Actividades Clase n°6:

**Actividad n°1:**

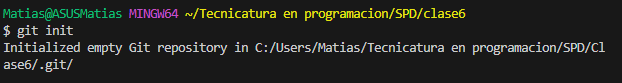
**Responder Cuestionario para asistencia (Aula del Campus)**

**Actividad n°2:(Trabajo Grupal) Ver consignas al final**

A)Realizar la siguiente actividad utilizando comandos vistos en la clases anteriores en la terminal Git Bach

1- Crear un nuevo repositorio Git con el nombre Clase6 desde la terminal Git Bach

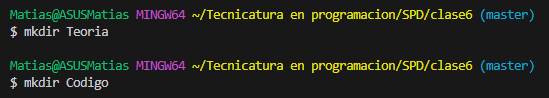
2-iniciar repositorio

3-configurar repositorio

4- crear 2 carpetas de nombre :

-Teoría

- Código

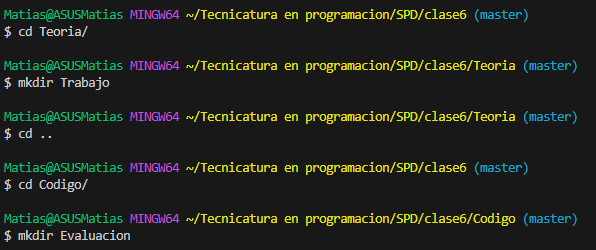


5-abrimos carpeta Teoría y creamos una subcarpeta:

- Trabajo

luego abrimos la carpeta Código y creamos una subcarpeta:

-Evaluación.



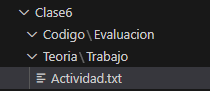
B)Realizar la siguiente actividad utilizando los comandos vistos en la terminal Git Bash y en el editor de texto que utilicen.

1- Abrimos el editor de texto, creamos un nuevo documento con el nombre :

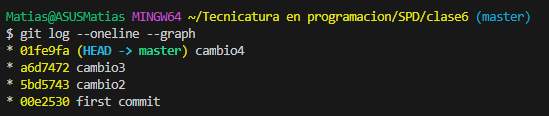
-Actividad

Guardamos documento en nuestra carpeta trabajo.

Trabajo <-Actividad



2- Realizamos cuatro modificaciones en el archivo y commiteamos cada cambio realizado utilizando comandos vistos.

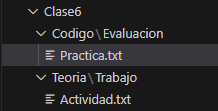


3- Creamos otro documento con el nombre en nuestro editor de texto con el nombre:

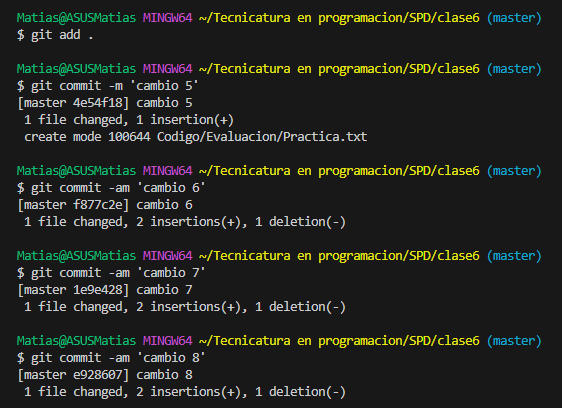
-Practica

Guardamos el documento Practica en la carpeta Evaluación

Evaluación <- Practica



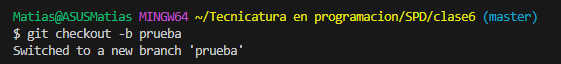
4- realizamos 4 modificaciones en el documento Práctica, añadimos cambios y commit por separado



C)Realizar la siguiente actividad utilizando los comandos vistos las clases anteriores:

1- Creamos una rama auxiliar con el nombre :

-Prueba



2- Creamos una nueva carpeta con el nombre :

-Integrador

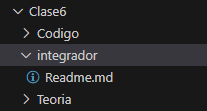


3- Abrimos un documento en nuestro editor de texto con el nombre :

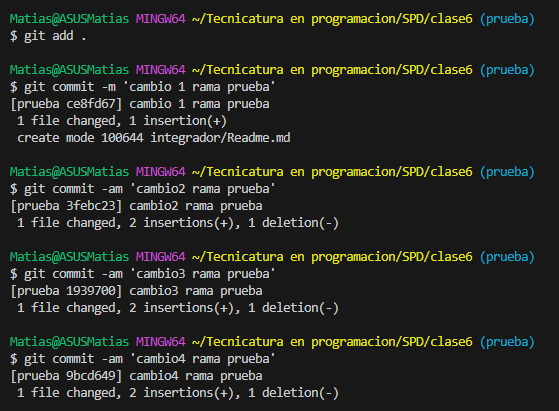
-Readme

Guardamos documento en nuestra carpeta de nombre Integrador

Integrador <- Readme

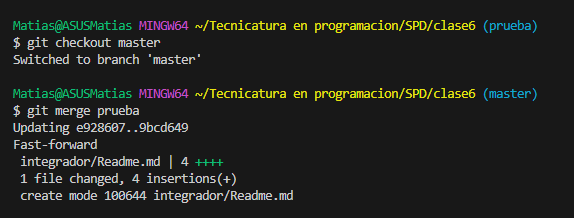


4- Realizamos 4 modificaciones en nuestro documento Readme, añadimos cada modificación y commit por separado.



5 - Fusionamos Rama Prueba con nuestra Rama Clase6 (Master o Main)

6- Tomar captura de la fusión de las ramas.



D) Responder las siguientes preguntas : desarrollar y utilizar comandos vistos ( se puede acompañar de capturas )​

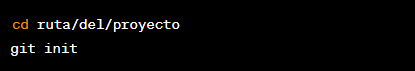
**1- ¿Cómo creamos una carpeta desde Git Bash?** ​

Para crear una carpeta desde Git Bash, puedes utilizar el comando mkdir seguido del nombre de la carpeta que deseas crear. Por ejemplo, si deseas crear una carpeta llamada "mi\_carpeta", puedes ejecutar el siguiente comando:



**2- ¿Cómo iniciamos nuestro Repositorio Git?​**

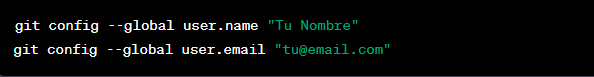
Para iniciar un repositorio Git, debes navegar hasta la ubicación de tu proyecto en Git Bash utilizando el comando cd (change directory) y luego ejecutar el comando git init.



**3- ¿Qué comando utilizamos al configurar usuario y email ? ​**

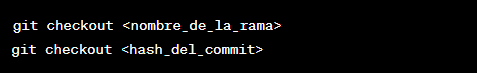
El comando utilizado para configurar el nombre de usuario y el correo electrónico en Git es:

Debes reemplazar "Tu Nombre" con tu nombre y "tu@email.com" con tu dirección de correo electrónico.



**4- ¿Cuál es el comando que me permite viajar en el tiempo de los diferentes commit? ​**

El comando que te permite viajar en el tiempo y revisar diferentes commits es git checkout. Puedes utilizarlo para cambiar entre ramas o para desplazarte a commits específicos. Por ejemplo:



**5- ¿A qué llamamos Rama Master o Main?​**

La rama "Master" o "Main" se refiere a la rama principal de un repositorio Git. Es la rama predeterminada creada cuando se inicializa un repositorio y suele ser utilizada como la rama principal donde se encuentran las versiones estables del código.

**6- ¿Cuál es la diferencia entre Rama Auxiliar y Rama Master o Main?** ​

La diferencia entre una rama auxiliar y la rama "Master" o "Main" es que la rama "Master" o "Main" es la rama principal del repositorio y suele contener las versiones estables del código. Las ramas auxiliares se crean a partir de la rama principal y se utilizan para desarrollar nuevas características o solucionar problemas sin afectar directamente la rama principal. Una vez que las modificaciones en una rama auxiliar son probadas y consideradas estables, pueden fusionarse con la rama principal.

**7- ¿Qué comando utilizamos para cambiar de una rama a otra?​**

Para cambiar de una rama a otra en Git, se utiliza el comando git checkout seguido del nombre de la rama a la que deseas cambiar. Por ejemplo:



**8- ¿Se puede modificar la Rama Master?**

En Git, la rama "Master" o "Main" es la rama principal y por defecto no se puede modificar directamente. Sin embargo, puedes crear nuevas ramas a partir de la rama principal y realizar modificaciones en esas ramas. Luego, si deseas incorporar esas modificaciones a la rama principal, puedes fusionar las ramas utilizando el comando git merge. ​

**9- ¿Cuál es el comando para crear una nueva rama?​**

El comando para crear una nueva rama en Git es git branch seguido del nombre de la nueva rama que deseas crear. Por ejemplo:



Esto creará una nueva rama llamada "nueva\_ rama" basada en la rama actual en la que te encuentras.

**10- Mencionar 3 funcionalidades que tiene el comando Git Checkout ​**

* Cambiar de rama: Puedes utilizar git checkout <nombre\_de\_la\_rama> para cambiar de una rama a otra.
* Deshacer cambios locales: Si has realizado modificaciones en tus archivos pero aún no has confirmado los cambios, puedes utilizar git checkout -- <nombre\_del\_archivo> para descartar los cambios y restaurar el archivo a su estado anterior.
* Viajar en el tiempo: Como se mencionó anteriormente, puedes utilizar git checkout <hash\_del\_commit> para desplazarte a un commit específico en el historial del repositorio.

**11- ¿Cuál es el comando que me permite ver de manera gráfica los commit y las ramas creadas? ​**

El comando que te permite ver de manera gráfica los commits y las ramas creadas es git log con la opción --graph. Puedes ejecutar el siguiente comando



O puede ser también de la siguiente manera:



Esto mostrará un gráfico con las ramas y los commits en tu repositorio.

**12- Realizar una captura del último punto realizado, se puede utilizar captura de la actividad n° 4​.**