Configuración

Necesitas un nombre de usuario y una dirección de correo electrónico válida para trabajar con Git. Así es como se configura la identificación:

|  |  |
| --- | --- |
| Crea un nombre | git config --global user.name "Tu nombre" |
| Introduce tu dirección de correo electrónico | git config --global user.email "correo\_de\_ejemplo@ejemplo.es" |

Crear repositorio

Crea un nuevo repositorio o descarga un repositorio existente.

|  |  |
| --- | --- |
| Crear y nombrar un nuevo repositorio local | git init nombre\_de\_ejemplo |
| Copiar un repositorio existente y su historial con [git clone](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-clone/" \o "Git clone" \t "_top) | git clone "http://web-de-ejemplo.es" |

Realizar cambios

Puedes hacer, seguir y añadir cambios.

|  |  |
| --- | --- |
| Mostrar el estado del directorio | git status |
| Mostrar objetos | git show |
| Añadir un archivo | git add archivo\_de\_ejemplo |
| Añadir todos los archivos de un repositorio | git add \* |
| Mostrar todos los archivos nuevos o modificados con [git diff](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-diff/" \o "Git diff" \t "_top) | git diff |
| Mostrar cambios en archivos en staged | git diff --staged |
| Seguimiento de los cambios después de la confirmación | git diff HEAD |
| Mostrar las diferencias entre el Index y el estado actual de un archivo | git reset archivo\_de\_ejemplo |
| Añadir permanentemente al historial de versiones los archivos actualmente indexados con [git commit](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-commit/" \o "Git commit" \t "_top) | git commit -m "Explicaciones de los cambios". |

Ramas

Te permite agrupar los cambios en ramas e integrar los nuevos cambios.

Crear y editar

|  |  |
| --- | --- |
| Crear una nueva rama | git branch nombre\_de\_ejemplo |
| Lista de todas las ramas | git branch --list |
| Borrar rama | git branch -d |
| Eliminar la rama remota | git push origin --delete |
| Cambiar el nombre de la rama | git branch -m |

Cambiar de rama

|  |  |
| --- | --- |
| Cambiar a otra rama con [git checkout](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-checkout/" \o "Git checkout" \t "_top) | git checkout otra\_rama |
| Crear una nueva rama y cambiar a ella | git checkout -b |

Fusionar y buscar

|  |  |
| --- | --- |
| Fusionar el historial de una rama con la rama actual | git merge nombre\_de\_ejemplo |
| Registrar el repositorio externo y el historial de intercambio | git fetch "http://www.web-de-ejemplo.es" |
| Registrar todas las ramas | git fetch --all |
| Registrar el repositorio local | git fetch origin |

Caché

|  |  |
| --- | --- |
| Transferir los archivos actuales sin confirmar la rama con [git stash](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-stash/" \o "Git stash" \t "_top) | git stash |
| Guardar los cambios con una explicación | git stash save "Insertar explicación aquí". |
| Lista de stashes | git stash list |
| Volver a realizar cambios en el estado de la memoria | git stash apply |
| Seguimiento de los cambios en el estado de la memoria | git stash show |
| Usar y borrar directamente Stash | git stash pop |
| Descartar los resultados intermedios | git stash drop |
| Borrar toda la memoria disponible | git stash clear |
| Guardar en una rama separada | git stash branch rama\_de\_ejemplo |

Push y pull

Con [git push](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-push/" \o "Git Push" \t "_top) y [git pull](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-pull/" \o "Git pull" \t "_top) transfieres los commit al servidor remoto o los recuperas del servidor.

|  |  |
| --- | --- |
| Integrar la rama externa en la rama local actual | git push "http://www.web-de-ejemplo.es" rama\_local\_de\_ejemplo |
| Transferencia de datos al servidor remoto | git push origin master |
| Forzar push | git push -f |
| Retirar la rama remota a través de push | git push origin --delete ejemplo de rama |
| Extraer el historial desde un repositorio externo | git pull |
| Extraer datos del servidor | git pull origin master |

Registro

Puedes consultar el historial de un commit con [git log](https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/git-log/" \o "Git log" \t "_top).

|  |  |
| --- | --- |
| Mostrar todos los commit de una rama | git log |
| Limitar el número de commit (en este ejemplo a tres) | git log -3 |
| Buscar commit de un autor específico | git log --author= "nombre\_de\_ejemplo" |
| Mostrar los commit de un periodo de tiempo específico | git log <since>…<until> |
| Mostrar los commit con un archivo especial | git log --dato\_de\_ejemplo |

Revocar los cambios

Tienes dos opciones para revertir los cambios: git revert o git reset. Con este último, puedes elegir entre las opciones “soft”, “hard” y “mixed”.

|  |  |
| --- | --- |
| Cambiar el commit existente | git commit --amend |
| Eliminar archivo del staging area | git reset HEAD archivodeejemplo o git restore --staged archivodeejemplo |
| Descartar los cambios locales de un archivo en el staging area | git checkout --archivodeejemplo o git restore archivodeejemplo |