

Instructivo para mergear dos ramas en Git usando la consola Bash de VS Code

Objetivo

Supongamos que quieres **fusionar** la rama **feature-x** dentro de **master**.

Paso 1: Verificá tus ramas

```
git branch
```

Muestra todas las ramas locales. Asegurate de que tenés las ramas necesarias.

Paso 2: Ir a la rama donde vas a hacer el merge

```
git checkout master
```

Esto te posiciona en la rama que va a **recibir los cambios**.

Paso 3: Traer los últimos cambios del repositorio remoto (por si alguien subió algo)

```
git pull origin master
```

Paso 4: Hacer el merge

```
git merge feature-x
```

Este comando intenta combinar la rama **feature-x** dentro de **master**.

Posibilidad 1: Merge exitoso

Si todo va bien, vas a ver algo como:

Merge made by the 'recursive' strategy.

 ¡Listo! Podés hacer push:

```
git push origin master
```

Posibilidad 2: Conflictos de merge

Si hay archivos modificados en ambas ramas, Git va a mostrar:

Auto-merging archivo.txt

CONFLICT (content): Merge conflict in archivo.txt

Automatic merge failed; fix conflicts and then commit the result.

¿Qué hacer ante un conflicto?

1. **Abrió el archivo con conflicto.** Git lo marca así:

```
<<<<<<< HEAD
```

Este es el contenido actual en master.

```
=====
```

Este es el contenido nuevo en feature-x.

```
>>>>>>> feature-x
```

2. **Resuelve el conflicto.** Elegí qué parte quieres conservar, combina o escribí una nueva versión.
3. **Eliminá los indicadores** `<<<<<<<`, `=====`, `>>>>>>>`.
4. **Agregá el archivo a staging:**

```
git add archivo.txt
```

5. **Hacé commit del merge:**

```
git commit
```

6. **Subí los cambios al repositorio remoto:**

```
git push origin master
```



Situaciones conflictivas comunes

Situación	Causa	Solución
Mismo archivo, misma línea editada en ambas ramas	Conflicto al hacer <code>merge</code>	Editá manualmente el archivo y resolvé
Archivo borrado en una rama y modificado en otra	Git no sabe si mantenerlo o borrarlo	Elegí una opción usando <code>git checkout --ours</code> o <code>--theirs</code>
Merge automático pero con muchos cambios	Dificultad para revisar todo	Usá <code>git diff</code> antes y después del merge para ver los cambios



Tips extra

- Usá `git status` para ver qué archivos están en conflicto.
- Podés usar herramientas visuales como el editor de VS Code para resolver conflictos: te muestra botones para aceptar los cambios de una u otra rama.
- Si querés **abortar** el merge:

`git merge --abort`