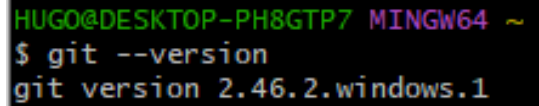


Práctica - Uso de Git y GitHub

Fundamentos de Git I

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

```
git --version
```



```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~
$ git --version
git version 2.46.2.windows.1
```

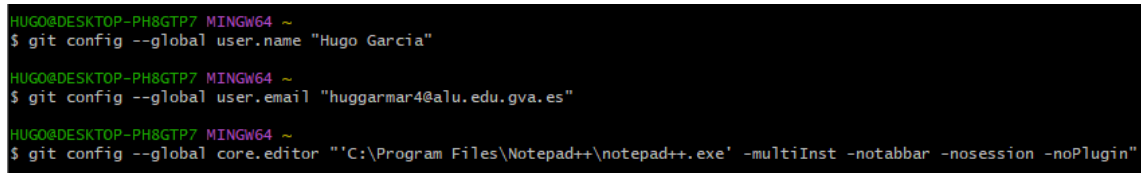
2. Realiza la configuración de Git según lo indicado en el tema (nombre, correo electrónico y editor de preferencia). Adjunta una captura de pantalla con el resultado de la ejecución de los comandos de configuración.

```
git config --global user.name "Hugo Garcia"
```

```
git config --global user.email "huggarmar4@alu.edu.gva.es"
```

```
git config --global core.editor "'C:\Program Files\Notepad++\notepad++.exe'
-multiInst -notabbar -nosession -noPlugin"
```

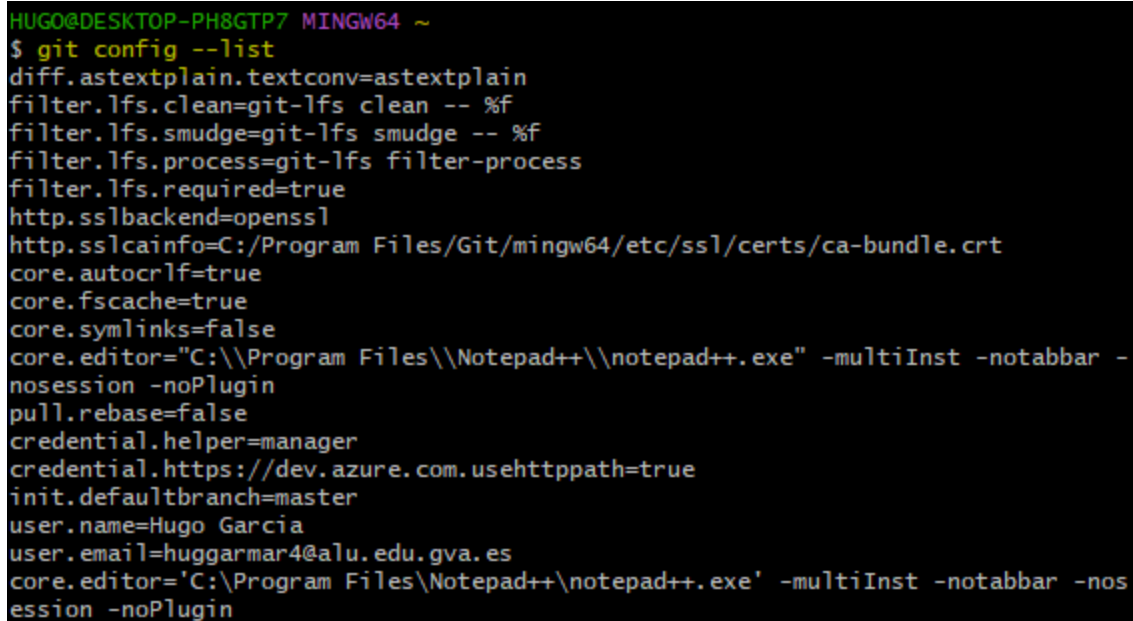
```
git config --list
```



```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~
$ git config --global user.name "Hugo Garcia"

HUGO@DESKTOP-PH8GTP7 MINGW64 ~
$ git config --global user.email "huggarmar4@alu.edu.gva.es"

HUGO@DESKTOP-PH8GTP7 MINGW64 ~
$ git config --global core.editor "'C:\Program Files\Notepad++\notepad++.exe' -multiInst -notabbar -nosession -noPlugin"
```



```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
core.editor="C:\\Program Files\\Notepad++\\notepad++.exe" -multiInst -notabbar -nosession -noPlugin
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=Hugo Garcia
user.email=huggarmar4@alu.edu.gva.es
core.editor='C:\Program Files\Notepad++\notepad++.exe' -multiInst -notabbar -nosession -noPlugin
```

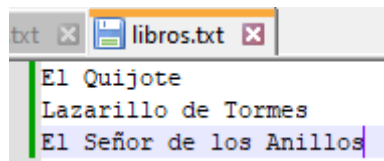
3. Crea una carpeta denominada S1R1. Realiza las siguientes acciones en ella:

- a) Crea un repositorio Git.

```
git init
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1
$ git init
Initialized empty Git repository in C:/Users/HUGO/Desktop/S1R1/.git/
```

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



```
libros.txt
El Quijote
Lazarillo de Tormes
El Señor de los Anillos
```

- c) Haz un primer commit.

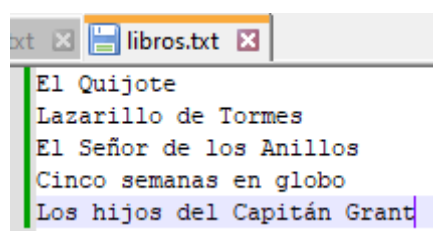
```
git add libros.txt
```

```
git commit -m "Primer commit"
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git add libros.txt

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git commit -m "Primer commit"
[master (root-commit) 1c5d515] Primer commit
1 file changed, 3 insertions(+)
create mode 100644 libros.txt
```

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



```
libros.txt
El Quijote
Lazarillo de Tormes
El Señor de los Anillos
Cinco semanas en globo
Los hijos del Capitán Grant
```

- e) Haz un segundo commit.

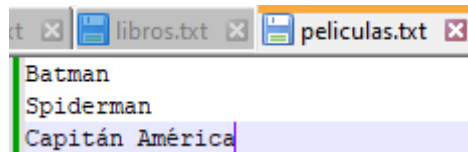
```
git add libros.txt
```

```
git commit -m "Segundo commit"
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git add libros.txt

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git commit -m "Segundo commit"
[master 907a339] Segundo commit
1 file changed, 3 insertions(+), 1 deletion(-)
```

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



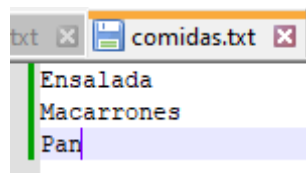
- g) Haz una captura de pantalla del comando git status.

git status

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    peliculas.txt

nothing added to commit but untracked files present (use "git add" to track)
```

- 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.

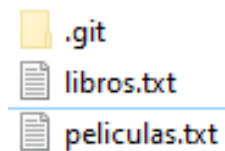
git add .

git commit -m "Tercer commit comidas.txt y peliculas.txt"

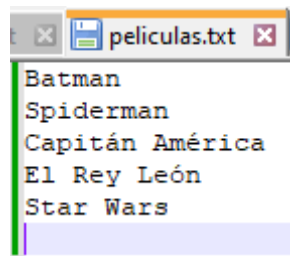
```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git add .

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git commit -m "Tercer commit comidas.txt y peliculas.txt"
[master 6b7cebe] Tercer commit comidas.txt y peliculas.txt
2 files changed, 6 insertions(+)
create mode 100644 comidas.txt
create mode 100644 peliculas.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 trabajo.

git status

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    comidas.txt
        modified:   peliculas.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

- m) Añade los cambios al área de preparación.

git add.

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git add .
```

- 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.

git status

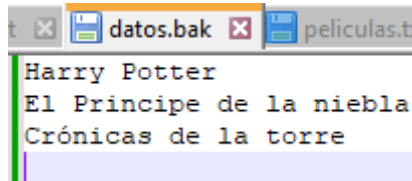
```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    comidas.txt
        modified:   peliculas.txt
```

- o) Haz un cuarto commit.

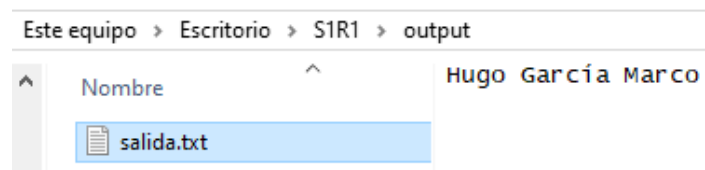
git commit -m "Cuarto commit, eliminar y modificar"

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git commit -m "Cuarto commit - eliminar y modificar"
[master 6964b6e] Cuarto commit - eliminar y modificar
2 files changed, 2 insertions(+), 3 deletions(-)
delete mode 100644 comidas.txt
```

- p) Crea un archivo denominado datos.bak. Añade tres títulos de libros a dicho archivo.
!!IMPORTANTE! No añadas el archivo al área de preparación ni hagas ningún commit.



- q) Crea una subcarpeta denominada output. Crea un archivo denominado salida.txt en su interior. Escribe tu nombre y apellidos en dicho archivo. **!!IMPORTANTE!** No añadas los archivos al área de preparación ni hagas ningún commit.



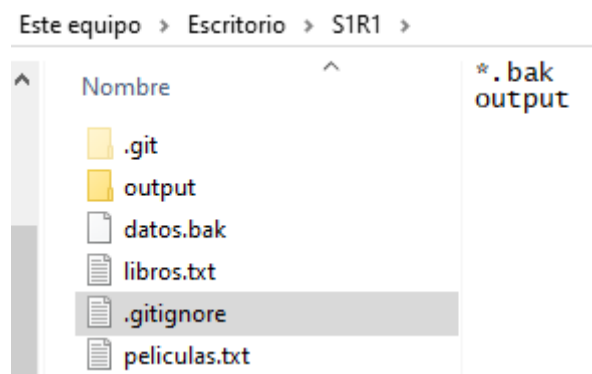
- r) Haz una captura de pantalla del comando git status. Deben aparecer los archivos datos.bak y output/salida.txt como archivos nuevos (color rojo).

git status

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    datos.bak
    output/

nothing added to commit but untracked files present (use "git add" to track)
```

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



- t) 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.

git status

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  .gitignore

nothing added to commit but untracked files present (use "git add" to track)
```

- u) Haz un último commit para incluir el archivo .gitignore en el repositorio.

git add .gitignore

git commit -m "Último commit"

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git add .gitignore

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git commit -m "Último commit"
[master f1b3a2b] Último commit
1 file changed, 2 insertions(+)
create mode 100644 .gitignore
```

- v) Haz una captura de pantalla que muestre el histórico de cambios del repositorio.

git log

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S1R1 (master)
$ git log
commit f1b3a2b180dab3b06d1043240d662721dd50840e (HEAD -> master)
Author: Hugo García <huggarmar4@alu.edu.gva.es>
Date: Thu Oct 3 10:25:13 2024 +0200

    Último commit

commit 6964b6e00282c3fb1233bb70fc8f66ee41f1f8e6
Author: Hugo García <huggarmar4@alu.edu.gva.es>
Date: Thu Oct 3 10:11:56 2024 +0200

    Cuarto commit - eliminar y modificar

commit 6b7cebe31336fe77e048bdc3923ce8e51c42799f
Author: Hugo García <huggarmar4@alu.edu.gva.es>
Date: Thu Oct 3 10:02:42 2024 +0200

    Tercer commit comidas.txt y películas.txt

commit 907a3395fac18913557b63bdf2d320b45eebb673
Author: Hugo García <huggarmar4@alu.edu.gva.es>
Date: Thu Oct 3 09:52:47 2024 +0200

    Segundo commit

commit 1c5d515244bea2ae3932f0c31308bb32e84f2707
Author: Hugo García <huggarmar4@alu.edu.gva.es>
Date: Thu Oct 3 09:48:38 2024 +0200

    Primer commit
```

Fundamentos de Git II

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

- a) Crea un repositorio Git.

git init

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1
$ git init
Initialized empty Git repository in C:/Users/HUGO/Desktop/S2R1/.git/
```

- b) Crea un fichero denominado actores.txt. Añade tres nombres de actores cada uno en una línea distinta.

cat actores

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ cat actores.txt
Will Smith
Harrison Ford
Brad Pitt
```

- c) Haz un primer commit.

git commit -a -m "Primer commit 3 actores"

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git commit -a -m "Primer commit 3 actores"
[master (root-commit) ada154a] Primer commit 3 actores
1 file changed, 3 insertions(+)
create mode 100644 actores.txt
```

- d) Crea una rama denominada test.

git branch test

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git branch test
```

- e) Cambia a la rama test

git checkout test

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git checkout test
Switched to branch 'test'
```

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

```
git add .
git commit -m "Segundo commit actrices.txt"
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test)
$ cat actrices.txt
Blanca Suárez
Gal Gadot
Emma Watson

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test)
$ git add .
warning: in the working copy of 'actrices.txt', LF will be replaced by CRLF the next time Git
touches it

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test)
$ git commit -m "Segundo commit actrices.txt"
[test 051d0fc] Segundo commit actrices.txt
1 file changed, 3 insertions(+)
create mode 100644 actrices.txt
```

- g) Haz una captura de pantalla del resultado del comando git log --graph --all.

```
git log --graph --all
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test)
$ git log --graph --all
* commit 051d0fc02ae199d250db13c7cb01a7fb0305b9e2 (HEAD -> test)
| Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
| Date:   Fri Oct 4 11:46:21 2024 +0200
|
|     Segundo commit actrices.txt
|
* commit ada154ae586851dd3ef851ca0372a7bab3a379ef (master)
| Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
| Date:   Fri Oct 4 11:41:17 2024 +0200
|
|     Primer commit 3 actores
```

- h) Cambia a la rama master.

```
git checkout master
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test)
$ git checkout master
Switched to branch 'master'
```

- i) 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.

```
git merge test
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git merge test
Updating ada154a..051d0fc
Fast-forward
 actrices.txt | 3 +++
1 file changed, 3 insertions(+)
create mode 100644 actrices.txt
```


- j) Crea una segunda rama denominada test2. La rama test2 apunta al mismo commit que la rama master en este momento.

```
git branch test 2
git log --graph --all
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git branch test2

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git log --graph --all
* commit 051d0fc02ae199d250db13c7cb01a7fb0305b9e2 (HEAD -> master, test2, test)
| Author: Hugo García <huggarmar4@alu.edu.gva.es>
| Date:   Fri Oct 4 11:46:21 2024 +0200
|
|     Segundo commit actrices.txt
|
* commit ada154ae586851dd3ef851ca0372a7bab3a379ef
| Author: Hugo García <huggarmar4@alu.edu.gva.es>
| Date:   Fri Oct 4 11:41:17 2024 +0200
|
|     Primer commit 3 actores
```

- k) En la rama master, añade una actriz al fichero actrices.txt y haz un commit.

```
git add .
git commit -m "Tercer commit"
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ cat actrices.txt
Blanca Suárez
Gal Gadot
Emma Watson
Megan Fox

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git add .

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git commit -m "Tercer commit actrices.txt + Megan Fox"
[master 1105944] Tercer commit actrices.txt + Megan Fox
1 file changed, 1 insertion(+)
```

- l) Cambia a la rama test2

```
git checkout test2
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test2)
$ git checkout test2
Already on 'test2'
```

- m) En la rama test2, añade una actriz al fichero actrices.txt y haz otro commit.

```
git add .
git commit -m "Cuarto commit"
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test2)
$ cat actrices.txt
Blanca Suárez
Gal Gadot
Emma Watson
Penélope Cruz

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test2)
$ git add .

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test2)
$ git commit -m "Cuarto commit actrices.txt + Penélope Cruz"
[test2 000bb0a] Cuarto commit actrices.txt + Penélope Cruz
1 file changed, 1 insertion(+)
```

- n) 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.

```
git log --graph --all
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (test2)
$ git log --graph --all
* commit 000bb0adeb1368427793d34edfcf0b92fe5f0a44 (HEAD -> test2)
| Author: Hugo García <huggarmar4@alu.edu.gva.es>
| Date:   Fri Oct 4 11:58:19 2024 +0200
|
|     Cuarto commit actrices.txt + Penélope Cruz
|
| * commit 1105944c6b50d587a835500f538694f976110005 (master)
| / Author: Hugo García <huggarmar4@alu.edu.gva.es>
|   Date:   Fri Oct 4 11:53:52 2024 +0200
|
|     Tercer commit actrices.txt + Megan Fox
|
| * commit 051d0fc02ae199d250db13c7cb01a7fb0305b9e2 (test)
| | Author: Hugo García <huggarmar4@alu.edu.gva.es>
| | Date:   Fri Oct 4 11:46:21 2024 +0200
| |
| |     Segundo commit actrices.txt
| |
| * commit ada154ae586851dd3ef851ca0372a7bab3a379ef
|   Author: Hugo García <huggarmar4@alu.edu.gva.es>
|   Date:   Fri Oct 4 11:41:17 2024 +0200
|
|     Primer commit 3 actores
```

- o) Cambia a la rama master

```
git checkout master
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git checkout master
Already on 'master'
```

- p) 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.

```
git merge test2
git status
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git merge test2
Auto-merging actrices.txt
CONFLICT (content): Merge conflict in actrices.txt
Automatic merge failed; fix conflicts and then commit the result.

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master|MERGING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   actrices.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

- q) Resuelve el conflicto incorporando los dos nombres de actrices.

```
Git status
git add .
git status
git commit -m "Quinto commit"
```

```
GNU nano 8.2
Blanca Suárez
Gal Gadot
Emma Watson
<<<<<< HEAD
Megan Fox
=====
Penélope Cruz
>>>>>> test2
```

```

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master|MERGING)
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   actrices.txt

no changes added to commit (use "git add" and/or "git commit -a")

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master|MERGING)
$ git add .

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master|MERGING)
$ git status
On branch master
All conflicts fixed but you are still merging.
  (use "git commit" to conclude merge)

Changes to be committed:
  modified:   actrices.txt

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master|MERGING)
$ git commit -m "Quinto commit cambios 2 ramas"
[master 6bd6b55] Quinto commit cambios 2 ramas

```

- r) 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 caminos anteriores.

git log --graph --all

```

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R1 (master)
$ git log --graph --all
*   commit 6bd6b553c7f19e37fe2bcadc1fadcd7da7a4628b0 (HEAD -> master)
|/  Merge: 1105944 000bb0a
|   Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
|   Date:   Fri Oct 4 12:18:48 2024 +0200
|
|       Quinto commit cambios 2 ramas
|
| *   commit 000bb0adeb1368427793d34edfcf0b92fe5f0a44 (test2)
|/  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
|   Date:   Fri Oct 4 11:58:19 2024 +0200
|
|       Cuarto commit actrices.txt + Penélope Cruz
|
| *   commit 1105944c6b50d587a835500f538694f976110005
|/  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
|   Date:   Fri Oct 4 11:53:52 2024 +0200
|
|       Tercer commit actrices.txt + Megan Fox
|
| *   commit 051d0fc02ae199d250db13c7cb01a7fb0305b9e2 (test)
|/  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
|   Date:   Fri Oct 4 11:46:21 2024 +0200
|
|       Segundo commit actrices.txt
|
| *   commit ada154ae586851dd3ef851ca0372a7bab3a379ef
|/  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
|   Date:   Fri Oct 4 11:41:17 2024 +0200
|
|       Primer commit 3 actores

```

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.

git init --bare

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2-remoto
$ git init --bare
Initialized empty Git repository in C:/Users/HUGO/Desktop/S2R2/S2R2-remoto/
```

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

git clone S2R2-remoto/ S2R2

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2
$ git clone S2R2-remoto/ S2R2
Cloning into 'S2R2'...
warning: You appear to have cloned an empty repository.
done.
```

- a) Crea un archivo denominado `directores.txt`. Añade el nombre de tres directores de cine.

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ cat directores.txt
Santiago Segura
Juan Antonio Bayona
Pedro Almodóvar
```

- b) Haz un commit.

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git add .
warning: in the working copy of 'directores.txt', LF will be replaced by CRLF the next time G
it touches it

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git commit -m "Primer commit directores.txt"
[master (root-commit) 138b272] Primer commit directores.txt
1 file changed, 3 insertions(+)
create mode 100644 directores.txt
```

- c) Realiza un push al repositorio remoto. Adjunta captura de pantalla del resultado

```
git push origin master
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 287 bytes | 287.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To C:/Users/HUGO/Desktop/S2R2/S2R2-remoto/
* [new branch]      master -> master
```

- d) Crea una rama denominada version1.

```
git branch version1
git branch
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git branch version1

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git branch
* master
  version1
```

- e) Cambia a la rama version1.

```
git checkout version1
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git checkout version1
Switched to branch 'version1'
```

- f) En la rama version1 añade el nombre de dos directores de cine más al archivo directores.txt y haz un commit de los cambios.

```
git add .
git commit -m "Segundo commit"
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (version1)
$ cat directores.txt
Santiago Segura
Juan Antonio Bayona
Pedro Almodóvar
Steven Spielberg
Woody Allen

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (version1)
$ git add .
warning: in the working copy of 'directores.txt', LF will be replaced by CRLF the next time G
it touches it

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (version1)
$ git commit -m "Segundo commit directores.txt + 2 directores"
[version1 c3ae8df] Segundo commit directores.txt + 2 directores
1 file changed, 2 insertions(+)
```

- g) Realiza un push de la rama al repositorio remoto de manera que quede asociada a la rama remota del mismo nombre. Adjunta captura de pantalla del resultado.

```
git push --set-upstream origin version1
```

```
git log --graph --all
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (version1)
$ git push --set-upstream origin version1
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 349 bytes | 349.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To C:/Users/HUGO/Desktop/S2R2/S2R2-remoto/
 * [new branch]      version1 -> version1
branch 'version1' set up to track 'origin/version1'.
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (version1)
$ git log --graph --all
* commit c3ae8df4b1f22d67a72936339db65e2057d8cefb (HEAD -> version1, origin/version1)
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:44:16 2024 +0200

      Segundo commit directores.txt + 2 directores

* commit 138b2725d909c184ef3c942b63064ea47e35b71e (origin/master, master)
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:38:14 2024 +0200

      Primer commit directores.txt
```

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

```
git clone S2R2-remoto/ S2R3
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2
$ git clone S2R2-remoto/ S2R3
Cloning into 'S2R3'...
done.
```

- a) Muestra en la consola el contenido del fichero directores.txt y el resultado del comando git status. Debe mostrar tres directores.

```
git status
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ cat directores.txt
Santiago Segura
Juan Antonio Bayona
Pedro Almodóvar

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

- b) Cambia a la rama version1. Muestra el resultado del comando. Comprueba que se crea una rama local version1 con el contenido de la rama remota origin/version1 y enlazada con ella. Al clonar el repositorio la rama no existía (solo se clona la rama principal, master), pero al cambiar a una rama que existe en el remoto se produce su creación local y enlazado con su correspondiente remota.

```
git checkout version1
git branch git log --graph --all
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ git checkout version1
Switched to a new branch 'version1'
branch 'version1' set up to track 'origin/version1'.

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (version1)
$ git branch
  master
* version1

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (version1)
$ git log --graph --all
* commit c3ae8df4b1f22d67a72936339db65e2057d8cefb (HEAD -> version1, origin/version1)
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:44:16 2024 +0200

      Segundo commit directores.txt + 2 directores

* commit 138b2725d909c184ef3c942b63064ea47e35b71e (origin/master, origin/HEAD, master)
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:38:14 2024 +0200

      Primer commit directores.txt
```

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

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (version1)
$ cat directores.txt
Santiago Segura
Juan Antonio Bayona
Pedro Almodóvar
Steven Spielberg
Woody Allen
```

- d) Cambia a la rama master.

```
git checkout master
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (version1)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
```


- e) Incorpora los cambios de la rama version1 a la rama master.

```
git merge version1
git log --graph --all
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ git merge version1
Updating 138b272..c3ae8df
Fast-forward
  directores.txt | 2 ++
  1 file changed, 2 insertions(+)

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ cat directores.txt
Santiago Segura
Juan Antonio Bayona
Pedro Almodóvar
Steven Spielberg
Woody Allen

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ git log --graph --all
* commit c3ae8df4b1f22d67a72936339db65e2057d8cefb (HEAD -> master, origin/version1, version1)
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:44:16 2024 +0200
  |
  |     Segundo commit directores.txt + 2 directores
  |
  * commit 138b2725d909c184ef3c942b63064ea47e35b71e (origin/master, origin/HEAD)
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:38:14 2024 +0200
  |
  |     Primer commit directores.txt
```

- f) Sube la rama master actualizada al servidor. Adjunta captura de pantalla del resultado del comando.

```
git push origin master
git log --graph --all
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ git push origin master
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To C:/Users/HUGO/Desktop/S2R2/S2R2-remoto/
  138b272..c3ae8df  master -> master

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R3 (master)
$ git log --graph --all
* commit c3ae8df4b1f22d67a72936339db65e2057d8cefb (HEAD -> master, origin/version1, origin/master, origin/HEAD, version1)
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:44:16 2024 +0200
  |
  |     Segundo commit directores.txt + 2 directores
  |
  * commit 138b2725d909c184ef3c942b63064ea47e35b71e
  Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
  Date:   Fri Oct 4 12:38:14 2024 +0200
  |
  |     Primer commit directores.txt
```

5. Vuelve de nuevo a la carpeta S2R2 y realiza las siguientes acciones:

- a) Obtén los cambios que hay en el repositorio remoto sin fusionarlos en la rama local. Adjunta captura de pantalla del resultado del comando utilizado.

git fetch

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (version1)
$ git fetch
From C:/Users/HUGO/Desktop/S2R2/S2R2-remoto
 138b272..c3ae8df  master    -> origin/master
```

- b) Actualiza la rama master local con el contenido de la rama master del repositorio remoto. Adjunta captura de pantalla del resultado del comando utilizado.

git pull origin master
git log --graph --all

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git pull origin master
From C:/Users/HUGO/Desktop/S2R2/S2R2-remoto
 * branch          master      -> FETCH_HEAD
Updating 138b272..c3ae8df
Fast-forward
 directores.txt | 2 ++
 1 file changed, 2 insertions(+)
```

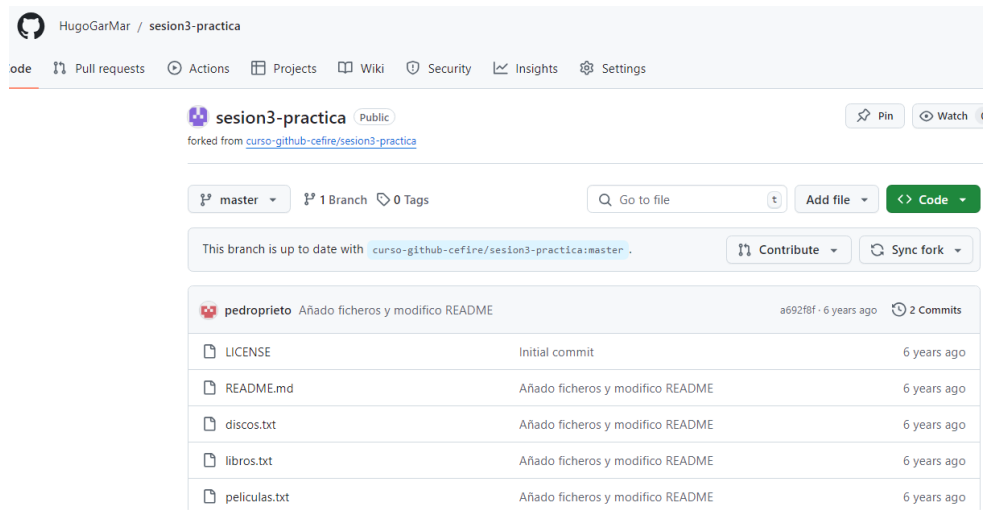
```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ git log --graph --all
* commit c3ae8df4b1f22d67a72936339db65e2057d8cefb (HEAD -> master, origin/version1, origin/master, version1)
 | Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
 | Date:   Fri Oct 4 12:44:16 2024 +0200
 |
 |     Segundo commit directores.txt + 2 directores
 |
 |
 * commit 138b2725d909c184ef3c942b63064ea47e35b71e
 | Author: Hugo Garcia <huggarmar4@alu.edu.gva.es>
 | Date:   Fri Oct 4 12:38:14 2024 +0200
 |
 |     Primer commit directores.txt
```

- c) Comprueba que aparecen los 5 nombres de directores esperados.

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/S2R2/S2R2 (master)
$ cat directores.txt
Santiago Segura
Juan Antonio Bayona
Pedro Almodóvar
Steven Spielberg
Woody Allen
```

4. GitHub

1. Crea una cuenta en GitHub
2. Añade tu dirección de correo de educación
3. Solicita un descuento para uso educativo: <https://education.github.com/pack>
4. Haz un fork del repositorio localizado en la siguiente url: <https://github.com/curso-github-cefire/sesion3-practica>. A partir de este momento todas las tareas que se indican se deben realizar en tu repositorio (el que has clonado mediante el fork).



git clone URL

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git clone https://github.com/HugoGarMar/sesion3-practica.git
```

- a) Realiza un primer commit para poner tu nombre y apellidos en el fichero README.md

```
git add README.md
git commit -m "Primer commit"
```

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ cat README.md
# Uso de GitHub en la práctica docente
## Práctica 3.2
Repositorio de prácticas para la sesión 3 del curso

Incluye a continuación tu nombre y apellidos:
Hugo García Marco

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git add README.md

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git commit -m "Primer commit nombre y apellidos"
[master 788911d] Primer commit nombre y apellidos
1 file changed, 1 insertion(+)
```

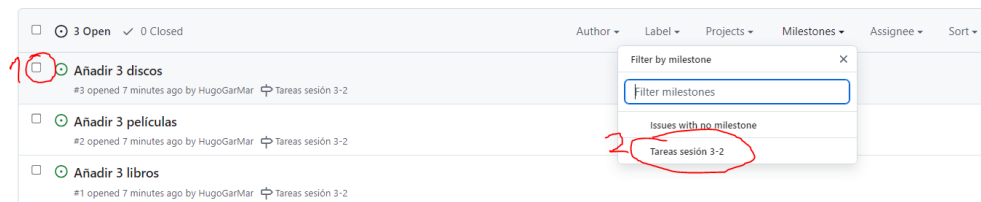
- b) Crea 3 issues con los siguientes títulos. Si no ves la pestaña de issues, actívala desde los ajustes (settings) del repositorio.

Settings > General > Features > issues

- Anadir 3 libros
- Anadir 3 películas
- Anadir 3 discos



- c) Crea una milestone denominada Tareas sesion 3-2 que contenga los 3 issues creados.

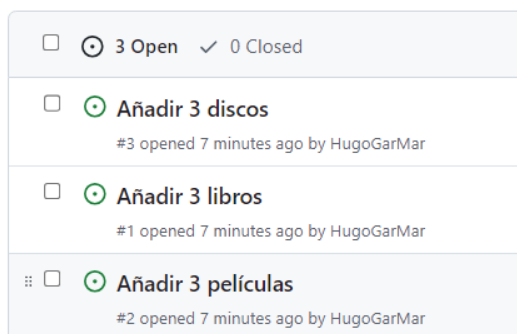


ProTip! Exclude your own issues with `-author:HugoGarMar`.

Labels Milestones

Tareas sesión 3-2

No due date 0% complete



- 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.

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ cat peliculas.txt
Orgullo y Prejuicio
La boda de mi mejor amiga
Alta Fidelidad
Harry Potter 1
La vida de Pi
Aquaman

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ cat libros.txt
Juego de Tronos
El Juego de Ender
La Historia Interminable
Harry Potter
El principe de la niebla
Cinco semanas en globo

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ cat discos.txt
The Beatles
Ella & Louis
Jagged Little Pill
Rolling Stones
Fito & Fitipaldis
Extremoduro
```

git commit -m "Libros Close #1"

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git add libros.txt

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git commit -m "Libros. close #1"
[master ea7b568] Libros. close #1
1 file changed, 3 insertions(+)

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git add peliculas.txt

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git commit -m "Peliculas. close #2"
[master d4775ac] Peliculas. close #2
1 file changed, 3 insertions(+)

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git add discos.txt

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git commit -m "Discos. close #3"
[master fc9e083] Discos. close #3
1 file changed, 3 insertions(+)

```

- e) Haz una captura de pantalla de los comandos que has utilizado para subir los cambios a GitHub.

git push origin master

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git push origin master
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 16 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (9/9), 1.06 KiB | 1.06 MiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/HugoGarMar/sesion3-practica.git
78891ld..fc9e083 master -> master
```



- f) Incluye 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.

```
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ ls capturas/
Práctica.docx

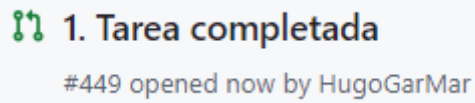
HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git add .

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   "capturas/Pr\303\241ctica.docx"

HUGO@DESKTOP-PH8GTP7 MINGW64 ~/Desktop/GIT/sesion3-practica (master)
$ git commit -m "Carpeta capturas y fichero word con capturas"
[master 224ca9d] Carpeta capturas y fichero word con capturas
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 "capturas/Pr\303\241ctica.docx"
```

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



5. Entrega de la tarea

Crea un documento con las capturas pedidas (Fundamentos de Git I, Fundamentos de Git II y GitHub) y guárdalo en formato PDF.

Guarda dicho documento en la carpeta capturas del repositorio y sube los cambios a GitHub. La entrega de la tarea se realiza a través de la pull request en GitHub.