

1) ¿Diferencie la herramienta Git de Github?

Git es un software de control de versiones. Es de software libre y se puede mantener un historial completo de versiones, utilizando ramas o branches para acceder a cada versión de manera organizada.

Github es una compañía sin fines de lucro, en cuyo sitio web se utiliza Git para que desarrolladores almacenen su código y accedan a cada versión de manera organizada.

Por ello, la diferencia entre Git y Github es que Git es un software que facilita el control de versiones para desarrolladores y Github utiliza en su sitio web el software Git

2) ¿Qué es un branch?

Una branch es una rama independiente de la rama principal (o master) para poder programar y probar diferentes versiones de un mismo código, sin tener que sobrescribir o borrar el archivo original.

3) ¿Cómo se crea un nuevo Branch?

Para crear un nuevo branch en Git, primero se necesita tener una rama principal (o master), la cual se crea al iniciar un nuevo proyecto. Para crear una nueva branch, se usa el comando “git branch” seguido del nombre que se le quiera dar a esa rama, por ejemplo “git branch nueva_rama”

En el caso de Github, es necesario estar en el repositorio, luego dar click en “Branch:master” y en la ventana emergente poner nombre al nuevo branch y dar click en enter.

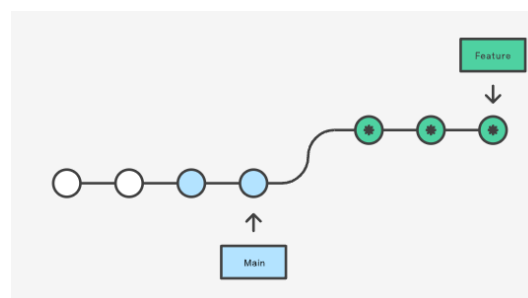
4) ¿Qué es un commit?

Un commit es un comando que guarda todos los cambios con una descripción, para recordar los cambios hechos. Si varios commits editan partes distintas del proyecto, no se sobrescriben entre sí.

5) ¿Qué es la operación “git stash”?

Un git stash es un comando que guarda temporalmente los cambios efectuados en el código en el que se está trabajando. Se guardan en un área llamada “stash” donde se almacenan temporalmente estos cambios, sin enviarlos al repositorio.

6) Explique de forma gráfica: ¿Qué sucede en mi Branch local cuando hago “git rebase master”?



Al utilizar el comando git rebase master, la rama feature se mueve hasta llegar al extremo de la rama main, incorporando las nuevas configuraciones que se encuentran en el main. Asimismo se reescribe el historial del proyecto creando nuevas configuraciones para cada confirmación de la rama original

7) ¿Qué es una Prueba Unitaria o Unittest en el contexto de desarrollo de software?

Una Prueba Unitaria es una prueba donde se aísla una parte del código y se comprueba si funciona a la perfección y normalmente se lleva a cabo en la etapa de desarrollo del software

8) Bajo el contexto de pytest. ¿Qué es un “assert”?

Esta función se utiliza normalmente para realizar un test de nuestro código, destinado al uso de test unitarios, en pytest lo que logra es verificar valores en las pruebas de python; lo que se crea es una afirmación, entonces si la función otorga un valor diferente el valor falla y se especifica en donde ocurrió el error.

9) ¿Qué es Flake 8?

Flake8 es una librería de Python que nos permite verificar el código fuente contra el PEP8, que es una normativa de convenciones estilísticas a seguir a la hora de escribir código en Python, errores de programación y complejidad ciclomática.

10) Explique la diferencia entre un “log de error” y un “valor de error de retorno”

Un Log Error es un registro en donde se encuentran los errores que ha sufrido una aplicación, sistema operativo o servidor mientras están en funcionamiento, mientras que un valor de error de retorno es un error que impide al sistema seguir ejecutando la función, por lo que finaliza devolviendo un código de error