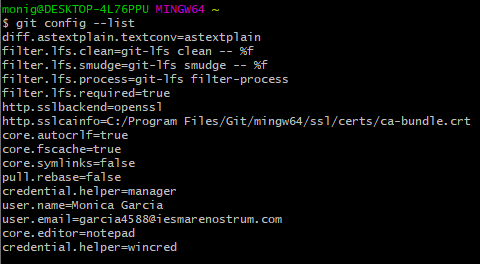
**1. Fundamentos Git I**

**1. Instala Git en tu sistema operativo. Haz una captura de pantalla en la que aparezca el resultado de la ejecución del comando git --version.**

****

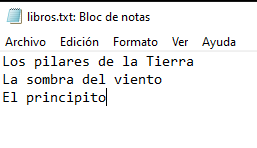
**2. Realiza la configuración de Git según lo indicado en el tema (nombre, correo electrónico, editor de preferencia y almacenamiento de credenciales). Haz una captura de pantalla con el resultado de la ejecución de los comandos de configuración**

**3. Crea una carpeta denominada S1R1. Realiza las siguientes acciones en ella:**

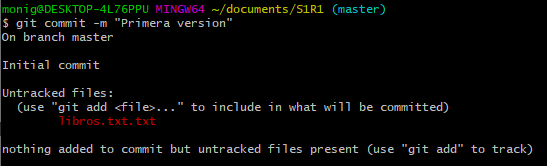
**a) Crea un repositorio Git.**

****

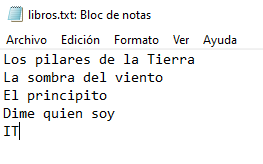
**b) Crea un fichero denominado libros.txt. Añade tres títulos de libros cada uno en una línea distinta.**

****

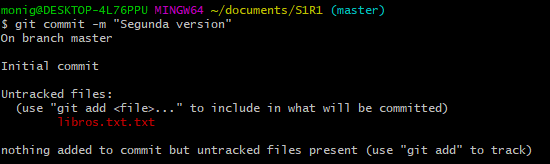
**c) Haz un primer commit.**

****

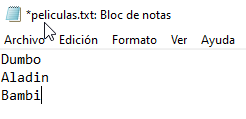
**d) Añade dos libros al archivo libros.txt.**

****

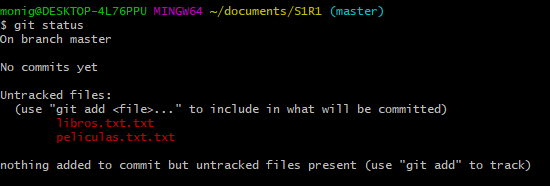
**e) Haz un segundo commit.**

****

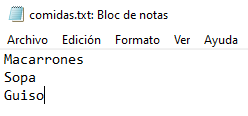
**f ) Crea un fichero denominado peliculas.txt. Añade tres títulos de películas a dicho archivo.**

****

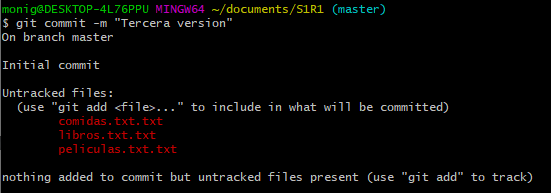
**g) Haz una captura de pantalla del comando git status.**

****

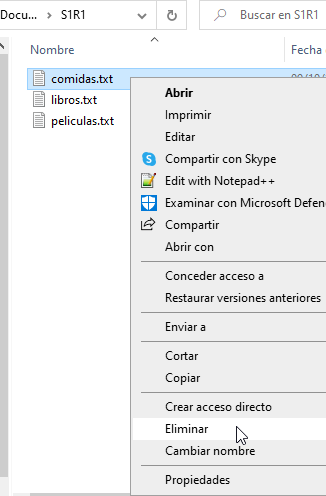
**h) Crea un fichero denominado comidas.txt. Añade tres nombres de comidas a dicho archivo.**

****

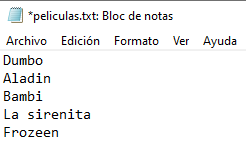
**i) Haz un tercer commit que incluya los archivos peliculas.txt y comidas.txt.**



**j) Elimina el archivo comidas.txt desde el navegador de archivos.**

****

**k) Añade dos películas más al archivo peliculas.txt.**

****

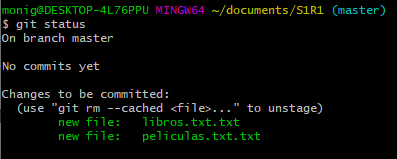
**l) Haz una captura de pantalla que muestre los cambios en el directorio de Ptrabajo.**

****

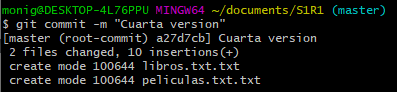
**m) Añade los cambios al área de preparación.**

****

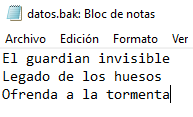
**n) Haz una captura de pantalla del comando git status. Debe indicar que se ha borrado el archivo comidas.txt y que se ha modificado el archivo peliculas.txt.**

****

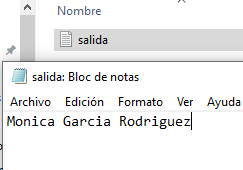
**ñ) Haz un cuarto commit.**

****

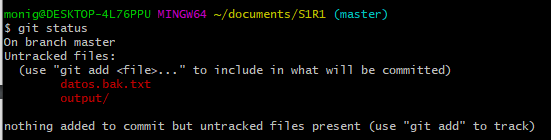
**o) Crea un archivo denominado datos.bak. Añade tres títulos de libros a dicho archivo.**

****

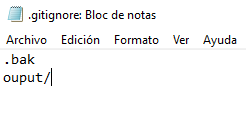
**p) Crea una subcarpeta denominada output. Crea un archivo denominado salida.txt en su interior. Escribe tu nombre y apellidos en dicho archivo.**

****

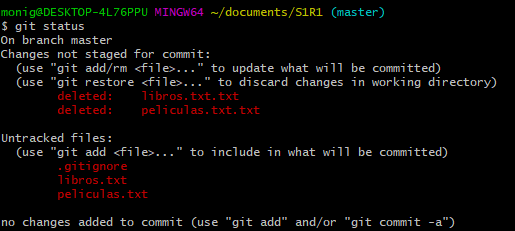
**q) Haz una captura de pantalla del comando git status. Deben aparecer los archivos datos.bak y output/salida.txt como archivos nuevos.**

****

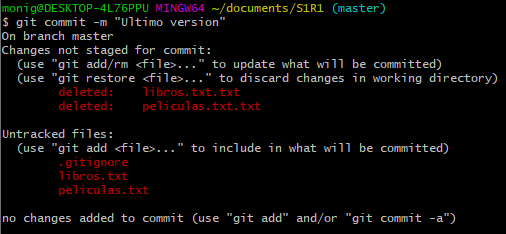
**r ) Crea un archivo .gitignore para que los ficheros con extensión .bak y el contenido de la carpeta output/ no se incluyan en el repositorio.**

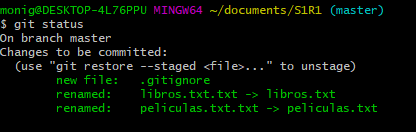
****

**s) Haz una nueva captura de pantalla del comando git status. Ahora no deben aparecer los archivos datos.bak y output/salida.txt como archivos nuevos, sino que en su lugar debe aparecer únicamente el archivo .gitignore.**

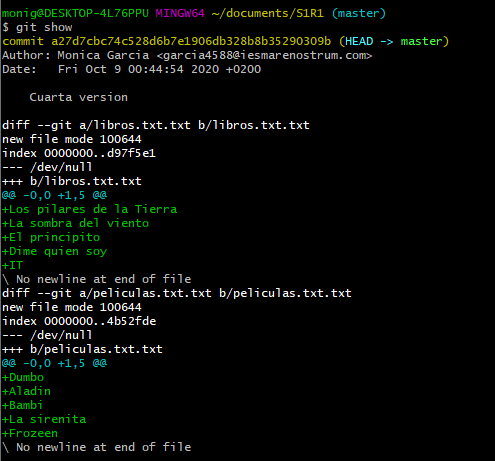
****

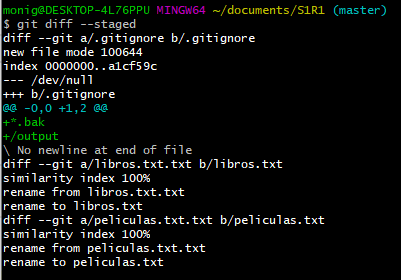
**t) Haz un último commit para incluir el archivo .gitignore en el repositorio.**

****

****

**u) Haz una captura de pantalla que muestre el histórico de cambios del repositorio**

****

****

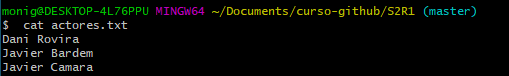
**2. Fundamentos Git II**

**1. Crea una carpeta denominada S2R1. Realiza las siguientes acciones en ella:**

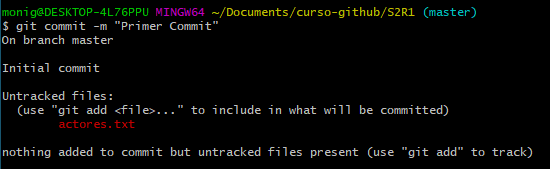
**a) Crea un repositorio Git.**

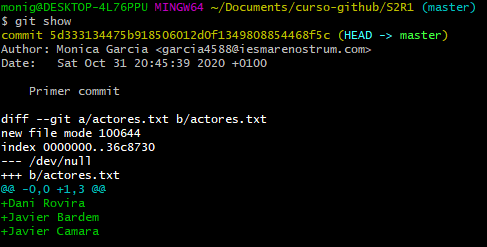
****

**b) Crea un fichero denominado actores.txt. Añade tres nombres de actores cada no en una línea distinta.**

****

**c) Haz un primer commit.**

****

****

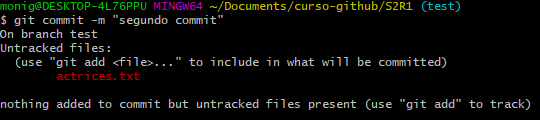
**d) Crea una rama denominada test.**

****

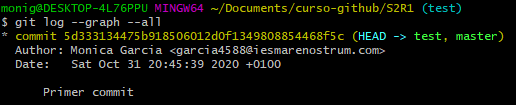
**e) En la rama test crea un fichero denominado actrices.txt. Añade tres nombres de actrices y realiza un commit en dicha rama.**

****

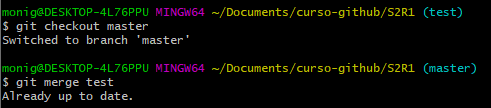
****

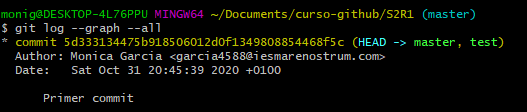
****

**f) Haz una captura de pantalla del resultado del comando git log –graph –all**

****

**g) Incorpora los cambios de la rama test a la rama master. Haz una captura de pantalla de los comandos que has utilizado y de su resultado.**

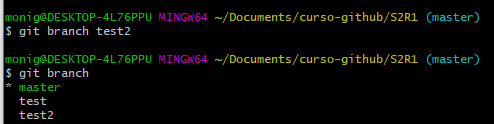
****

****

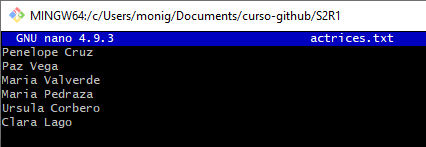
**h) Posiciónate en la rama master.**

****

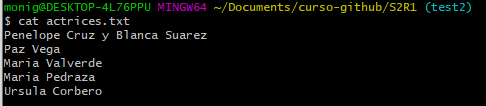
**i) Crea una segunda rama denominada test2**

****

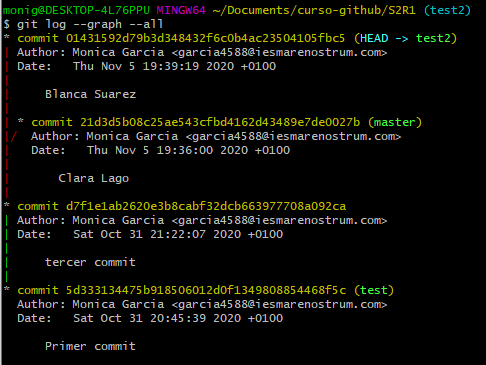
**j) En la rama master, añade una actriz al fichero actrices.txt**

****

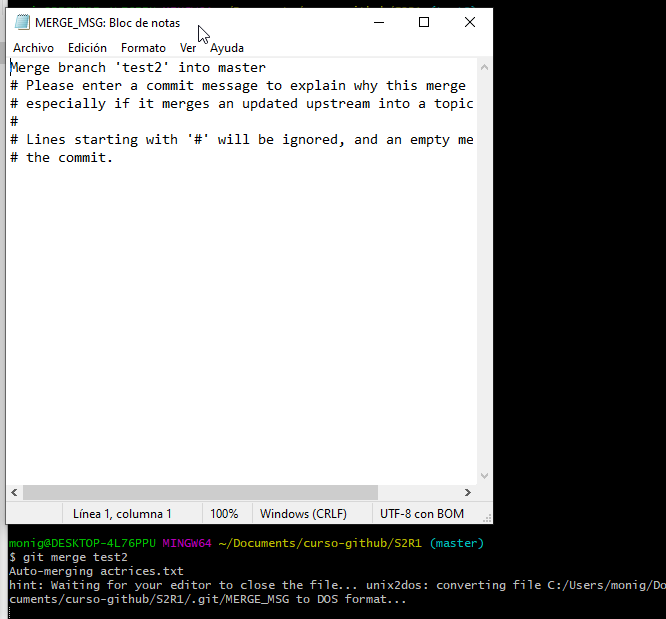
**k) En la rama test2, añade una actriz al fichero actrices.txt**

****

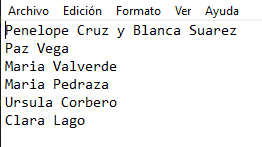
**l) Haz una captura de pantalla del resultado del comando git log –graph –all. Debe haber dos caminos distintos: uno para la rama master y otro para la rama test2.**

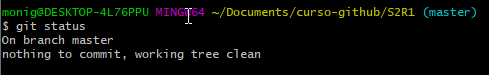
****

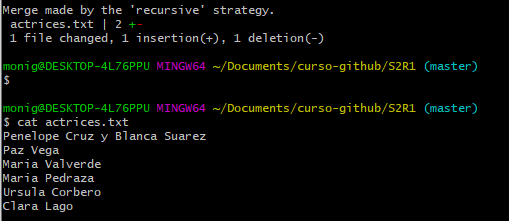
**m) Incorpora los cambios de la rama test2 a la rama master. ¿Se produce un conflicto? De ser así realiza una captura del comando git status.**

****

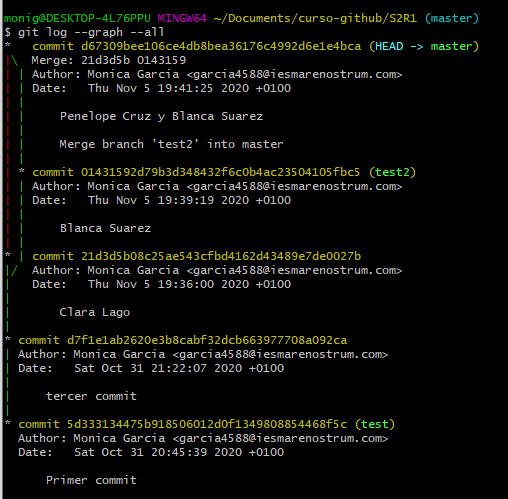
**n) Resuelve el conflicto incorporando los dos nombres de actrices**

****

****

****

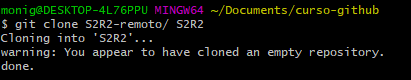
**ñ) Haz una captura de pantalla del resultado del comando git log –graph –all. Observa que se ha creado un nuevo commit que integra los dos cambios anteriores.**

****

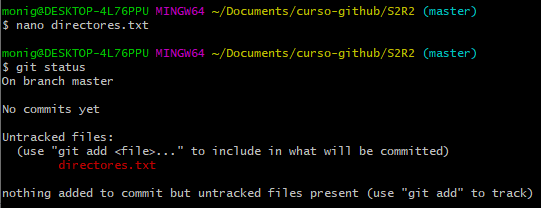
**2. Crea una carpeta denominada S2R2-remoto. Inicializa un repositorio Git en su interior mediante el comando git init –bare. Esta carpeta se utilizará como repositorio remoto.**

****

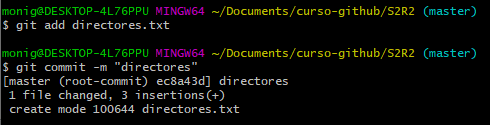
**3. Clona el repositorio S2R2-remoto en una carpeta denominada S2R2. Haz una captura de pantalla del resultado del comando de clonado. A continuación realiza las siguientes acciones en el repositorio S2R2.**

****

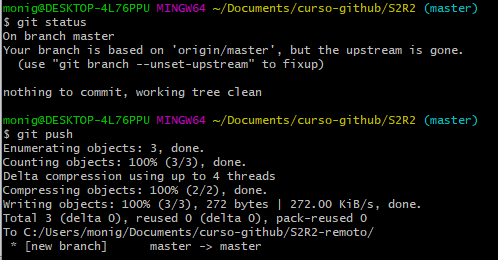
1. **Crea un archivo denominado directores.txt. Añade el nombre de tres directores de cine.**

****

1. **Haz un commit.**

****

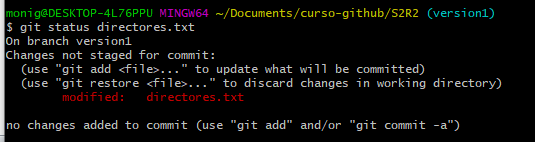
1. **Realiza un push a repositorio remoto. Haz una captura de pantalla del resultado.**

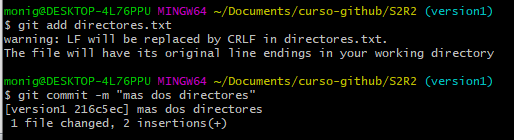
****

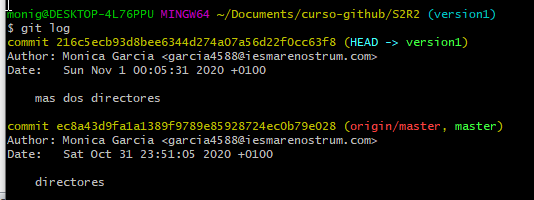
1. **Crea una rama versión1.**

****

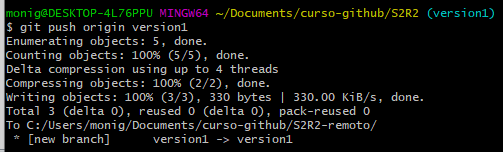
1. **En la rama versión1 añade el nombre de dos directores de cine más el archivo directores.txt.**

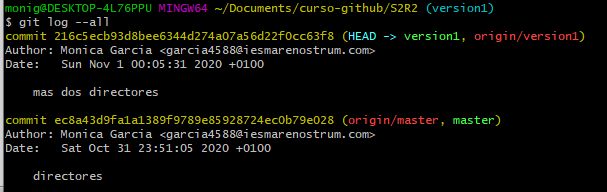
****

****

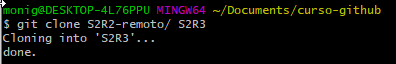
****

1. **Realiza un push de la rama el repositorio remoto de manera que quede asociada a la rama remota del mismo nombre. Haz una captura de pantalla del resultado.**

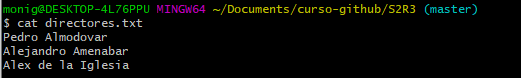
****

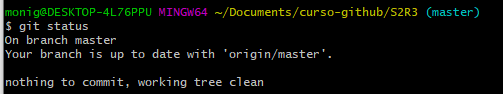
****

**4. Clona el repositorio S2R2-remoto en euna segunda carpeta denominada S2R3. Realiza las siguientes acciones sobre ella:**

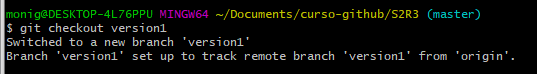
****

1. **Muestra en la consola el contenido del fichero directores.txt y el resultado del comando git status.**

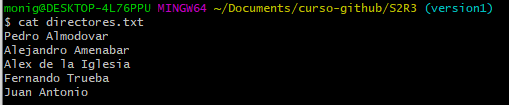
****

****

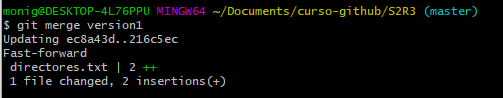
1. **Cambia a la rama version1. Muestra el resultado del comando.**

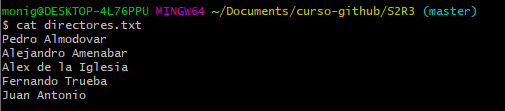


**c) Muestra el contenido del fichero directores.txt por la pantalla. Comprueba que se muestran los 5 nombres de directores esperados. Haz una captura de pantalla.**

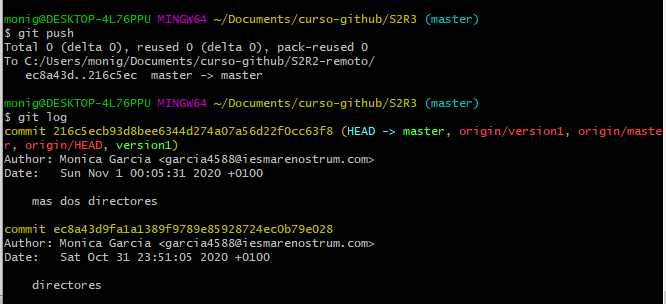
****

1. **Incorpora los cambios de la rama version1 a la rama master.**

****

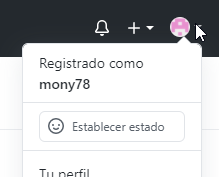
****

1. **Sube la rama master actualizada al servidor. Haz una captura de pantalla del resultado del comando.**

****

**3. GitHub**

**1. Crea una cuenta en GitHub**

****

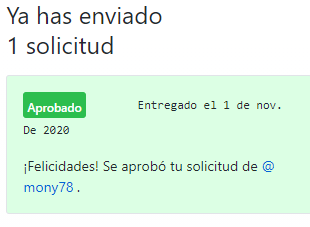
Ya tengo una cuenta

**2. Añade tu dirección de correo de educación**

****

**3. Solicita un descuento para uso educativo en https://education.github.**

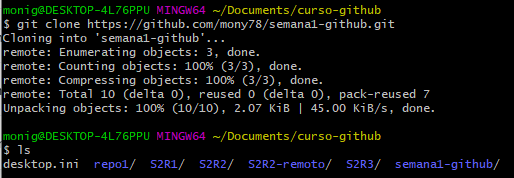
**com/discount\_requests/new**

****

**4. Haz un fork del repositorio localizado en la siguiente url: https://**

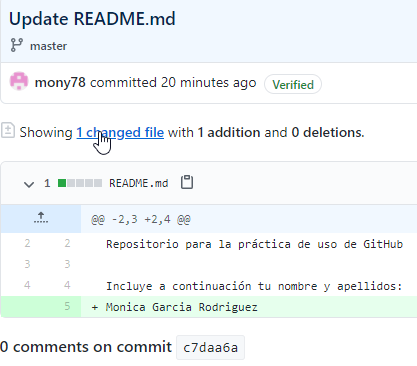
**github.com/interfacesweb-19-20/semana1-github**

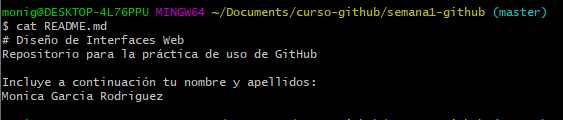
****

****

**a) Realiza un primer commit para poner tu nombre y apellidos en**

**el fichero README.md**

****

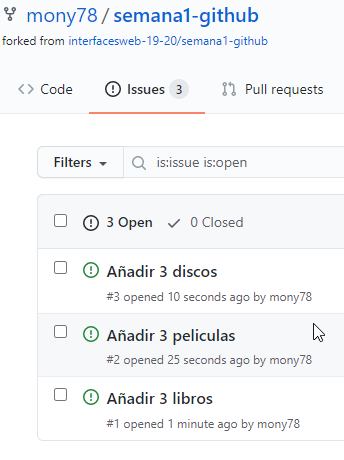
****

**b) Crea 3 issues con los siguientes títulos:**

**Añadir 3 libros**

**Añadir 3 películas**

**Añadir 3 discos**

****

**c) Crea una milestone denominada Tareas que contenga los 3 issues**

**creados**

****

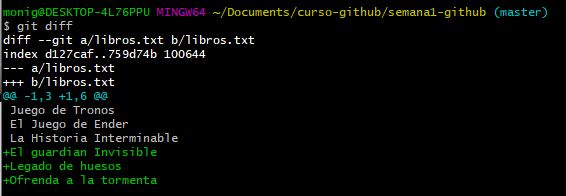
**d) Modifica los ficheros correspondientes y realiza 3 commits para**

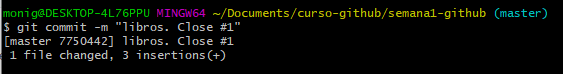
**realizar cada una de las tareas que se indican en los issues. El**

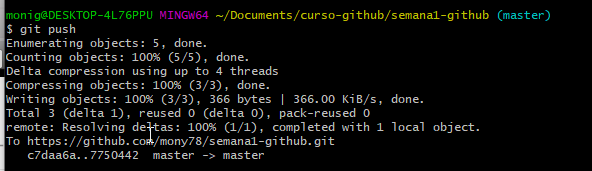
**mensaje del commit debe hacer que se cierren los issues correspondientes de manera automática.**

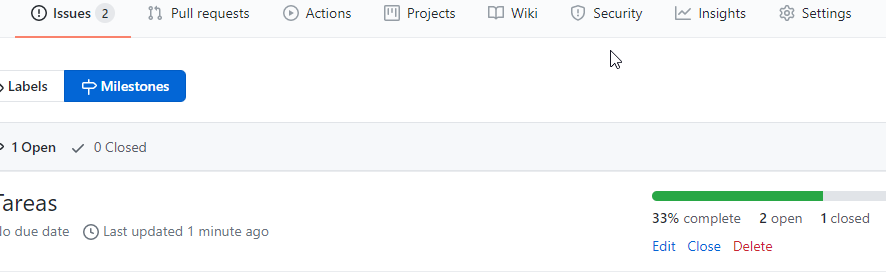
**e) Haz una captura de pantalla de los comandos que has utilizado**

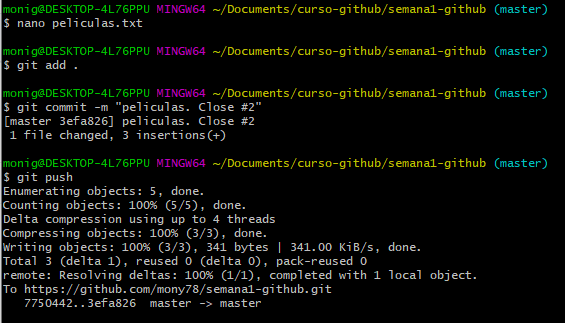
**para hacer los commits y subir los cambios a GitHub.**

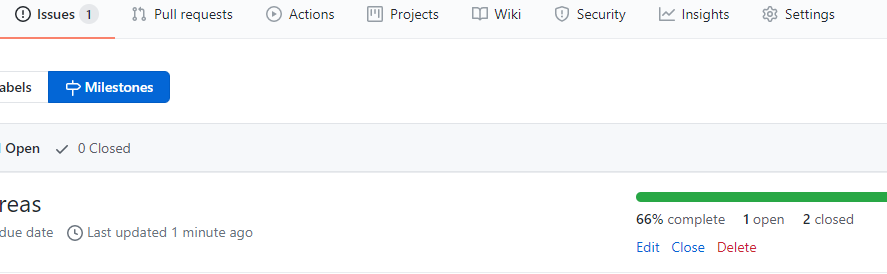
****

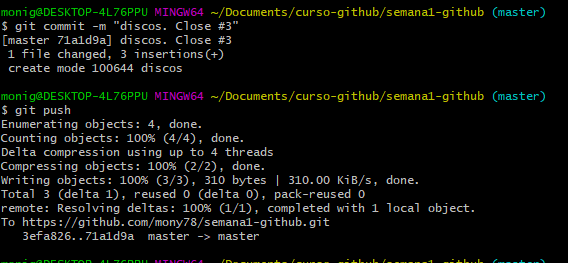
****

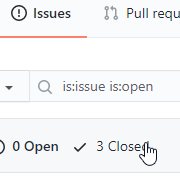
****

****

****

****

****

****

**f ) Incluye todas las capturas de pantalla en el repositorio dentro de la carpeta capturas. Añádelas también al repositorio de**

**manera que queden guardadas en tu repositorio en GitHub.**

**g) Realiza una pull request indicando en el mensaje que has completado la tarea.**