

**1. ¿Qué comando utilizaste en el paso 11? ¿Por qué?**

**Comando:** git reset --hard HEAD~1

**Comentario:** La opción --hard asegura que los cambios se descarten completamente del último commit y del *working directory*. HEAD~1 Indica que deseas mover el puntero HEAD un commit hacia atrás, es decir, al commit anterior.

**2. ¿Qué comando o comandos utilizaste en el paso 12? ¿Por qué?**

**Comando:** git reflog

**Comando:** git reset --hard <hash code commit>

**Comentario:** Con el git reflog puedo observar un registro los movimientos de HEAD y ahí puedo buscar el hash del commit al cual necesito ir.

**3. El merge del paso 13, ¿Causó algún conflicto? ¿Por qué?**

No, no causó conflicto, la rama styled ya ha sido actualizada con todos los cambios de main anteriormente, y como los cambios realizados en el archivo fueron en líneas diferentes, es decir en secciones que no se superponen, Git puede fusionar los cambios sin conflictos, incluso no realiza el merge porque entiende que no hay nada nuevo que mergear

**4. El merge del paso 19, ¿Causó algún conflicto? ¿Por qué?**

Si, causó un conflicto, porque como los cambios realizados en el archivo fueron en líneas de secciones que si había superposición.

**5. El merge del paso 21, ¿Causó algún conflicto? ¿Por qué?**

No, no causó conflicto. Git puede fusionar los cambios sin conflictos los cambios fueron en líneas diferentes, es decir en secciones que no se superponen.

**6. ¿Qué comando o comandos utilizaste en el paso 25?**

**Comando:** git graph

**Comentario:** El parámetro graph es un alias configurado de forma global con los siguientes valores log --graph --decorate --pretty=oneline por lo que el comando completo seria **git log --graph --decorate --pretty=oneline**

**7. El merge del paso 26, ¿Podría ser fast forward? ¿Por qué?**

Si, porque en la rama MAIN no tiene commits que la rama de TITLE no tenga.

**8. ¿Qué comando o comandos utilizaste en el paso 27?**

**Comando:** git reset HEAD~1

**9. ¿Qué comando o comandos utilizaste en el paso 28?**

**Comando:** git restore git-nuestro.md

10. ¿Qué comando o comandos utilizaste en el paso 29?

**Comando:** `git branch -D title`

**Comentario:** No se puede realizar con el parámetro -d porque hay cambios en la rama title que no fueron mergeados.

11. ¿Qué comando o comandos utilizaste en el paso 30?

**Comando:** `git reflog`

**Comando:** `git reset --hard <hash code commit>`

12. ¿Qué comando o comandos usaste en el paso 32?

**Comando:** `git reflog`

**Comando:** `git checkout <hash code commit>`

**Comentario:** Solo vamos al commit, es decir no llevamos el HEAD, en caso de llevar el HEAD a ese commit entonces seria con reset

13. ¿Qué comando o comandos usaste en el punto 33?

**Comando:** `git reflog`

**Comando:** `git checkout <hash code commit>`

**Comentario:** Solo vamos al commit, es decir no llevamos el HEAD, en caso de llevar el HEAD a ese commit entonces seria con reset