

### 1) ¿Qué es un Git?

Es un software de control de versiones que fue diseñado por Linus Torvalds, pensado en almacenamiento y confiabilidad de versiones de aplicaciones cuando se tiene una gran cantidad de archivos de código fuente.

### 2) ¿Qué es Github?

Es una plataforma de desarrollo colaborativo de software para guardar proyectos utilizando el sistema de control de versiones Git.

### 3) ¿Qué es un branch?

Un "branch" o una rama (en español), es una ramificación que se crea cuando uno desea para almacenar sus archivos, aquí llamado instantáneas que son copias puntuales de los archivos completos, tal y como se encuentran en ese momento. Entonces al final de todo, una rama viene siendo un puntero que se crea con la dirección de donde está almacenada esa copia.

Estas se utilizan para desarrollar funcionalidades separadas de otras.

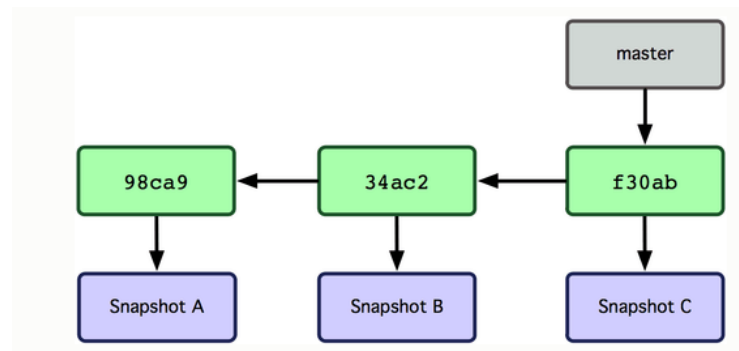


Figura 1. Ejemplo de branch.

### 4) ¿Qué es un commit?

Un "commit" es la acción de guardar o subir tus archivos a tu repositorio remoto, se puede decir que es una actualización de tus cambios, también puede hacerse al local, todo depende donde hayas creado tu repositorio.

### 5) ¿Qué se entiende cuando se dice que un archivo esta “staged”?

Hacer un “stage” es simplemente preparar el archivo para realizar un “commit”.

Sería guardar los cambios que se están realizando en la parte en que se está trabajando.

### 6) ¿Qué hace el comando git checkout?

El comando git checkout se utiliza para crear ramas o cambiar entre ellas.

Ejemplo para crear rama:

```
command git checkout -b <branch-name>
```

Ejemplo para cambiar entre ellas:

```
git checkout <branch-name>
```

### 7) ¿Qué hace el comando git stash?

Git stash ayuda a guardar los cambios que no están por ser “committed” inmediatamente, sino solo temporalmente. Se le puede llamar como guardado rápido provisional.

### 8) ¿Qué hace el comando git add?

El comando git add funciona para guardar archivos en el index. Por ejemplo, para guardar un archivo .txt en el index se haría de la siguiente manera.

```
git add temp.txt
```

### 9) ¿Qué es Pytest?

Es una librería de testing para Python. Nos ayuda a testear o probar la infraestructura o código de nuestro programa en Python.

#### **10) Bajo el contexto de pytest, ¿Qué es un assert?**

Se utiliza para verificar las expectativas y valores del aserto que uno espera de la función en el código. Si el aserto falla e retornara el valor que si da respuesta a la función especificada.

#### **11) ¿Qué es flake8?**

Es una herramienta muy útil, utilizada para revisar si hay variables declaradas que no están siendo utilizadas en el programa, como también imports y otros parámetros que no se están utilizando, pero si están declaradas. Esto nos ayuda a tener un código más limpio y eficiente.

## Bibliografía:

[1]"Git - Guardado rápido provisional", *Git-scm.com*, 2019. [Online]. Available: <https://git-scm.com/book/es/v1/Las-herramientas-de-Git-Guardado-r%C3%A1pido-provisional>. [Accessed: 10- Sep- 2019].

[2]"Git - Procedimientos básicos para ramificar y fusionar", *Git-scm.com*, 2019. [Online]. Available: <https://git-scm.com/book/es/v1/Ramificaciones-en-Git-Procedimientos-b%C3%A1sicos-para-ramificar-y-fusionar>. [Accessed: 10- Sep - 2019].

[3]"Git - ¿Qué es una rama?", *Git-scm.com*, 2019. [Online]. Available: <https://git-scm.com/book/es/v1/Ramificaciones-en-Git-%C2%BFQu%C3%A9-es-una-rama%3F>. [Accessed: 10- Sep - 2019].

[4]"Introducción — Conociendo GitHub 0.1 documentation", *Conociendogithub.readthedocs.io*, 2019. [Online]. Available: <https://conociendogithub.readthedocs.io/en/latest/data/introduccion/>. [Accessed: 10- Sep - 2019].

[5]"Una referencia visual de Git", *Marklodato.github.io*, 2019. [Online]. Available: <https://marklodato.github.io/visual-git-guide/index-es.html>. [Accessed: 10- Sep - 2019].