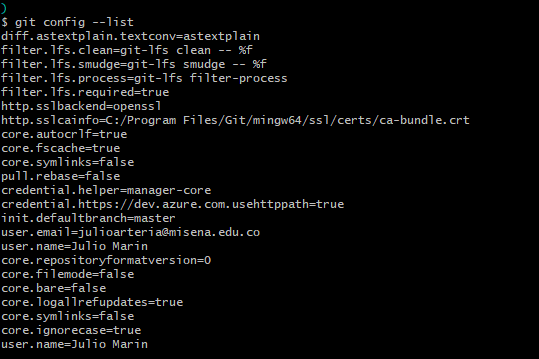
Taller de Git

**Nombre Aprendiz:**

Creación y actualización de repositorios:

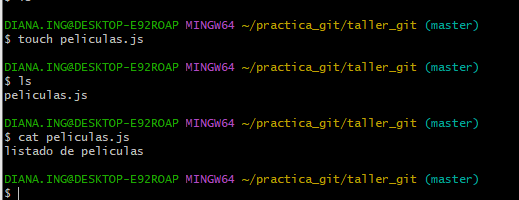
Ejercicio 1:

Configurar Git definiendo el nombre del usuario y el correo electrónico.



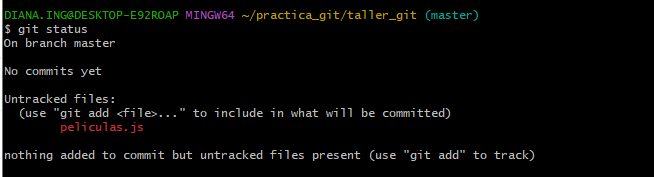
Ejercicio 2:

Crear un repositorio nuevo con el nombre Películas y mostrar su contenido usando la consola de comandos.



Ejercicio 3:

1. Comprobar el estado del repositorio

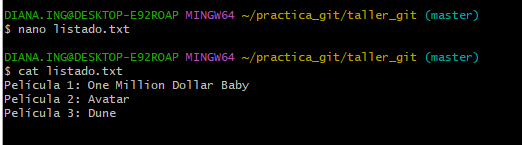


1. Crear un archivo de texto con nombre listado.txt con el siguiente contenido:

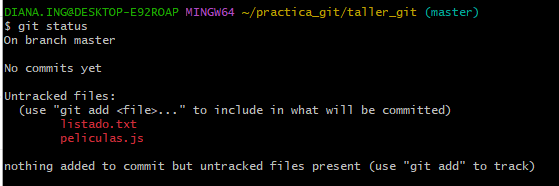
Película 1: One Million Dollar Baby

Película 2: Avatar

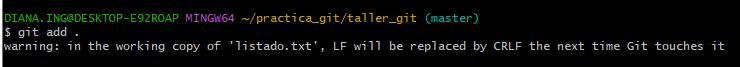
Película 3: Dune



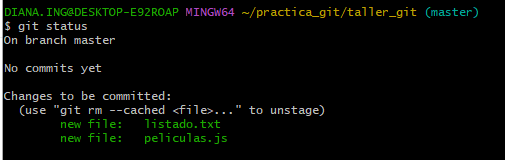
1. Comprobar de nuevo el estado del repositorio.



1. Agregar el archivo a la zona de intercambio temporal.

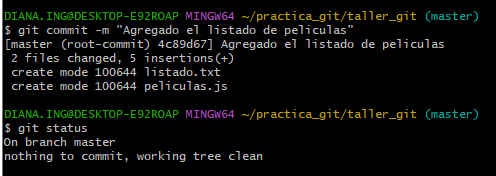


1. Volver a comprobar una vez más el estado del repositorio.



Ejercicio 4:

Realizar un commit de los últimos cambios con el mensaje “Agregado el listado de películas” y ver el estado del repositorio.



Ejercicio 5:

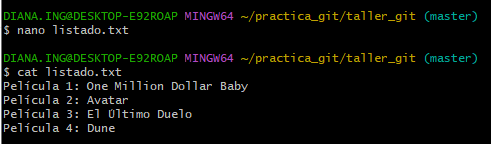
1. Cambiar el archivo listado.txt para que contenga lo siguiente:

Película 1: One Million Dollar Baby

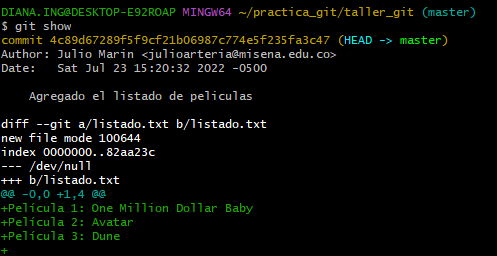
Película 2: Avatar

Película 3: El Último Duelo

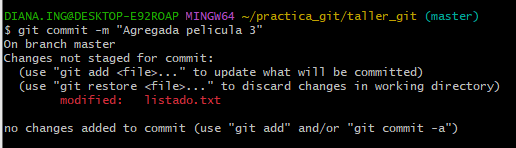
Película 4: Dune



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

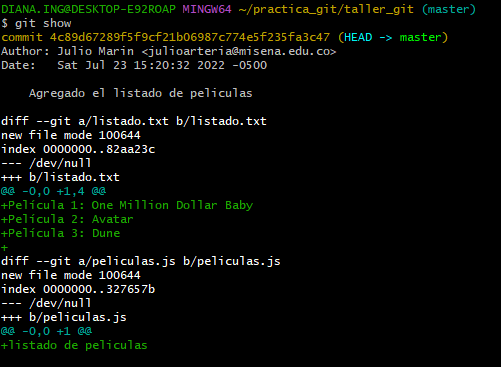


1. Hacer un commit de los cambios con el mensaje “Agregada película 3”.



Ejercicio 6:

1. Mostrar los cambios de la última versión del repositorio con respecto a la anterior. (git diff)



1. Cambiar el mensaje del último commit por “Agregada película 3 sobre caballeros y batallas”

($ git commit –amend) cambia el ultimo commit

1. Volver a mostrar los últimos cambios del repositorio.

($git diff)

Ejercicios de manejo del historial de cambios

Ejercicio 1:

1. Mostrar el historial de cambios del repositorio
2. Crear la carpeta capítulos y crear dentro de ella el archivo listado.txt vacío y el archivo capitulo1.txt con el siguiente texto.

*Git es un sistema de control de versiones ideado por Linus Torvalds.*

1. Agregar los cambios a la zona de intercambio temporal o Staging Area.
2. Hacer un commit de los cambios con el mensaje “Agregado índice y capítulo 1”.
3. Volver a mostrar el historial de cambios del repositorio

Ejercicio 2

1. Crear el archivo capitulo2.txt en la carpeta capítulos con el siguiente texto:

*El flujo de trabajo básico con Git consiste en: 1- Hacer cambios en el repositorio. 2- Agregar los cambios a la zona de intercambio temporal. 3- Hacer un commit de los cambios.*

1. Agregar los cambios a la zona de intercambio temporal.
2. Hacer un commit de los cambios con el mensaje “Agregado capítulo 2.”
3. Mostrar las diferencias entre la última versión y dos versiones anteriores.

Ejercicio 3:

1. Crear el archivo capitulo3.txt en la carpeta capítulos con el siguiente texto:

*Git permite la creación de ramas lo que permite tener distintas versiones del mismo proyecto y trabajar de manera simultánea en ellas.*

1. Agregar los cambios a la zona de intercambio temporal.
2. Hacer un commit de los cambios con el mensaje “Agregado capítulo 3.”
3. Mostrar las diferencias entre la primera y la última versión del repositorio.

Ejercicio 4:

1. Agregar al final del archivo listado.txt la siguiente línea:

*Película 5: Tiempos Violentos*

1. Agregar los cambios a la zona de intercambio temporal.
2. Hacer un commit de los cambios con el mensaje “Agregado capítulo 5 al listado.”.
3. Mostrar quién ha hecho cambios sobre el archivo listado.txt.

Ejercicios de deshacer cambios

Ejercicio 1:

1. Eliminar la última línea del archivo listado.txt y guardarlo.
2. Comprobar el estado del repositorio.
3. Deshacer los cambios realizados en el archivo listado.txt para volver a la versión anterior del archivo.
4. Volver a comprobar el estado del repositorio.

Ejercicio 2:

1. Eliminar la última línea del archivo listado.txt y guardarlo.
2. Agregar los cambios a la zona de intercambio temporal.
3. Comprobar de nuevo el estado del repositorio.
4. Quitar los cambios de la zona de intercambio temporal, pero mantenerlos en el directorio de trabajo.
5. Comprobar de nuevo el estado del repositorio.
6. Deshacer los cambios realizados en el archivo listado.txt para volver a la versión anterior del archivo.
7. Volver a comprobar el estado del repositorio.

Ejercicio 3:

1. Eliminar la última línea del archivo listado.txt y guardarlo.
2. Eliminar el archivo capitulos/capitulo3.txt.
3. Agregar un archivo nuevo captitulos/capitulo4.txt vacío.
4. Agregar los cambios a la zona de intercambio temporal.
5. Comprobar de nuevo el estado del repositorio.
6. Quitar los cambios de la zona de intercambio temporal, pero mantenerlos en el directorio de trabajo.
7. Comprobar de nuevo el estado del repositorio.
8. Deshacer los cambios realizados para volver a la versión del repositorio.
9. Volver a comprobar el estado del repositorio.

Ejercicio 4:

1. Eliminar la última línea del archivo listado.txt y guardarlo.
2. Eliminar el archivo capítulos/capitulo3.txt.
3. Agregar los cambios a la zona de intercambio temporal y hacer un commit con el mensaje “Borrado accidental.”
4. Comprobar el historial del repositorio.
5. Deshacer el último commit pero mantener los cambios anteriores en el directorio de trabajo y la zona de intercambio temporal.
6. Comprobar el historial y el estado del repositorio.
7. Volver a hacer el commit con el mismo mensaje de antes.
8. Deshacer el último commit y los cambios anteriores del directorio de trabajo volviendo a la versión anterior del repositorio.
9. Comprobar de nuevo el historial y el estado del repositorio.

Ejercicios de Ramas

Ejercicio 1:

Crear una nueva rama bibliografía y mostrar las ramas del repositorio.

Ejercicio 2

1. Crear el archivo capitulos/capitulo4.txt y Agregar el texto siguiente

*En este capítulo veremos cómo usar GitHub para alojar repositorios en remoto.*

1. Agregar los cambios a la zona de intercambio temporal.
2. Hacer un commit con el mensaje “Añadido capítulo 4.”
3. Mostrar la historia del repositorio incluyendo todas las ramas.

Ejercicio 3

1. Cambiar a la rama bibliografía.
2. Crear el archivo bibliografia.txt y Agregar la siguiente referencia

*Chacon, S. and Straub, B. Pro Git. Apress.*

1. Agregar los cambios a la zona de intercambio temporal.
2. Hacer un commit con el mensaje “Añadida primera referencia bibliográfica”.
3. Mostrar la historia del repositorio incluyendo todas las ramas.

Ejercicio 4:

1. Fusionar la rama bibliografía con la rama master.
2. Mostrar la historia del repositorio incluyendo todas las ramas.
3. Eliminar la rama bibliografía.
4. Mostrar de nuevo la historia del repositorio incluyendo todas las ramas.

Ejercicio 5:

1. Crear la rama bibliografia.
2. Cambiar a la rama bibliografia.
3. Cambiar el archivo bibliografia.txt para que contenga las siguientes referencias:

*Scott Chacon and Ben Straub. Pro Git. Apress.*

*Ryan Hodson. Ry’s Git Tutorial. Smashwords (2014)*

1. Agregar los cambios a la zona de intercambio temporal y hacer un commit con el mensaje “Añadida nueva referencia bibliográfica.”
2. Cambiar a la rama master.
3. Cambiar el archivo bibliografia.txt para que contenga las siguientes referencias:

*Chacon, S. and Straub, B. Pro Git. Apress.*

*Loeliger, J. and McCullough, M. Version control with Git. O’Reilly.*

1. Agregar los cambios a la zona de intercambio temporal y hacer un commit con el mensaje “Añadida nueva referencia bibliográfica.”
2. Fusionar la rama bibliografia con la rama master.
3. Resolver el conflicto dejando el archivo bibliografia.txt con las referencias:

*Chacon, S. and Straub, B. Pro Git. Apress.*

*Loeliger, J. and McCullough, M. Version control with Git. O’Reilly.*

*Hodson, R. Ry’s Git Tutorial. Smashwords (2014)*

1. Agregar los cambios a la zona de intercambio temporal y hacer un commit con el mensaje “Resuelto conflicto de bibliografía.”
2. Mostrar la historia del repositorio incluyendo todas las ramas.