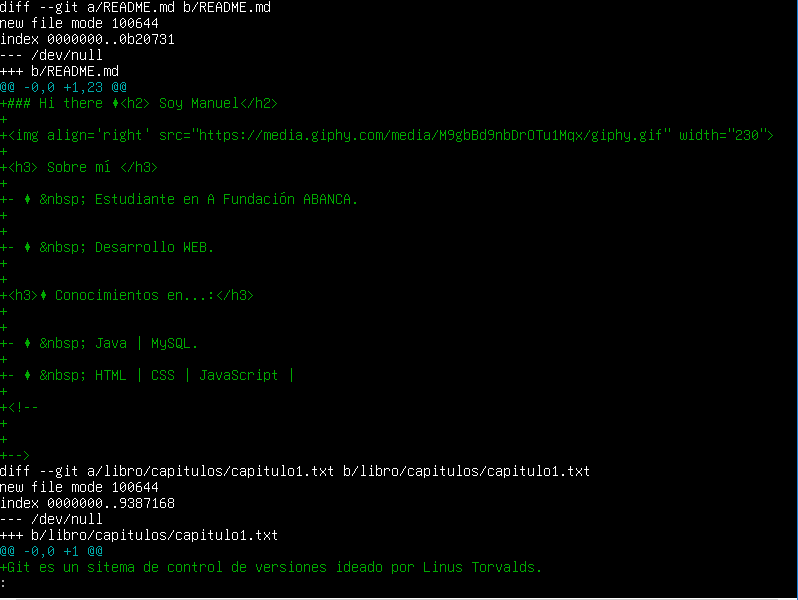
EJERCICIO 3

Creamos el tercer capítulo, añadiendo el nuevo mensaje



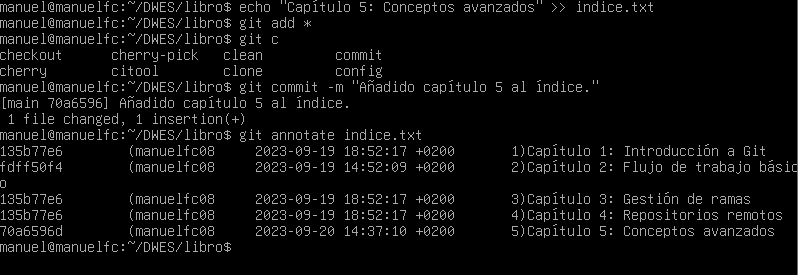
Después comparamos el primer commit “vacio” de hash 661d4d8 con el HEAD

Dónde nos muestra los cambios que se han añadido con respecto de la primera a la última versión.



EJERCICIO 4

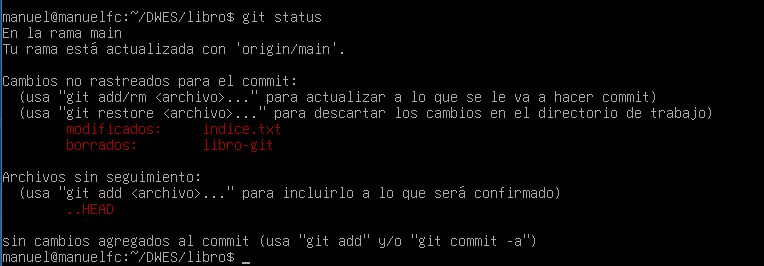
Hemos creado un repositorio llamado libro, compuesto por capítulos, añadimos el quinto capítulo, y el índice, que lógicamente tienen todos los libros.

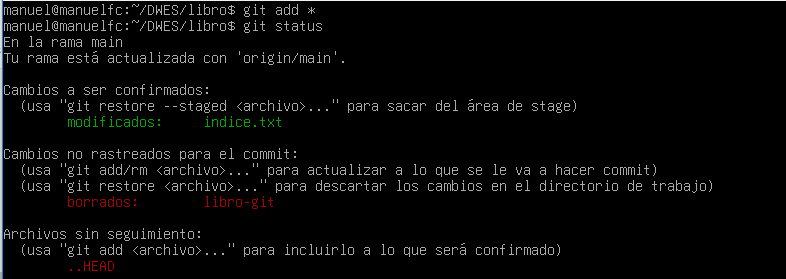


DESHACER CAMBIOS

EJERCICIO 1

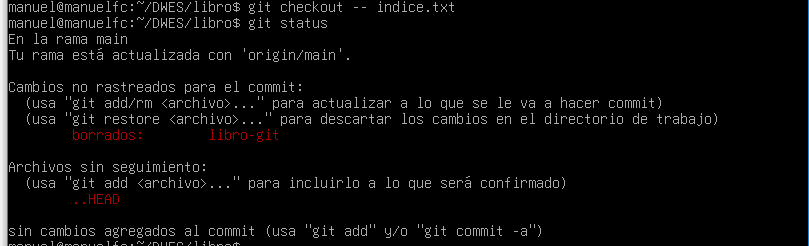
Accedemos al editor nano y al borrar el capítulo 5, última línea añadida hasta el momento muestra el siguiente estado actual:

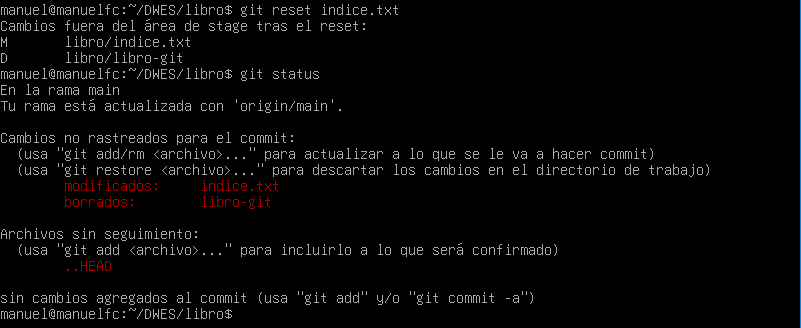


A continuación, revertimos los cambios por medio de un comando nuevo y como se ve en el estado, índice.txt ya no aparece en rojo como elemento borrado.

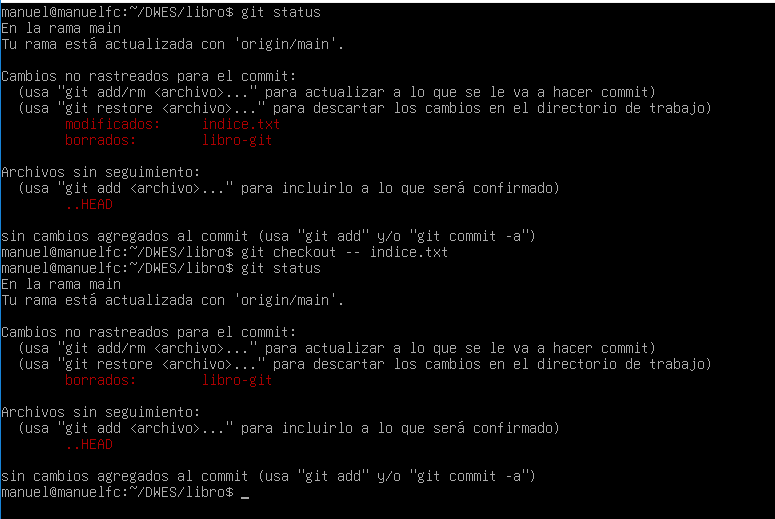
EJERCICIO 2

Comenzamos alterando el fichero índice y añadiendo los cambios, que supone que el último capítulo, no conste.

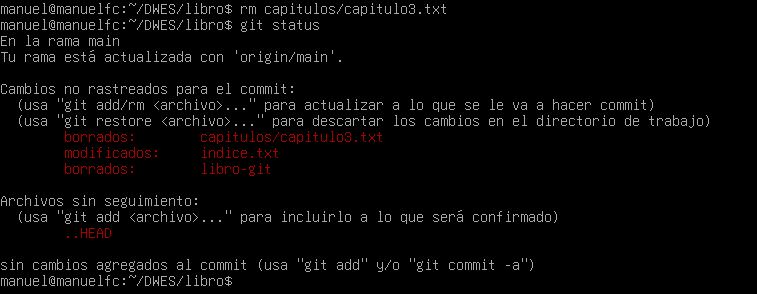


En esta operación anterior teníamos hecho un git add, que temporalmente mantiene el posible cambio para aplicar en el repositorio remoto. Así que por medio de git reset se anula esa temporalidad, pero continuamos manteniendo el estado actual del área de trabajo, es decir, la última línea del índice sigue borrada.

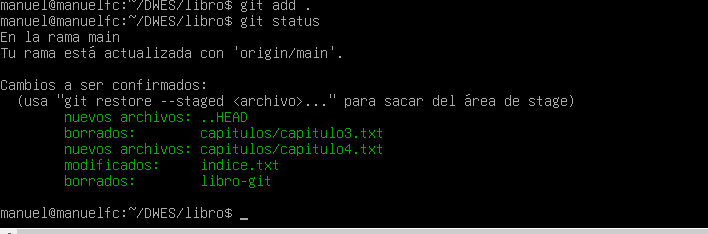
Así que, por medio de git checkout – índice.txt, revertimos el cambio, quedando el índice como se encontraba en origen. Sin ningún capítulo borrado.



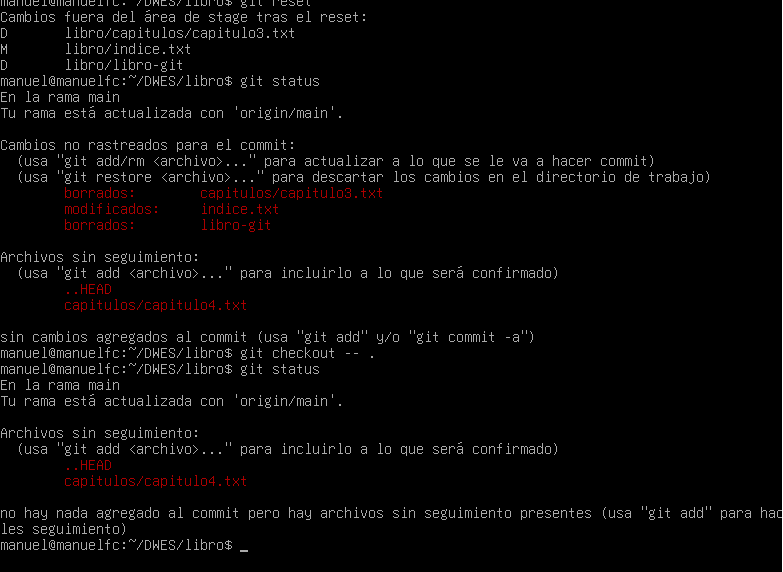
EJERCICIO 3

Se elimina desde el editor de Debian el capitulo3 como muestra el estado.

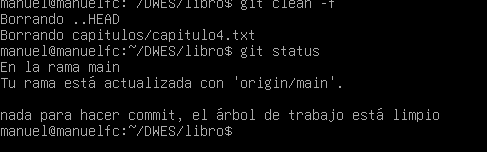
Comprobamos como dice el terminal que los cambios a ser confirmados son:



Que estaría pendiente de commit y push si fuese preciso. Pero, al restablecer los cambios de la zona de intercambio temporal, vemos que se anula la situación de borrado del capítulo, quedando aún pendiente de confirmación, el añadido del capitulo 4.

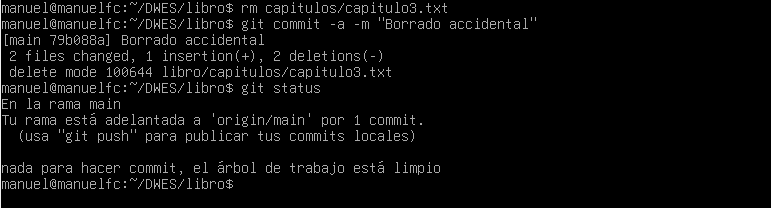


Finalmente, con el comando git clean -f, eliminados la situación temporal de confirmar el capitulo4, regresando al estado inicial del repositorio local de nuestra área de trabajo.

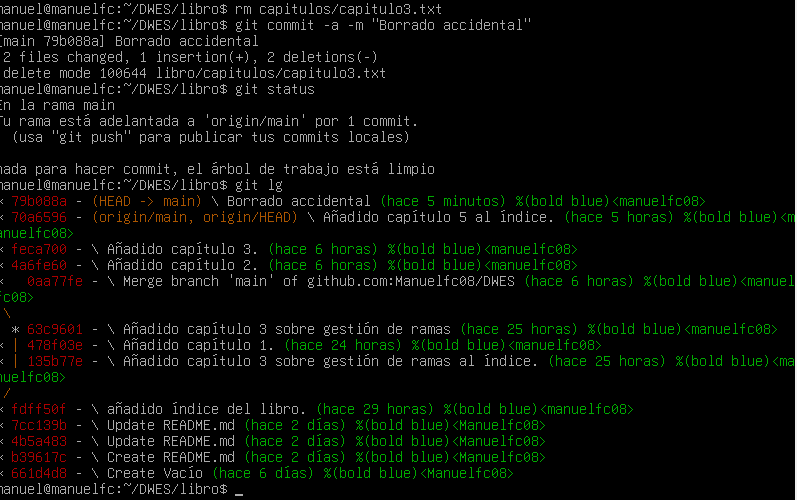


EJERCICIO 4

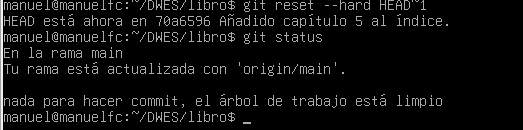
Eliminamos última línea índice y capítulo3, queda en estado temporal, añadimos el mensaje con git commit -a -m (esto no viene en el ejercicio, si no se pone la -m da un error).



Vemos el historial para comprobar la situación.



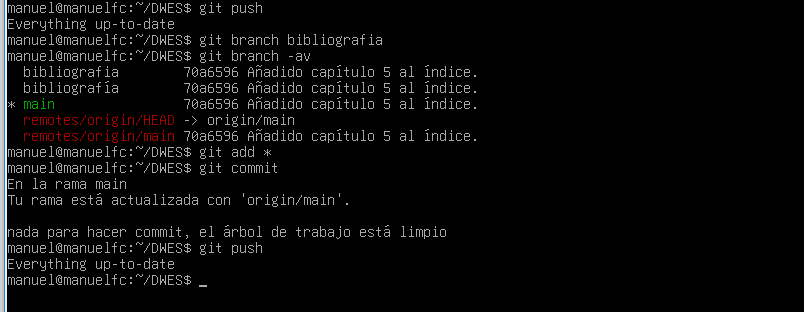
Restablecemos los cambios.



RAMAS

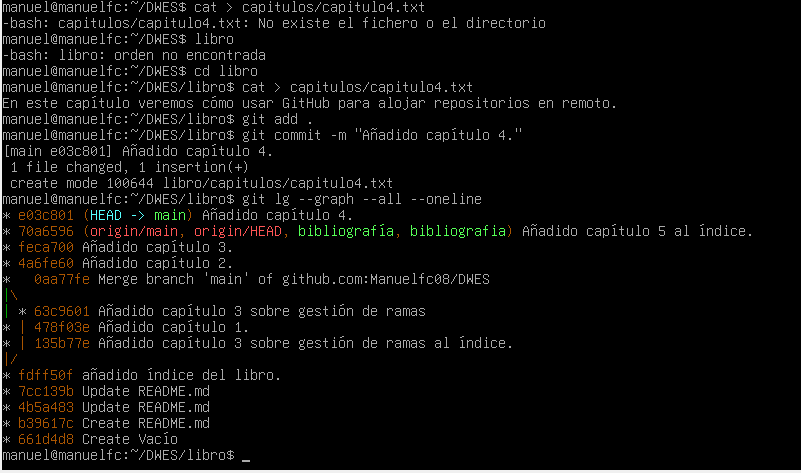
EJERCICIO 1

Se crea una rama y se muestran las que hay.



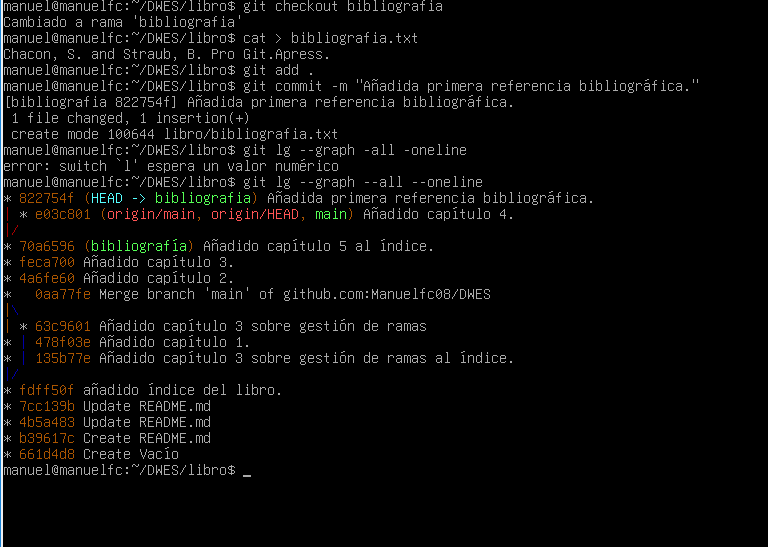
EJERCICIO 2

Creamos un nuevo capítulo, le añadimos un sms. Listamos todas las ramas existentes y subimos los cambios al repo remoto.



EJERCICIO 3

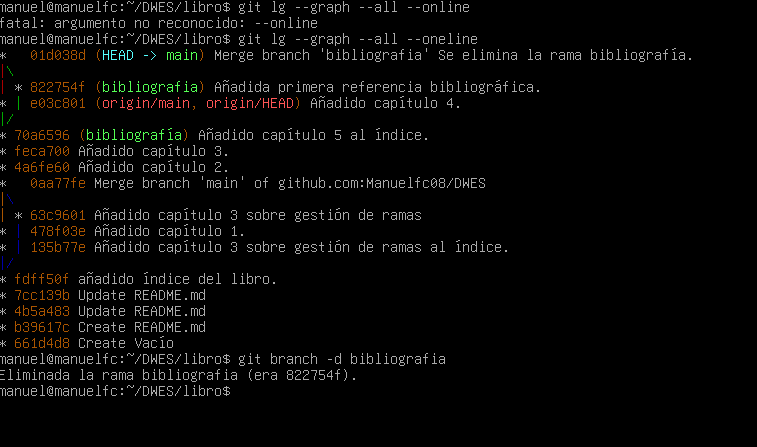
Nos posicionamos en la bibliografía, añadimos un fichero de texto, con una referencia bibliográfica. Posteriormente se muestra el resumen del histórico.



EJERCICIO 4

En el ejercicio anterior intenté hacer un push, pero como la rama bibliografía no estaba fusionada con la principal, me dio un sms de error. Así pues, en este ejercicio 4 precisamente se va a fusionar la rama bibliografía con la principal.

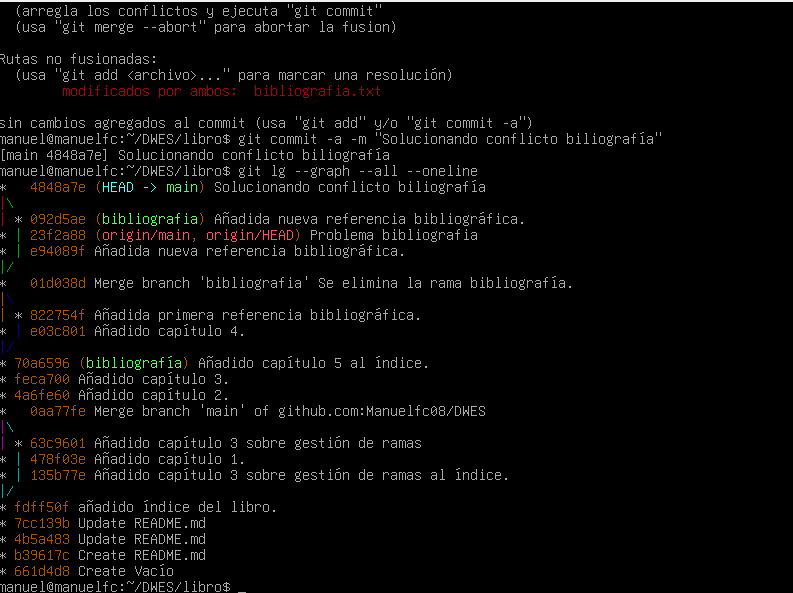
Eliminamos la rama creada de bibliografía y mostramos el historial



EJERCICIO 5

En este ejemplo vemos como fusionar ramas y solucionar este conflicto, es decir, se trabaja en una rama bibliografía, al cambiar de rama y editar el fichero también llamado bibliografía, al añadir los cambios y querer hacer un push, surge el conflicto entre ramas. Hacemos merge para fusionar y corregimos el archivo.



Listamos resumen