

**Madrigal Sánchez, Clara Catalina**  
**Rojas Hernández, Fabricio José**

**2019023699**  
**2014008214**

**Tarea #1:**  
**GitHub, Pytest y Flake 8**

**I: Respuestas**

**1. Diferencie la herramienta Git de Github.**

Git es un software local, enfocado en el uso individual, mientras que Github es una plataforma basada en la nube.

**2. ¿Qué es un branch?**

Es una sección aislada del repositorio en el que se pueden hacer pruebas sin alterar el código completo. Aunque un repositorio tiene una branch por defecto, se pueden crear más. También se pueden unir branches utilizando un pull request.

**3. ¿Cómo se crea un nuevo branch?**

Para crear un nuevo branch, primero se debe ingresar a la página principal del repositorio, luego se selecciona el menú de selectores de branches, se nombra el branch nuevo y se oprime el botón “Create branch”.

**4. ¿Qué es un commit?**

Un commit son todos los cambios que se le generan a los archivos en una branch. Cada commit tiene un ID único que identifica los cambios específicos, cuándo fueron hechos los cambios y quién realizó los cambios. Cabe destacar que cuando se crea un commit, se debe incluir un mensaje que describa los cambios.

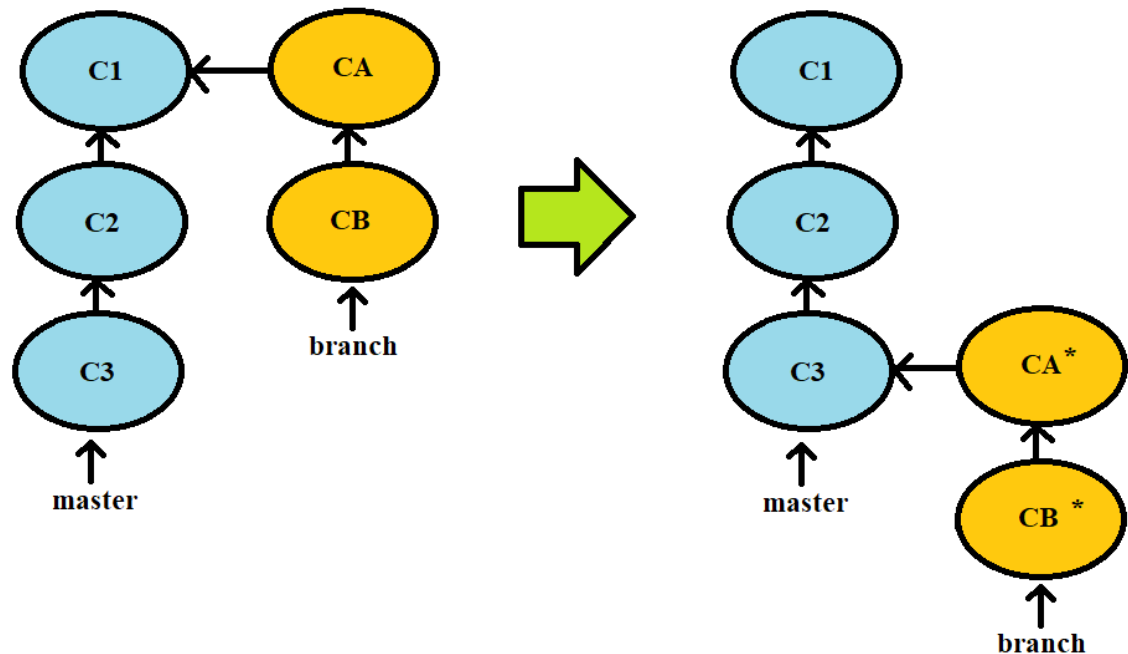
**5. ¿Qué es la operación “git stash”?**

Es una operación que “almacena” temporalmente los cambios realizados en el código y se puede decidir si los cambios se mantienen o se descartan. En otras palabras, un stash es un cambio sin commit. Si se usa la versión de escritorio de Github, cualquier cambio que no se guarde, se le aplicará un stash.

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

El “rebase to master” hace que el branch local, aparte de mantener los cambios realizados desde que se creó el branch, incorpore los cambios de la versión del master actual con respecto a la versión del master que existía cuando se creó el branch (asumiendo que no hay conflictos).

### Rebase to master



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

Unit Testing, o “unittest”, se le llama a una técnica de prueba de código en que se analizan los componentes/módulos individuales de código para verificar su funcionamiento y si son aptas para uso o no. Entonces, en vez de analizar todo el software, se analiza individualmente los módulos que lo componen.

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

Bajo el contexto de pytest, un “assert” es una operación nativa de Python que requiere un análisis introspectivo.

Cuando llamada, tal como cuando se corre el código normalmente, pytest debe verificar que lo que recibe ésta de parámetro designado para entrada resulte en un *true*. La manera en que determina esto depende del valor ingresado en sí, el esperado del resultado y la operación utilizada dentro del assert.

De fallar el assert, dependiendo a lo que se le pidió analizar, pytest determina el por qué falló (en otras palabras, por qué la entrada tuvo *false* de resultado), el testeó del programa es finalizado e imprime en la consola una sección detallando la parte del código que hizo que fallara el assert, junto con el código de error determinado por la operación y sus parámetros de entrada (tal como cuando se corre el código normalmente).

## 9. ¿Qué es Flake 8?

Flake8 es una utilidad de línea de comando para Python que, al ser ejecutada, identifica errores e inconsistencias de formato en el código que se le pide analizar. En otras palabras, aparte del hecho de que, en Python, los espaciados y justificados son vitales para la sintáxis, se puede utilizar Flake8 para asegurarse que, además, el formato del código sea consistente a lo largo de todos los proyectos que se realizan.

Note que es imperativo que el Flake8 utilizado sea de la versión correspondiente a la versión de Python en la que se escribió el documento.

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

Un log de error es un registro de los errores críticos y fatales que fueron encontrados durante la ejecución de un programa, y usualmente es mostrado al usuario al finalizar la ejecución del programa. Un código de error, mientras tanto, son números generados por el sistema al detectar un fallo o falta de ejecución de una acción durante el programa. Usualmente, las funciones corridas retornan diferentes valores dependiendo de si fueron exitosas o no, y los códigos que imprimen ya son determinados, pues ambos el sistema operativo y los programadores pueden hacer que el programa imprima diferentes códigos de error según la excepción/tipo de error que ocurra.

En resumen, un log de error es el registro de los errores durante la ejecución del programa, y el código de retorno representa el error en sí, pues se genera cuando ocurre un error.

## II. Referencias

[1] GitHub Docs, “Quickstart”, *docs.github.com* [Online]. Recuperado de: <https://docs.github.com/en/get-started/quickstart> (Accesado el 13 de Feb, 2022)

[2] GitHub Docs. “Creating and deleting branches within your repository”, *docs.github.com* [Online]. Recuperado de: <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/creating-and-deleting-branches-within-your-repository> (Accesado el 14 de Feb, 2022)

[3] GitHub Docs. “About branches”, *docs.github.com* [Online]. Recuperado de: <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-branches> (Accesado el 15 de Feb., 2022)

[4] Git Community, “Git Branching - Rebasing”, *git-scm.com* [Online]. Recuperado de: <https://git-scm.com/book/en/v2/Git-Branching-Rebasing> (Accesado 13 de Feb., 2022)

[5] Tutorials Point, “What is Unit Testing?”, *tutorialspoint.com* [Online]. Recuperado de: [https://www.tutorialspoint.com/software\\_testing\\_dictionary/unit\\_testing.htm](https://www.tutorialspoint.com/software_testing_dictionary/unit_testing.htm) (Accesado 13 de Feb., 2022)

[6] GeeksforGeeks, “Unit Testing in Python – Unittest”, *geeksforgeeks.org*, 4 de Oct. 2017 [Online]. Recuperado de: <https://www.geeksforgeeks.org/unit-testing-python-unittest/> (Accesado 13 de Feb., 2022)

[7] G. Szabo, “Python Pytest assertion error reporting”, *code-maven.com*, 5 de Mayo, 2017 [Online]. Recuperado de: <https://code-maven.com/pytest-assert-error-reporting> (Accesado 14 de Feb., 2022)

[8] Pytest Development Team, “The writing and reporting of assertions in tests”, *docs.pytest.org* [Online]. Recuperado de: <https://docs.pytest.org/en/6.2.x/assert.html> (Accesado 14 de Feb., 2022)

[9] Dev Null, “What is Flake8 and why we should use it?”, *Python Pandemonium*, 30 de En., 2017 [Online]. Recuperado de: <https://medium.com/python-pandemonium/what-is-flake8-and-why-we-should-use-it-b89bd78073f2> (Accesado 14 de Feb., 2022)

[10] Java TPoint, “Linux Error Codes”, *javatpoint.com* [Online]. Recuperado de: <https://www.javatpoint.com/linux-error-codes> (Accesado 14 de Feb., 2022)

[11] M. McMahon, “What Is an Error Log?”, *easytechjunkie.com*, 30 de En., 2022 [Online]. Recuperado de: <https://www.easytechjunkie.com/what-is-an-error-log.htm> (Accesado 15 de Feb., 2022)