[Git - Trabajar con Remotos (git-scm.com)](https://git-scm.com/book/es/v2/Fundamentos-de-Git-Trabajar-con-Remotos)

**2.5 Fundamentos de Git - Trabajar con Remotos**

Ejemplo práctico vamos a utilizar la terminal, fingir que tenemos un repositorio remoto, aunque en la realidad va estar en un repositorio local. Vamos a crear una carpeta con el nombre remoto01 y nos posicionamos en dicha carpeta.

Creamos un directorio dentro del RRemoto1 con la condición --bare, (esta condición permite realizar operaciones **push** sobre él. Es decir, un repositorio que no está activo para trabajar en el directamente, sí está activo no se podrían realizar operaciones **push** sobre el.

(En adelante, en otras prácticas, sí haremos prácticas con Github)

Vamos a inicializar el repositorio RRemoto1 y después sobre el vamos hacer un clonado.

Terminal 1:

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1

$ git init --bare

Initialized empty Git repository in C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1/

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1 (BARE:main)

$ cd ..

Por defecto Cuando se clona se crea una copia del repositorio con el mismo nombre, para evitar esto le vamos a indicar otro nombre en nuestro caso repo3

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED

$ git clone RRemoto1/ repo3

Cloning into 'repo3'...

warning: You appear to have cloned an empty repository.

done.

Si se abre el repo3 es un repo usual y encontraremos una carpeta **.git**

no hace falta hacerlo solo lo comento a título de curiosidad

nos posicionamos en la carpeta repo3:

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED

$ cd repo3

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git status

On branch main

No commits yet

nothing to commit (create/copy files and use "git add" to track)

creamos un archivo **lista1A.md** y le vamos añadir contenido, lo vamos hacer en en visual estudio code para que sea más representativo a título de vista, sobre frutas y escribimos un listado de frutas, vamos hacer lo de siempre modificar, guardar y preparar commits. Vamos a ello entonces.

Nos queda un buen ratito,

¡Ei la tercera fila atentos!

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ touch lista1A.md

Hacemos un git status para ver donde se posiciona nuestros archivos en el ciclo de vida git.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git status

On branch main

No commits yet

Untracked files:

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

lista1A.md

nothing added to commit but untracked files present (use "git add" to track)

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git add .

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git commit -m "melocotones y peras"

[main (root-commit) 0dd78ae] melocotones y peras

1 file changed, 2 insertions(+)

create mode 100644 lista1A.md

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git log

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5 (HEAD -> main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git status

On branch main

Your branch is based on 'origin/main', but the upstream is gone.

(use "git branch --unset-upstream" to fixup)

nothing to commit, working tree clean

podemos ver que la rama está basada en origin/main, pero el upstream a desaparecido, esto significa que el repo esta vacio.

Importante Origin es el nombre del remoto, y main es la rama remota

Para confirmar pondremos el comando git remote y nos va da respuesta de origin.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git remote

origin

si ponemos el mismo comando seguido de – v nos va responder con la rutas que apuntan a la carpeta donde se han guardado los remotos.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git remote -v

origin C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1/ (fetch)

origin C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1/ (push)

puesto que de momento el repositorio este vacio vamos a llenarlo con el comando git push (opcional con el nombre del remoto y de la rama)

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git push origin main

Enumerating objects: 3, done.

Counting objects: 100% (3/3), done.

Writing objects: 100% (3/3), 239 bytes | 239.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1/

\* [new branch] main -> main

Esta flecha : (main -> main) nos indica que la rama main apunta a la main

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git push

Everything up-to-date

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git log --graph --all

\* commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5 (HEAD -> main, origin/main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

(origin/main) esto indica una rama distinta es decir la rama remota del repositorio remoto hay también una una rama main y que ambas están sincronizadas en linea

melocotones y peras

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git add .

Vamos hacer un nuevo cambio dentro de la lista1A.md añadiento fruta albaricoque

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git commit -m "albaricoques"

[main 106d185] albaricoques

1 file changed, 1 insertion(+)

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

a continuación veremos después de realizar un ***git log*** que nuestra el resultado de abajo que la rama local master esta en unn punto siguente a la rama origin-main

$ git log

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (HEAD -> main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5 (origin/main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

nothing to commit, working tree clean

sí volvemos hacer un push se va a subir la rama origin/main en la misma línea con main.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Writing objects: 100% (3/3), 271 bytes | 271.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1/

0dd78ae..106d185 main -> main

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git log

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (HEAD -> main, origin/main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

vamos a similar como que nos da información de un repositorio remoto.

Para ello nos vamos a realizar nuevo repositorio remoto le vamos a guardar en otra carpeta y se llamara ***repo3Bis***

Si se mira tenemos dos carpetas repo3 y repo3Bis que están basadas en el repositorio remoto con nombre de RRemoto1

Otra cosa importante, tanto repo3 como reo3Bis tienen el mismo contenido dado que hemos clonado el uno del otro.

Ahora vamos a trabajar en dos repositorios a la vez y vamos realizar modificaciones en uno observando como podemos y que comandos hacen falta realizar para que surga el efecto total de sincronizar y obtener los mismos cambios en los dos repositorios.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ cd ..

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED

$ git clone RRemoto1/ repo3Bis

Cloning into 'repo3Bis'...

done.

Vamos hacer un git log para ver que estamos en el mismo punto de partida o sea tenemos la misma información en los dos repositorios, lo suyo seria poner los dos terminales uno al lado del otro e ir observando.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git log

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (HEAD -> main, origin/main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git add .

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

modified: lista1A.md

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git commit -m "uva"

[main 8481161] uva

1 file changed, 1 insertion(+)

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git status

On branch main

Your branch is ahead of 'origin/main' by 1 commit.

(use "git push" to publish your local commits)

nothing to commit, working tree clean

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git log

commit 8481161c3783bf8e58a2b4e1752e2c9e2bc8adb3 (HEAD -> main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 23:42:19 2023 +0200

uva

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (origin/main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

Si se publica desde aquí el comando git push, la rama se posicionado arriba y si alguien clona obtendrá los tres cambios.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Writing objects: 100% (3/3), 272 bytes | 272.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1/

106d185..8481161 main -> main

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git log

commit 8481161c3783bf8e58a2b4e1752e2c9e2bc8adb3 (HEAD -> main, origin/main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 23:42:19 2023 +0200

uva

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

\*\*\* nos queda casi nada, ahora tenemos que obtener los platanos con el comando **git pull** **=** nos va hacer la operación de **fetch+merge**

y lo hará con fast -forward

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git pull

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), 265 bytes | 15.00 KiB/s, done.

From C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1

8481161..7450eda main -> origin/main

Updating 8481161..7450eda

Fast-forward

lista1A.md | 1 +

1 file changed, 1 insertion(+)

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3 (main)

$ git log

commit 7450eda4ef7509eaa784a3a9d378ba30abed98c8 (HEAD -> main, origin/main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Fri Oct 6 00:07:31 2023 +0200

platanos

commit 8481161c3783bf8e58a2b4e1752e2c9e2bc8adb3

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 23:42:19 2023 +0200

uva

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

:

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Terminal 2:

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git status

On branch main

Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

si hacemos el git log vemos que están en sincronía todavía en la rama local co el del repositorio remoto. Esto se explica que el Git no sabe si hay cambios nuevos en el servidor. Para ello vamos a realizar unos comandos.

Se hará con el comando fetch

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git log

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (HEAD -> main, origin/main, origin/HEAD)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git log

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (HEAD -> main, origin/main, origin/HEAD)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git log

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (HEAD -> main, origin/main, origin/HEAD)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

aqui vemos el efecto del comando fetch y que va detectar un cambio:

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git fetch

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), 252 bytes | 31.00 KiB/s, done.

From C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1

106d185..8481161 main -> origin/main

Vemos que origin/main va por delante perola rama main va detrás.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git log --all

commit 8481161c3783bf8e58a2b4e1752e2c9e2bc8adb3 (origin/main, origin/HEAD)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 23:42:19 2023 +0200

uva

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481 (HEAD -> main)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

hacemos un git status y nos confirma que la main va por detrás (behind) de la rama local.

Importante después del comando git fetch el git se ha conectado, le a permitido saber que tiene un nuevo commit que está allí pero no nos actualiza nuestra rama local con los nuevos cambios simplemente nos avisa.

Quedaría hacer la fusión de la rama remota en nuestra rama local

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git status

On branch main

Your branch is behind 'origin/main' by 1 commit, and can be fast-forwarded.

(use "git pull" to update your local branch)

nothing to commit, working tree clean

Bueno vamos a ver si entendemos, nos sugiere hacer un git pull para conseguir incorporar estos cambios de la rama main arriba.

Una opción seria hacer el git pul o otra opción sería hacer una fusión manual con git merge.

Terecra fila, pregunta del examen.

Vale si ejecutamos el git merge vamos a pedirle que traiga lo que hay en la rama remota origin/main con la rama en que estoy en la main.

Ahora si hemos sincronizado la rama local con los cambios que hay.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git merge origin/main

Updating 106d185..8481161

Fast-forward

lista1A.md | 1 +

1 file changed, 1 insertion(+)

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git log

commit 8481161c3783bf8e58a2b4e1752e2c9e2bc8adb3 (HEAD -> main, origin/main, origin/HEAD)

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 23:42:19 2023 +0200

uva

commit 106d1854b5d05cbdbcf165eb79113f0a17bbc481

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:50:52 2023 +0200

albaricoques

commit 0dd78aedd33d4c5231eeac9e5d0053b74672e5b5

Author: andrei <a.micleusanu@edu.gva.es>

Date: Thu Oct 5 22:41:29 2023 +0200

melocotones y peras

si hacemos un git fetch nos dira que no hay nada nuevo, o si hacemos con git pul nos dira no nada nuevo , esta todo actualizado.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git fetch

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git pull

Already up to date.

Ahora vamos abrir el archivo lista1A desde el repo3Bis. Y le vamos añadir una nueva fruta, los platanos.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git status

On branch main

Your branch is up to date with 'origin/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: lista1A.md

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

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git add .

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git commit -m "platanos"

[main 7450eda] platanos

1 file changed, 1 insertion(+)

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 12 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 285 bytes | 285.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

To C:/Users/andri/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/RRemoto1/

8481161..7450eda main -> main

El comando git push si no hay nada modificado nos dira todo actualizado.

\*\*\* Ahora volvemos en repo2 y veremos que no esta n los platanos.

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$ git push

Everything up-to-date

andri@LAPTOP-US3E7NTC MINGW64 ~/Desktop/23\_24\_Benigasló/Entornos de Desarrollo/Repos\_GIT\_ED/repo3Bis (main)

$