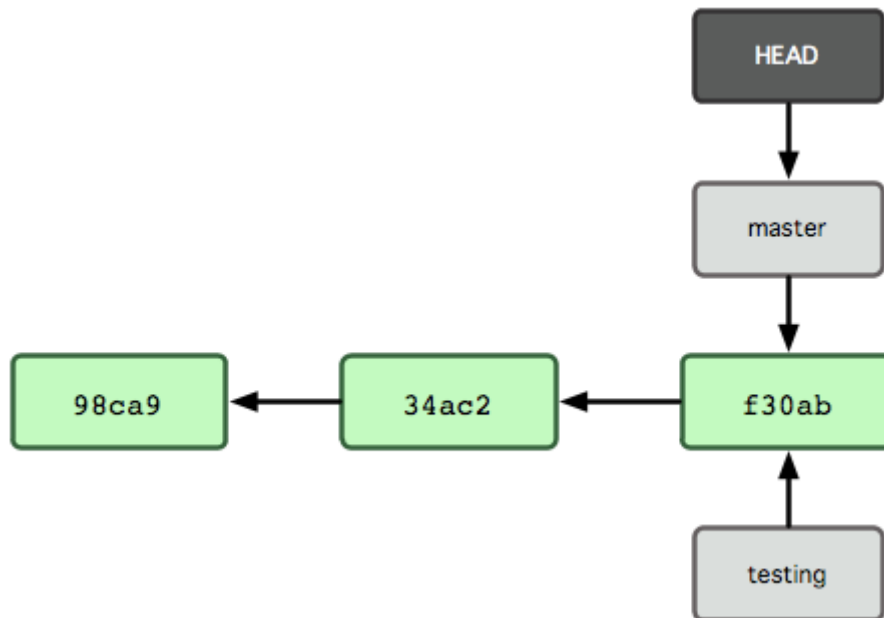


# ramas

A diferencia de otros sistemas de control de versiones, las ramas permiten trabajar a partir de una instantánea en paralelo, sin tener impacto sobre el código del cual se generó la instantánea.

Las Ramas no son más que “punteros” a una determinada instantánea:



## Crear Ramas

Tenemos dos formas

```
> git status
On brach master ...
git branch testing
```

OJO! Aquí sólo hemos creado la rama, pero seguimos en la rama master.

O bien , situados en la rama a partir de la cuál queremos crear otra rama ( una instantánea )

```
# Estando por ejemplo en la rama master
> git status
On brach master ...
> git checkout -b testing
```

Con esta forma hacemos dos pasos en uno : Crear la rama y cambiarnos a ella.

Recuerda que al crear una rama, la estás creando en tu repositorio local. Si quieres que esa rama aparezca en tu repositorio remoto, tendrás que hacer un push con algún commit en la nueva rama.

## Cambiar de Rama

Como hemos visto en la sección anterior , para cambiarnos de rama, tenemos que ejecutar el siguiente comando:

```
git checkout testing
```

siendo testing el nombre de la rama a la que queremos cambiar.

OJO! Para cambiar de rama, siempre deberíamos tener la rama desde la que nos cambiamos con una instantánea. Si tenemos ficheros modificados pero no añadidos al área de intercambio o ficheros en el área de intercambio no “commiteados”, nos aparecerá un mensaje de error, evitando que nos cambiemos de rama.

## Comparar Ramas

Para comparar dos instantáneas ( o dos ramas ), se puede emplear el comando :

```
git diff commit1..commit2
```

siendo commit1 y commit2 los hashes (SHA-1) de las instantáneas que quieres comparar. Por defecto, si no incluimos los commits en el commando , comparará HEAD , con la última instantánea conocida.

OJO! El orden en el que especifiquemos los hashes de los commits influye en cómo vas a ver las diferencias entre ficheros

## Fusionar Ramas

La forma de incorporar cambios de una rama a otra es fusionar o “mergear” ramas.

El procedimiento es siempre el mismo. Vamos a ponernos en un escenario donde queremos incorporar a la rama master los cambios que hemos hecho en otra rama cambios\_titulos .

Incorpora cambios a tu rama “fuente” (cambios\_titulos ) y haz commit.

```
git checkout cambios_titulos
vim index.html
...
git add index.html
git commit -m "Cambios en los títulos de la home"
```

Cambias a la rama destino (master), donde deseas que se incorporen los cambios.

```
git checkout master
```

Ejecutas :

```
git merge cambios_titulos
```

Recuerda , que cuando abrimos una nueva rama, lo que realmente se está haciendo es partir de una instantánea en ese momento , y al hacer checkout HEAD apuntará a la instantánea en ese momento. Si la instantánea de la que partimos no sufre ningún cambio, cuando hagamos el merge , no habrá ningún problema. Si por el contrario , en el momento de hacer el merge , la rama master apunta a otra instantánea distinta a la original desde la que partimos , pueden generarse “conflictos” y git nos avisará de ello, si no es capaz de poder fusionar las ramas por sí mismo.

## Resolución de Conflictos

En la sesión recrearemos un escenario en el que se produzcan conflictos a la hora de fusionar ramas y veremos cómo resolverlo.

From:

<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:

<https://miguelangel.torresegea.es/wiki/info:cursos:openwebinars:git:ramas>

Last update: **08/07/2018 02:03**

