CONCEPTOS BASICOS

* Directorio de trabajo: espacio donde se realizan los trabajos, es decir, allí donde hacemos las modificaciones.
* Espacio temporal: Donde almacenamos los trabajos realizados antes de marcarlos como definitivos.
* Repositorio: Una vez tenemos claro que los datos son correctos, los convertimos en definitivos desde el espacio temporal al repositorio.

Comando: git add

Comando: git commit -m “texto”

Repositorio

Espacio temporal

Directorio de trabajo

RAMAS

Una rama es una derivación de MASTER que nos permite realizar cambios sin que afecte al repositorio original (MASTER).

Cuando se crea una rama, coge todos los commit que hay actualmente en MASTER y los copia a la rama creada. Desde la RAMA creada podremos seguir haciendo commits que no se verán reflejados en MASTER u otras RAMAS.

**Cuando vamos cambiando de ramas, los archivos físicamente cambian a como estaban en esa rama.**

Cuando queremos aplicar los cambios a MASTER, deberemos fusionar la rama con MASTER y los cambios se añadirán a MASTER y si hay conflictos, lo notificará y deberemos modificar los cambios de la rama para que no entren en conflicto con los de MASTER

RAMA1

MASTER

MASTER

MASTER

Comando: **git merge rama 1**

Las modificaciones hechas en rama 1 pasarán a estar en MASTER pero sigue existiendo RAMA1

Comando: **git Branch rama1**

En este punto pasamos a tener 2 ramas iguales

REPOSITORIOS REMOTOS

Nos permite vincular un repositorio local con uno remoto a través de un alias. Es decir, el repositorio MASTER estará vinculado con el repositorio remoto.

MASTER LOCAL

PRUEBAS REMOTO

<https://aprendeconalf.es/docencia/git/manual/creacion-actualizacion-repositorios/>

COMANDOS GIT

|  |  |  |
| --- | --- | --- |
| Comando | [GRUPO ] Funcion | Ejemplo |
| Git init | [CREACION] Crear directorio de trabajo | Git init practica1 |
| Git add | [CREACION] Añade TODOS los cambios del directorio de trabajo al espacio temporal | Git add . |
|  | [CREACION] Añade un cambio un archivo o carpeta del directorio de trabajo al espacio temporal | Git add prueba1.txt |
| Git commit -m “mensaje” | [CREACION] Añade los cambios del espacio temporal al repositorio | Git commit -m “Añadido prueba1.txt” |
| Git commit -- amend -m “mensaje” | [CREACION] Modifica el texto añadido al ultimo commit hecho | Git commit --amend -m “Añadido prueba1.1.txt” |
| Git status | [HISTORIAL]Comprueba el estado del directorio de trabajo. **Los cambios pendientes de subir al repositorio ubicados en el espacio temporal y los archivos pendientes de añadir al espacio temporal** | Git status |
| Git log | [HISTORIAL] Muestra información detallada de cada uno de los commit realizados | Git log |
| Git log --oneline | [HISTORIAL] Muestra en una línea cada commit con su comentario | Git log --oneline |
| Git show | [HISTORIAL] Muestra información del último commit y su comparación con el anterior | Git show |
| Git show commit | [HISTORIAL] Muestra información de un commit especifico y su comparación con el anterior | Git show 060c29c 1\* |
| Git annotate archivo | [HISTORIAL] Visualiza el contenido del archivo y nos indica en que commit se añadió cada línea de información | Git annotate prueba1.txt |
| Git diff | [HISTORIAL] Muestra las diferencias entre el directorio de trabajo y la zona de intercambio | Git diff |
| Git diff --cached | [HISTORIAL] Muestra las diferencias entre la zona de espacio temporal y el repositorio | Git diff --cached |
| Git diff head | [HISTORIAL] Muestra las diferencias entre el directorio de trabajo y el ultimo commit | Git diff head |
| Git diff head~N | [HISTORIAL] Muestra las diferencias entre el directorio de trabajo y el commit número especificado en N | Git diff head~3 |
| Git checkout -- fichero | [CAMBIOS] Restaura el archivo especificado del directorio de trabajo a como estaba en el último commit | Git checkout -- prueba1.txt |
| Git checkout . | [CAMBIOS] Restaura **TODOS** los archivos del directorio de trabajo a como estaban en el último commit | Git checkout . |
| Git checkout códigohead -- fichero | [CAMBIOS] Restaura el archivo especificado del directorio de trabajo a como estaba en el commit especificado. **TAMBIEN VALIDO CON .** | Git checkout 060c29c – prueba1.txt |
| Git reset -- fichero | [CAMBIOS] Elimina el archivo especificado del área temporal | Git reset -- prueba1.txt |
| Git reset \* | [CAMBIOS] Elimina TODOS los archivos del área temporal | Git reset . |
| Git reset codigohead | [CAMBIOS] Elimina TODOS los commit desde el especificado hacia adelante. **No borra el especificado** | Git reset c477198 |
| Git Branch nombrerama | [RAMAS] crea una rama nueva | Git Branch rama1 |
| Git checkout nombrerama | [RAMAS] cambia a la rama especificada | Git checkout rama1 |
| Git merge nombrerama | [RAMAS] fusiona la rama especificada con la que estás actualmente | Git merge rama1 (ESTANDO EN MASTER) |
| Git Branch -d nombrerama | [RAMAS] borra la rama especificada siempre que haya sido fusionada antes | Git Branch -d rama1 |
| Git Branch -D nombrerama | [RAMAS] borra la rama especificada aunque no haya sido fusionada antes | Git Branch -D rama1 |
| Git Branch | [RAMAS] muestra un listado de ramas creadas | Git branch |
| Git remote add alias URL | [REMOTO] Crea un enlace entre tu repositorio local y uno remoto a través de un alias. | git remote add remoto https://github.com/Cristia-IG/pruebas.git |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1\*: podemos conseguir el código del commit haciendo “Git log –oneline”

COMANDO DE LINUX GENERALES

|  |  |  |
| --- | --- | --- |
| Comando | Funcion | Ejemplo |
| Cd | Desplazarse por los directorios |  |
| Mkdir | Crear carpetas |  |
| Rm | Eliminar archivos |  |
| Rmdir | Eliminar carpetas |  |
| Ls | Visualizar el contenido de una carpeta |  |

CASOS PRACTICOS

Git reset