



Push, pull & pull request

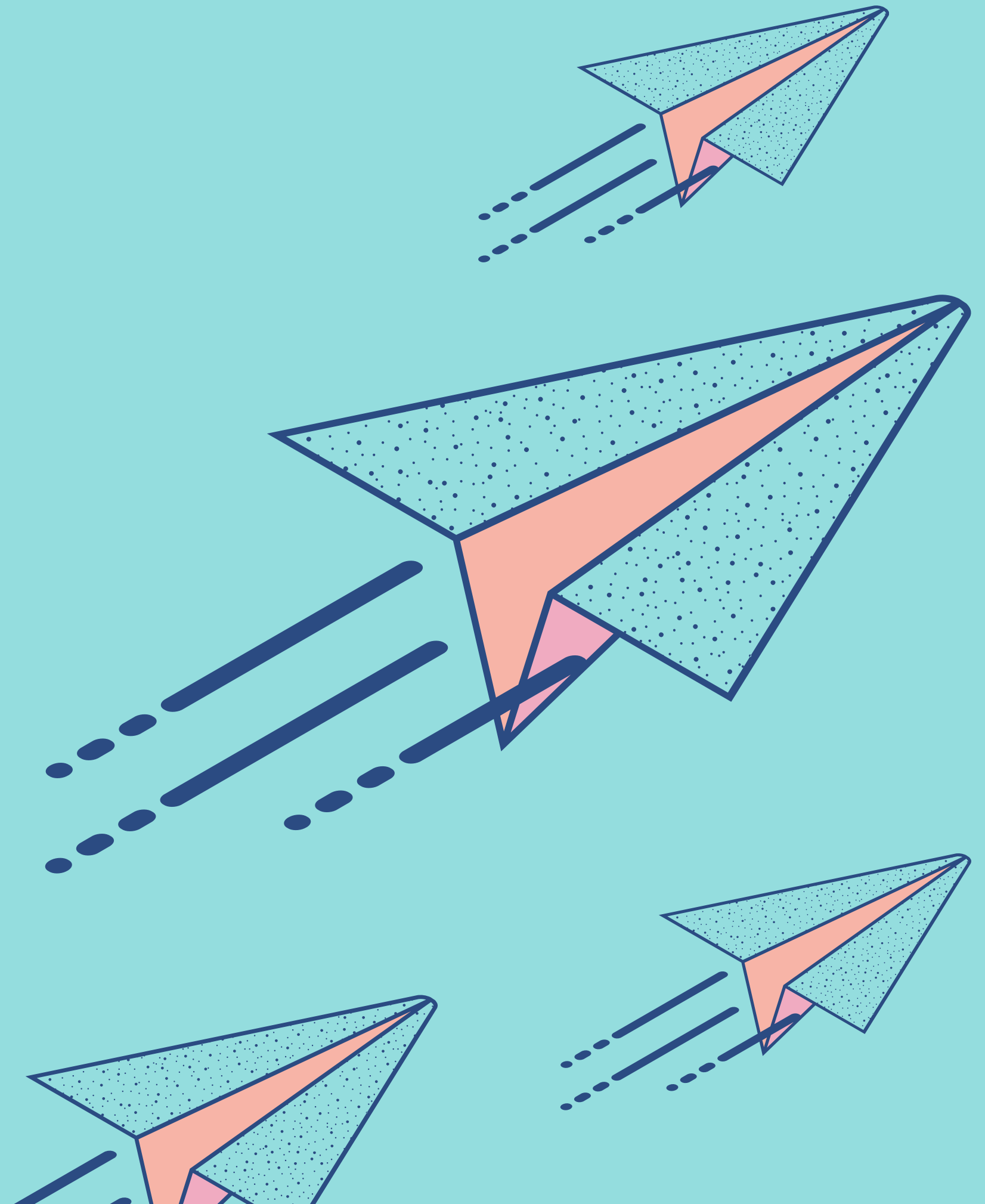


¿Cuál es la diferencia
entre git push y git pull?



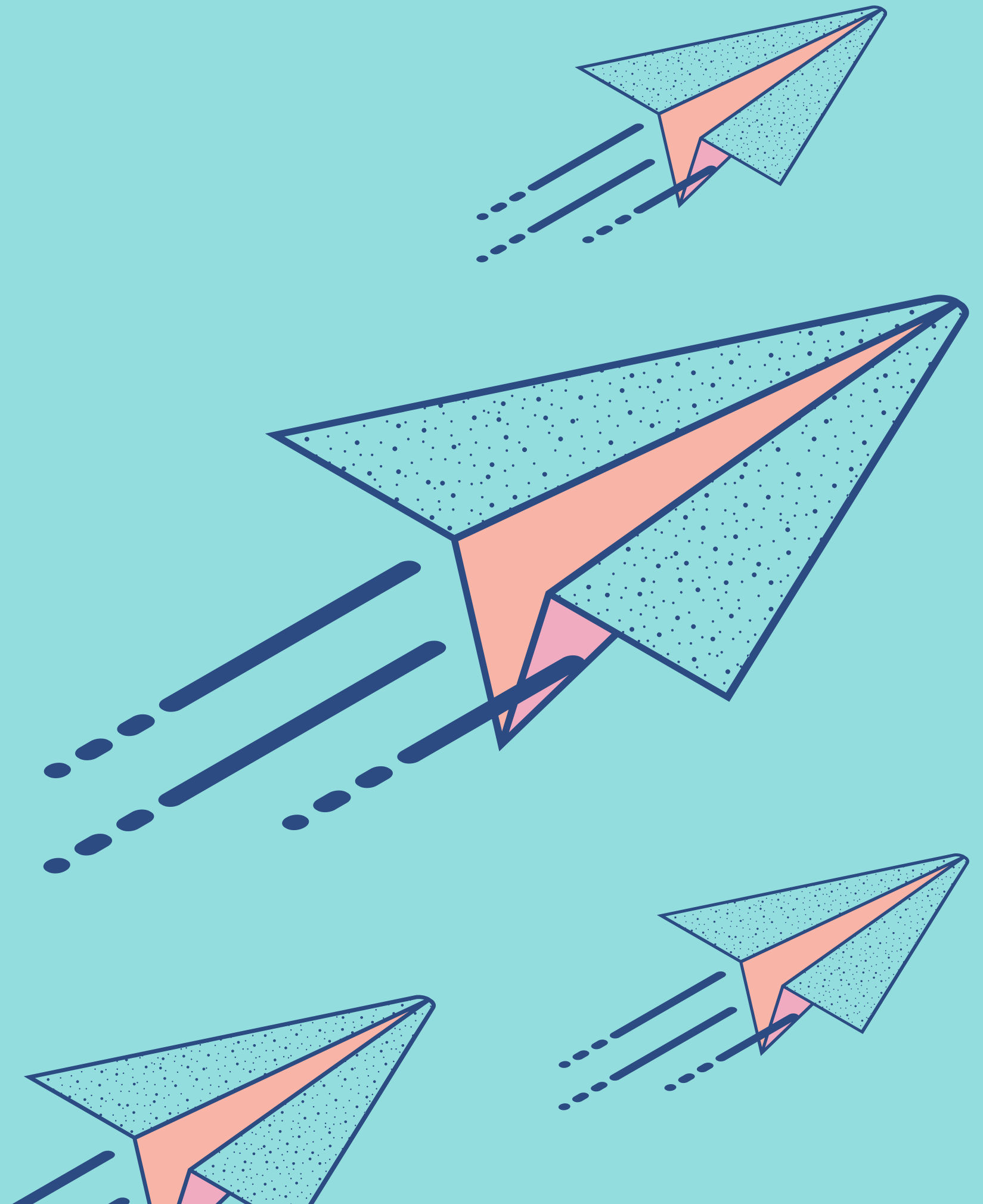
git push

Nos sirve para **empujar** cualquier cambio o modificación del repositorio local al repositorio remoto.



Experimentos con git push

1. Desde otra rama hacer push al main
2. La duda del git push -u
3. El peligro de git push -f
4. Borrar ramas remotas



Comandos git push

git push

git push -u origin <rama>

git push --all

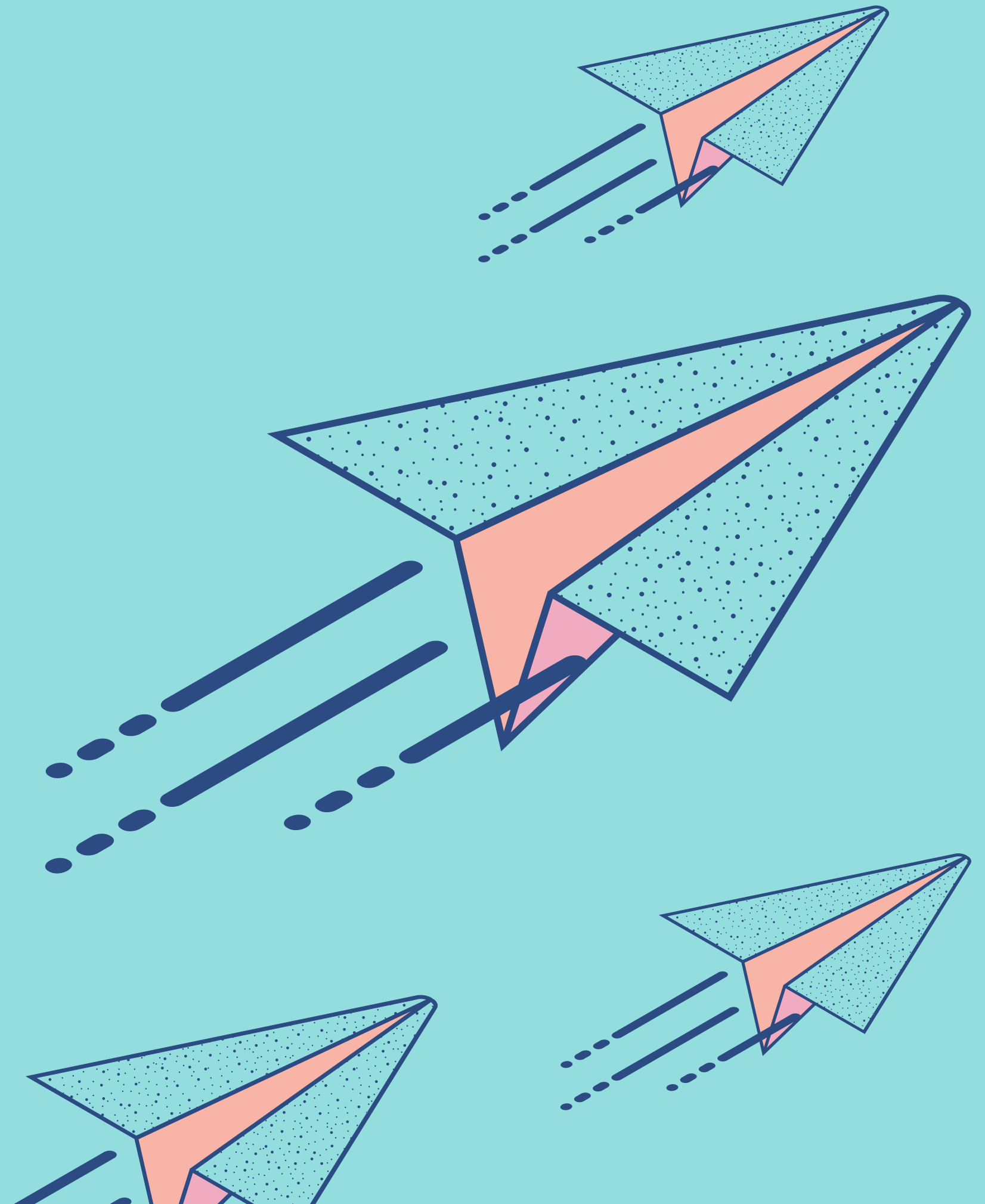
git push origin <rama1> <rama2>
<ramaN>

git push -f

git push -d origin <rama>

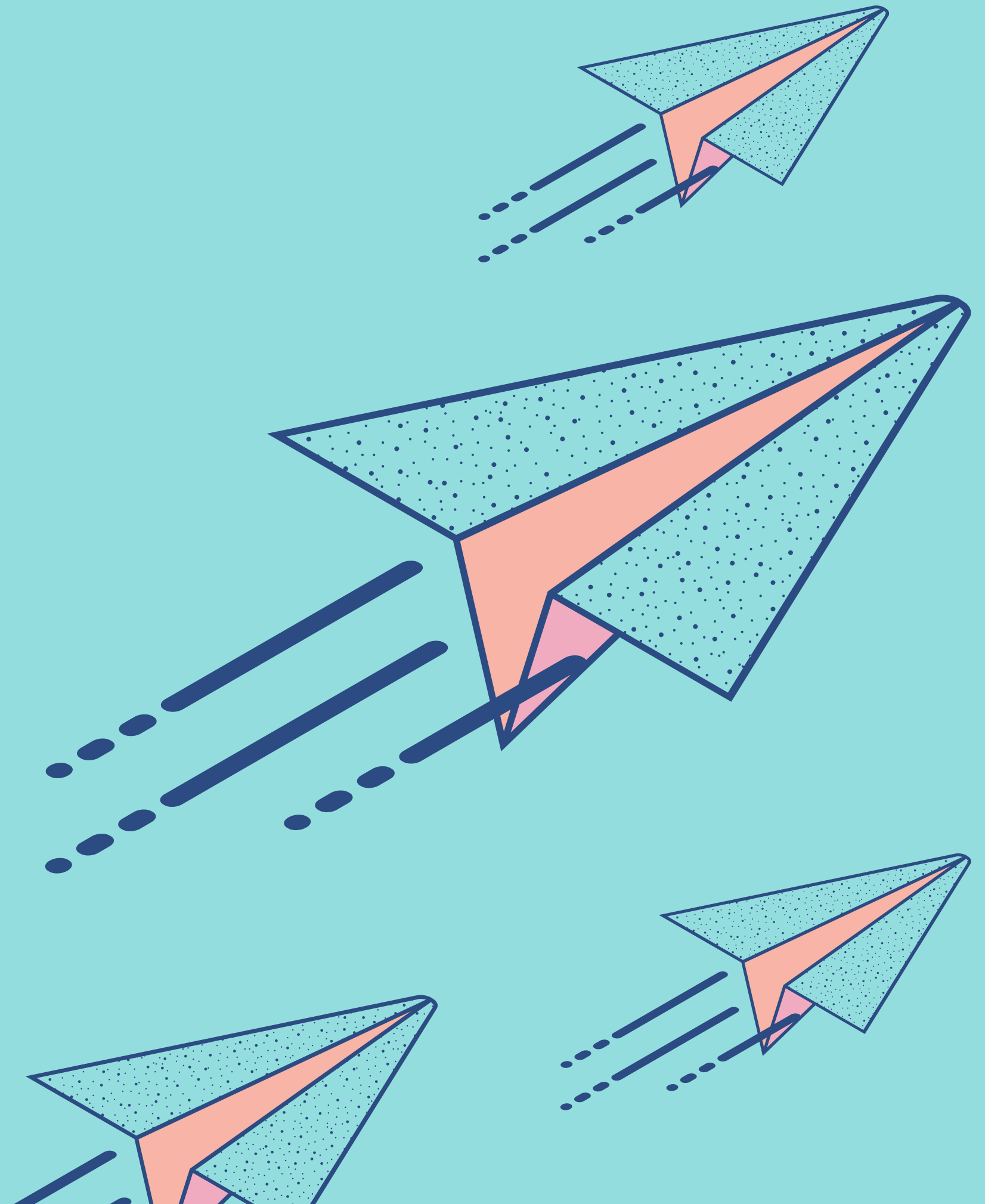
git pull

Nos sirve para **jalar** o **descargar** los cambios o modificaciones del repositorio remoto al repositorio local.



Experimentos con git pull

1. Traernos cambios de otras ramas a la actual
2. Traernos los cambios de varias ramas a una sola
3. Quiero traerme cambios solo con git pull
4. ¿Se pueden producir conflictos con git pull?



Comandos git pull

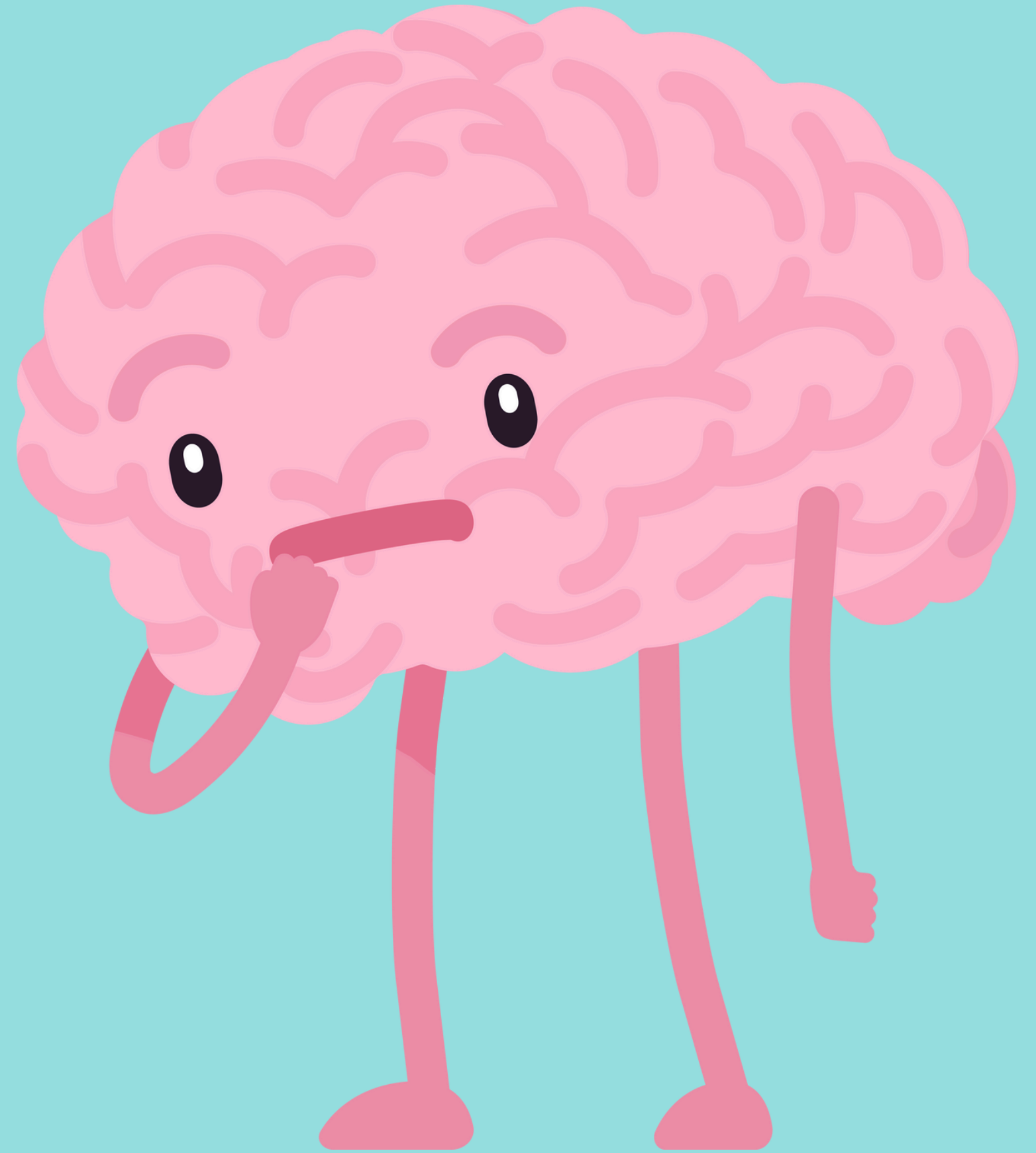
```
git pull
```

```
git pull --set-upstream origin  
<rama>
```

```
git pull --all
```

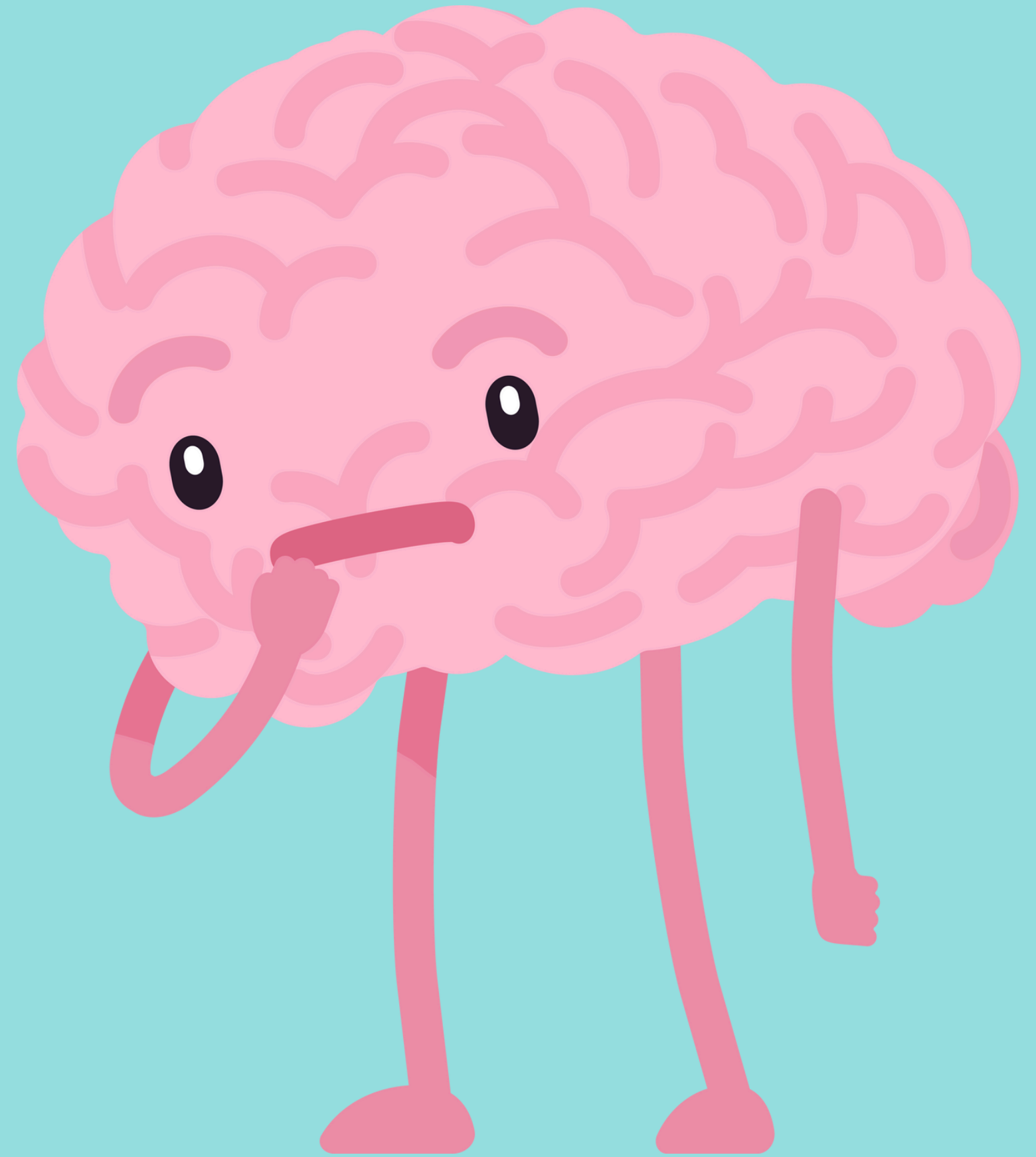
```
git pull origin <rama1> <rama2>  
<ramaN>
```


**¿Qué es una
Pull Request?**



¿Qué es una Pull Request?

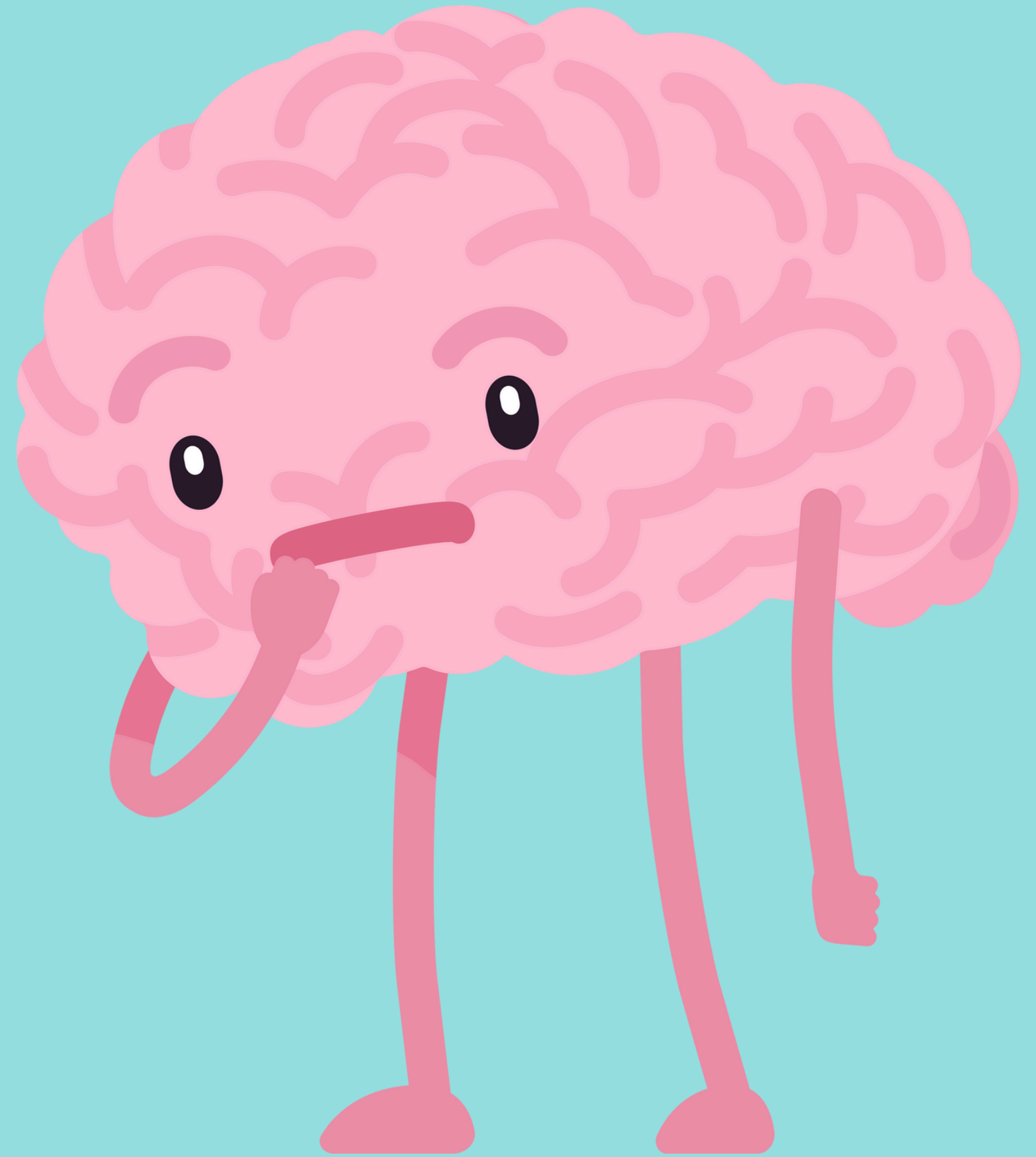
Una Pull Request, o de forma abreviada **PR**, es una petición de cambios que se envía al repositorio original.



¿Cómo se hace una PR?

Tenemos que subir nuestra rama con **git push** y a continuación hay 2 maneras:

1. La rama la subiste recientemente y te aparece la opción en GitHub
> Code
2. Irnos al apartado Pull Request



Hacer una buena PR

1. **Enfoca tu código en una sola cosa**, es mucho más fácil revisar y aceptar una Pull Request que hace una sola cosa a revisar una Pull Request que aprovecha a hacer muchas cosas.
2. **Explica tu Pull Request**, y si una imagen vale más que mil palabras. ¿Qué puede valer un GIF o un vídeo mostrando la funcionalidad? Existen muchas aplicaciones y herramientas que nos permiten grabar la pantalla.



Revisar una PR

1. Proporciona siempre feedback positivo
2. Concreción y claridad
3. **Entiende el contexto**, es posible que a veces tengamos que poner paños calientes o parches a nuestro código y que, pese a no ser el más bonito, sí que cumpla su cometido.

