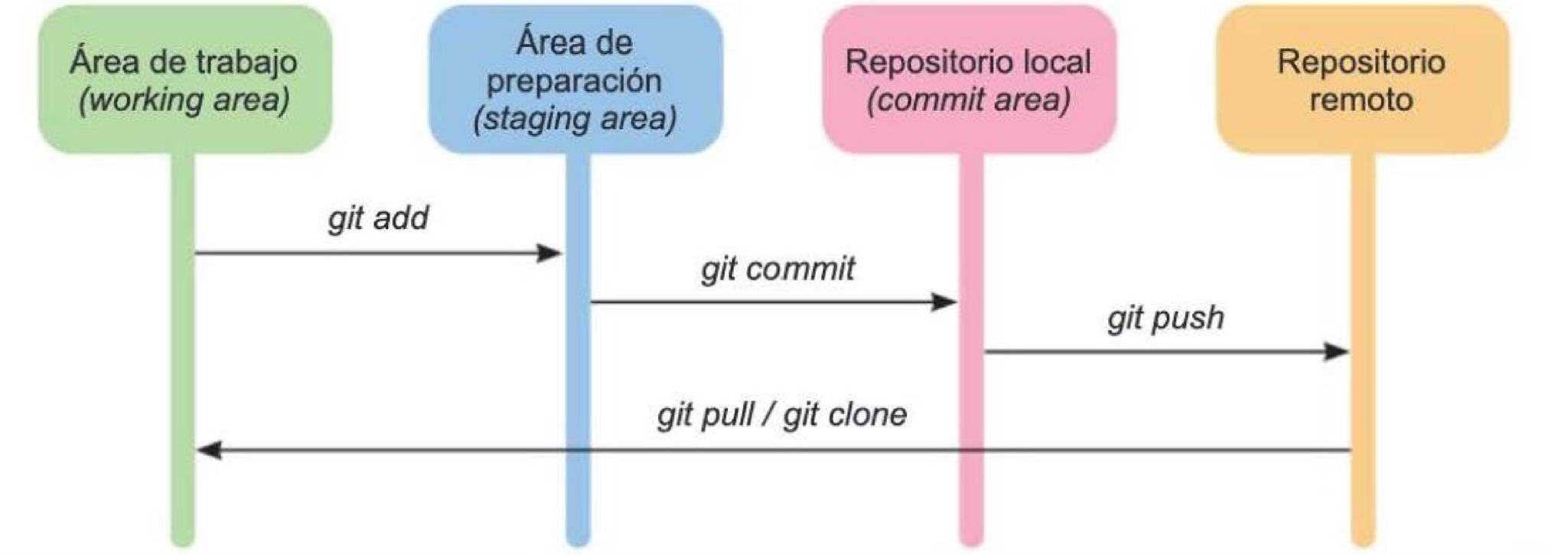
# Git

Clase 06

# Eliminar archivos

# Eliminar archivos



# Eliminar archivos

¿Cómo eliminamos archivos de nuestro repositorio?

rm, git add VS git rm

git rm --cached

# ¿Qué pasa al mover archivos?

# ¿Qué pasa al mover archivos?

Vamos a ver qué pasa cuando renombramos un archivo

Hay que realizar dos pasos

Opcional: git mv antiguo nuevo

# git log

# git log

El comando git log en Git es utilizado para revisar el historial de commits que se han realizado en un repositorio.

Este comando es extremadamente útil para rastrear cambios, entender la historia del desarrollo y realizar diagnósticos

# git log

```
git log --oneline
```

# git log

```
git log --oneline --reverse
```

# git log

git log --oneline --graph

Muy útil ahora cuando empecemos con las ramas

# Sacar ficheros de staging

# git restore

```
git restore --staged nombre_fichero
```

# Etiquetas

# Etiquetas

git tag vX.X hash_del_commit

git tag -a vX.X -m “Comentario”

git log --oneline

git checkout vX.X

git tag (listar todas las etiquetas)

git tag -n (para ver versiones y comentarios)

# Etiquetas

Con git checkout puedo navegar entre etiquetas

# Ramas II

# Ramas

Para qué se utilizan

- Varios desarrolladores
- Varias versiones
- Desarrollo
- Poducción
- Errores

# Ramas

branch, git checkout -b

git branch -m (renombrar ramas)

checkout, switch

merge

# Ramas

Comandos basicos de Git

## git ¿Que es un Branch y cómo funciona un Merge en Git?

### Ideas

**Master:**
Todo lo que esta en esta rama va a producción.

**Development:**
Las nuevas features, características y experimentos

**HotFix:**
Aqui van los errores se solucionan tan pronto como sea posible.

Si las ramas tienen algun conflicto para unirse git te avisara y te pedira que los corrijas, los conflictos pueden deberse a que se modficaron las mismas lineas del archivo en las dos ramas.

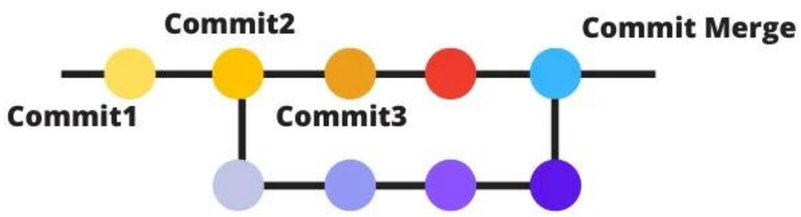

### Notas Clase

Cuando creas tu repositorio en tu carpeta de trabajo se crea por defecto una rama (Branch) principal llamada master.

Branch se puede ver como el mapa lineal de los commits que haz realizados al archivo.


Usas checkout para crear una nueva rama desde el commit que desees de tu rama master, esta rama te sirve para hacer experimentos o reparar errores de tu código principal sin afectar al mismo.


Y para unir los cambios de esta rama de prueba con master utilizas merge, de esta manera las dos ramas se unirán formando una nueva rama.




### Resumen

Branch es aquella que representan los commits como un mapa lineal de tiempo, es posible crear nuevas ramas para realizar modificaciones de nuestro código sin afectar la rama principal.

Merge es el comando que se usa para unir dos ramas, generalmente los merge se hacen desde la rama master, se debe tomar en cuenta que puede haber conflicto entre las ramas.

@YisusJoe

# Ramas

- Usar ramas
- Comparar ramas y fusionarlas
- Resolver conflictos
- Qué es el stash

# Tipos de merge

# Fast-Forward VS 3 way merge

# Ejercicio práctico

# Ramas

- Comparar ramas con
  - git log  rama..rama
- Comparar cambios con diff
  - git diff rama..rama

# ‹Despedida›

**Email**

bienvenidosaez@gmail.com

**Instagram**

@bienvenidosaez

**Youtube**

youtube.com/bienvenidosaez