

# PROGRAMACION 1

## Práctico 2: Git y GitHub

### Actividad 1:

¿Qué es Github?

GitHub es una plataforma basada en la nube que permite almacenar y gestionar repositorios de código usando Git. Es ampliamente utilizada para el desarrollo colaborativo, control de versiones y despliegue de proyectos.

¿Cómo crear un repositorio en GitHub?

1. Ingresa a GitHub.
2. Haz clic en el ícono "+" (arriba a la derecha) y selecciona "New repository".
3. Especifica un nombre, descripción y visibilidad (público o privado).
4. Opcionalmente, inicializa con un README.md.
5. Haz clic en "Create repository".

¿Cómo crear una rama de Git?

Para crear una rama en Git se utiliza el siguiente comando: `git branch nombre-rama`

¿Cómo cambiar una rama en Git?

Se utiliza el siguiente comando: `git checkout rama`

¿Cómo fusionar ramas en Git?

Para fusionar se utiliza el comando: `git merge rama`

¿Cómo crear un commit en Git?

Se utiliza: `git commit -m "Mensaje del commit"`

¿Cómo enviar un commit a GitHub?

Para enviar el commit: `git push origin nombre-rama`

¿Qué es un repositorio remoto?

Un repositorio remoto es una versión alojada en un servidor (como GitHub) de un repositorio Git. Permite que varias personas colaboren en el mismo proyecto.

¿Cómo agregar un repositorio remoto a Git?

```
git remote add origin https://github.com/usuario/repo.git
```

¿Cómo empujar cambios a un repositorio remoto?

```
git push origin nombre-rama
```

¿Cómo tirar de cambios de un repositorio remoto?

```
git pull origin nombre-rama
```

¿Qué es un fork de repositorio?

Un fork es una copia de un repositorio de otro usuario en tu cuenta de GitHub. Te permite hacer cambios sin afectar el original y luego proponer contribuciones.

¿Cómo crear un fork de un repositorio?

1. Ve al repositorio en GitHub.
2. Haz clic en el botón Fork (arriba a la derecha).
3. Espera a que se cree en tu cuenta.

¿Cómo enviar una solicitud de extracción (pull request) a un repositorio?

1. Sube tus cambios a GitHub 

```
git push origin nombre-rama
```
2. Ve al repositorio en GitHub.
3. Haz clic en la pestaña "Pull Requests".
4. Clic en "New pull request".
5. Selecciona la rama donde hiciste los cambios y compárala con "main" (o la rama de destino).
6. Escribe un título y descripción de los cambios.
7. Haz clic en "Create pull request".

¿Cómo aceptar una solicitud de extracción?

1. Ve a la pestaña Pull Requests en GitHub.
2. Abre la solicitud pendiente.
3. Revisa los cambios y comenta si es necesario.
4. Si todo está bien, haz clic en Merge pull request.
5. Confirma con Confirm merge.

¿Qué es una etiqueta en Git?

Una etiqueta (tag) en Git es una referencia a un punto específico del historial, como una versión de software (v1.0.0)

¿Cómo crear una etiqueta en Git?

```
git tag -a v1.0.0 -m "Versión 1.0.0"
```

¿Cómo enviar una etiqueta a GitHub?

`git push origin v1.0.0` o `git push origin --tags` para enviar todas las etiquetas.

¿Qué es un historial de Git?

El historial de Git es un registro de todos los cambios realizados en el repositorio, con información de commits, autores y fechas.

¿Cómo ver el historial en Git?

Para buscar se puede utilizar el comando: `git log`

¿Cómo buscar en el historial de Git?

1. Buscar por palabra clave: `git log --grep="palabra clave"`
2. Buscar por autor: `git log --author="nombre"`
3. Buscar cambios en un archivo específico: `git log -- archivo.txt`

¿Cómo borrar el historial de Git?

¿Qué es un repositorio privado en GitHub?

Es un repositorio que solo los usuarios con permisos pueden ver y modificar.

¿Cómo crear un repositorio privado en GitHub?

1. Ve a GitHub.
2. Clic en "+" → New repository".
3. Escribe el nombre del repositorio.
4. Selecciona Private.
5. Clic en Create repository.

¿Cómo invitar a alguien a un repositorio privado de GitHub?

1. Abre el repositorio en GitHub.
2. Ve a Settings → Collaborators.
3. Clic en "Add people".
4. Escribe el nombre de usuario de GitHub y agrégalo.

¿Qué es un repositorio público en GitHub?

Es un repositorio accesible para cualquier persona en GitHub.

¿Cómo crear un repositorio público en GitHub?

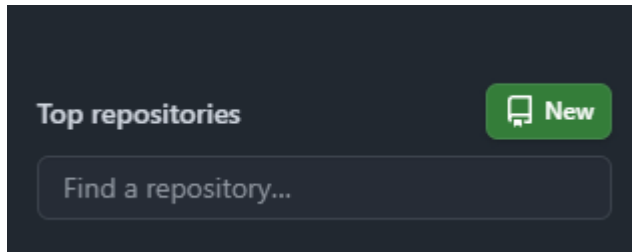
1. Ve a GitHub.
2. Clic en "+" → New repository".
3. Escribe el nombre del repositorio.
4. Selecciona Public.
5. Clic en Create repository.

¿Cómo compartir un repositorio público en GitHub?

1. Copiar la URL del repositorio con: `git remote get-url origin`.
2. Comparte el enlace.

## Actividad 2:

### 1. Crear un repositorio:




### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

*Required fields are marked with an asterisk (\*).*


**Owner \*** **Repository name \***


 Diexvel /

✔ Tp2-Git-2 is available.

Great repository names are short and memorable. Need inspiration? How about [glowing-fishstick](#) ?

**Description** (optional)

☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

**Initialize this repository with:**


☒ **Add a README file**  
This is where you can write a long description for your project. [Learn more about READMEs.](#)


**Add .gitignore**

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

**Choose a license**

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set  **main** as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

**Create repository**

## 2. Agregando un Archivo:

```

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacio
n 1/REPOSITARIOS
$ git clone https://github.com/Diexvel/Tp2-Git-2.git
Cloning into 'Tp2-Git-2'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (6/6), done.

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS
$ cd Tp2-Git-2

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ echo "Este el archivo de Diego Velardes" > mi-archivo.txt

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ git add.
git: 'add.' is not a git command. See 'git --help'.

The most similar command is
    add






Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ git add .
warning: in the working copy of 'mi-archivo.txt', LF will be replaced by CRLF the next time Git touches it

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ git commit -m "Agregando mi primer archivo"
[main 13a5e7f] Agregando mi primer archivo
1 file changed, 1 insertion(+)
create mode 100644 mi-archivo.txt

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 354 bytes | 354.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Diexvel/Tp2-Git-2.git
8cbd4d5..13a5e7f  main -> main

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ |

```

 <b>Diexvel</b> Agregando mi primer archivo	13a5e7f · 3 minutes ago	 <b>3 Commits</b>
 README.md	Initial commit	20 minutes ago
 archivo.txt	Create archivo.txt	14 minutes ago
 mi-archivo.txt	Agregando mi primer archivo	3 minutes ago

### 3. Creando Branchs:

```

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ git branch nueva-rama

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ git branch
* main
  nueva-rama

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (main)
$ git checkout nueva-rama
Switched to branch 'nueva-rama'

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (nueva-rama)
$ echo "Este archivo es de una nueva rama" > rama-nueva.txt

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (nueva-rama)
$ git add .
warning: in the working copy of 'rama-nueva.txt', LF will be replaced by CRLF the next time Git touches it

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (nueva-rama)
$ git commit -m "creacion de nueva rama con archivo nuevo"
[nueva-rama c9e7873] creacion de nueva rama con archivo nuevo
1 file changed, 1 insertion(+)
 create mode 100644 rama-nueva.txt

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/Tp2-Git-2 (nueva-rama)
$ git push origin nueva-rama
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 329 bytes | 329.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'nueva-rama' on GitHub by visiting:
remote:   https://github.com/Diexvel/Tp2-Git-2/pull/new/nueva-rama
remote:
To https://github.com/Diexvel/Tp2-Git-2.git
 * [new branch]      nueva-rama -> nueva-rama

```

main          2 Branches          0 Tags			<input type="text" value="Go to file"/>	<input type="button" value="Add file"/>	<input type="button" value="Code"/>
<b>Diexvel</b> Merge pull request #1 from Diexvel/nueva-rama <span>b95df47 · now</span> <span>5 Commits</span>					
	README.md	Initial commit	40 minutes ago		
	archivo.txt	Create archivo.txt	35 minutes ago		
	mi-archivo.txt	Agregando mi primer archivo	24 minutes ago		
	rama-nueva.txt	creacion de nueva rama con archivo nuevo	6 minutes ago		

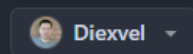
## Actividad 3:

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner \*



Repository name \*

/ ejercicio-conflicto

✔ ejercicio-conflicto is available.

Great repository names are short and memorable. Need inspiration? How about [miniature-guacamole](#) ?

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:



Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set main as the default branch. Change the default name in your [settings](#).

You are creating a public repository in your personal account.

Create repository



```

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS
$ git clone https://github.com/Diexvel/ejercicio-conflicto.git
Cloning into 'ejercicio-conflicto'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS
$ cd ejercicio-conflicto

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(main)
$

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(main)
$

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(main)
$ git checkout -b feature-branch
Switched to a new branch 'feature-branch'

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(feature-branch)
$ git add README.md

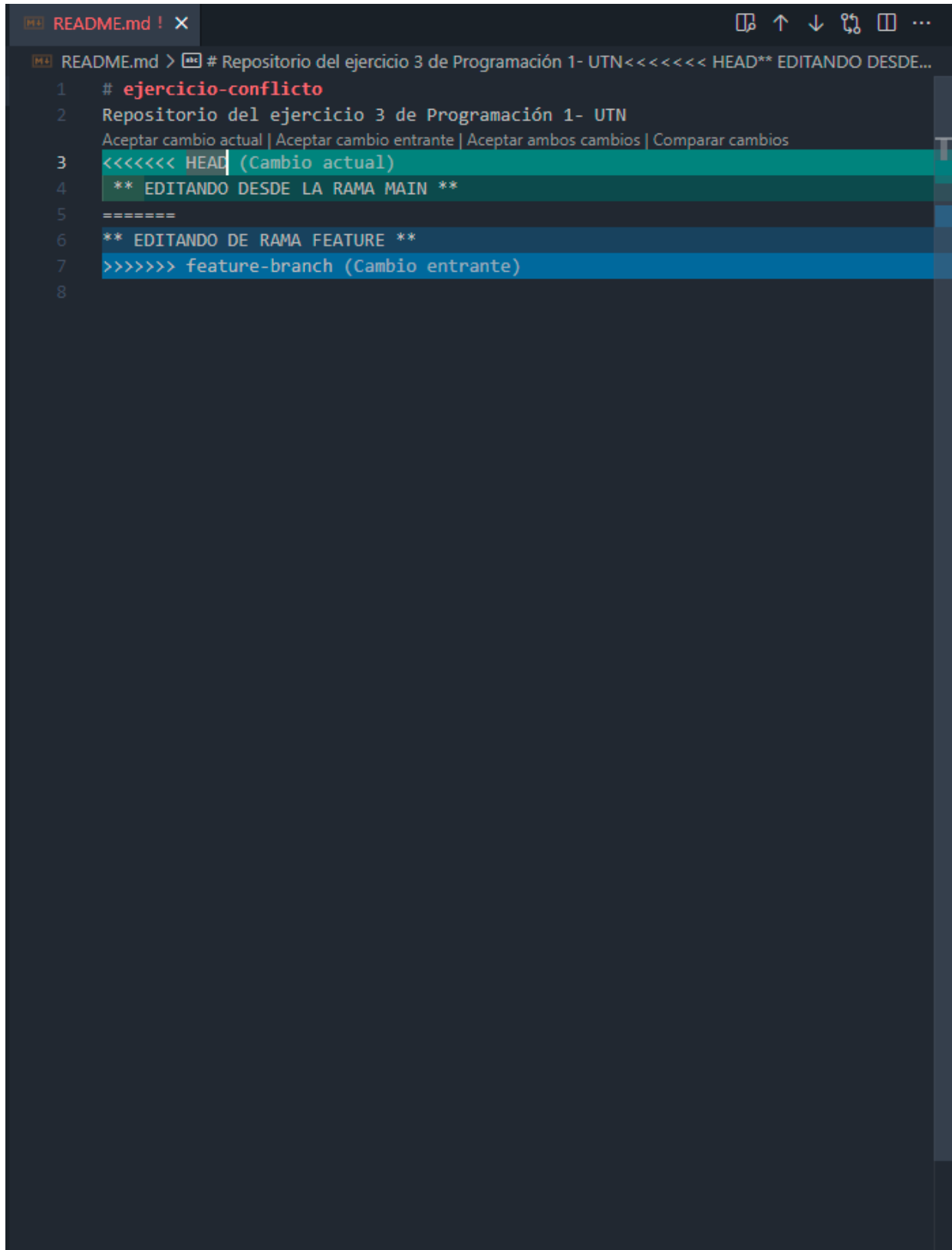
Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(feature-branch)
$ git commit -m "agregando linea en el README"
[feature-branch ab5768b] agregando linea en el README
1 file changed, 1 insertion(+)

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(feature-branch)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(main)
$ git add README.md

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITARIOS/ejercicio-conflicto
(main)
$ git commit -m "agregando linea en el README desde la rama main"
[main fd64e64] agregando linea en el README desde la rama main
1 file changed, 1 insertion(+)

```





```
README.md ! X
# Repositorio del ejercicio 3 de Programación 1- UTN<<<<<< HEAD** EDITANDO DESDE...
1 # ejercicio-conflicto
2 Repositorio del ejercicio 3 de Programación 1- UTN
  Aceptar cambio actual | Aceptar cambio entrante | Aceptar ambos cambios | Comparar cambios
3 <<<<<< HEAD (Cambio actual)
4 ** EDITANDO DESDE LA RAMA MAIN **
5 =====
6 ** EDITANDO DE RAMA FEATURE **
7 >>>>>> feature-branch (Cambio entrante)
8
```

```
Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITORIOS/ejercicio-conflicto (main|Merging)
$ git add README.md

Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITORIOS/ejercicio-conflicto (main|Merging)
$ git commit -m "Resolviendo el conflict"
[main 64a4a82] Resolviendo el conflict



Usuario@DESKTOP-PCTESR3 MINGW64 /d/TECNICATURA/MATERIAS 1ER SEMESTRE/Programacion 1/REPOSITORIOS/ejercicio-conflicto (main)
$ git push origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (9/9), 786 bytes | 262.00 KiB/s, done.
Total 9 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/Diexvel/ejercicio-conflicto.git
e579fac..64a4a82 main -> main
```

 **Diexvel** Resolviendo el conflict 64a4a82 · 2 minutos ago 🕒 4 Commits

 README.md

Resolviendo el conflict

2 minutos ago

 README 

## ejercicio-conflicto

Repositorio del ejercicio 3 de Programación 1- UTN \*\* EDITANDO DE RAMA FEATURE \*\* \*\* EDITANDO DESDE LA RAMA MAIN \*\*