

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Isai Mónica Marquy	1/7	Electiva	24/1/2023

Title: Comandos del GIT

Keyword

Topic: Clone

Copy  
Repository

Project

Remote

Repository  
Local

Notes:

El comando clone se utiliza para Copiar un repositorio remoto en tu computadora. Cuando lo ejecutas crea una copia exacta del repositorio, incluyendo su historial de cambios, ramas, y etiquetas. Se puede usar con URLs HTTP, SSH o incluso rutas locales.

Questions

¿Cómo puedo  
usar clone

Para trabajar  
en diferentes

ramas desde  
el inicio del  
proyecto?

Después de clonar, tendrás el repositorio en una carpeta en tu máquina, luego para hacer cambios tu siguiente comando es `[git clone <URL del repositorio>]`.

También puedes especificar el nombre de la carpeta donde se clonará con: `[git clone <URL> <nombre carpeta>]`.

El uso de los comandos viene usado de GIT

Summary:

El comando git clone sirve para Copiar un proyecto de Internet a tu computadora. Así puedes trabajar en el proyecto junto con otra persona. Solo necesitas la dirección del Proyecto y URL.

By Carlos Pichardo Vique



NAME

Isai Mojica

PAGES

2/7

SPEAKER/CLASS

Electiva

DATE - TIME

24/1/23

Title:

Comandos del GIT

Keyword

Enviar  
Cambios  
Ramas  
Repasos  
Remoto  
Sincronizar  
Conflicto.

Questions

¿Que pasa  
si intentas  
hacer git  
push pero  
tu rama  
tiene esta  
desincronización?

Topic: Push

Notes:

El comando push se utiliza para enviar los cambios realizados en tu repositorio local hacia el repositorio remoto. Esto incluye commits, ramas nuevas y actualizaciones.

Cuando trabajas en equipo push permite compartir tus avances con la demás. Por defecto, los cambios se mandan a la rama actual en el repositorio remoto.

La sintaxis es [git push <remote> <rama>]. El nombre remoto suele ser origin y la rama principal su su nombre o master.

También puedes hacer todos los cambios con [git push -all].

Summary:

El comando push sirve para enviar tu trabajo al proyecto en internet. Es como guardar tus avances para que otros los vean. Solo debes asegurarte de que no choquee con lo que otros hicieron.



NAME

Ivan Mojica

PAGES

3/7

SPEAKER/CLASS

Electronica

DATE - TIME

24/1/23

Title:

Codigoz del GIT

Keyword

Actualizar  
Descargar  
Cambios.  
Fusion  
Conflicto  
Incorporar

Topic: Pull

Notes:

El comando pull se utiliza para actualizar tu repositorio local con los últimos cambios del repositorio remoto. Consiste en dos acciones: descargar los cambios [git fetch] y fusionarlos con tus propios cambios [git merge].

Questions

¿Que pasa si voy en pull pero me da error?  
¿Como se resuelve?  
¿Como se evita?

Es útil cuando trabajas en equipo, ya que te asegura que tu trabajo este sincronizado con lo que otros han hecho.

La sincronización se [git pull <nombre-remota> <branch>] donde el nombre tiene su origen. Si no especifica una rama, toma la que este usando.

Es importante este git pull para de realizar tus propios cambios.

Summary:

El comando Pull sirve para traer los cambios del repositorio de Internet a tu computadora. Es como actualizar tu copia para que sea igual que la de tus compis.



NAME  
Isaia Mojica

PAGES  
4/7

SPEAKER/CLASS  
Electura

DATE - TIME  
24/1/23

Title: Codigo del GIT

Keyword

Guardar  
Cambios  
Área  
Mensaje  
Describe  
Local  
Commit  
version.  
Questions

Topic: Commit

Notes:

La utilidad Commit para guardar cambios en el historial del repositorio de forma permanente. Representa un punto seguro en el tiempo que registra qué se modificó y por qué. Cada commit incluye un mensaje que describe brevemente los cambios realizados, lo cual es importante para mantener un historial claro y comprensible.

La sintaxis es "[Commit -m 'Mensaje']" ya no usar el -m, git asume el editor de texto para que escribas el mensaje.

Antes de hacer un commit, debes añadir los archivos al área de preparación usando git add.

Summary:

El comando commit guarda tus avances como si fueran una foto de tu trabajo. Puedes escribir una nota descriptiva acerca de los cambios, como "añadiendo un nuevo archivo".

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Icar Mojica	5/7	Electura	24/1/23

Title: *Comandos del GIT*

Keyword

Topic: *Branch*

*Branch  
Create  
Commit  
Push  
Merge  
Delete*

Notes: El comando Branch se utiliza para gestionar los ramos en un repositorio. Una rama es una línea separada del trabajo, útil para desarrollar nuevas características sin afectar la versión principal del proyecto.

*Desarrollando*

Con el comando `git branch` puedes ver los ramos existentes, crear nuevos, eliminar o cambiar entre ellos. La sintaxis para crear una nueva rama es `[git branch <nombre-rama>].` Para eliminar todos los ramos simplemente usa `git branch -D`.

Questions

Para las Commits [git checkout <nombre-rama>] o el comando `git commit` de [git push <nombre-rama>].

Summary:

El comando `git branch` te ayuda a crear caminos diferentes para trabajar del repositorio. El comando `git branch` te ayuda a crear una nueva rama lo que ya tienes.



NAME	PAGES	SPEAKER/CLASS	DATE-TIME
Isai Mayra	6/7	Electronica	23/1/24

Title: *Codigos del git*

Keyword	Topic: <i>Fetch</i>
Descargar Repository sin fuerza Cambios Central Preservacion.	Notes: El comando <i>Fetch</i> se utiliza para descargar actualizaciones del repository. Renueva tu repository local, pero sin fusionarlo automaticamente con tu rama actual.  Con este comando, puedes ver los nuevos commits, ramas o etiquetas que se han añadido en el remote, pero todavia tienes el control sobre cuando integras.
Questions	

La sintaxis es: *[git Fetch <remote-name>]*  
 Donde el remote suele ser *origin*.  
 Si no especificas una rama, descargara informacion de todos los ramas.

Summary: El comando *Fetch* trae los cambios del proyecto de internet a tu computadora, pero no los mezcla. Es como ver que hay de nuevo sin de *pull* fue cosas.

By Carlos Ricardo Ciro

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Isar Mejia	7/7	Electura	23/1/24

Title: Codigos del GIT

<b>Keyword</b>	<b>Topic:</b> tag
Etiquetas. Puntos. Lanzamientos Anuncios Historias Compartir.	<p><b>Notes:</b> Se utiliza para crear puntos de referencia en el historial de un repositorio. Estos puntos son como "etiquetas" que marcan versiones importantes, como lanzamientos de software. Existen dos tipos de etiquetas: ligeras y anotadas.</p> <p>La sintaxis para crear una etiqueta con rigidez es <code>[git tag &lt;nombre-etiqueta&gt;]</code>. Para una anotacion es <code>[git tag -a &lt;nombre-etiqueta&gt; -m "Mensaje"]</code>.</p> <p>Puedes crear todas las etiquetas con <code>git tag</code> y eliminar una con <code>git tag -d</code> para eliminar etiquetas con un repositorio remoto, debe usar <code>git push origin</code>.</p>
<b>Questions</b>	

**Summary:** El comando `git tag` se usa para crear una etiqueta en un repositorio local. Por ejemplo, puedes decir "este es la version 1.0" así como decir "esta es la nueva version".