

3) Realizar la siguiente actividad:

Paso 1: Crear un repositorio en GitHub


- Ve a GitHub e inicia sesión en tu cuenta.
- Haz clic en el botón "New" o "Create repository" para crear un nuevo repositorio.
- Asigna un nombre al repositorio, por ejemplo, conflict-exercise.
- Opcionalmente, añade una descripción.
- Marca la opción "Initialize this repository with a README".
- Haz clic en "Create repository".

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 *

 EugeniaDemarchi ▾

Repository name *

/ conflict_exercise

✔ conflict_exercise is available.

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

Description (optional)

TUPaD_UTN_Programacion_1_T.P2_Trabajao_Colaborativo_2025



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

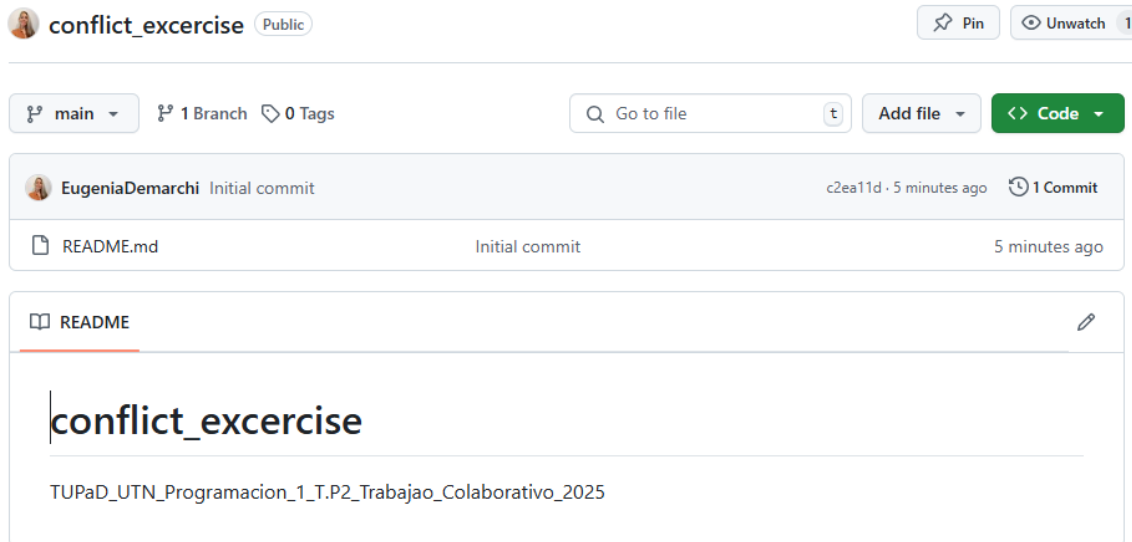
License: None ▾

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



Paso 2:

Clonar el repositorio a tu máquina local

- Copia la URL del repositorio (usualmente algo como https://github.com/tuusuario/conflict_excercise.git).
- Abre la terminal o línea de comandos en tu máquina.
- Clona el repositorio usando el comando:

git clone https://github.com/tuusuario/conflict_excercise.git

- Entra en el directorio del repositorio: cd conflict-exercise

```
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3> git clone https://github.com/EugeniaDemarchi/conflict_excercise.git
Cloning into 'conflict_excercise'...
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.
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3> cd conflict_excercise
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_excercise>
```

Paso 3:

Crear una nueva rama y editar un archivo

- Crea una nueva rama llamada feature-branch:

git checkout -b feature-branch

- Abre el archivo README.md en un editor de texto y añade una línea nueva, por ejemplo:
Este es un cambio en la feature branch.

- Guarda los cambios y haz un commit:

git add README.md

git commit -m "Added a line in feature-branch"

Paso 4: Volver a la rama principal y editar el mismo archivo

- Cambia de vuelta a la rama principal (main):

git checkout main

- Edita el archivo README.md de nuevo, añadiendo una línea diferente:

Este es un cambio en la main branch.

- Guarda los cambios y haz un commit:

git add README.md git commit -m "Added a line in main branch"

```
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise> git checkout -b feature_branch
Switched to a new branch 'feature_branch'
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise> git add .
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise> git commit -m 'Added a line in feature_branch'
[feature_branch b4215aa] Added a line in feature_branch
1 file changed, 3 insertions(+)
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise> git add .
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise> git commit -m 'Added a line in the main branch'
[main a43b124] Added a line in the main branch
1 file changed, 3 insertions(+)
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise>
```

Paso 5:

Hacer un merge y generar un conflicto

- Intenta hacer un merge de la feature-branch en la rama main:

git merge feature-branch

- Se generará un conflicto porque ambos cambios afectan la misma línea del archivo

README.md. Paso 6: Resolver el conflicto

```
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_exercise> git merge feature_branch
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
```

- Abre el archivo README.md en tu editor de texto. Verás algo similar a esto:

<<<<<<< HEAD

Este es un cambio en la main branch.

=====

Este es un cambio en la feature branch

.>>>>>>> feature-branch

```

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<<< HEAD (Current Change)
Este es un cambio en la main branch.
=====
Este es un cambio en la feature branch.
>>>>>>> feature_branch (Incoming Change)

```

- Decide cómo resolver el conflicto. Puedes mantener ambos cambios, elegir uno de ellos, o fusionar los contenidos de alguna manera.
- Edita el archivo para resolver el conflicto y guarda los cambios (Se debe borrar lo marcado en verde en el archivo donde estes solucionando el conflicto. Y se debe borrar la parte del texto que no se quiera dejar).

```

# conflict_excercise

TUPaD_UTN_Programacion_1_T.P2_Trabajao_Colaborativo_2025

Este es un cambio fusionado.

```

- Añade el archivo resuelto y completa el merge:

`git add README.md git commit -m "Resolved merge conflict"`

```

CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_excercise> git add .
PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_excercise> git commit -m 'Resolved merge conflict'
[main a3daf96] Resolved merge conflict

```

Paso 7: Subir los cambios a GitHub

- Sube los cambios de la rama main al repositorio remoto en GitHub:

`git push origin main`

```

PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_excercise> 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), 813 bytes | 406.00 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/EugeniaDemarchi/conflict_excercise.git
c2ea11d..a3daf96  main -> main

```

- También sube la feature-branch si deseas:

`git push origin feature-branch`

```

PS C:\Users\Usuario\Desktop\PROGRAMACION\UTN\1ER_CUATRIMESTRE\PROGRAMACION_1\GIT\respuesta3\conflict_excercise> git push origin feature_branch
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature_branch' on GitHub by visiting:
remote:   https://github.com/EugeniaDemarchi/conflict_excercise/pull/new/feature_branch
remote:
To https://github.com/EugeniaDemarchi/conflict_excercise.git
* [new branch]   feature_branch -> feature_branch

```

Paso 8: Verificar en GitHub

- Ve a tu repositorio en GitHub y revisa el archivo README.md para confirmar que los cambios se han subido correctamente.

The screenshot shows the GitHub interface for a repository named 'conflict_excercise'. At the top, the repository name is displayed with a 'Public' badge. To the right are buttons for 'Pin' and 'Unwatch', with a notification count of '1'. Below this is a navigation bar showing 'main' as the selected branch, '2 Branches', and '0 Tags'. A search bar labeled 'Go to file' is present, along with 'Add file' and 'Code' buttons. The commit history section shows a commit by 'EugeniaDemarchi' titled 'Resolved merge conflict' with hash 'a3daf96' and timestamp '3 minutes ago'. Below the commit, a file named 'README.md' is listed with the same title and timestamp. The README content is displayed below, featuring the title 'conflict_excercise', the subtitle 'TUPaD_UTN_Programacion_1_T.P2_Trabajao_Colaborativo_2025', and the text 'Este es un cambio fusionado.'

conflict_excercise Public

Pin Unwatch 1

main 2 Branches 0 Tags

Go to file Add file Code

EugeniaDemarchi Resolved merge conflict a3daf96 · 3 minutes ago 4 Commits

README.md Resolved merge conflict 3 minutes ago

README

conflict_excercise

TUPaD_UTN_Programacion_1_T.P2_Trabajao_Colaborativo_2025

Este es un cambio fusionado.

- Puedes revisar el historial de commits para ver el conflicto y su resolución.