



Taller Git

Ejercicio 1

- 1) Clonen el repositorio <https://github.com/jfdoppler/marolio>
- 2) Completen el archivo “marolio.txt” (pueden ayudarse con el link dentro del archivo)
- 3) Guarden y hagan un commit con los cambios

Ayuda: [cheat-sheet](https://github.github.com/training-kit/downloads/es_ES/github-git-cheat-sheet.pdf) de comandos (https://github.github.com/training-kit/downloads/es_ES/github-git-cheat-sheet.pdf)

Ejercicio 2

- 1) Creen un repositorio local vacío usando el comando **git init**
- 2) Creen un repositorio en GitHub
- 3) Digamosle al repositorio local que apunte a este repositorio remoto usando **git remote add origin [URL]**
- 4) ¿Qué devuelve el comando **git remote -v**? ¿Qué quiere decir?
- 5) Hagan un push al repositorio remoto: **git push origin master** (¿por qué **origin**?)

Ejercicio 3 (de a 2)

- 1) Elijan uno de los repositorios de Github que crearon recién en el ejercicio 2.
- 2) Quien sea el dueño (**D**) del repo, le va a dar acceso a su colaborador (**C**) (settings, manage access)
- 3) Ahora **C** va a clonar el repositorio en su compu (va a tener un repo **local**)
- 4) Cambien, cada uno por algo distinto, el contenido de la **primera línea**
- 5) Ahora **D** committee sus cambios y pushee al remoto
- 6) Ahora **C**... ¿que pasó?

- 7) Para enmendar, **C** tiene que hacer un pull (**git pull origin master**) que lo pone al día
- 8) A veces git puede mechar todo sin problemas, pero si hay inconsistencias, nos pide que resolvamos

```
README: Bloc de notas
Archivo Edición Formato Ver Ayuda
Hola
<<<<<<< HEAD
Bye
=====
Ciao
>>>>>> 84fe8ed1ff056ee5b238a2b72d87c27e84c70a99
Hola
```

Ejercicio 4 (de a 2)

- 1) A va a crear un repo con un archivo de texto llamado “películas.txt” y le va a dar acceso de push a B. En el archivo va a escribir los nombres de 5 películas
- 2) A y B se van a crear nuevas ramas en su repo. A y B le van a asignar una nota a cada una de las películas.
- 3) Ambos van a pushear al remoto (B en su nueva rama): **git push -u origin [rama]**
- 4) B va a hacer un pull en la rama master (¡recordar hacer checkout!)
- 5) B va a hacer un merge en la rama master. Si hay conflictos en las notas, ¡se promedia!
Luego hace un push: **git push -u origin master**
- 6) Ahora A puede pullear y estar al día.
- 7) Intercambien roles y de nuevo!

Ejercicio 5 (de a 2)

- 1) Trabajamos de a 2: Mantenedor, Colaborador (M, C)
- 2) M clona el siguiente repo y lo sube a Github: [jfdoppler/taller-git](https://github.com/jfdoppler/taller-git)
- 3) C hace un “fork” del repo de M y lo baja
- 4) C le dice a su repo, donde está el “upstream”: **git remote add upstream [URL]**
- 5) ¿Qué dice el “git remote -v”?
- 6) C completa en el archivo los nombres o funciones de los comandos de git
- 7) C commitea los cambios y hace push
- 8) ¡Revisen los repos! ¿En qué repositorio remoto están los cambios de C?
- 9) Para que los cambios lleguen al repositorio original (del cual C hizo un fork) tiene que hacer un **pull request**
- 10) Si está todo ok, ¡M lo acepta!