

Práctica con la consola de Git

Beatriz Cruz Carrillo

Configuración inicial

```
Git CMD
C:\Users\bea_c>git config --global user.name Beatriz Carrillo
C:\Users\bea_c>git config --global user.email tenebeca@gmail.com
C:\Users\bea_c>
```

Crear repositorio nuevo

```
C:\Users\bea_c>mkdir git-practica
C:\Users\bea_c>cd git-practica
C:\Users\bea_c\git-practica>git init
Initialized empty Git repository in C:/Users/bea_c/git-practica/.git/
C:\Users\bea_c\git-practica>
```

Primer archivo y primer commit

```
C:\Users\bea_c\git-practica>echo "# Proyecto de práctica con Git" > README.md
C:\Users\bea_c\git-practica>git add README.md
C:\Users\bea_c\git-practica>git commit -m "Primer commit: agregar README"
[master (root-commit) 9dcfe57] Primer commit: agregar README
1 file changed, 1 insertion(+)
create mode 100644 README.md
C:\Users\bea_c\git-practica>
```

Revisar historial y estado

```
C:\Users\bea_c\git-practica>git status
On branch master
nothing to commit, working tree clean

C:\Users\bea_c\git-practica>git log --oneline --graph
* 9dcfe57 (HEAD -> master) Primer commit: agregar README

C:\Users\bea_c\git-practica>
```

Crear y modificar archivos

```
C:\Users\bea_c\git-practica>echo "print('Hola Git!')" > app.py

C:\Users\bea_c\git-practica>git add app.py

C:\Users\bea_c\git-practica>git commit -m "Agregar archivo app.py con saludo"
[master b7ad6ec] Agregar archivo app.py con saludo
1 file changed, 1 insertion(+)
create mode 100644 app.py

C:\Users\bea_c\git-practica>echo "print('Modificado')" >> app.py

C:\Users\bea_c\git-practica>git add app.py

C:\Users\bea_c\git-practica>git commit -m "Modificar app.py para imprimir mensaje nuevo"
[master b93c158] Modificar app.py para imprimir mensaje nuevo
1 file changed, 1 insertion(+)

C:\Users\bea_c\git-practica>
```

Crear y cambiar ramas

```
C:\Users\bea_c\git-practica>git branch nueva-funcionalidad

C:\Users\bea_c\git-practica>git checkout nueva-funcionalidad
Switched to branch 'nueva-funcionalidad'

C:\Users\bea_c\git-practica>
```

```
C:\Users\bea_c\git-practica>echo "print('Nueva funcionalidad')" >> app.py
C:\Users\bea_c\git-practica>git add app.py
C:\Users\bea_c\git-practica>git commit -m "Agregar nueva funcionalidad en app.py"
[nueva-funcionalidad 94251a7] Agregar nueva funcionalidad en app.py
1 file changed, 1 insertion(+)
C:\Users\bea_c\git-practica>
```

Fusionar ramas

```
C:\Users\bea_c\git-practica>git checkout main
error: pathspec 'main' did not match any file(s) known to git
C:\Users\bea_c\git-practica>git branch
  master
* nueva-funcionalidad
C:\Users\bea_c\git-practica>git checkout master
Switched to branch 'master'
C:\Users\bea_c\git-practica>
```

```
C:\Users\bea_c\git-practica>git checkout master
Switched to branch 'master'
C:\Users\bea_c\git-practica>git merge nueva-funcionalidad
Updating b93c158..94251a7
Fast-forward
 app.py | 1 +
1 file changed, 1 insertion(+)
C:\Users\bea_c\git-practica>
```

Conectar con un repositorio remoto

```
C:\Users\bea_c\git-practica>git remote add origin https://github.com/bnocturnum/Aplicaciones-Distribuidas

C:\Users\bea_c\git-practica>git branch -M master

C:\Users\bea_c\git-practica>git push -u origin master
info: please complete authentication in your browser...
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (12/12), 1.12 KiB | 88.00 KiB/s, done.
Total 12 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/bnocturnum/Aplicaciones-Distribuidas
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

```
C:\Users\bea_c\git-practica>
```

```
C:\Users\bea_c\git-practica>git pull origin master
From https://github.com/bnocturnum/Aplicaciones-Distribuidas
 * branch            master      -> FETCH_HEAD
Already up to date.

C:\Users\bea_c\git-practica>git push origin master
Everything up-to-date

C:\Users\bea_c\git-practica>
```

Ejercicio guiado

Se crea un repositorio nuevo para trabajar esta actividad:

```
C:\Users\bea_c>mkdir git-conflicto

C:\Users\bea_c>cd git-conflicto

C:\Users\bea_c\git-conflicto>git init
Initialized empty Git repository in C:/Users/bea_c/git-conflicto/.git/

C:\Users\bea_c\git-conflicto>echo "print('Hola desde main')" > app.py

C:\Users\bea_c\git-conflicto>git add app.py

C:\Users\bea_c\git-conflicto>git commit -m "Commit inicial con app.py"

[master (root-commit) c07c630] Commit inicial con app.py
 1 file changed, 1 insertion(+)
 create mode 100644 app.py

C:\Users\bea_c\git-conflicto>
```

Dentro de este nuevo repositorio se crean dos ramas: A y B. Ambas ramas realizan cambios a un archivo, para después hacer commit en él.

```
C:\Users\bea_c\git-conflicto>git checkout -b feature-a
Switched to a new branch 'feature-a'

C:\Users\bea_c\git-conflicto>echo "print('Cambio desde feature A')" >> app.py

C:\Users\bea_c\git-conflicto>git add app.py

C:\Users\bea_c\git-conflicto>git commit -m "Feature A: agregar mensaje en app.py"
[feature-a eedf2a3] Feature A: agregar mensaje en app.py
1 file changed, 1 insertion(+)

C:\Users\bea_c\git-conflicto>
```

```
C:\Users\bea_c\git-conflicto>git checkout master
Switched to branch 'master'

C:\Users\bea_c\git-conflicto>git checkout -b feature-b
Switched to a new branch 'feature-b'

C:\Users\bea_c\git-conflicto>echo "print('Cambio desde feature B')" >> app.py

C:\Users\bea_c\git-conflicto>git add app.py

C:\Users\bea_c\git-conflicto>git commit -m "Feature b: agregar mensaje en app.py"
[feature-b b7c4d0a] Feature b: agregar mensaje en app.py
1 file changed, 1 insertion(+)

C:\Users\bea_c\git-conflicto>
```

Al hacer merge, el sistema responde que hay un conflicto:

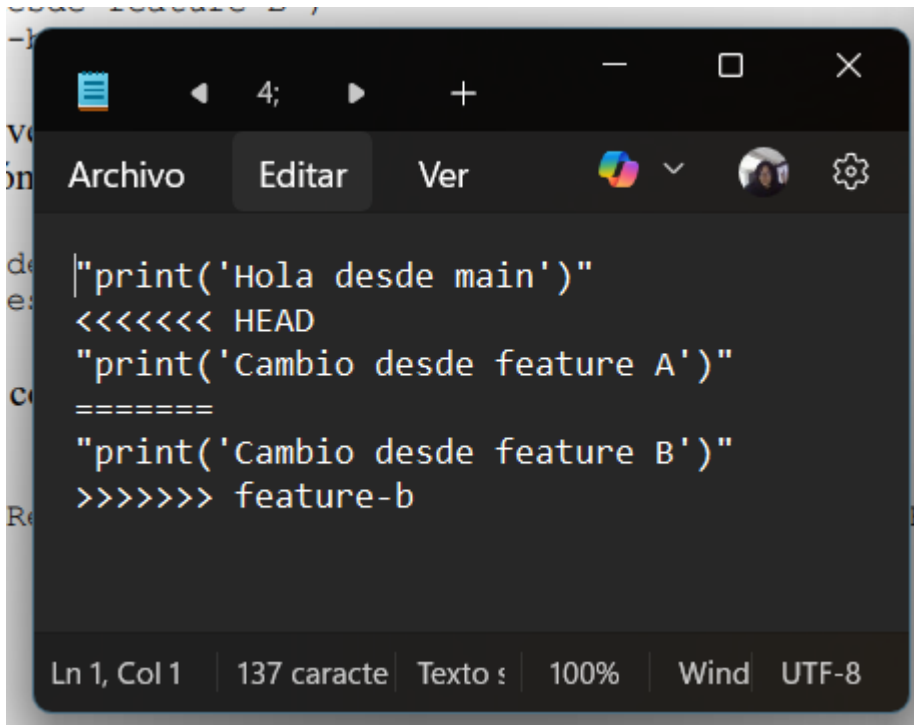
```
C:\Users\bea_c\git-conflicto>git checkout master
Switched to branch 'master'

C:\Users\bea_c\git-conflicto>git merge feature-a
Updating c07c630..eedf2a3
Fast-forward
 app.py | 1 +
1 file changed, 1 insertion(+)
```

```
C:\Users\bea_c\git-conflicto>git merge feature-b
Auto-merging app.py
CONFLICT (content): Merge conflict in app.py
Automatic merge failed; fix conflicts and then commit the result.

C:\Users\bea_c\git-conflicto>|
```

Si se abre el archivo app.py en alguna interfaz, podemos visualizar lo siguiente:



Al verificar desde la consola el estado de repositorio, nos muestra el error. Después de solucionar la situación, se realiza otro commit. Finalmente podemos ver el log actual.

```
C:\Users\bea_c\git-conflicto>git status  
On branch master  
You have unmerged paths.  
  (fix conflicts and run "git commit")  
  (use "git merge --abort" to abort the merge)  
  
Unmerged paths:  
  (use "git add <file>..." to mark resolution)  
        both modified:   app.py  
  
no changes added to commit (use "git add" and/or "git commit -a")  
  
C:\Users\bea_c\git-conflicto>git add app.py  
  
C:\Users\bea_c\git-conflicto>git commit -m "Resuelto, se hizo combinación"  
[master 8047810] Resuelto, se hizo combinación  
  
C:\Users\bea_c\git-conflicto>git log --oneline --graph  
*   8047810 (HEAD -> master) Resuelto, se hizo combinación  
|\  
| * b7c4d0a (feature-b) Feature b: agregar mensaje en app.py  
| * eedf2a3 (feature-a) Feature A: agregar mensaje en app.py  
|/  
* c07c630 Commit inicial con app.py  
  
C:\Users\bea_c\git-conflicto>
```