

git branch [-a]

//para ver la rama en la que estoy parado

verde: la rama que estoy,

gris: las ramas locales

rojo: las ramas remotas

git status

//para ver el estado de los repositorios

git log [--oneline] [--decorate] [--graph]

//para ver los commits.

archivos en rojo (nuevo y con modificaciones)

git diff archivo

//para ver los cambios que tiene.

git switch rama

//para cambiar de rama.

parado en rama (feature)

git switch -c ramaNueva

//para crear una ramaNueva a partir de la rama feature

parado en la rama (feature)

git switch -c ramaNueva origin/feature

//para crear una ramaNueva a partir de la rama remota feature

ejecutar antes: git fetch -a (para bajar todas las ramas)

git switch -c ramaNueva master

crea una ramaNueva a partir master

git config --global user.name 'Nombre Apellido'

//nombre que aparece en el commit

git config --global user.email 'correo@gmail.com'

correo que aparece en el commit

```
git config --global color.ui true
//configurar
```

```
git config --global pull.rebase true
//opción de configuración
```

```
git config --global credential.username "usuariogit"
```

```
git branch --set-upstream-to=origin/master master
//vincular la rama origin/master (remota) a la rama master local
(estando en master) git pull = git pull origin master
```

```
git config --global alias.lodg 'log --oneline --decorate --graph'
//crear un alias
git lodg = git log --oneline --decorate --graph
git config --global alias.conflict 'diff --name-only --diff-filter=U'
//alias
git conflict = git diff --name-only --diff-filter=U
```

```
git config --global --get-regexp alias
//git config --global --unset alias.lodg
//borrar alias
git config --global --list
//listar config
git config --global --unset user.email
```

```
git pull = git fetch -a && git merge origin/master
```

```
//flujo de trabajo
git switch master
```

git fetch -a
git pull

git switch -c nuevaRama
//nuevaRama

git status

git add .

git commit -m "mensaje"

git lodg
//veo el commit con mensaje

puedo seguir haciendo cambios
git status

git add .
git commit -m "segundo mensaje"

git lodg

2 Opciones

git switch master

git pull

ahora me cambio mezclar los cambios

git merge nuevaRama

sino hay conflictos

git push origin master

si hay conflictos

git conflict

//ver archivos con conflictos

resolverlos

git status

git add .

git commit -m "resolviendo conflictos"

git status

//indica opción hacer

git rebase --continue

//si no quiero los cambios, borro la rama

git branch -D nuevaRama