

## Tarea de Git: Uso de `amend` y `reset --soft`

### Objetivo

- Practicar cómo corregir commits con `git commit --amend`.
- Aprender a retroceder en el historial manteniendo cambios en staging con `git reset --soft`.

### Enunciado

Realiza los siguientes pasos en tu terminal de Git.

#### 1. Inicializa un repositorio

```
mkdir tarea-git  
cd tarea-git  
git init
```

#### 2. Primer commit con error en el mensaje

```
echo "<h1>Hola mundo</h1>" > index.html  
git add index.html  
git commit -m "Creeeando archivo HTML"  
git commit --amend -m "Creando archivo HTML"
```

#### 3. Segundo commit olvidando un archivo y corrigiéndolo

```
echo "body {background: lightblue;}" > style.css  
echo "console.log('Hola');" > script.js  
git add style.css  
git commit -m "Añadiendo CSS"  
git add script.js  
git commit --amend --no-edit
```

#### 4. Tercer commit y retroceso con reset --soft

```
echo "Datos iniciales" > data.txt  
git add data.txt  
git commit -m "Añadiendo datos iniciales"  
git reset --soft HEAD~1
```

#### 5. Modificar archivo y commit final

```
echo "Datos corregidos" > data.txt  
git add data.txt  
git commit -m "Añadiendo datos corregidos"
```

### Entregable

- Un repositorio de Git con el historial limpio y corregido.
- Muestra el resultado de `git log --oneline` al final de la práctica.