Desarrollo de Software - CC3S2 Diego Akira García Rojas Actividad 4

En los siguientes comandos, creo un repositorio de Git llamado "Actividad 4" y genero mi primer commit con un archivo README.md.

```
Cursos/DS/Actividad 4

> git init
Inicializado repositorio Git vacío en /home/ak13a/Documentos/Cursos/DS/Actividad
4/.git/
```

```
Actividad 4 on | main [+]

> git commit "Initial commit with README.md"
error: ruta especificada 'Initial commit with README.md' no concordó con n
ingún archivo conocido por git
(base)
Actividad 4 on | main [+]

> git commit -m "Initial commit with README.md"
[main (commit-raíz) 7524d98] Initial commit with README.md

1 file changed, 1 insertion(+)
create mode 100644 README.md
(base)
Actividad 4 on | main
> git status
En la rama main
nada para hacer commit, el árbol de trabajo está limpio
```

Pregunta: ¿Cuál es la salida del comando "git log --graph --pretty=format: '%x09 %h %ar ("%an") %s'"?

Respuesta: Formatea el log de commits en forma de gráfico. A la derecha del grafo, están los commits separados por un espacio tabulador. Primero está el hash acortado del commit, la fecha relativa del commit, el nombre del autor del commit y el título del commit.

En los siguientes comandos agrego otros archivos a la rama principal y pruebo la creación, fusión y eliminación de ramas.

```
Actividad 4 on | main
) echo " CONTRIBUTING" > CONTRIBUTING.md
(base)
Actividad 4 on | main [?]
) echo " README\n\Welcome to the project" > README.md
(base)
Actividad 4 on | main [!?]
) git add .
(base)
Actividad 4 on | main [+]
) git commit -m "Set up the repository base documentation"
[main fb3fd5c] Set up the repository base documentation
2 files changed, 2 insertions(+)
create mode 100644 CONTRIBUTING.md
```

```
Actividad 4 on | main
) echo "print('Hello World')" > main.py
(base)
Actividad 4 on | main [?] via | v3.12.7
) git add .
(base)
Actividad 4 on | main [+] via | v3.12.7
) git commit -m "Add main.py"
[main dfc1894] Add main.py
1 file changed, 1 insertion(+)
create mode 100644 main.py
```

```
Actividad 4 on <sup>↑</sup> main via <sup>1885</sup> v3.12.7

› git log --oneline
dfc1894 (HEAD -> main) Add main.py
fb3fd5c Set up the repository base documentation
7524d98 Initial commit with README.md
```

```
Actividad 4 on / main via 👫 v3.12.7
) git branch feature/new-feature
(base)
Actividad 4 on 🎙 main via 🔠 v3.12.7
) git checkout feature/new-feature
Cambiado a rama 'feature/new-feature'
(base)
Actividad 4 on / feature/new-feature via 🔠 v3.12.7
) git checkout main
Cambiado a rama 'main'
(base)
Actividad 4 on 🎖 main via 🚻 v3.12.7
) git merge feature/new-feature
Ya está actualizado.
(base)
Actividad 4 on 🎖 main via 🚻 v3.12.7
) git branch -d feature/new-feature
Eliminada la rama feature/new-feature (era dfc1894).
```

Preguntas:

- ¿Cómo te ha ayudado Git a mantener un historial claro y organizado de tus cambios? Cada cambio en el repositorio del proyecto puede ser "fotografiado" con un commit y ser visto en el log de commits. De esta forma puedo ver todos los cambios realizados a través de commits en el respositorio y sé exactamente qué se ha agregado y qué falta trabajar sin necesidad de revisar cada archivo del proyecto.
- ¿Qué beneficios ves en el uso de ramas para desarrollar nuevas características o corregir errores? Trabajar directamente sobre una única rama, la principal, ralentiza el trabajo y puede resultar en conflicitos fácilmente. Si dos desarrolladores trabajan en características que modifican los mismos archivos y ralizan los mismos cambios sobre la misma rama, es probable que ninguna característica funcione por tener conflictos entre ambos cambios. Por lo tanto, trabajar con ramas permite realizar los cambios que se quieran sin temor a que tus cambios se solapen con el de otro desarrollador.
- Realiza una revisión final del historial de commits para asegurarte de que todos los cambios se han registrado correctamente.

```
Actividad 4 on 7 main via 🚻 v3.12.7
) git status
En la rama main
nada para hacer commit, el árbol de trabajo está limpio
Actividad 4 on ∮ main via  v3.12.7
) git log
commit dfc189455198defaacdf290cf9522c5d51d3a5df (HEAD -> main)
Author: Akira <ak13a@tutamail.com>
Date: Mon Apr 7 11:26:56 2025 -0500
    Add main.py
Author: Akira <ak13a@tutamail.com>
Date: Mon Apr 7 11:26:04 2025 -0500
    Set up the repository base documentation
Author: Akira <ak13a@tutamail.com>
Date: Mon Apr 7 11:23:43 2025 -0500
    Initial commit with README.md
```

- Revisa el uso de ramas y merges para ver cómo Git maneja múltiples líneas de desarrollo.