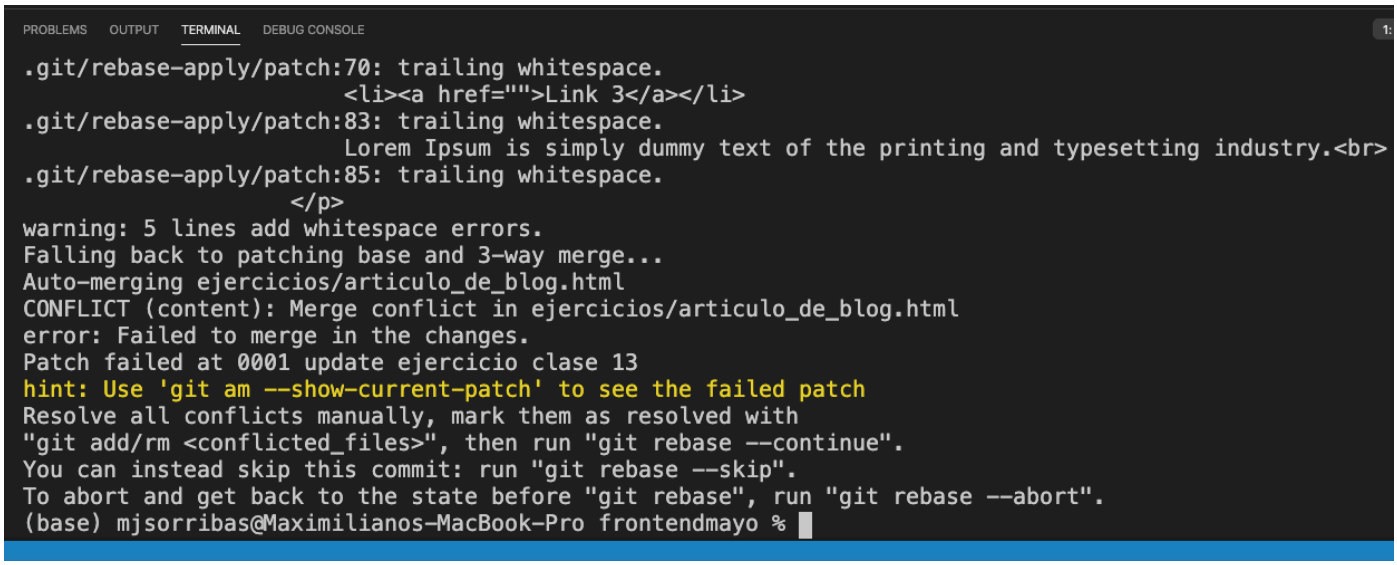
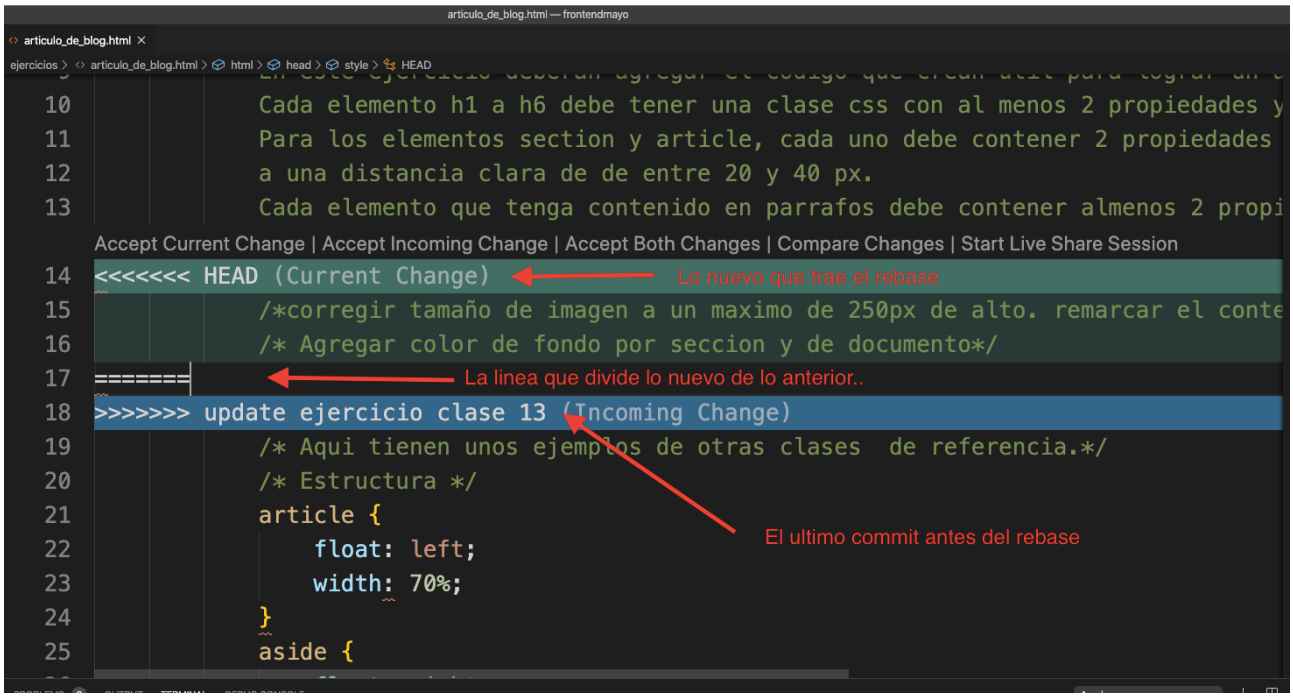
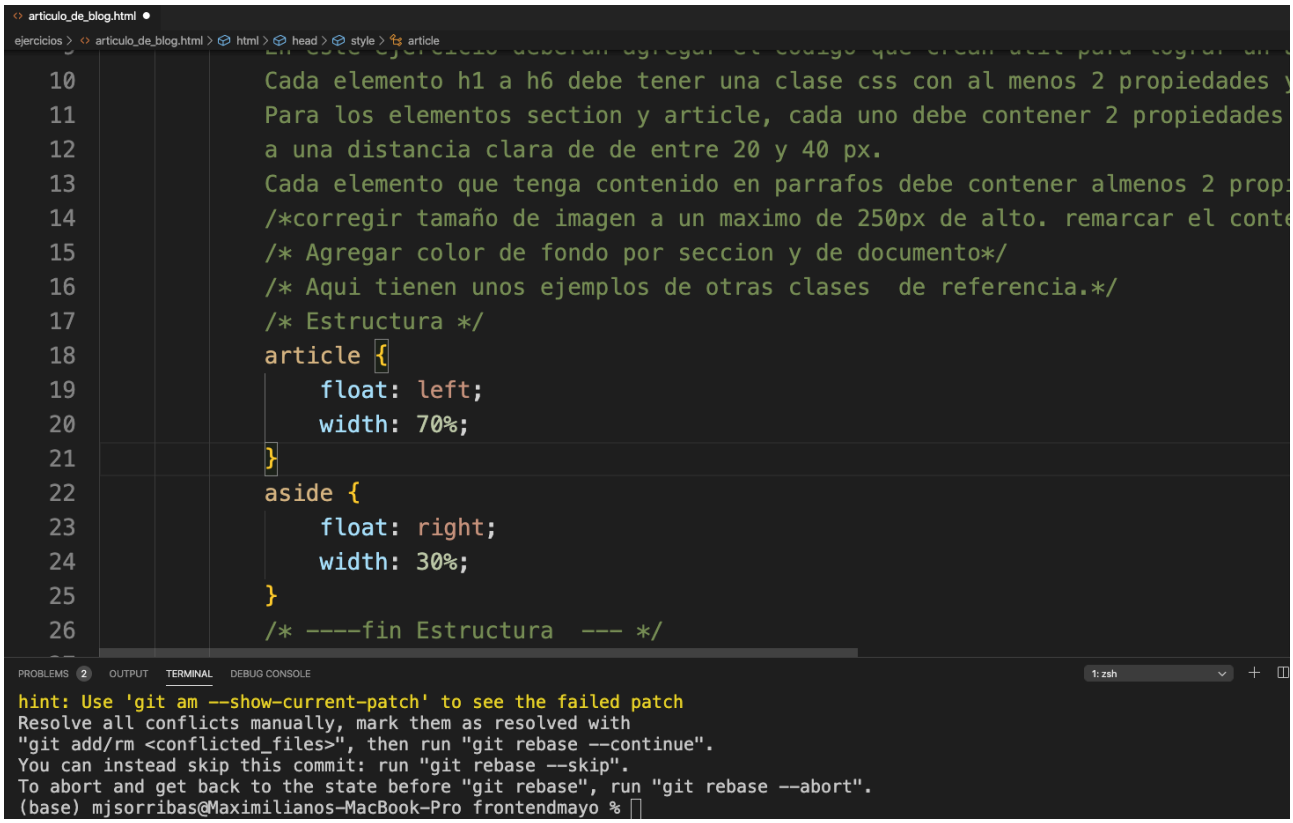
Como saber que reparar en nuestro rama cuando hago un rebase y hay conflicto



Miramos que archivos es el conflictivo (la consola nos dice CONFLICT (content): Merge conflict in ejercicios/articulo_de_blog.html). Abrimos el archivo con conflicto



Tratamos de revisar el cambio .. y vemos que solo agrega unas lianas (HEAD) Eliminamos en este ejemplo la linea que dice HEAD (linea14), la separación (linea17) y el incoming change.



Luego de limpiar nuestro archivo. Nos quedará algo así:

Ahora que quedaron los conflictos resueltos guardamos el archivo.
Ejecutamos git add carpeta/nombre_demi_archivo
Ejecutamos git rebase —continue

En el caso de que de este error:

```
(base) mjsorribas@Maximilianos-MacBook-Pro frontendmayo % git add ejercicios/articulo_de_blog.html
(base) mjsorribas@Maximilianos-MacBook-Pro frontendmayo % git rebase --continue
Applying: update ejercicio clase 13
No changes - did you forget to use 'git add'?
If there is nothing left to stage, chances are that something else
already introduced the same changes; you might want to skip this patch.
Resolve all conflicts manually, mark them as resolved with
"git add/rm <conflicted_files>", then run "git rebase --continue".
You can instead skip this commit: run "git rebase --skip".
To abort and get back to the state before "git rebase", run "git rebase --abort".
```

Lo que hacemos es aplicar lo que nos pide que es un git rebase —skip.

```
(base) mjsorribas@Maximilianos-MacBook-Pro frontendmayo % git rebase --skip
Applying: consigna
```

Si por un camino u otro llegaron finalizar el rebase

Ejecutamos git push —force

Empezamos a escribir nuestro código.