

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Wener V. Pacheco F.	1/7	Electiva C1-2025	24-01-2025

Title:

Comandos de Git

Keyword

clone
Repositorio
Remoto
Branch
Depth
SSH
HTTPS

Topic: Comando "Clone"

Notes: El comando "git clone" se utiliza para crear una copia exacta (clonar) de un repositorio remoto en el sistema local.

trae el historial del proyecto, incluyendo las ramas y los archivos.

`git clone <URL> [directorio]`

Para clonar un repositorio en específico:
`git clone -b <rama> <URL> [directorio]`

Questions

¿Cómo clonar solo una rama en específico?

¿Se puede clonar un repositorio privado? Si, pero pidiendo los permisos

Para clonar una rama específica:
`git clone --single-branch --branch <rama> <URL>`

Para clonar sin historial completo:
`git clone --depth <profundidad> <URL>`

Para clonar usando SSH:
`git clone git@<host>:usuario/<repositorio>.git`

Summary:

El comando "git clone" crea una copia exacta de un repositorio remoto en tu máquina local, incluyendo archivos, historial y ramas. Se puede usar con HTTPS o SSH para autenticación, clonar una rama o limitar el historial de opciones.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Wener V. Pacheco E.	2/7	Electiva C1-2025	24-01-2025

Title:

Comandos de Git

Keyword

push
Remoto
original
Branch
Forzar
Tags
Colaborativo

Topic:

Comando "push"

Notes:

El comando "git push" se usa para enviar los cambios que hiciste en tu repositorio local al repositorio remoto.

Es un paso super necesario para compartir nuestro trabajo con otros colaboradores o guardar nuestros avances en un servidor.

`git push <nombre-remoto> <rama>`

Questions

¿Qué pasa si no uso el remoto predeterminado?
¿Cómo enviar cambios a una rama distinta del remoto?
¿Cuándo se utiliza "Force"?

Para enviar una rama específica:

`git push origin main`

Para subir todas las ramas:

`git push --all`

Para forzar el envío:

`git push --Force`

Para establecer el repositorio remoto predeterminado: `git push -u origin main`

Summary:

El comando "git push" sube los cambios guardados en tu repositorio local al servidor remoto, permitiendo compartir nuestros proyectos o respaldarlos. Podemos especificar la rama, enviar todas las ramas o incluso forzar cambios con opciones adicionales. Es esencial combinarlo con "git pull" para evitar conflictos.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Wener V. Pacheco E.	3/7	Electiva C1-2025	24-01-2025

Title:

Comandos de Git

Keyword

Pull
Fetch
Merge
Rebase
Conflicts
Origin
Remote

Topic:

Comando "pull"

Notes:

El comando "git pull" se utiliza para traer y fusionar los cambios más recientes de un repositorio remoto con tu repositorio local.

Combina dos comandos en uno: git Fetch (traer cambios) y git merge (integrarlos).

git pull <nombre-remoto> <rama>

Questions

¿Cómo evitar conflictos al hacer "pull"?
¿Cuál es la diferencia entre "merge" y "rebase"?
¿Cuándo usar "--no-commit"?

Para traer una rama específica y fusionarla:
git pull origin main

Para hacer un rebase en lugar de una fusión: git pull --rebase

Para traer solo cambios sin fusionarlos:
git remote

Para evitar conflictos grandes:
git pull --no-commit

Summary:

El comando "git pull" trae los cambios de un repositorio remoto y los combina automáticamente con los de tu repositorio local. Es ideal para mantener su trabajo actualizado con el equipo. Puede usar opciones como "--rebase" para un historial más limpio o "--no-commit" para controlar la fusión manualmente.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Werner V. Pacheco B.	4/7	Electiva C1-2025	24-01-2025

Title: Comandos de Git

Keyword	Topic: Comando "commit"
Commit add mensaje amend Historial Archivos Allow Snapshot	Notes: El comando "git commit" se utiliza para guardar los cambios realizados en los archivos en el historial del repositorio. Cada commit representa un "punto de control" que permite rastrear el progreso y revertir cambios, si es necesario. git commit -m "Mensaje a enviar"
Questions	Para editar el mensaje del último commit: git commit --amend Para hacer commit solo de ciertos archivos: git commit <archivo 1> <archivo 2> ¿Qué diferencia hay entre commit y push? ¿Qué pasa si no añado archivos con 'git add' antes del 'commit'? ¿Incluir cambios no incluidos en el anterior? git commit -a -m "actualización" Hacer un commit vacío (sin cambios): git commit --allow-empty -m "proyecto"

Summary: El comando "git commit" guarda los cambios preparados en un repositorio como un punto de control en el historial. Es importantes escribir mensajes claros para explicar los cambios. Se puede usar la opción de "--amend" para modificar el último "commit" o "-a" para incluir automáticamente todos los cambios.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Wener V. Pacheco B.	5/7	Electiva C1-2025	24-01-2025

Title:

Comandos de Git

Keyword

Branch
checkout
Delete
Remote
merge

Topic: Comando "Branch"

Notes:

El comando "git branch" se utiliza para gestionar las ramas de un repositorio.

Las ramas permiten trabajar en diferentes líneas de desarrollo sin afectar la rama principal (main o master).
Para ver ramas existentes:
git branch

Questions

¿Qué diferencia

hay entre una rama local y una remota?
Para eliminar una rama:
git branch -d <nombre-rama>

¿Cómo fusionar ramas sin conflictos?

Para renombrar una rama:
git branch -m <nombre-rama>

¿Cómo listar todas las ramas?

Para ver todas las ramas:
git branch -a

Summary:

El comando "git branch" permite gestionar ramas en un proyecto, creando espacios separados para nuevas funcionalidades o pruebas. Se puede crear, eliminar, renombrar y listar ramas, tanto locales como remotas. Trabajar con ramas organizadas mejora los cambios y evita problemas en el desarrollo colaborativo.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Wener V. Pacheco F.	6/7	Electiva CI-2025	24-01-2025

Title: Comandos de Git

Keyword

Fetch
Remote
tags
Branches
Sync
Origin

Topic: Comando "Fetch"

Notes: El comando "git fetch" se utiliza para descargar los últimos cambios de un repositorio remoto, sin fusionarlos automáticamente con el repositorio local.

Para descargar cambios del remoto predeterminado: `git Fetch`

Para especificar un remoto y una rama:
`git Fetch <nombre-remoto> <rama>`

Questions

¿Cuál es la diferencia entre `git Fetch --all` y `git pull`?

¿Cómo saber qué cambios hay después de un `git Fetch`?

Para traer todas las ramas y etiquetas:
`git Fetch --all`

Para obtener etiquetas específicas:
`git Fetch --tags`

Para comprobar diferencias entre ramas:
`git log HEAD..origin/<rama>`

Para obtener cambios sin actualizar rama:
`git Fetch --no-tags`

Summary:

El comando "git Fetch" descarga actualizaciones del repositorio remoto sin fusionarlas automáticamente. Te permite revisar los cambios antes de integrarlos, ejerciendo más control sobre el flujo de trabajo. Es ideal para mantener un repositorio sincronizado sin apertar las ramas locales.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Wenera V. Pacheco F.	7/7	Electiva C1-2025	24-01-2025

Title:

Comandos de Git

Keyword	Topic: Comando "Tags"
tag Anotado Lightweight Versioning release push	Notes: El comando "git tags" se utiliza para marcar puntos específicos en el historial del repositorio, como versiones importantes, lanzamientos o hits. Los tags (etiquetas) son inmutables y facilitan identificar commits clave de forma permanente. Para crear un tag ligero: git tag <nombre-etiqueta> Para crear un tag anotado: git tag -a <nombre-etiqueta> -m " " Para listar todas las etiquetas: git tag Para mostrar detalles de un tag en específico: git show <nombre-etiqueta>
Questions ¿Cuál es la diferencia entre un tag ligero y uno anotado? ¿Cuándo es útil usar tag en un proyecto?	

Summary:

El comando "git tag" marca puntos importantes en el historial del repositorio, como versiones o lanzamientos. Puedes usar tags ligeros o anotados según el nivel de detalle necesario. Los tags son útiles para mantener un control claro de versiones y compartir hits en el equipo.