

Git Exercices

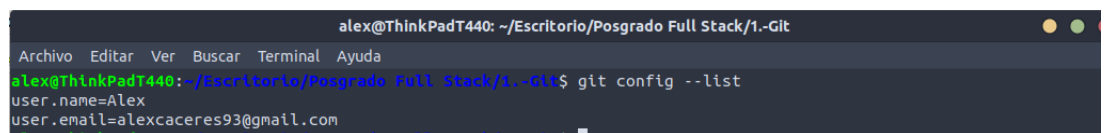
Alex Cáceres Gómez

UPC School - Full Stack Web Technologies

alexcaceres93@gmail.com

In this exercise, we are going to play with git and its commands. Describe the procedure and commands for the following:

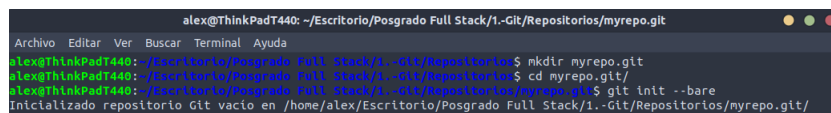
1. Install git in your (Linux) system and set your username and e-mail.



```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git
Archivo Editar Ver Buscar Terminal Ayuda
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git$ git config --list
user.name=Alex
user.email=alexcaceres93@gmail.com
```

Done.

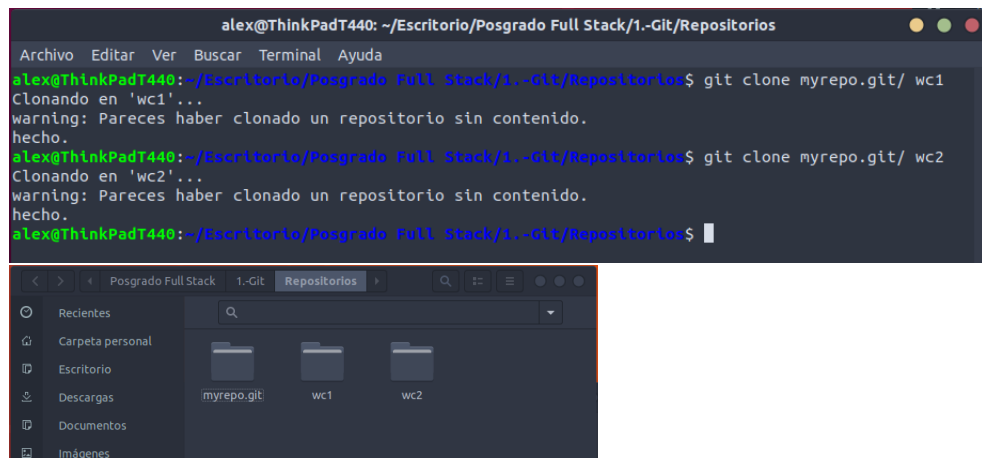
2. In your system, create a bare repository called myrepo.git.



```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/myrepo.git
Archivo Editar Ver Buscar Terminal Ayuda
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios$ mkdir myrepo.git
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios$ cd myrepo.git/
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/myrepo.git$ git init --bare
Iniciado repositorio Git vacío en /home/alex/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/myrepo.git/
```

Done.

3. In your system, create two clones of myrepo called wc1 and wc2.



```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios
Archivo Editar Ver Buscar Terminal Ayuda
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios$ git clone myrepo.git/ wc1
Clonando en 'wc1'...
warning: Parece haber clonado un repositorio sin contenido.
hecho.
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios$ git clone myrepo.git/ wc2
Clonando en 'wc2'...
warning: Parece haber clonado un repositorio sin contenido.
hecho.
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios$
```

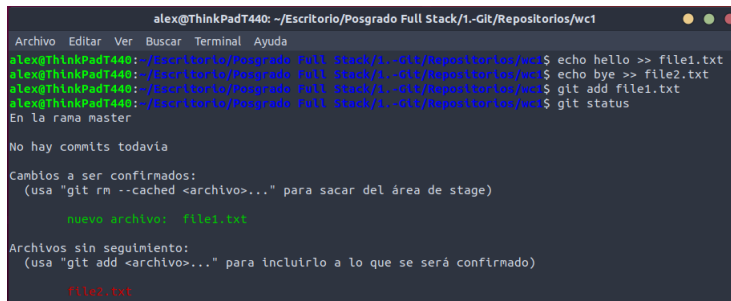
The file manager screenshot shows the 'Repositorios' directory containing three folders: 'myrepo.git', 'wc1', and 'wc2'.

*No se especifica en el enunciado, pero se ha configurado *myrepo.git* como repositorio remoto común a *wc1* y *wc2*

4. In `wc1`, create two files `file1.txt` and `file2.txt` where `file1.txt` has the content "hello" and `file2.txt` has the content "bye". Describe the command to stage your current version of `file1.txt`.

```
git add
```

Permite incluir las actualizaciones de un archivo en el *stage*, lo cual, prepara el archivo para realizarse un *commit*.



```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1
Archivo Editar Ver Buscar Terminal Ayuda
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ echo hello >> file1.txt
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ echo bye >> file2.txt
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git add file1.txt
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git status

En la rama master

No hay commits todavía

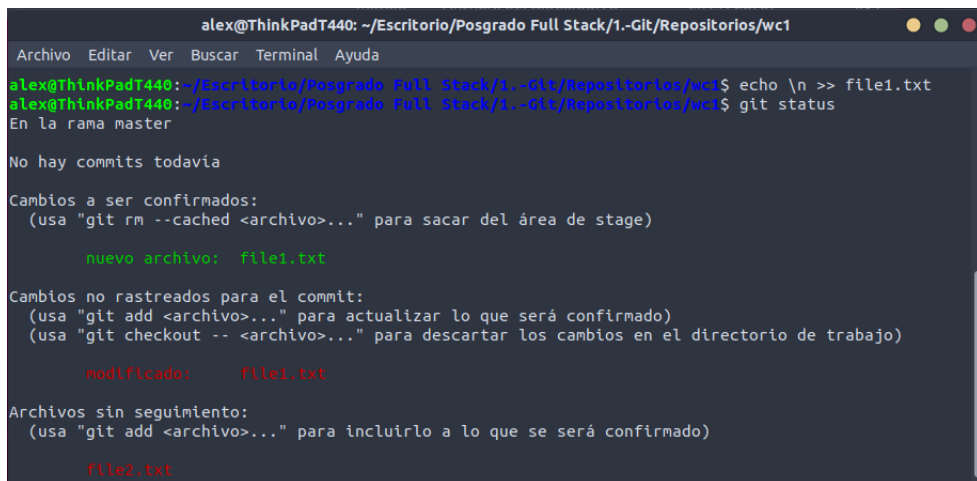
Cambios a ser confirmados:
(usa "git rm --cached <archivo>..." para sacar del área de stage)

nuevo archivo: file1.txt

Archivos sin seguimiento:
(usa "git add <archivo>..." para incluirlo a lo que se será confirmado)

file2.txt
```

5. Add a line to `file1.txt`. Which is the current state of `file1.txt`?



```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1
Archivo Editar Ver Buscar Terminal Ayuda
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ echo \n >> file1.txt
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git status

En la rama master

No hay commits todavía

Cambios a ser confirmados:
(usa "git rm --cached <archivo>..." para sacar del área de stage)

nuevo archivo: file1.txt

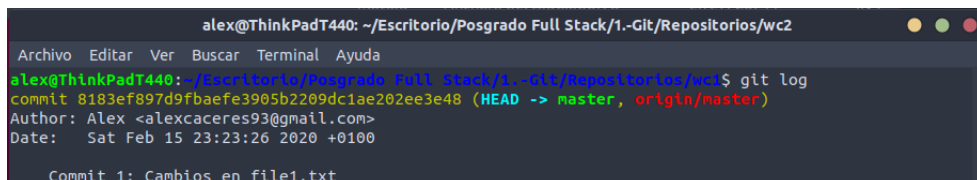
Cambios no rastreados para el commit:
(usa "git add <archivo>..." para actualizar lo que será confirmado)
(usa "git checkout -- <archivo>..." para descartar los cambios en el directorio de trabajo)

modificado: file1.txt

Archivos sin seguimiento:
(usa "git add <archivo>..." para incluirlo a lo que se será confirmado)

file2.txt
```

6. Now, stage all the changes made to `file1.txt` and create a commit.



```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc2
Archivo Editar Ver Buscar Terminal Ayuda
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git log
commit 8183ef897d9fbaefe3905b2209dc1ae202ee3e48 (HEAD -> master, origin/master)
Author: Alex <alexcaceres93@gmail.com>
Date: Sat Feb 15 23:23:26 2020 +0100

Commit 1: Cambios en file1.txt
```

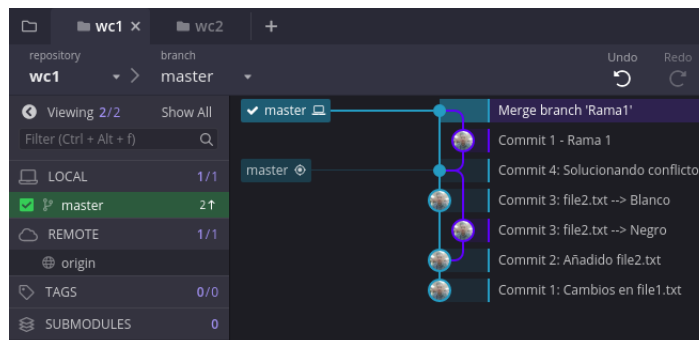
* Se ha creado un segundo commit para añadir el archivo `file2.txt` que todavía no se había pedido añadir y se ha hecho un *push* para actualizarlo en *myrepo.git*. En el repo `wc2` se ha hecho un *pull* para ir a la par con `wc1`, preparándolo así para crear el conflicto en el siguiente ejercicio.

7. Create a conflict between wc1 and wc2. Describe how you know that there is a conflict and how youfix it.

```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1
Archivo Editar Ver Buscar Terminal Ayuda
alex@ThinkPadT440:~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git push
To /home/alex/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/myrepo.git/
! [rejected]        master -> master (non-fast-forward)
error: fallo el push de algunas referencias a '/home/alex/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/myrepo.git/'
ayuda: Actualizaciones fueron rechazadas porque la punta de tu rama actual esta
ayuda: detrás de su contraparte remota. Integra los cambios remotos (es decir
ayuda: 'git pull ...') antes de hacer push de nuevo.
ayuda: Mira 'Note about fast-forwards' en 'git push --help' para mas detalles.
```

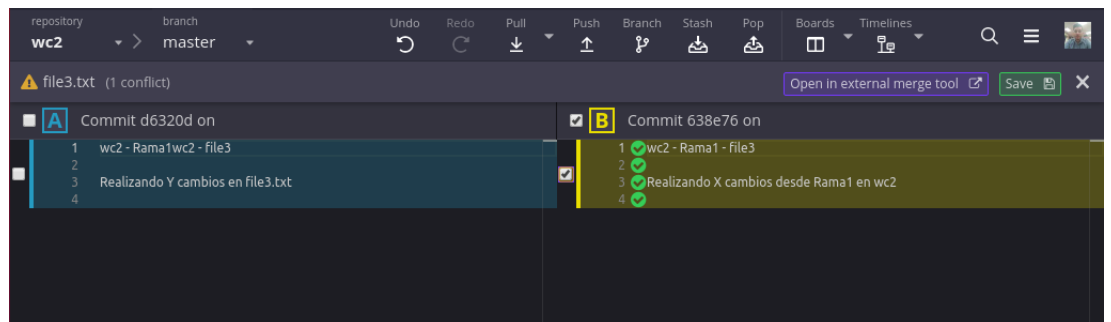
Se ha decidido realizar un *pull* en *wc1* para obtener los cambios realizados antes en *wc2* y, después, hacer *push* desde *wc1* para solucionar el conflicto.

8. Create a branch and commit some changes in it. Merge these changes in the master branch. Which type of merge is this (fast forward or 3 way merge)? Use some command or graphical tool to see the history graph. Finally, delete the created branch once merged.

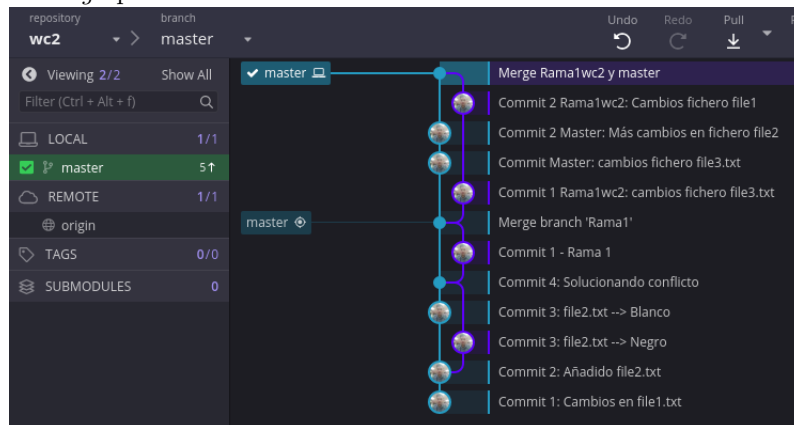


Se han realizado cambios únicamente en la *Rama 1* y se ha realizado el *merge fast-forward*, posteriormente, se ha eliminado Rama 1. *Esto se ha llevado a cabo en el repo wc1.

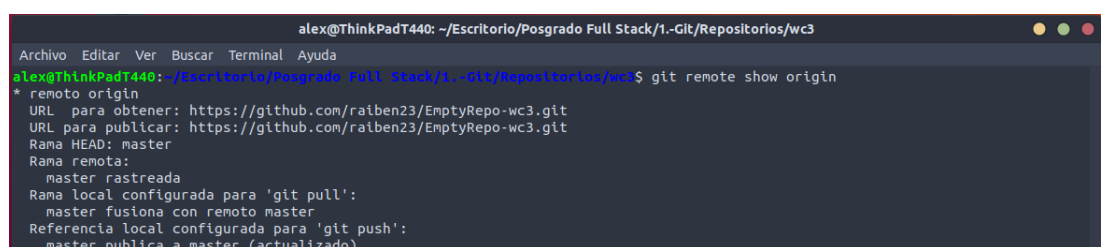
9. Create another new branch and commit modifications in both the new branch and master. Merge these changes in master. Which type of merge is this (fast forward or 3 way merge)? Use some-command or graphical tool to see the history graph. Finally, delete the created branch once merged.

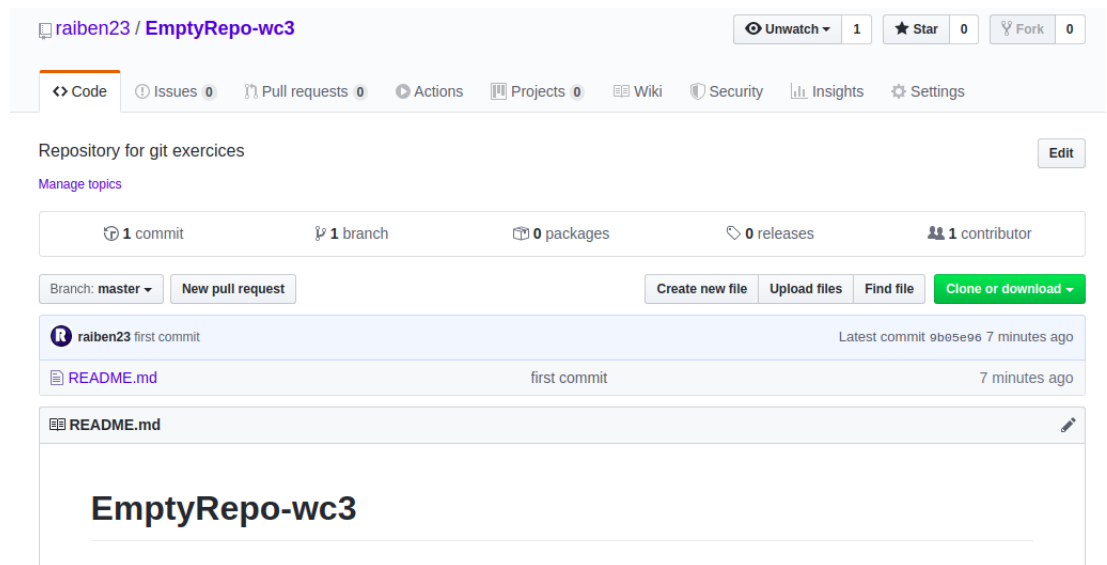


Se ha encontrado conflictos en el fichero file3, y se han aceptado los cambios que se han realizado en la Rama1, solucionando así el conflicto y realizando el *merge* posteriormente.



10. Create an account in Github.com. This account will be used during the course (and beyond?). From the Github control panel create a new empty repo and create a clone in your local system called wc3. Use a command to see the details of the origin of wc3.





11. Change the origin of your wc1 to point to your Github repository and push all your changes to this Github repo.



```
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git remote set-url origin https://github.com/raiben23/EmptyRepo-wc3.git
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git remote -v
origin  https://github.com/raiben23/EmptyRepo-wc3.git (fetch)
origin  https://github.com/raiben23/EmptyRepo-wc3.git (push)
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$ git push
Username for 'https://github.com': raiben23
Password for 'https://raiben23@github.com':
Everything up-to-date
alex@ThinkPadT440: ~/Escritorio/Posgrado Full Stack/1.-Git/Repositorios/wc1$
```

En primer lugar, se ha cambiado el remoto del repo wc1 por el enlace directo al repo de GitHub. Después, se ha ejecutado un *push* en el cual ha habido un conflicto ya que ambos repositorios divergían en 12 commits de diferencia, se ha solventado realizando un *git push -force* (No se, si es la mejor solución)

12. In pairs, make pull requests to each other. Describe how you do the pull request and how you acceptit.

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

Delete Conversor-1.0.jar #1

- Se ha realizado un *fork* del repositorio de interés
- Se han realizado los cambios pertinentes
- Se ha hecho un *commit* con los cambios (Por lo cual, mi repositorio proveniente del fork, va 1 commit por delante del de Víctor)
- Se ha enviado el *pull request* solicitando a Víctor que haga *merge* con los cambios realizados si lo cree conveniente.

- Mismo procedimiento que en el ejercicio anterior.