



Ejercicio 1 : Crear un repositorio y hacer contribuciones

1. Crear un repositorio en GitHub:
 - a. Crea un nuevo repositorio en tu cuenta de GitHub.
 - b. Puedes nombrarlo como deseas y agregar una descripción que explique su propósito.
2. Clonar el repositorio en tu computadora:
 - a. Utiliza el comando `git clone` para clonar el repositorio en tu computadora local.
3. Realizar cambios locales:
 - a. Crea un nuevo archivo o modifica uno existente en tu repositorio clonado.
 - b. Utiliza `git add` para agregar los cambios al área de preparación y luego `git commit` para confirmar los cambios.
4. Sincronizar con GitHub:
 - a. Utiliza `git push` para enviar los cambios al repositorio remoto en GitHub.
 - b. Verifica en GitHub que los cambios se reflejen correctamente en el repositorio remoto.

Ejercicio 2 : Crear un repositorio y trabajar ramas

En este ejercicio trabajarás con un repositorio Git que contiene dos archivos: `login.js` y `ui.css`. Comenzarás creando una rama llamada `mejora-login`, donde modificarás la función de inicio de sesión para incluir validación. Luego, desde la rama `main`, crearás otra rama llamada `cambio-ui`, donde modificarás `login.js` agregando una función de cierre de sesión (`logout`) y cambiarás el color de fondo en `ui.css`. Después de estos cambios paralelos, realizarás un merge en `main` de ambas ramas, lo que generará conflictos tanto en `login.js` (por modificar la misma función) como en `ui.css` (por cambiar la misma línea). Resolverás manualmente los conflictos editando los archivos y haciendo el `commit` correspondiente.

Ejercicio 3: Colaborar en un repositorio existente

1. **Fork de un repositorio:**
 - a. Encuentra un repositorio público en GitHub que te interese contribuir.
 - b. Haz un fork del repositorio para crear una copia bajo tu propia cuenta.
2. **Clonar tu fork localmente:**
 - a. Utiliza `git clone` para clonar tu fork del repositorio en tu computadora local.
3. **Agregar un nuevo feature o solucionar un issue:**
 - a. Implementa una nueva característica o soluciona un problema existente en el código.
 - b. Asegúrate de trabajar en una rama separada utilizando `git checkout -b nombre-de-la-rama`.
4. **Enviar un pull request:**
 - a. Utiliza `git push` para enviar tus cambios a tu fork en GitHub.
 - b. Desde GitHub, abre un pull request (PR) hacia el repositorio original.
 - c. Describe tus cambios y espera la revisión del propietario del repositorio original.