

Preguntas Teóricas

1. ¿Qué es Git?

Git, es un software de control específico de versión de fuente abierta, diseñado por Linus Torvalds, en el cual la base del código y sus versiones se encuentran disponibles en cualquier computadora.

2. ¿Qué es Github?

Es un servicio para que desarrolladores almacenen y administren su código. Posee un sitio web y un servicio de nube. Tiene como ventaja que es gratuito y, además, permite llevar un registro de control en cualquier cambio sobre el código.

3. ¿Qué es un branch?

En Git, un Branch es una forma de organización de la plataforma GitHub. Puede ser definido como un conjunto único de cambios o modificaciones que se hacen al proyecto sin alterar el código principal (master). El empleo de ramas resulta sumamente útil, sobre todo en el trabajo colaborativo, porque al no afectar el código principal, permiten al programador probar nuevas funcionalidades y cambios para el mismo y que, una vez que se tenga certeza que funcionan correctamente, pueden ser fácilmente incorporados a la rama principal para que se visualice dichas modificaciones en la funcionalidad del código.

4. ¿Qué es un commit?

Consiste en la actualización de los archivos que están siendo creados en git, logrando que los cambios que se hayan hecho sean guardados. Cuando se hacen cambios en el index, es posible ejecutar el comando commit, el cual hace que el archivo esté ahora guardado en el head, es decir, el comando es usado para direccionarse al head, y con esto es posible redireccionarlo al repositorio remoto.

5. ¿Qué es la operación cherry-pick?

En git, cherry-pick es un comando que permite incluir algunos cambios específicos en el actual Branch, elige una confirmación de una rama y la aplicar a otra. Por ejemplo, supongamos que una confirmación se aplica accidentalmente en la rama equivocada. Puedes cambiar a la rama correcta y ejecutar cherry-pick en la confirmación para aplicarla a donde pertenece.

6. ¿Qué hace el comando git checkout?

El comando git checkout se utiliza para la creación de nuevas ramas, o bien, para cambiar entre ellas (movilización entre ramas). Además, actualiza los archivos en el árbol de trabajo para que coincidan con la versión en el índice o el árbol especificado.

7. ¿Qué hace el comando git stash?

Este comando permite guardar temporalmente un archivo, para poder trabajar en alguna otra parte del código sin tener que preparar y actualizar ese archivo temporal.

8. ¿Compare las operaciones git fetch y git pull

Ambos se utilizan para descargar nuevos datos desde un repositorio. Sin embargo, git fetch realmente solo descarga nuevos datos de un repositorio, pero no integra ninguno de estos nuevos datos en sus archivos de trabajo, es decir, nunca manipulará, destruirá o estropeará nada. Por el contrario, git pull actualiza su rama head actual con los últimos cambios del servidor remoto. Esto significa que no solo descarga nuevos datos; también lo integra directamente en sus archivos de copia de trabajo actuales

9. ¿Qué hace el comando git reset ~HEAD?

El comando git reset es una herramienta compleja y versátil para deshacer cambios. A nivel superficial, git reset tiene un comportamiento similar a git checkout. Mientras que git checkout solo opera en el puntero de referencia HEAD, git reset moverá el puntero de referencia HEAD y el puntero de referencia de la rama actual. El comando git reset HEAD~2 retrocede la rama actual 2 confirmaciones.

10. ¿Qué es Pytest?

Pytest es una herramienta para Python, un marco de referencia que permite realizar códigos de prueba bajo el lenguaje Python. Es sumamente utilizado en la industria por diversas ventajas: su sintaxis es muy simple y fácil de comprender, puede correr pruebas en paralelo, puede correr una prueba en específico o un subgrupo de pruebas, puede omitir pruebas y detecta automáticamente una prueba.

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

Un assert es una función de pytest que permite verificar expectativas o valores en pruebas de Python. Por medio de introspección, pytest puede inferir el valor actual, el valor esperado y la operación utilizada con un assert, y devuelve un mensaje de error con su explicación y el tiempo de error.

12. ¿Qué es Flake 8?

Es una librería de Python que funciona para revisar el código en busca de errores de sintaxis y de forma; también permite revisar la “complejidad ciclomática” del código. Este término se refiere al número de caminos independiente que tiene el código base. En otras palabras, es una medida cuantitativa de la complejidad lógica de un programa.

Bibliografía:

Albert, M. (2017). ¿Qué es y para qué sirve Github? | Deusto Formación. Retrieved from <https://www.deustoformacion.com/blog/programacion-diseno-web/que-es-para-que-sirve-github>

Empezando - Fundamentos de Git. (2018). Retrieved from <https://git-scm.com/book/es/v1/Empezando-Fundamentos-de-Git>

Fundamentos de Git - Guardando cambios en el repositorio. (2018). Retrieved from <https://git-scm.com/book/es/v1/Fundamentos-de-Git-Guardando-cambios-en-el-repositorio>

Git Cherry Pick. (2019). Retrieved from <https://www.atlassian.com/es/git/tutorials/cherry-pick>

Git reset. (2019). Retrieved from <https://www.atlassian.com/es/git/tutorials/undoing-changes/git-reset>

Pytest: helps you write better programs. (2017). Retrieved from <https://docs.pytest.org/en/latest/>

¿Qué es GitHub? Una Guía para Principiantes sobre GitHub. (2018). Retrieved from <https://kinsta.com/es/base-de-conocimiento/que-es-github/>

The writing and reporting of assertions in tests. (2017). Retrieved from <https://docs.pytest.org/en/latest/assert.html#assert>

Trabajando con GIT, introducción al uso de los branch y git-completion.bash - Adictos al trabajo. (2011). Retrieved from <https://www.adictosaltrabajo.com/2011/06/27/git-branch-bash/>

What's the difference between git fetch and git pull? (2019). Retrieved from <https://www.git-tower.com/learn/git/faq/difference-between-git-fetch-git-pull>