git config: Preséntate a Git

MINGW64:/c/Users/windows10

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~ (develop)
$ git config --global user.name "AriusJoel1"

windows10@DESKTOP-UEI1KU9 MINGW64 ~ (develop)
$ git config --global user.name "AriusJoel1"

windows10@DESKTOP-UEI1KU9 MINGW64 ~ (develop)
$ |
```

Nos movemos a la carpeta:

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~ (develop)
$ cd C:\Users\windows10\Desktop\DesarrolloDeSoftware
bash: cd: C:Userswindows10DesktopDesarrolloDeSoftware: No such file or directory

windows10@DESKTOP-UEI1KU9 MINGW64 ~ (develop)
$ cd Desktop/

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop (develop)
$ cd DesarrolloDeSoftware/

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware (feature/advanc
ed-feature)
$
```

Creamos y nos movemos

MINGW64:/c/Users/windows10/Desktop/DesarrolloDeSoftware/actividad4   —   □   ✕

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~ (develop)
$ cd Desktop/

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop (develop)
$ cd DesarrolloDeSoftware/

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware (feature/advanc
ed-feature)
$ mkdir actividad4

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware (feature/advanc
ed-feature)
$ cd actividad4/

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (fea
ture/advanced-feature)
$ git init
Initialized empty Git repository in C:/Users/windows10/Desktop/DesarrolloDeSoftw
are/actividad4/.git/

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ |
```

Creamos el readme y verificamos

Commit y status

Agregamos el readme seguidamente del commit y mostramos los commits
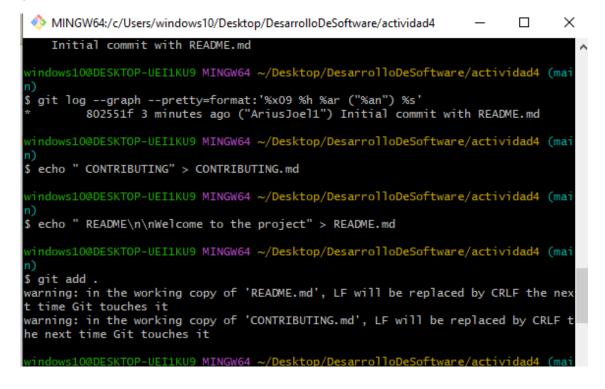
MINGW64:/c/Users/windows10/Desktop/DesarrolloDeSoftware/actividad4

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the nex
t time Git touches it

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git commit -m "Initial commit with README.md"
[main (root-commit) 802551f] Initial commit with README.md
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git log
commit 802551fa1436dc1b785f420c0e7dbdaa488a3c1a (HEAD -> main)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 00:31:13 2025 -0500

    Initial commit with README.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$
```

¿Cual es la salida de este comando?

MINGW64:/c/Users/windows10/Desktop/DesarrolloDeSoftware/actividad4

```
    Initial commit with README.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git log --graph --pretty=format:'%x09 %h %ar ("%an") %s'
*        802551f 3 minutes ago ("AriusJoel1") Initial commit with README.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ echo " CONTRIBUTING" > CONTRIBUTING.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ echo " README\n\nWelcome to the project" > README.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git add .
warning: in the working copy of 'README.md', LF will be replaced by CRLF the nex
t time Git touches it
warning: in the working copy of 'CONTRIBUTING.md', LF will be replaced by CRLF t
he next time Git touches it

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
```

Agregamos el código ejemplo python

Esta correctamente registrado:

```
$ git log --oneline
26fca00 (HEAD -> main) Set up the repository base documentation
802551f Initial commit with README.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$
```

Mostramos en que rama estamos (rama main)

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (ma
n)
$ git branch
* main

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (ma
```

Creamos la rama feature y cambiamos a dicha rama

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4
n)
$ git branch feature/new-feature
```

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git checkout feature/new-feature
A       main.py
Switched to branch 'feature/new-feature'
```

Creamos la rama, la cargamos y creamos otra rama

creando una nueva rama a partir de un commit específico

nos movemos a la rama feature

Creamos la rama develop, cambiamos a dicha rama y volvemos a crear y movernos a una nueva rama

Creamos una rama desde un commit específico

Cambiamos de rama:

Unimos las ramas y eliminamos la rama local

Preguntas:

¿Cómo te ha ayudado Git a mantener un historial claro y organizado de tus cambios?
Git permite registrar cada cambio con un mensaje descriptivo, lo que facilita seguir el historial y retroceder a versiones anteriores si es necesario. Así, se mantiene un control claro de todas las modificaciones.

¿Qué beneficios ves en el uso de ramas para desarrollar nuevas características o corregir errores? Las ramas permiten trabajar de manera aislada en nuevas características o correcciones, sin afectar el código principal. Esto facilita la colaboración y la integración de cambios sin conflictos.

EJERCICIO 1

Creamos rama nueva desde main:

Editamos el archivo main

Cambiamos a main:

```
windows10@DESKTOP-UEI1KU9 MIN
ture/advanced-feature)
$ git checkout main
A       main.py
Switched to branch 'main'
```

Editamos el archivo main:

```
Users > windows10 > Desktop > DesarrolloDeSoftware > activida
    print('Hello World - updated in main')

    def greet():
        print('Hello from advanced feature')

    greet()
```

Añadimos los cambios:

```
n)
$ git add main.py
warning: in the working copy of 'main.py', LF will be replaced by CRLF the nex
time Git touches it

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (m
n)
$ git commit -m "Update main.py message in main branch"
[main fd5de5a] Update main.py message in main branch
 1 file changed, 6 insertions(+)
 create mode 100644 main.py
```

Fusionamos ramas

```
$ git merge feature/advanced-feature
Already up to date.
```

commit

```
n)
$ git commit -m "Resolve merge conflict between main and feature/advanced-featur
e"
On branch main
nothing to commit, working tree clean
```

Viendo el historial de commits:

```
$ git log -p
commit fd5de5af5c794d12e0e8adeb1cea5fe44568e6e3 (HEAD -> main)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 01:10:21 2025 -0500

    Update main.py message in main branch

diff --git a/main.py b/main.py
new file mode 100644
index 0000000..945334b
--- /dev/null
+++ b/main.py
@@ -0,0 +1,6 @@
+print('Hello World - updated in main')
+
+def greet():
+    print('Hello from advanced feature')
+
+greet()

commit 26fca0045b3ec3124d75d2bf40c5eb97e7c5da2d (new-branch-name, feature/login,
 feature/another-new-feature, feature/advanced-feature, develop, base-branch-nam
```

Ejercicio 2:

Mostramos mis commits:

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git log --author="AriusJoel1"
commit fd5de5af5c794d12e0e8adeb1cea5fe44568e6e3 (HEAD -> main)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 01:10:21 2025 -0500

    Update main.py message in main branch

commit 26fca0045b3ec3124d75d2bf40c5eb97e7c5da2d (new-branch-name, feature/login,
 feature/another-new-feature, feature/advanced-feature, develop, base-branch-nam
 e)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 00:37:34 2025 -0500

    Set up the repository base documentation

commit 802551fa1436dc1b785f420c0e7dbdaa488a3c1a (hotfix/bugfix)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 00:31:13 2025 -0500

    Initial commit with README.md
```

Revertir un commit:

```
jwindows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
mRevert "Update main.py message in main branch"

 This reverts commit fd5de5af5c794d12e0e8adeb1cea5fe44568e6e3.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
$#
$# On branch main
# Changes to be committed:
g#      deleted:    main.py
#
# Untracked files:
#       dd
g#
```

```
MINGW64:/c/Users/windows10/Desktop/DesarrolloDeSoftware/actividad4          —    □    ×
Revert "Update main.py message in main branch"

This reverts commit fd5de5af5c794d12e0e8adeb1cea5fe44568e6e3.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch main
# Changes to be committed:
#       deleted:    main.p# Untrack
```

```
MINGW64:/c/Users/windows10/Desktop/DesarrolloDeSoftware/actividad4



Successfully rebased and updated refs/heads/main.
```

Mostrando commits:

```
n)
$ git log --graph --oneline --all
* 980c05c (HEAD -> main) Revert "Update main.py message in main branch"
* fd5de5a Update main.py message in main branch
* 26fca00 (new-branch-name, feature/login, feature/another-new-feature, f
advanced-feature, develop, base-branch-name) Set up the repository base d
ation
* 802551f (hotfix/bugfix) Initial commit with README.md
```

Ejercicio 3:

Crear nueva rama desde nuevo commit:

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/
n)
$ git branch bugfix/rollback-feature fd5de5a

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/
n)
$ git checkout bugfix/rollback-feature
Switched to branch 'bugfix/rollback-feature'
```

Modificar main.py

```python
1    print('Hello World - updated in main')
2
3    def greet():
4        print('Fixed bug in feature')
5
6    greet()
7
```

Agregamos el main.py y nos movemos a la rama main y combinamos las ramas

```
$ git add main.py

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (bug
fix/rollback-feature)
$ git commit -m "Fix bug in rollback feature"
[bugfix/rollback-feature bdb653d] Fix bug in rollback feature
 1 file changed, 1 insertion(+), 1 deletion(-)

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (bug
fix/rollback-feature)
$ git checkout main
Switched to branch 'main'

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git merge bugfix/rollback-feature
CONFLICT (modify/delete): main.py deleted in HEAD and modified in bugfix/rollbac
k-feature.  Version bugfix/rollback-feature of main.py left in tree.
Automatic merge failed; fix conflicts and then commit the result.
```

Rama eliminada:

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftw
n|MERGING)
$ git branch -D bugfix/rollback-feature
Deleted branch bugfix/rollback-feature (was bdb653d).
```

Ejercicio 4

Editamos el main

```
 main.py > ...
  1    print('Hello World - updated in main')
  2
  3    def greet():
  4        print('Fixed bug in feature')
  5
  6    greet()
  7    print('This change will be reset')
  8
```

Agregamos el main y agregamos el commit

```
n|MERGING)
$ git branch -D bugfix/rollback-feature
Deleted branch bugfix/rollback-feature (was bdb653d).

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/activid
n|MERGING)
$ git add main.py

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/activid
n|MERGING)
$ git commit -m "Introduce a change to be reset"
[main f5690e3] Introduce a change to be reset
```

Deshacemos el commit

```
n)
$ git reset --hard HEAD~1
HEAD is now at 980c05c Revert "Update main.py message in main branch"
```

Realizamos el cambio sin confirmar y finalmente lo deshacemos:

```
n)
$ echo "Another line in README" >> README.md

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (m
n)
$ git status
On branch main
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        dd

no changes added to commit (use "git add" and/or "git commit -a")

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (m
n)
$ git restore README.md
```

Ejercicio 5

Clonamos

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
n)
$ git clone "https://github.com/AriusJoel1/DesarrolloDeSoftware.git"
Cloning into 'DesarrolloDeSoftware'...
remote: Enumerating objects: 72, done.
remote: Counting objects: 100% (72/72), done.
remote: Compressing objects: 100% (62/62), done.
remote: Total 72 (delta 16), reused 4 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (72/72), 9.18 MiB | 5.28 MiB/s, done.
Resolving deltas: 100% (16/16), done.

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (mai
```

Creamos ramas:

```
n)
$ git branch feature/team-feature

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/
n)
$ git checkout feature/team-feature
Switched to branch 'feature/team-feature'
```

Creamos el archivo Python con su contenido

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/ac
ture/team-feature)
$ echo 'print("Collaboration is key!")' > collaboration.py
```

Subimos y creamos commit:

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (fea
ture/team-feature)
$ git add .
warning: in the working copy of 'collaboration.py', LF will be replaced by CRLF
the next time Git touches it
warning: in the working copy of 'dd', LF will be replaced by CRLF the next time
Git touches it

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (fea
ture/team-feature)
$ git commit -m "Add collaboration script"
[feature/team-feature aa2bcb8] Add collaboration script
 2 files changed, 54 insertions(+)
 create mode 100644 collaboration.py
 create mode 100644 dd

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (fea
ture/team-feature)
$
```

Subimos (pusheamos) los cambios al github

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (fea
ture/team-feature)
$ git remote add origin https://github.com/AriusJoel1/DesarrolloDeSoftware.git

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (fea
ture/team-feature)
$ git push origin feature/team-feature
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 4 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (15/15), 2.00 KiB | 684.00 KiB/s, done.
Total 15 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
remote:
remote: Create a pull request for 'feature/team-feature' on GitHub by visiting:
remote:      https://github.com/AriusJoel1/DesarrolloDeSoftware/pull/new/feature
/team-feature
remote:
To https://github.com/AriusJoel1/DesarrolloDeSoftware.git
 * [new branch]      feature/team-feature -> feature/team-feature

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (fea
```

Vamos a la rama main y creamos el archivo main.py con su contenido:

```
windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDe
ture/team-feature)
$ git checkout main
Switched to branch 'main'

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDe
$ echo "print('Cherry pick this!')" >> main.py
bash: !': event not found

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDe
$ echo 'print("Cherry pick this!")' >> main.py
```

Agregamos el main.py y el commit:

Aplicamos el commit con ese hash en la rama

Cambio sin confirmar:

Cambios en main.py

```
$ git cherry-pick --continue
On branch feature/cherry-pick
You are currently cherry-picking commit 980c05c.
  (all conflicts fixed: run "git cherry-pick --continue")
  (use "git cherry-pick --skip" to skip this patch)
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

nothing to commit, working tree clean
The previous cherry-pick is now empty, possibly due to conflict resolution.
If you wish to commit it anyway, use:

    git commit --allow-empty

Otherwise, please use 'git cherry-pick --skip'

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (feature/cherry-pick
|CHERRY-PICKING)
$ echo "This change is stashed" >> main.py

windows10@DESKTOP-UEI1KU9 MINGW64 ~/Desktop/DesarrolloDeSoftware/actividad4 (feature/cherry-pick
|CHERRY-PICKING)
$ git status
On branch feature/cherry-pick
You are currently cherry-picking commit 980c05c.
  (all conflicts fixed: run "git cherry-pick --continue")
  (use "git cherry-pick --skip" to skip this patch)
  (use "git cherry-pick --abort" to cancel the cherry-pick operation)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   main.py

no changes added to commit (use "git add" and/or "git commit -a")
```

Revisar el historial de commits

```
$ git log
commit a5fbfc7d62f2870d7c03a398616e71b251ada66a (HEAD -> feature/cherry-pick, main)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 02:11:23 2025 -0500

    Add cherry-pick example

commit 980c05c17cdcf134853a6b2934ba2569ac140dca
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 01:18:29 2025 -0500

    Revert "Update main.py message in main branch"

    This reverts commit fd5de5af5c794d12e0e8adeb1cea5fe44568e6e3.

commit fd5de5af5c794d12e0e8adeb1cea5fe44568e6e3
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 01:10:21 2025 -0500

    Update main.py message in main branch

commit 26fca0045b3ec3124d75d2bf40c5eb97e7c5da2d (new-branch-name, feature/login, feature/another
-new-feature, feature/advanced-feature, develop, base-branch-name)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 00:37:34 2025 -0500

    Set up the repository base documentation

commit 802551fa1436dc1b785f420c0e7dbdaa488a3c1a (hotfix/bugfix)
Author: AriusJoel1 <joel.seminario.s@uni.pe>
Date:   Mon Apr 14 00:31:13 2025 -0500

    Initial commit with README.md
```