Informe ejercicios de git

Dilan Alexander Robayo Tavera Julio Roberto Galvis

Servicio Nacional de Aprendizaje

Análisis y desarrollo de Software

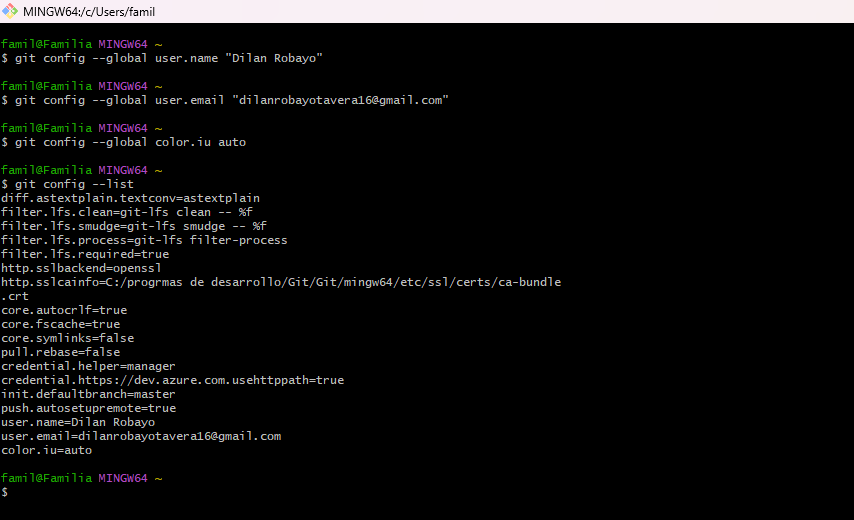
2024

Bogotá D.C

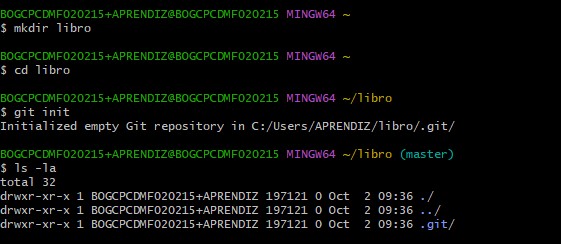
**Ejercicios de creación y actualización de repositorios**

# Ejercicio 1

Git config se utiliza para configurar el nombre de usuario del repositorio, el email y otros ajustes necesarios.



# Ejercicio 2: crear repositorio nuevo



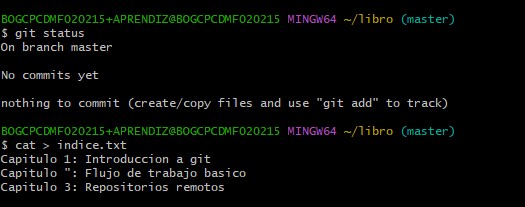
Mkdir: creación de directorio

cd : busca la ubicación para la creación del directorio

Git init: inicia repositorio

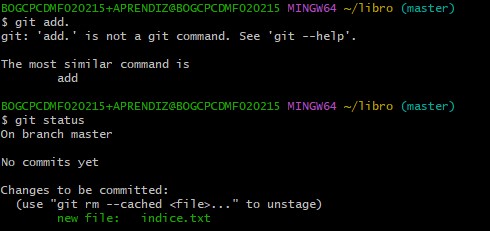
Ls -la: lista los archivos o directorios del repositorio actual e incluye detalles de fecha, propietario, permisos y etc.

**Ejercicio 3:**

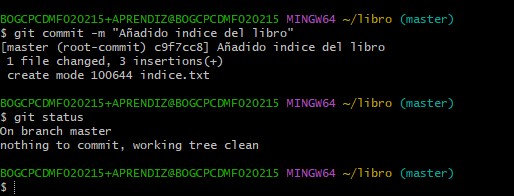


Git status: muestra el estado actual del repositorio

cat>: creación o sobreescribir el archivo

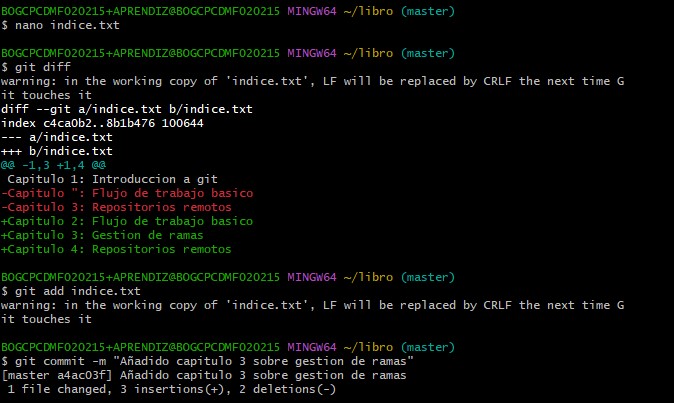


**Git add: rastreador de archivos nuevos Ejercicio 4:**



# Git commit: confirma los cambios en el repositorio local

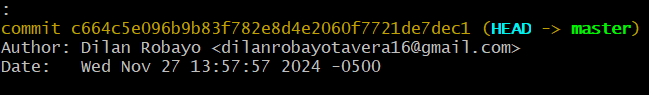
**Ejercicio 5:**

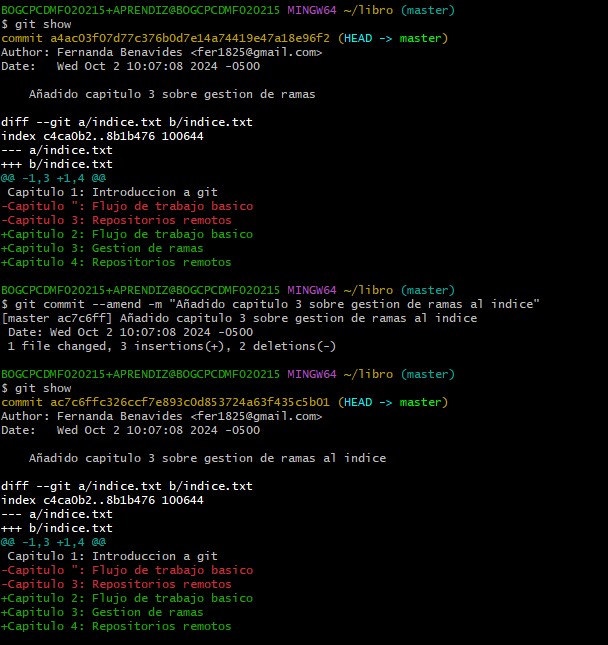


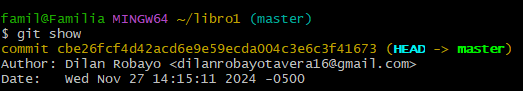
# Nano: editor de código desde la terminal

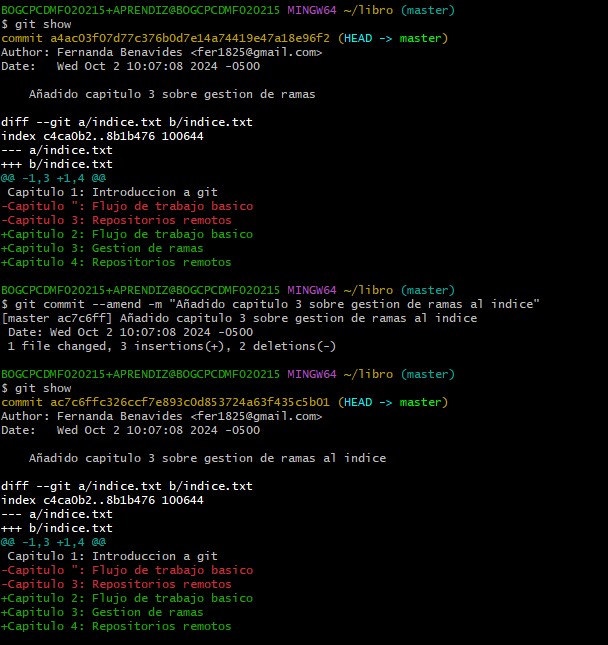
**Git diff: muestra la diferencia entre los cambios en el repositorio y el último commit.**

**Ejercicio 6:**









**Git show: muestra la información detallada sobre un commit en especifico**

**Ejercicios de manejo del historial de cambios**

# Ejercicio1

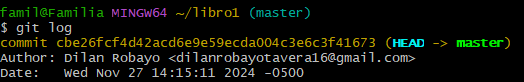
# 

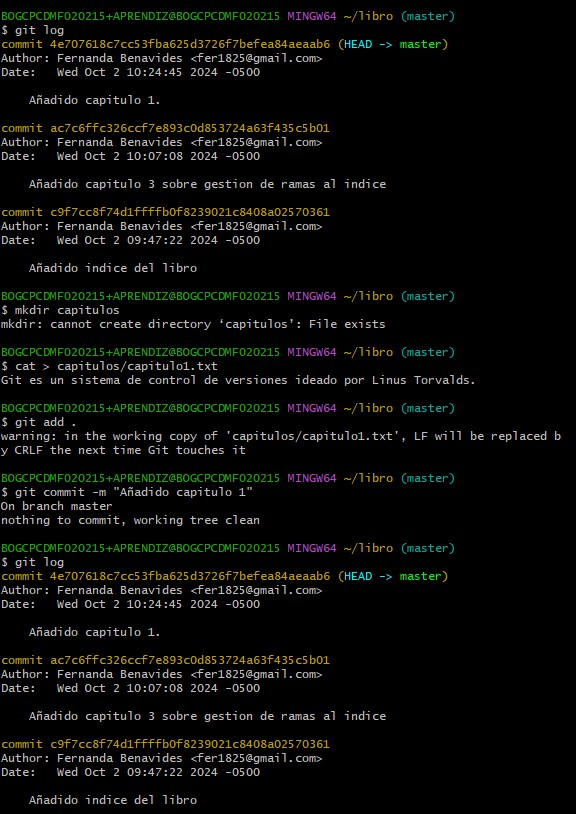
# 

# 

# 

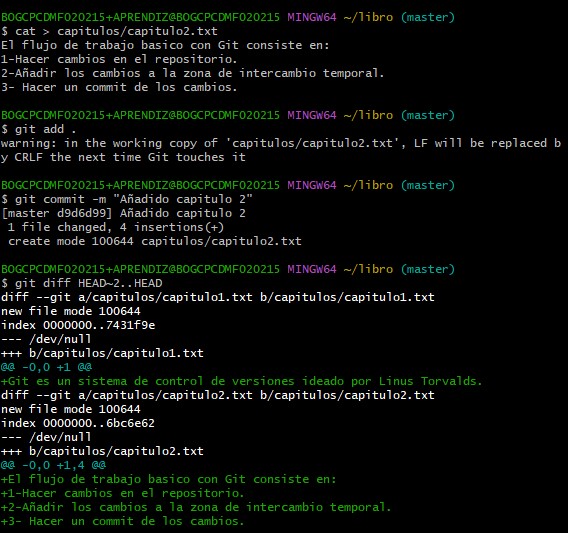
# 



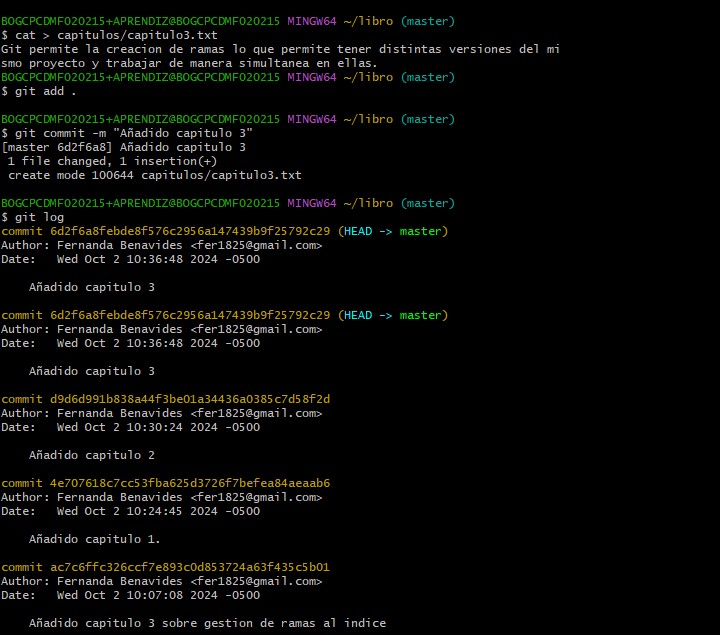


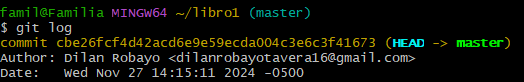
**Git log: muestra el historial de commits en un repositorio.**

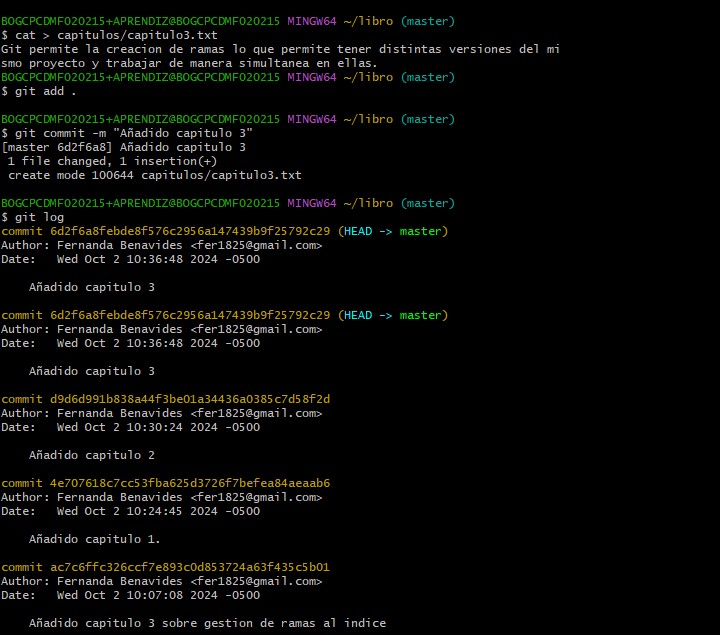
# Ejercicio 2

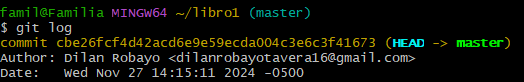


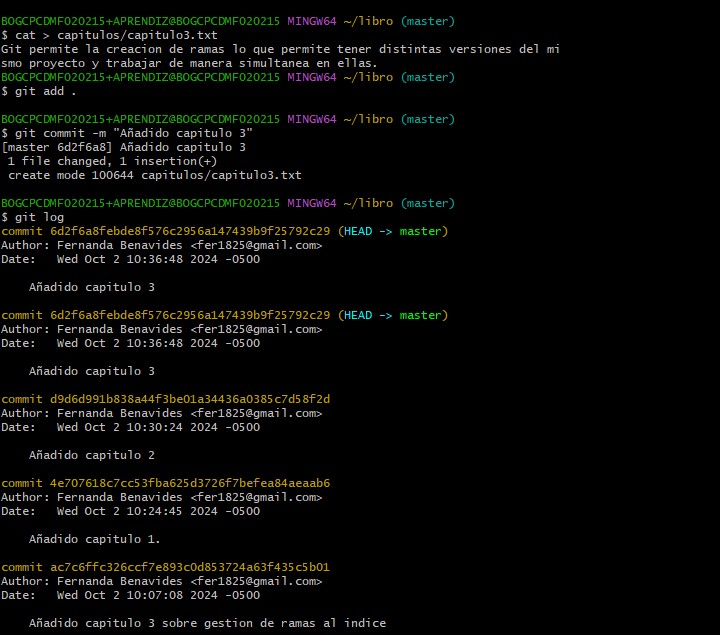
# Ejercicio 3

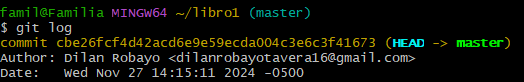


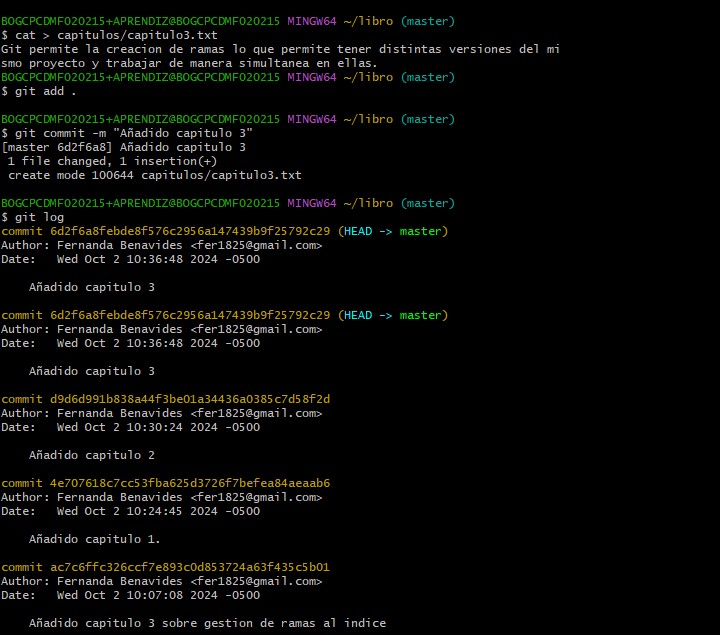


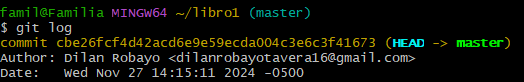


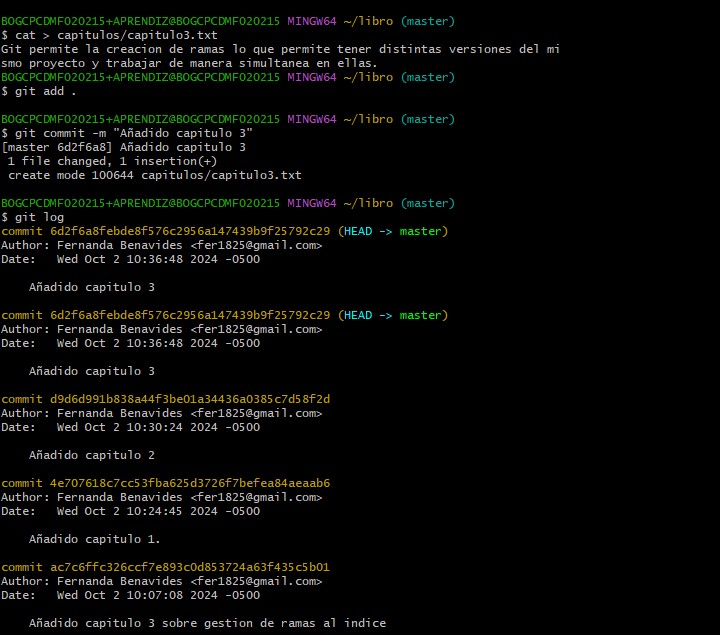


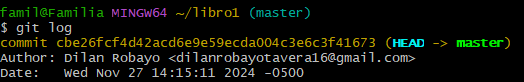


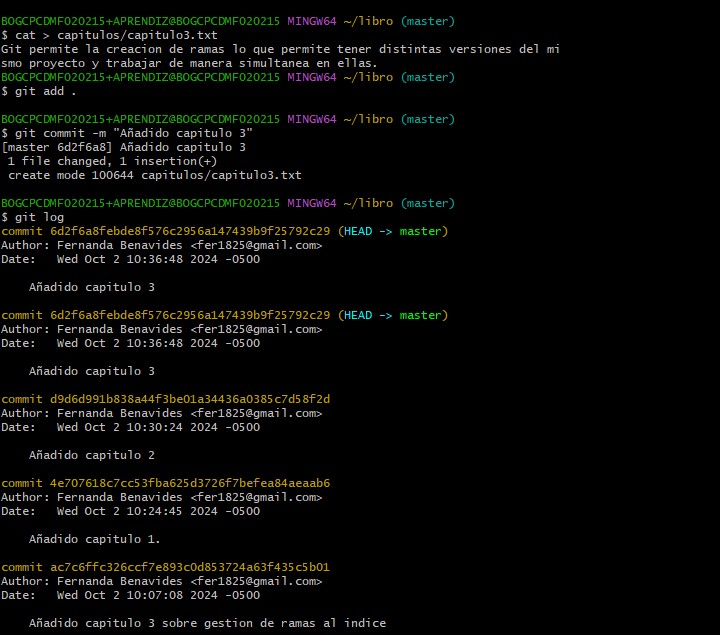


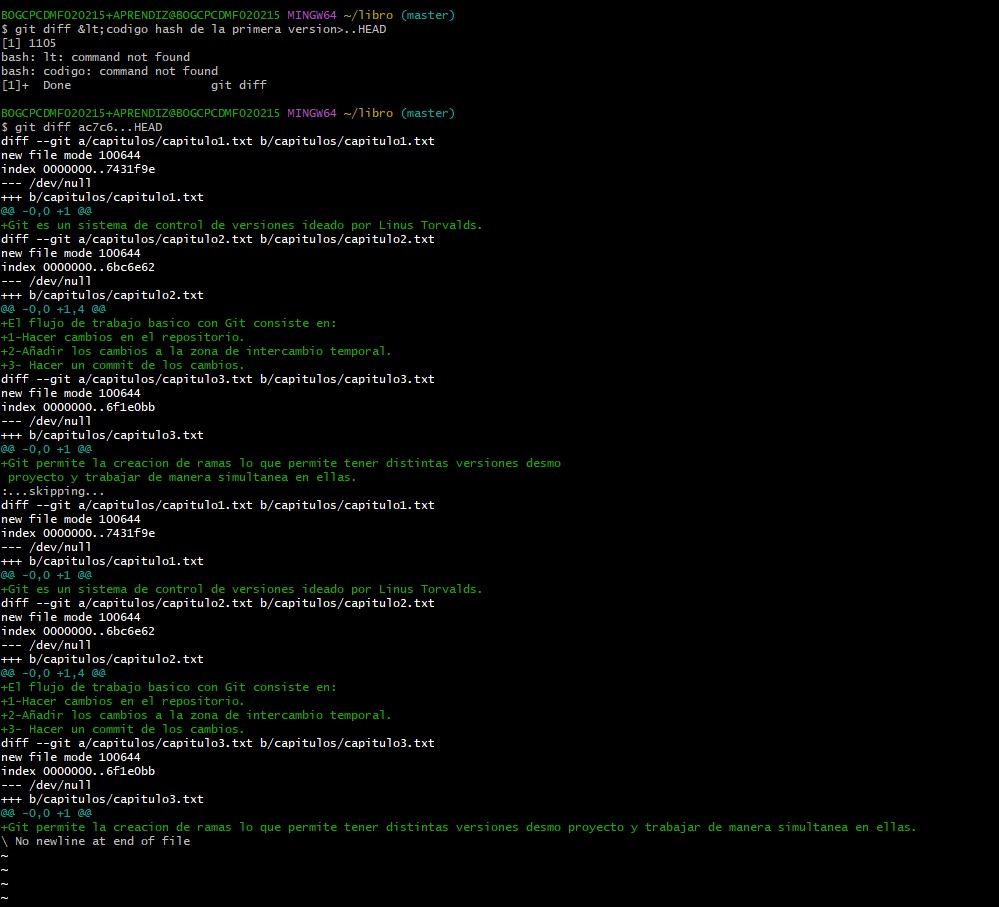


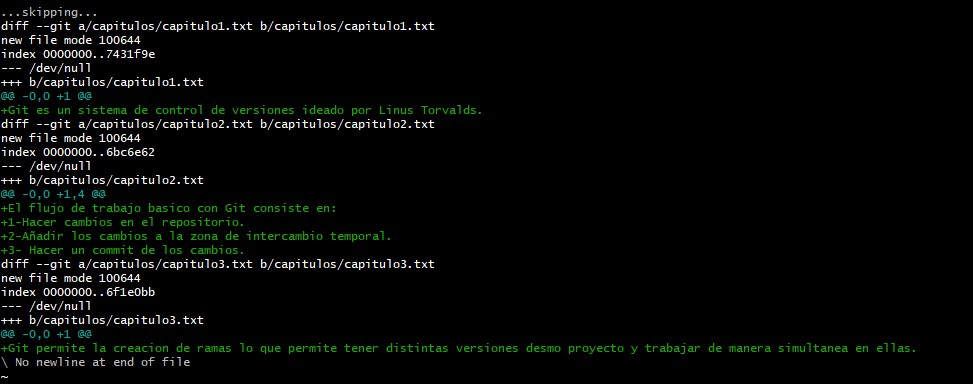


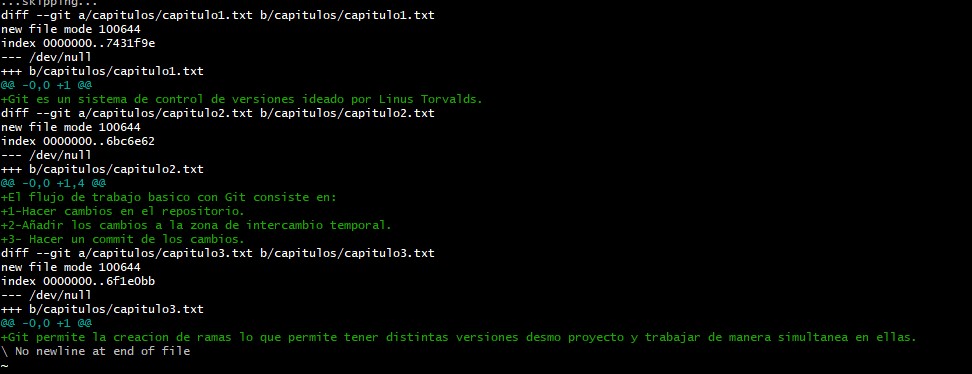




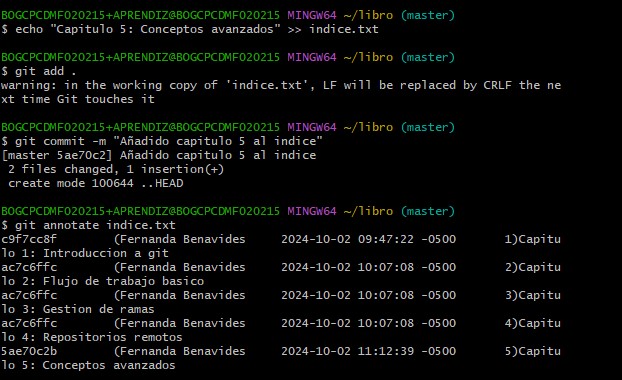








# Ejercicio 4



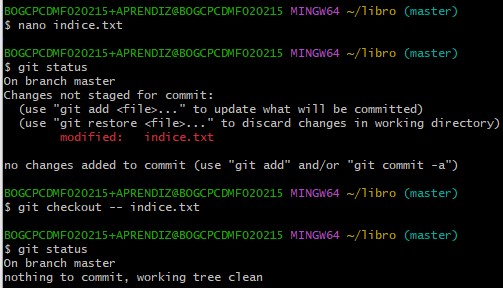
Echo: crea o sobreescribe directamente en el archivo.

Git annotate: para mostrar qué usuario realizó los cambios en cada línea de un archivo

específico.

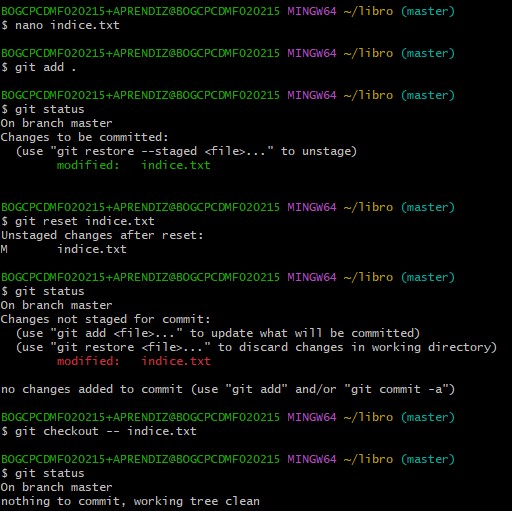
**Ejercicios de deshacer los cambios**

# Ejercicio 1



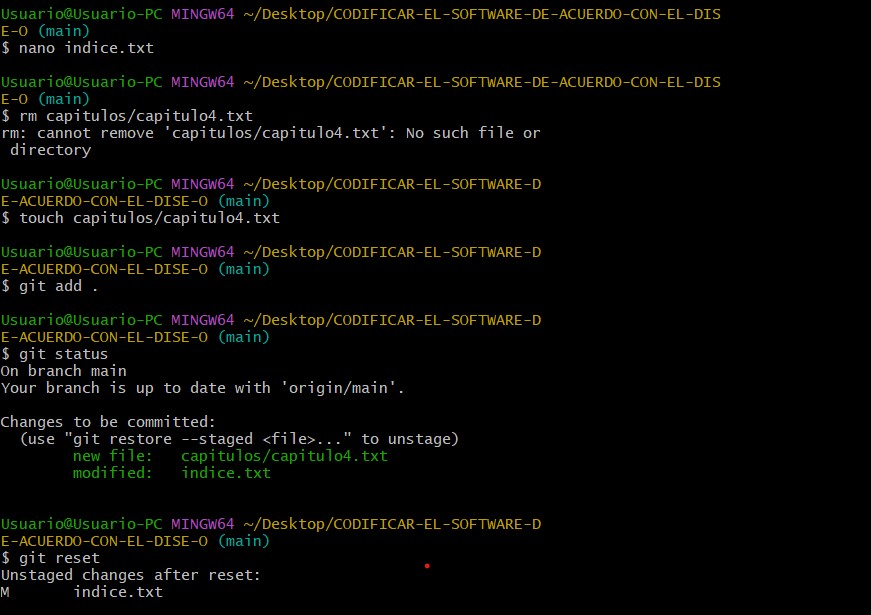
**Git checkout: restauración de cambios en archivos**

# Ejercicio 2



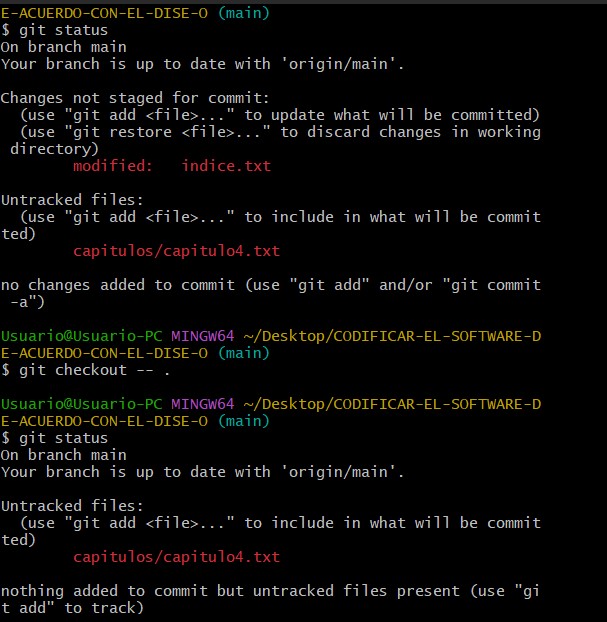
**Git reset: deshacer cambios en el área de staging**

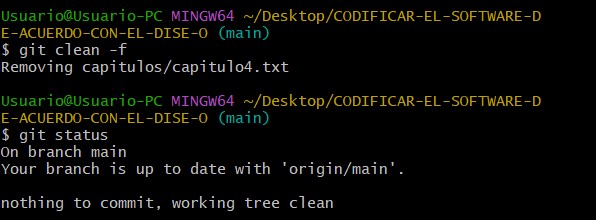
# Ejercicio 3



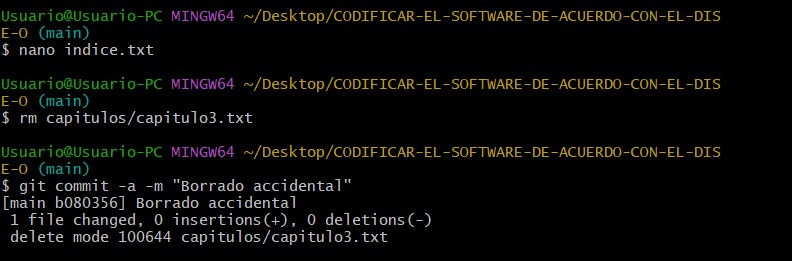
Rm: renové (eliminación de archivos y directorios)

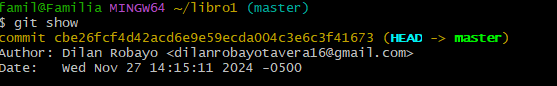
Touch: crea archivos vacíos y para gestionar marcas de tiempo de archivos existentes.

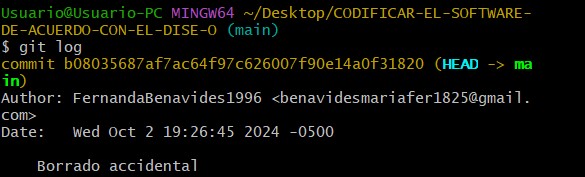


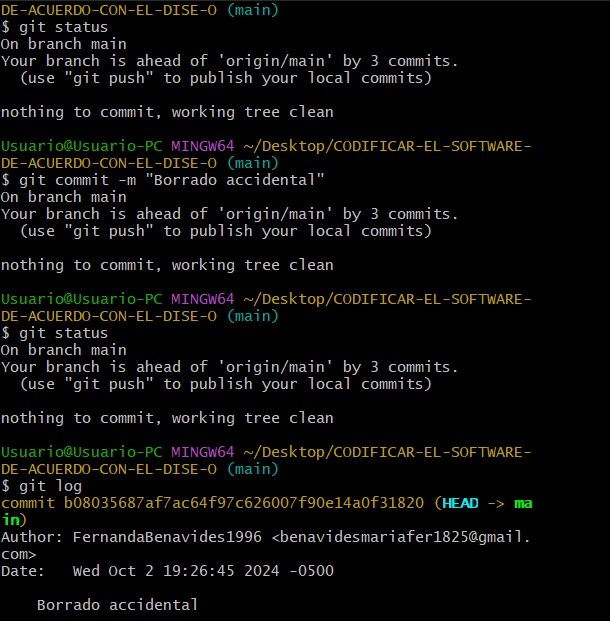


Git clean -f: eliminar archivos no rastreados del directorio de trabajo y la f es para forzar la eliminación. **Ejercicio 4**





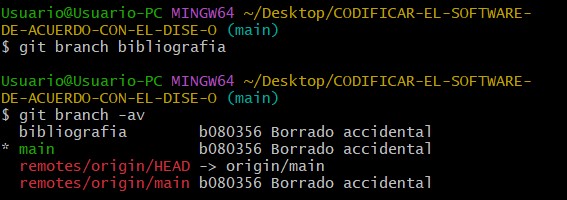






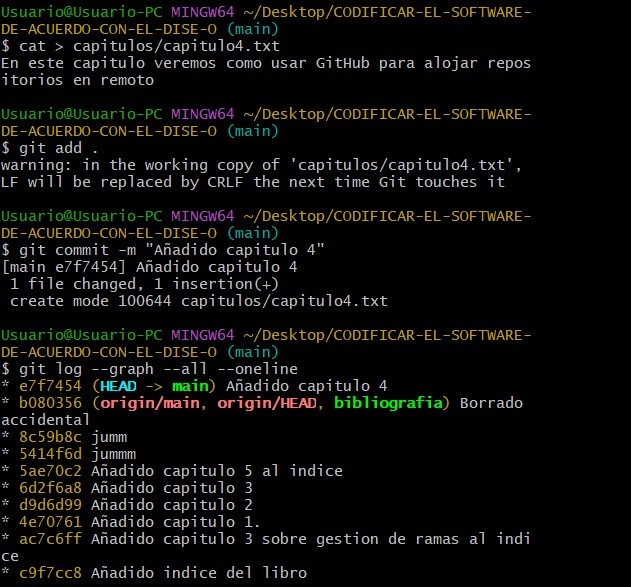
**Ramas**

# Ejercicio 1

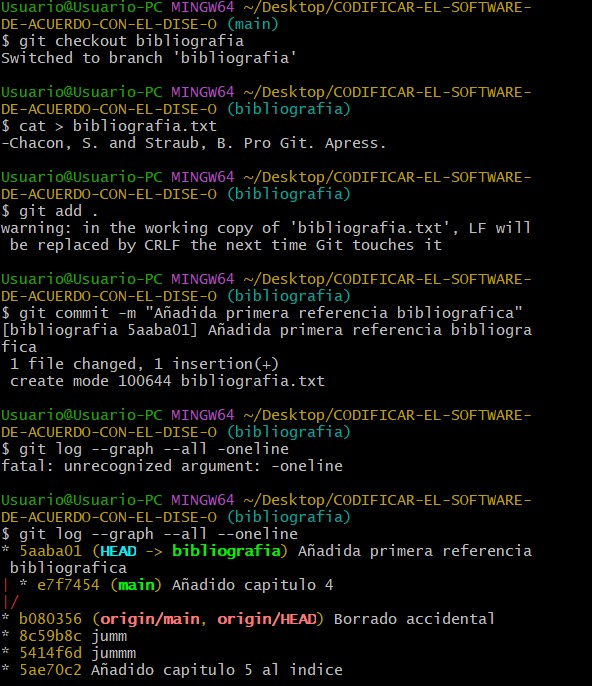


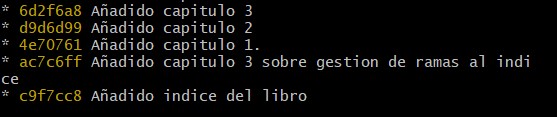
**Git branch: creación de ramas Git branch -av: mostrar una lista de todas las ramas en tu repositorio**

# Ejercicio 2

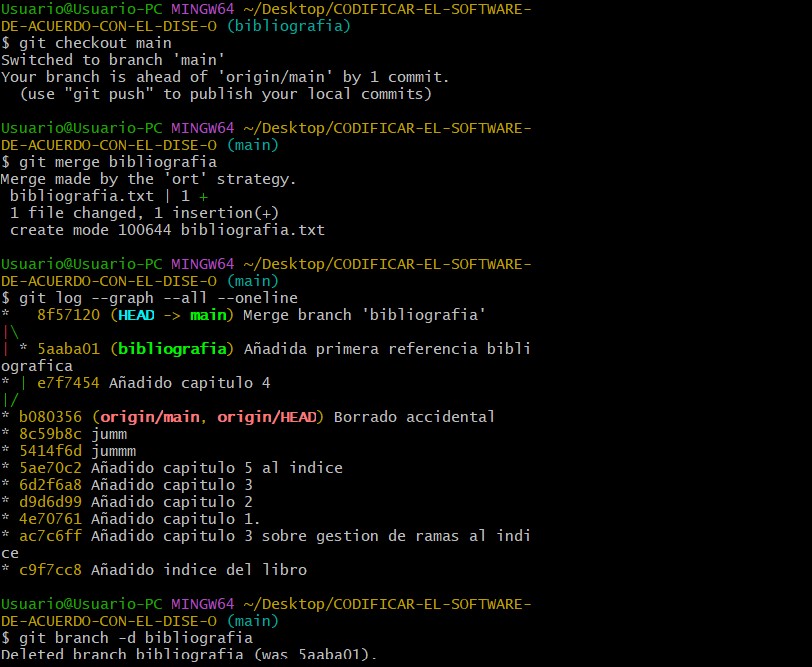


# Ejercicio 3



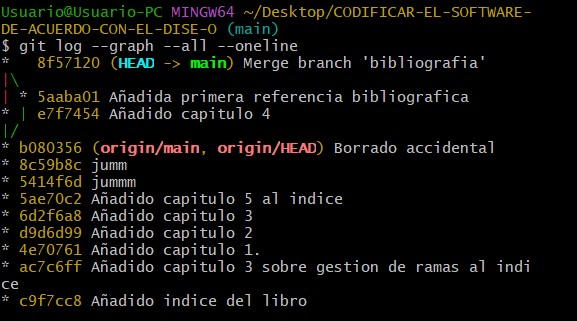


# Ejercicio 4

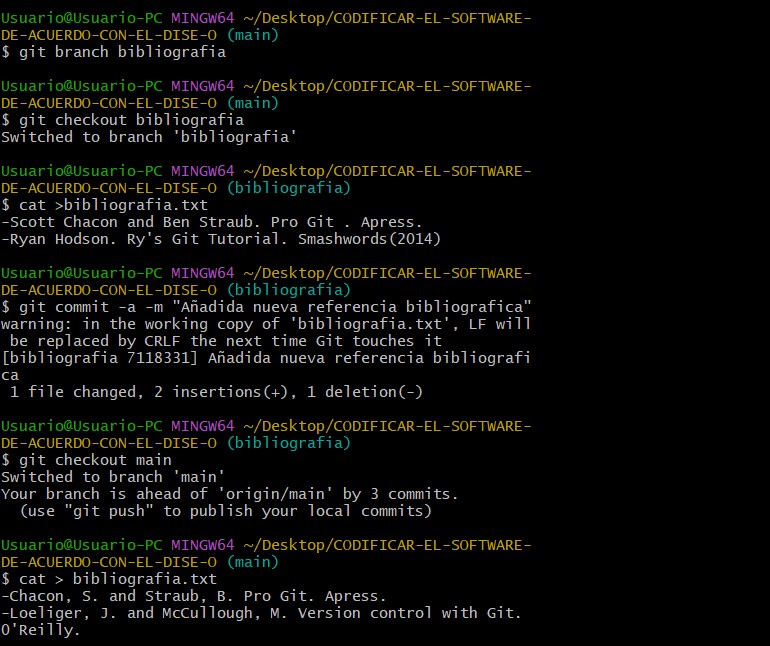


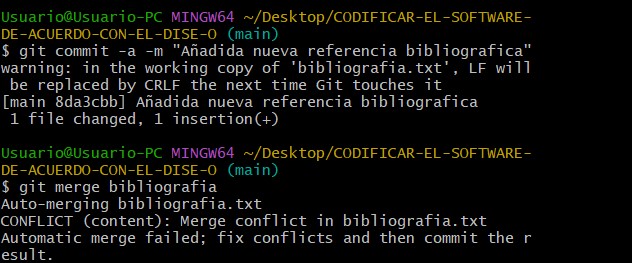
**Git merge: union de cambios entre ramas**

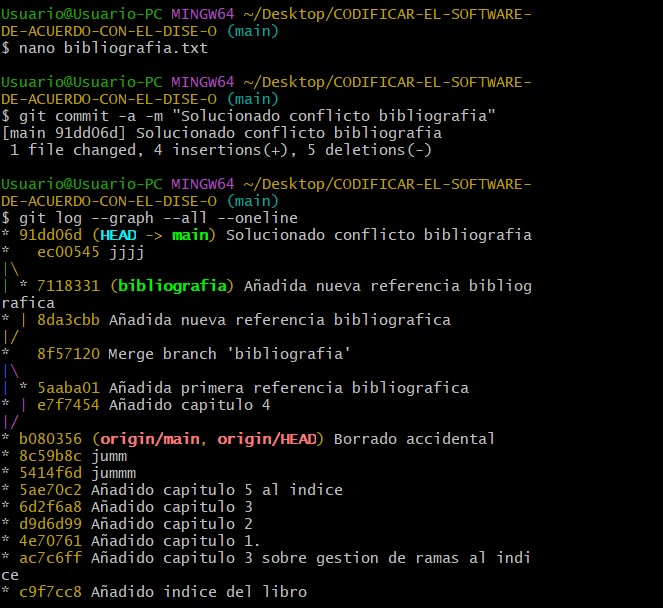
# Git branch -d: eliminación de ramas



# Ejercicio 5







**Repositorios remotos**

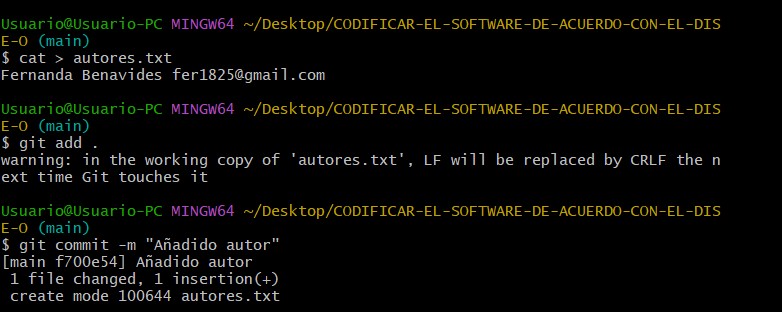
**Ejercicio 1**

**Git remote: gestiona y visualiza repositorios remotos cargados en la web**

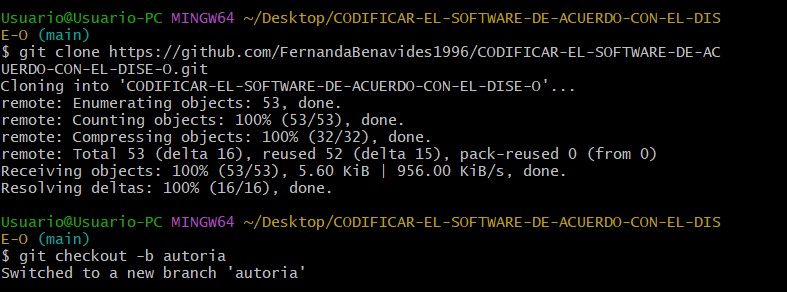
**Git remote -v : listado de repositorios con las respectiva url**

**Git push: empujar todo lo creado**

# ejercicio 3



# Ejercicio 4



# Git clone: clonacion del repositorio

