TALLER 1B

***Objetivo***

Al finalizar el taller deberá estar en capacidad de clonar un repositorio, usar las herramientas propias de un proyecto con control de versiones para recuperar una versión correcta.

***Consigna***

Clone los siguientes repositorios y resuelva los problemas que tienen para poder mostrar la página web completa.

1. Este repositorio en algún punto estuvo bien, pero algún programador se equivocó y ahora la página no se muestra correctamente. Arréglelo de modo que el historial sólo tenga UN SOLO commit

<https://github.com/Domiciano/PasaporteGITEjercicioA>

Súbalo a su perfil de GitHub como repositorio propio, ya arreglado.

1. Este repositorio en su penúltimo commit estaba todo bien, pero por error el programador eliminó código importante en el último commit. Deshaga este commit de modo que la acción correctiva quede también en el historial.

<https://github.com/Domiciano/PasaporteGITEjercicioB>

Súbalo a su perfil de GitHub como repositorio propio, ya arreglado.

1. Este repositorio en su último commit tiene perfecta la página *index.html*, pero hace falta el código de *user.html*. Sin embargo, en su penúltimo commit, el repositorio tiene perfecto *user.html*, pero *index.html* no está terminado. Arréglelo.

<https://github.com/Domiciano/PasaporteGITEjercicioC>

Los archivos del repositorio son:

index.html

user.html

css/index.css

css/user.css

img/icesi.html

img/banner.html

Súbalo a su perfil de GitHub como repositorio propio, ya arreglado.

1. Clone este repositorio

<https://github.com/Domiciano/PasaporteGITEjercicioD>

En el repositorio, verifique el historial y encontrará varios commits erróneos que están marcados con mensajes tipo: “Error X”. Su tarea es corregir el historial de este repositorio, de modo que no aparezcan estos commits ni tampoco los que tengan que ver con los errores como los commits que comienzan por “Corregido…”, eliminándolos con herramientas de versión.

Súbalo a su perfil de GitHub como repositorio propio, ya arreglado.

1. Clone este repositorio

<https://github.com/Domiciano/PasaporteGITEjercicioE>

* 1. Explore cada versión del historial usando git checkout
  2. Regrese a la normalidad usando git checkout master
  3. Sólo usando revert deshaga el commit E.
  4. Reflexione acerca de por qué razón esta acción provoca un conflicto.
  5. Antes de resolver el conflicto, use git status, para mirar qué otras opciones tiene
  6. Solucione el conflicto escogiendo el *Incoming Change* (Ver el proyecto siempre en VSCode), luego debe crear un nuevo commit luego de solucionar el conflicto.
  7. Luego revierta el commit B. Reflexione por qué sale el conflicto.
  8. Finalmente, restaure todo el proyecto usando un hard reset y párese en el commit G
  9. Revierta el commit B y reflexione por qué sólo hay un solo conflicto