

# CONTROL DE VERSIONES

*Maria Jose Salar Garcia*

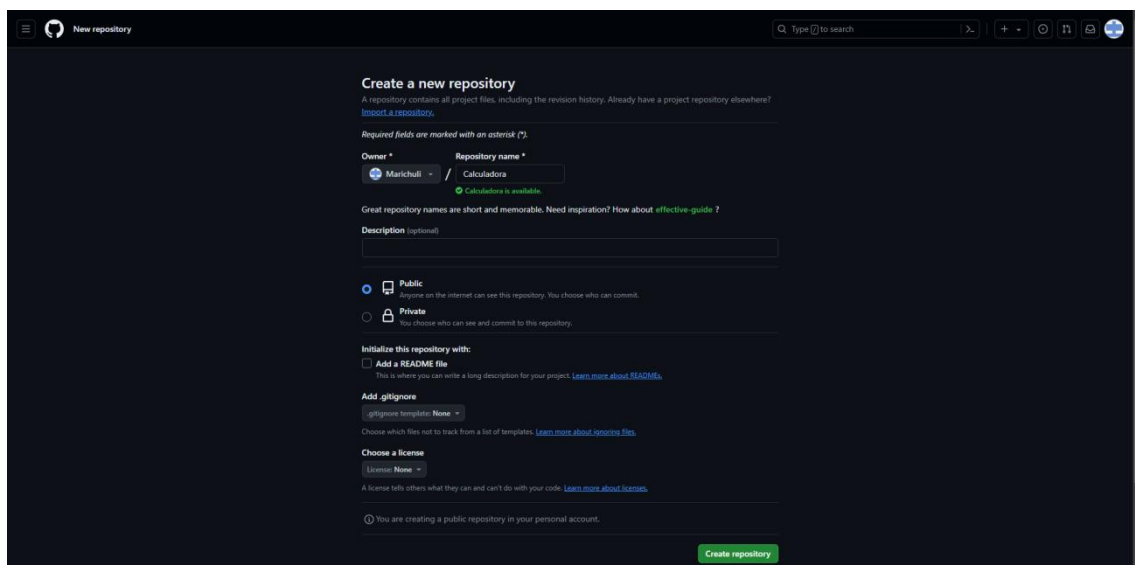
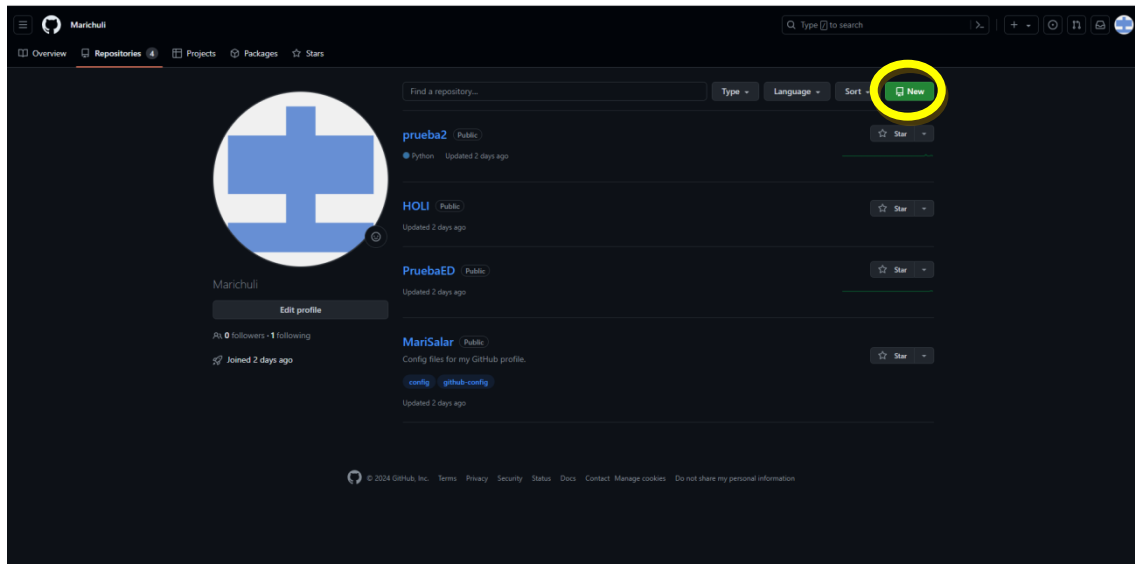
*Git y GitHub*

## **INDICE**

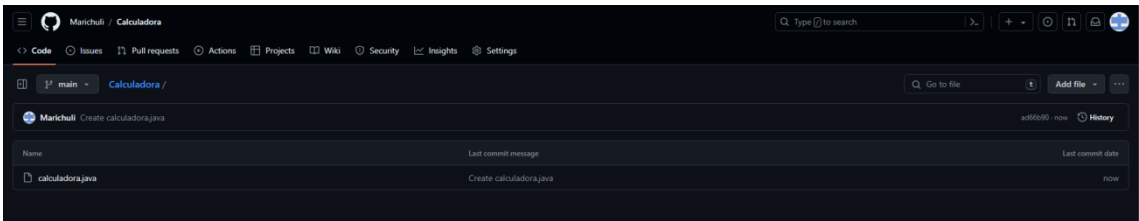
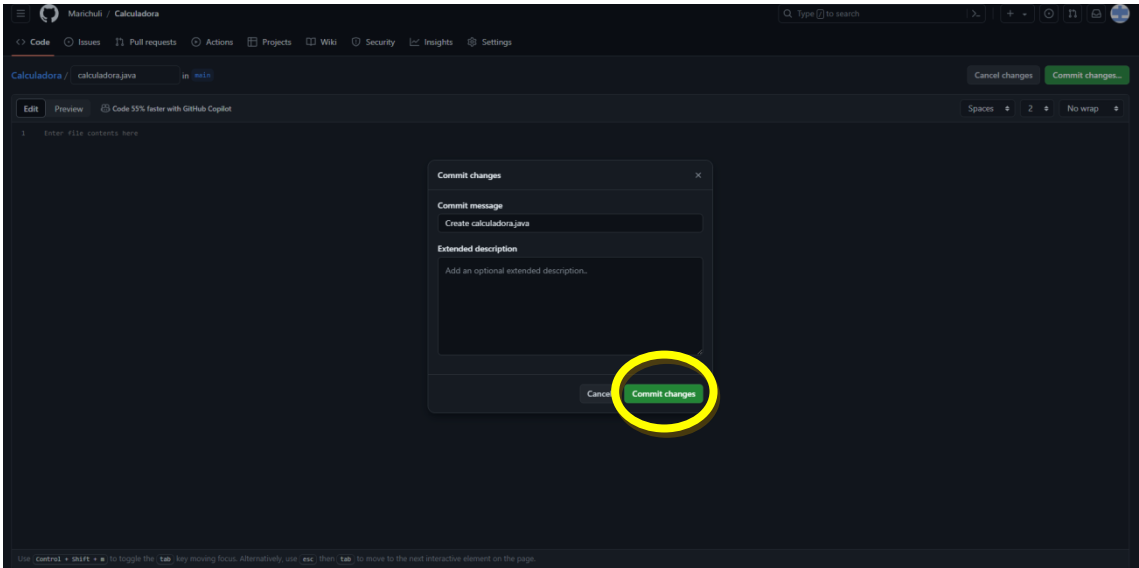
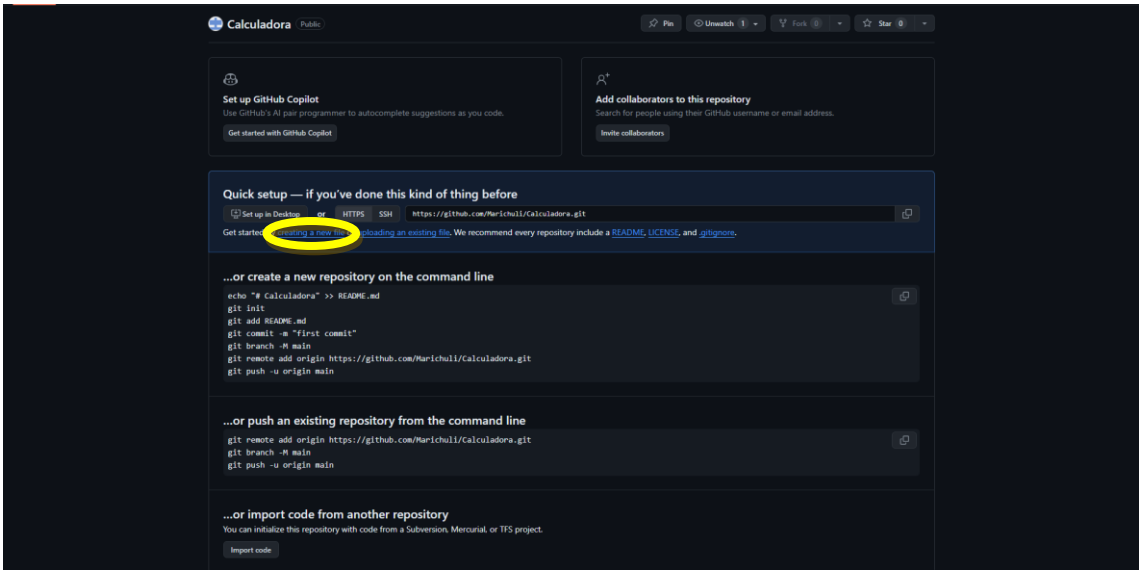
1. *Crear el repositorio Calculadora*
2. *Bajar el repositorio Calculadora a la copia local*
3. *Modificar la clase Calculadora*
4. *Subir los archivos a Github*
5. *Realizar cambios en la clase Calculadora directamente en el repositorio de  
GitHub*
6. *Modificar el proyecto Calculadora en local y volver a bajar el repositorio a la  
copia local.*
7. *Crear en el repositorio local Calculadora una rama llamada "rama2"*
8. *Identificar el archivo Calculadora.java de rama2 con un comentario que diga  
"rama2" y subir los cambios a GitHub.*
9. *Crear una situación de conflicto*
10. *Hacemos un fork del repositorio remoto de un compañero*
11. *Hacemos cambios en el fichero y los subimos al repositorio bifurcado*
12. *Solicitamos pull request al repositorio remoto original.*

## 1. Crear el repositorio Calculadora

Iniciamos sesión en Github y vamos al apartado “repositorios”, allí creamos un repositorio llamado Calculadora:



Dentro de éste, añadimos el fichero calculadora.java.



## 2. Bajar el repositorio Calculadora a la copia local

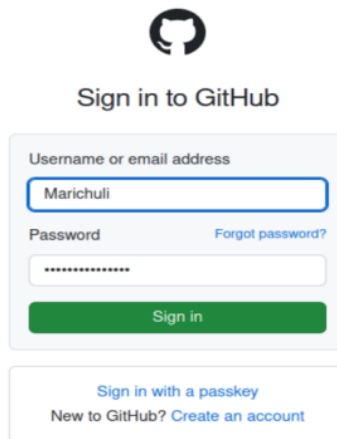
Abrimos la terminal de Linux, creamos una carpeta donde trabajaremos en el proyecto e iniciamos git dentro de esta.

```
administrador@Ubuntu22:~$ cd ProyectoEntornos/  
administrador@Ubuntu22:~/ProyectoEntornos$ git init  
Reinicializado el repositorio Git existente en /home/administrador/ProyectoEntornos/.git/  
administrador@Ubuntu22:~/ProyectoEntornos$  
  
administrador@Ubuntu22:~$ mkdir ProyectoEntornos  
administrador@Ubuntu22:~$ git init  
Reinicializado el repositorio Git existente en /home/administrador/.git/  
administrador@Ubuntu22:~$
```

Después, nos autenticamos con gh auth login

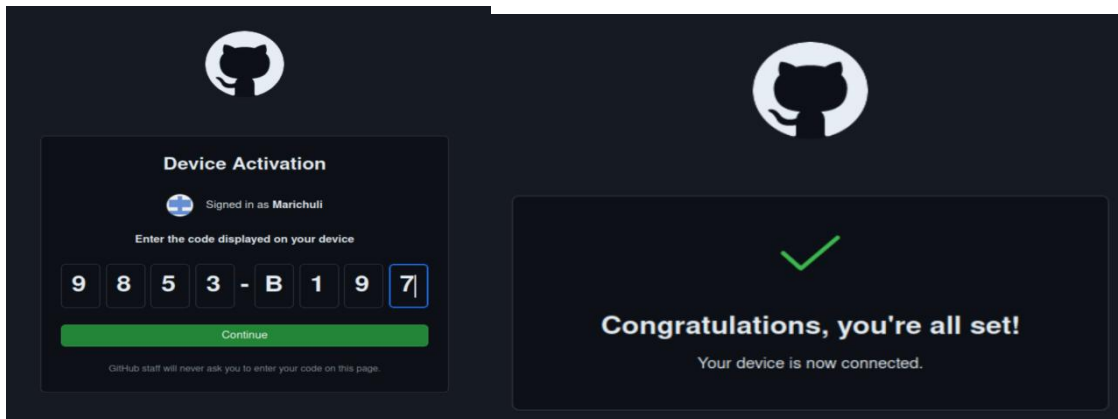
```
administrador@Ubuntu22:~$ gh auth login  
? What account do you want to log into? GitHub.com  
? You're already logged into github.com. Do you want to re-authenticate? Yes  
? What is your preferred protocol for Git operations? HTTPS  
? How would you like to authenticate GitHub CLI? Login with a web browser  
  
! First copy your one-time code: 9853-B197  
- Press Enter to open github.com in your browser...
```

Ponemos nuestro usuario y contraseña



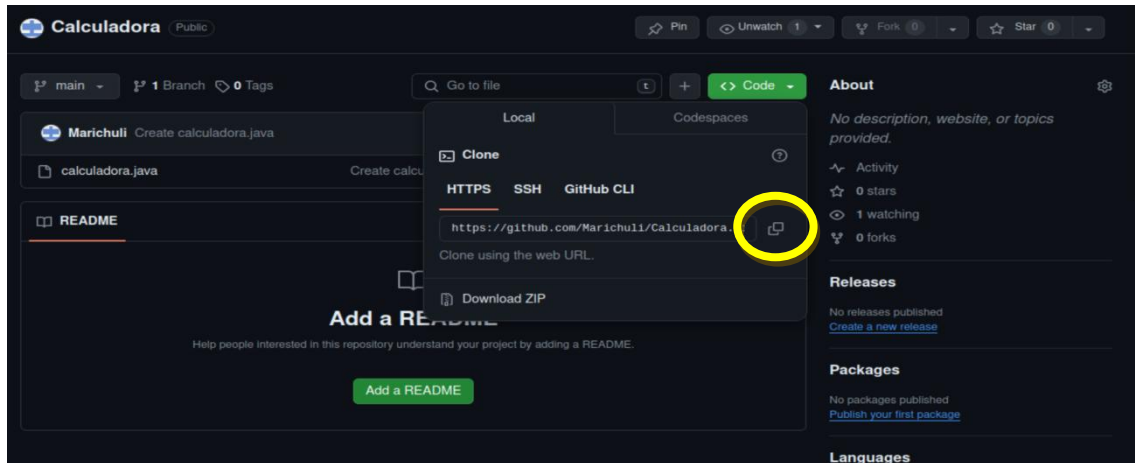
The image shows the GitHub sign-in interface. At the top is the GitHub logo and the text "Sign in to GitHub". Below this is a form with two input fields: "Username or email address" containing "Marichuli" and "Password" with masked characters. To the right of the password field is a link "Forgot password?". Below the inputs is a green "Sign in" button. At the bottom, there is a link "Sign in with a passkey" and another link "New to GitHub? Create an account".

E introducimos el código que nos ha dado gh en la terminal



The image shows the GitHub Device Activation screen. On the left, under the heading "Device Activation", it says "Signed in as Marichuli" and "Enter the code displayed on your device". Below this is a row of input boxes containing the code "9 8 5 3 - B 1 9 7", with the last box "7" highlighted. A green "Continue" button is below the code. At the bottom, a small note says "GitHub staff will never ask you to enter your code on this page." On the right, there is a large green checkmark and the text "Congratulations, you're all set!" followed by "Your device is now connected."

Nos traemos el repositorio calculadora a local, para ello. Vamos a nuestro repositorio y copiamos el enlace



Después en nuestra terminal ponemos clone (para traernos el repositorio) y el enlace copiado, y comprobamos que nos hemos traído correctamente el repositorio

```
administrador@Ubuntu22:~/ProyectoEntornos$ git clone https://github.com/Marichuli/Calculadora.git
Clonando en 'Calculadora'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 10 (delta 0), reused 3 (delta 0), pack-reused 0
Recibiendo objetos: 100% (10/10), listo.
```

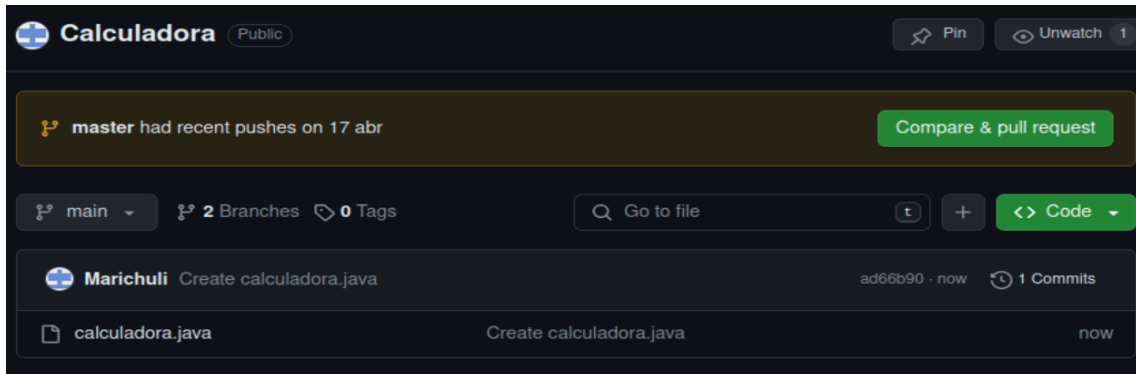
```
administrador@Ubuntu22:~/ProyectoEntornos$ ls
Calculadora
```



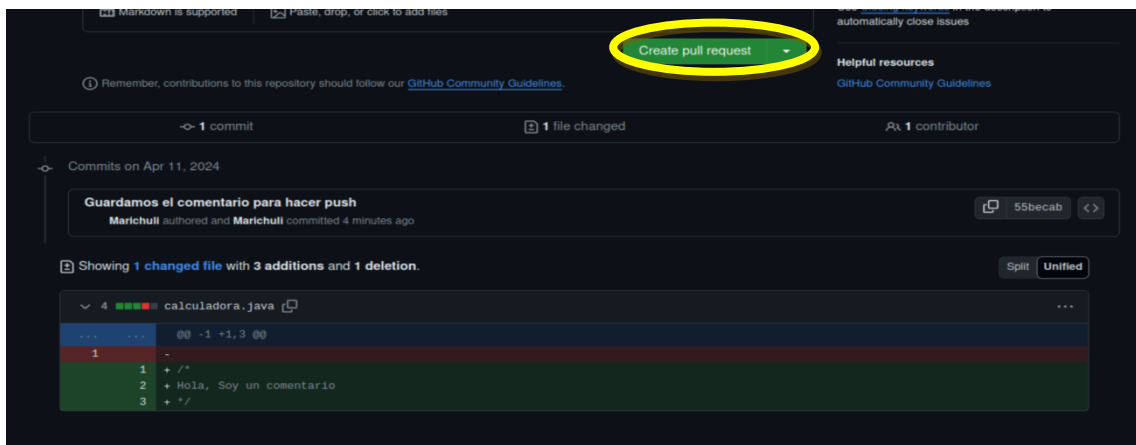
```
administrador@Ubuntu22:~/ProyectoEntornos$ git push https://github.com/Marichuli/Calculadora.git
Enumerando objetos: 5, listo.
Contando objetos: 100% (5/5), listo.
Escribiendo objetos: 100% (3/3), 305 bytes | 305.00 KiB/s, listo.
Total 3 (delta 0), reusados 0 (delta 0), pack-reusados 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/Marichuli/Calculadora/pull/new/master
remote:
To https://github.com/Marichuli/Calculadora.git
 * [new branch]      master -> master
```

Y comprobamos que los cambios se han actualizado en el repositorio.

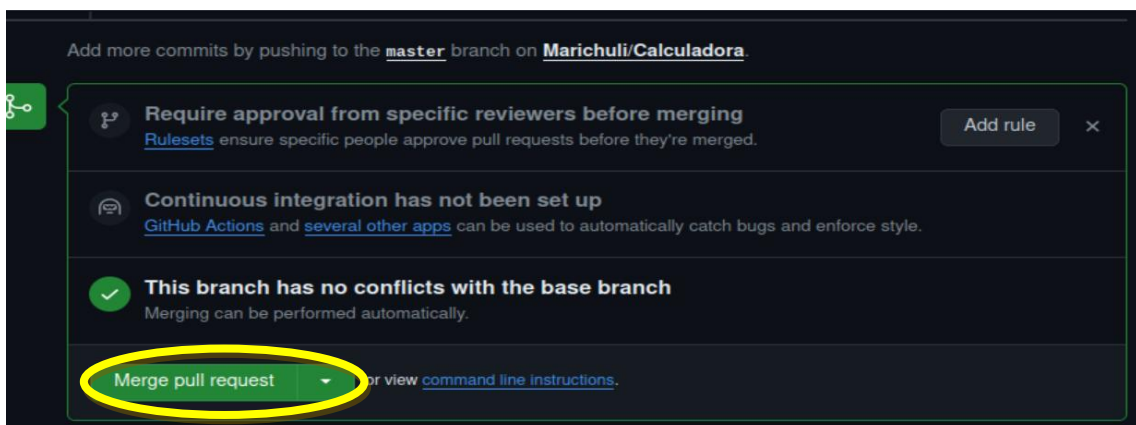
Si nos vamos a github, nos pide hacer pull request para integrar los cambios.



Si le damos a comparar y hacer pull request podemos ver los cambios que hemos realizado, así que los aprobamos:



Hacemos merge de los cambios



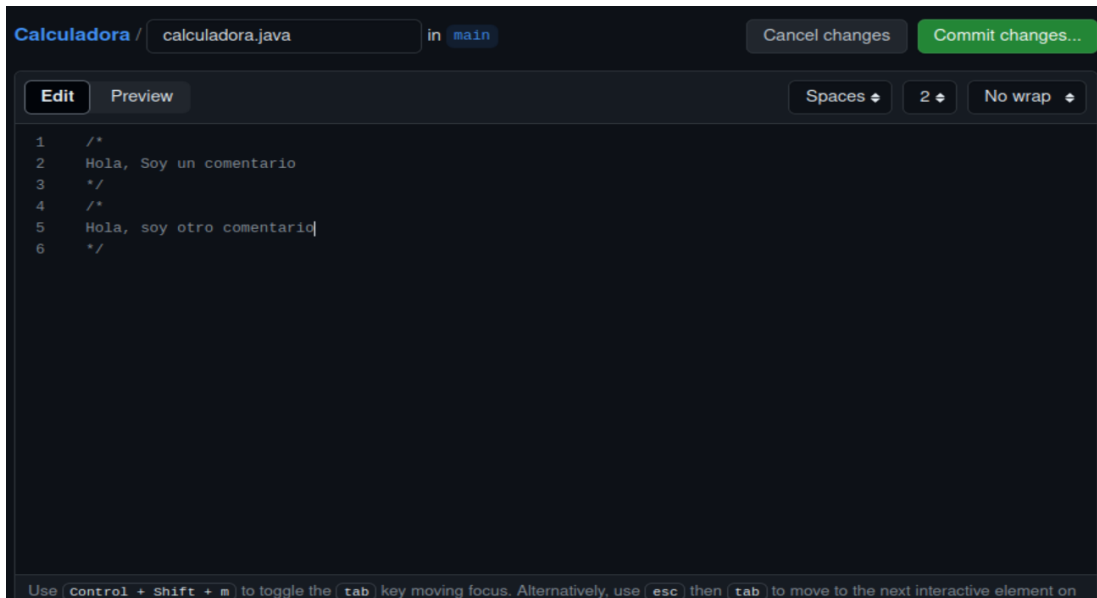


## 5. Realizar cambios en la clase Calculadora directamente en el repositorio de GitHub

Nos vamos al repositorio y le damos al lápiz



Añadimos otro comentario y hacemos commit changes



Actualizar los cambios en la copia local con git pull

```
administrador@Ubuntu22:~/ProyectoEntornos$ git pull https://github.com/Marichuli/Calculadora.git
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Desempaquetando objetos: 100% (4/4), 1.80 KiB | 461.00 KiB/s, listo.
Desde https://github.com/Marichuli/Calculadora
* branch      HEAD      -> FETCH_HEAD
Actualizando 55becab..f47866c
Fast-forward
 calculadora.java | 3 +++
1 file changed, 3 insertions(+)
```

Y comprobamos que los cambios se han actualizado.

```
administrador@Ubuntu22:~/ProyectoEntornos$ cat calculadora.java
/*
Hola, Soy un comentario
*/
/*
Hola, soy otro comentario
*/
```

## 6. Modificar el proyecto Calculadora en local y volver a bajar el repositorio a la copia local.

Volvemos a modificar calculadora

```
/*  
Hola, Soy un comentario  
*/  
/*  
Hola, soy otro comentario  
*/  
/*  
Hola, soy un tercer comentario  
*/
```

Y lo commiteamos

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git status  
En la rama main  
Tu rama está actualizada con 'origin/main'.  
  
Cambios no rastreados para el commit:  
  (usa "git add <archivo>..." para actualizar lo que será confirmado)  
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)  
      modificados:    calculadora.java  
  
sin cambios agregados al commit (usa "git add" y/o "git commit -a")  
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git add calculadora.java  
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git commit -m "tercer commit"  
[main 48b0441] tercer commit  
Committer: Marichuli <administrador@Ubuntu22.04>  
Tu nombre y correo fueron configurados automáticamente basados  
en tu usuario y nombre de host. Por favor verifica que sean correctos.  
Tú puedes suprimir este mensaje configurándolos de forma explícita:  
  
    git config --global user.name "Tu nombre"  
    git config --global user.email you@example.com  
  
Tras hacer esto, puedes arreglar tu identidad para este commit con:  
  
    git commit --amend --reset-author  
  
1 file changed, 3 insertions(+)
```

Al hacer git status nos informa que nuestra rama está adelantada

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git status  
En la rama main  
Tu rama está adelantada a 'origin/main' por 1 commit.  
  (usa "git push" para publicar tus commits locales)  
  
nada para hacer commit, el árbol de trabajo está limpio
```

Y hacemos git pull para traernos el repositorio

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ 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 1), reused 0 (delta 0), pack-reused 0
Desempaquetando objetos: 100% (3/3), 926 bytes | 926.00 KiB/s, listo.
Desde https://github.com/Marichuli/Calculadora
  48b0441..8de2c25  main      -> origin/main
Actualizando 48b0441..8de2c25
Fast-forward
 calculadora.java | 3 ---
 1 file changed, 3 deletions(-)
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ cat calculadora.java
/*
Hola, Soy un comentario
*/
/*
Hola, soy otro comentario
*/
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$
```

De esta forma, se elimina nuestro tercer comentario porque nos hemos bajado el repositorio que no tenia ese tercer comentario.

## 7. Crear en el repositorio local Calculadora una rama llamada "rama2"

Con git branch creamos la nueva rama y con git log comprobamos que ha sido creada

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git branch rama2
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git log
commit 8de2c2508f5e4ab091dc201861b706d6ab1740a0 (HEAD -> main, origin/main, origin/HEAD, rama2)
Author: Marichuli <166368051+Marichuli@users.noreply.github.com>
Date:   Wed Apr 17 12:28:32 2024 +0200

    Update calculadora.java
```

## 8. Identificar el archivo Calculadora.java de rama2 con un comentario que diga "rama2" y subir los cambios a GitHub.

Hacemos git checkout rama2 para pasar a ese rama

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git checkout rama2
Cambiado a rama 'rama2'
```

Modificamos calculadora, poniendo un comentario

```
/*
Hola, Soy un comentario
*/
/*
Hola, soy otro comentario
*/
/*
Estoy en rama2
*/
```

Guardamos los cambios con git add y git commit

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git status
En la rama rama2
Tu rama está actualizada con 'origin/rama2'.

Cambios no rastreados para el commit:
  (usa "git add <archivo>..." para actualizar lo que será confirmado)
  (usa "git restore <archivo>..." para descartar los cambios en el directorio de trabajo)
    modificados:    calculadora.java

sin cambios agregados al commit (usa "git add" y/o "git commit -a")
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git add calculadora.java
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git commit -m "commit para rama2"
[rama2 b91b2b2] commit para rama2
Committer: Marichuli <administrador@Ubuntu22.04>
Tu nombre y correo fueron configurados automáticamente basados
en tu usuario y nombre de host. Por favor verifica que sean correctos.
Tú puedes suprimir este mensaje configurándolos de forma explícita:

    git config --global user.name "Tu nombre"
    git config --global user.email you@example.com

Tras hacer esto, puedes arreglar tu identidad para este commit con:

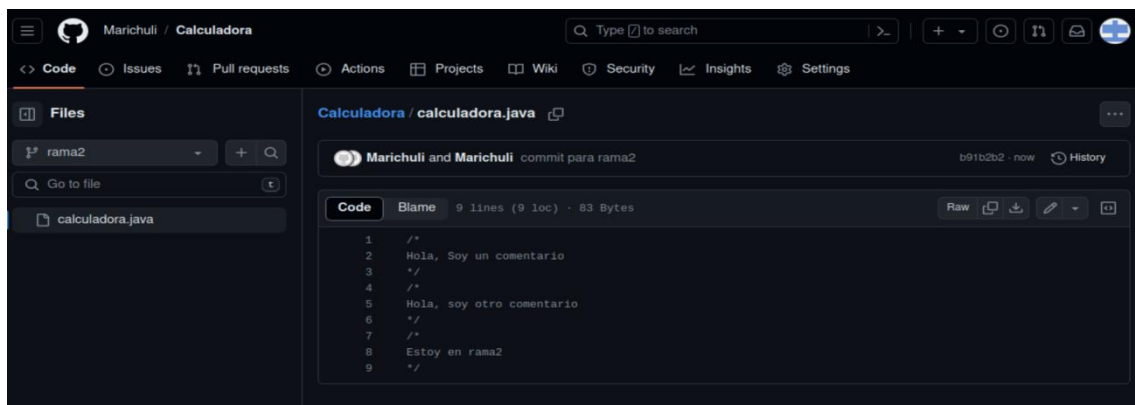
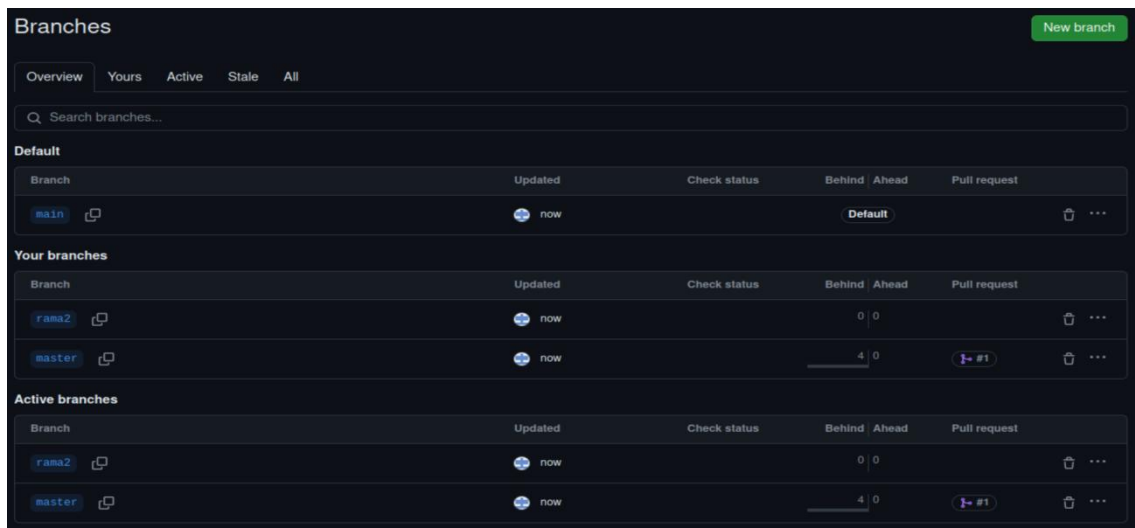
    git commit --amend --reset-author

1 file changed, 3 insertions(+)
```

Subimos la nueva rama a github con git push --set-upstream origin rama2 para subir la rama2

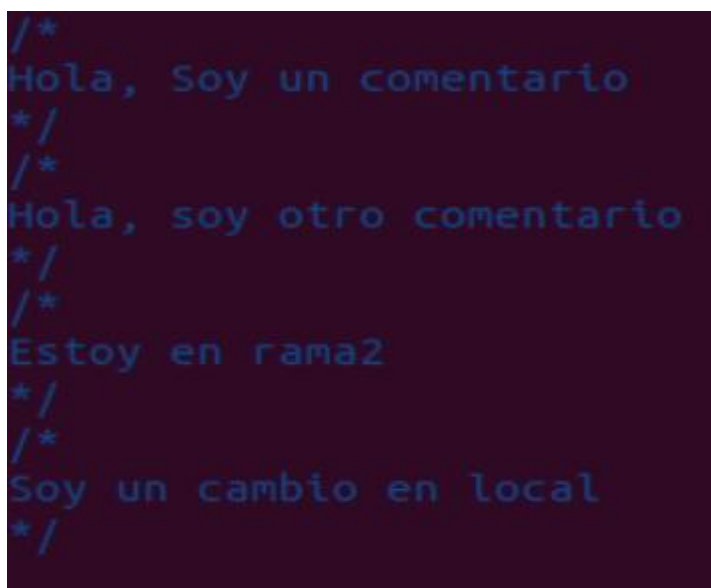
```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git push --set-upstream origin rama2
Total 0 (delta 0), reusados 0 (delta 0), pack-reusados 0
remote:
remote: Create a pull request for 'rama2' on GitHub by visiting:
remote:   https://github.com/Marichuli/Calculadora/pull/new/rama2
remote:
To https://github.com/Marichuli/Calculadora.git
 * [new branch]    rama2 -> rama2
Rama 'rama2' configurada para hacer seguimiento a la rama remota 'rama2' de 'origin'.
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$
```

Comprobar que los cambios se han actualizado en la rama anteriormente creada.



## 9. Crear una situación de conflicto

En primer lugar, modificamos el repositorio en local



Le hacemos git add para trackearlo y git commit para guardar los cambios



```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git add calculadora.java
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git commit calculadora.java
[rama2 3dbbf72] commit con conflicto
Committer: Marichuli <administrador@Ubuntu22.04>
Tu nombre y correo fueron configurados automáticamente basados
en tu usuario y nombre de host. Por favor verifica que sean correctos.
Tú puedes suprimir este mensaje configurándolos de forma explícita:

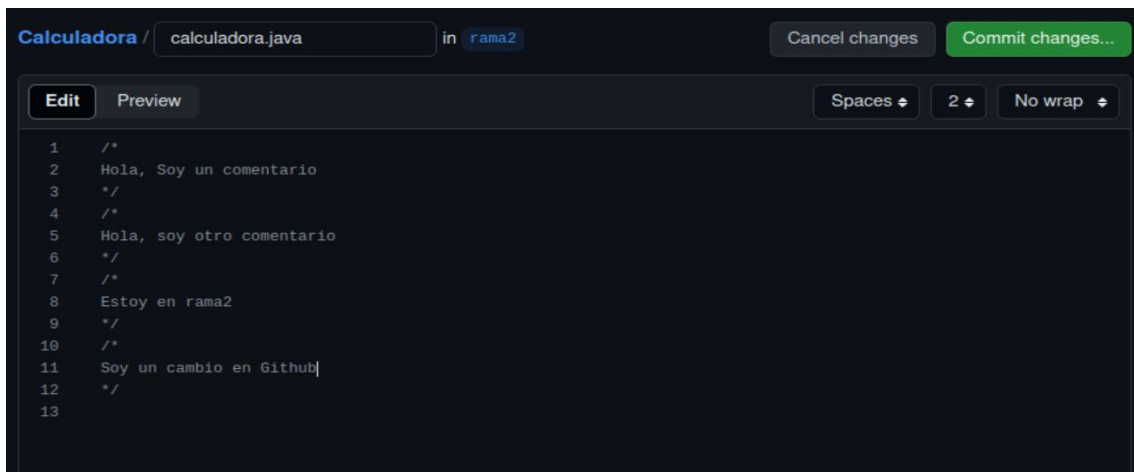
    git config --global user.name "Tu nombre"
    git config --global user.email you@example.com

Tras hacer esto, puedes arreglar tu identidad para este commit con:

    git commit --amend --reset-author

1 file changed, 3 insertions(+)
```

Luego, modificamos el repositorio de Github en este caso en la rama2



```
Calculadora / calculadora.java in rama2
Cancel changes Commit changes...

Edit Preview Spaces 2 No wrap

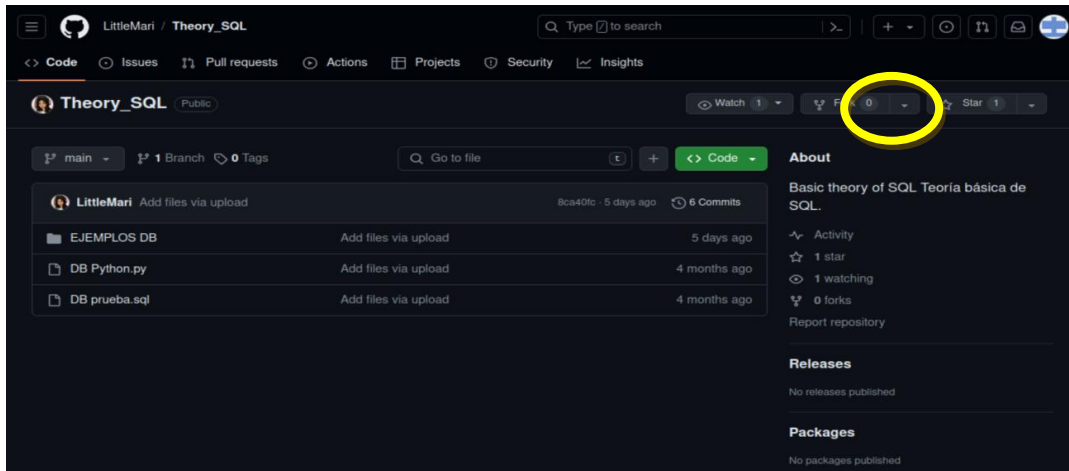
1  /*
2  Hola, Soy un comentario
3  */
4  /*
5  Hola, soy otro comentario
6  */
7  /*
8  Estoy en rama2
9  */
10 /*
11 Soy un cambio en Github
12 */
13
```

Finalmente subimos el archivo de local a github, donde habrá conflicto

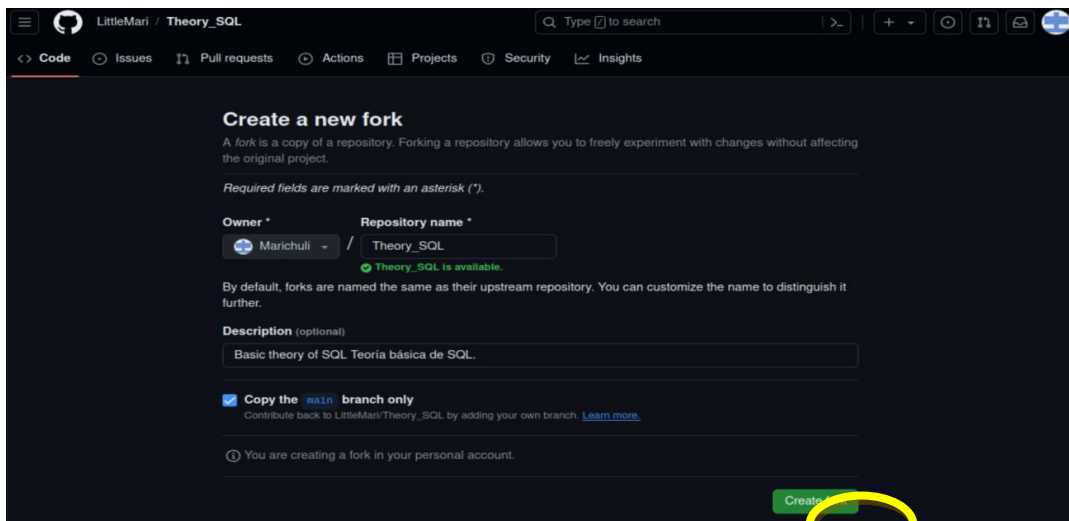
```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git push
To https://github.com/Marichuli/Calculadora.git
! [rejected]        rama2 -> rama2 (fetch first)
error: falló el push de algunas referencias a 'https://github.com/Marichuli/Calculadora.git'
ayuda: Actualizaciones fueron rechazadas porque el remoto contiene trabajo que
ayuda: no existe localmente. Esto es causado usualmente por otro repositorio
ayuda: realizando push a la misma ref. Quizás quieras integrar primero los cambios
ayuda: remotos (ej. 'git pull ...') antes de volver a hacer push.
ayuda: Mira 'Notes about fast-forwards' en 'git push --help' para detalles.
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$
```

## 10. Hacemos un fork del repositorio remoto de un compañero

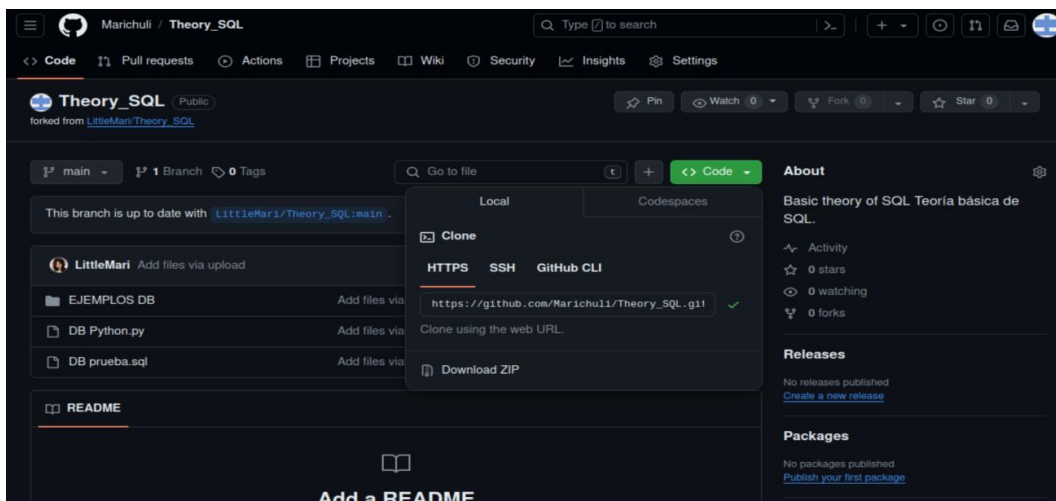
Vamos al repositorio de nuestro compañero y le damos a “fork”



Después lo creamos, dándole a “create fork”



Una vez tenemos el fork en nuestra cuenta, lo clonamos en local para poder trabajar con él. Para ello copiamos el enlace del repositorio y hacemos git clone



```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora$ git clone https://github.com/Marichuli/Theory_SQL.git
Clonando en 'Theory_SQL'...
remote: Enumerating objects: 35, done.
remote: Counting objects: 100% (35/35), done.
remote: Compressing objects: 100% (35/35), done.
remote: Total 35 (delta 15), reused 0 (delta 0), pack-reused 0
Recibiendo objetos: 100% (35/35), 91.55 KiB | 655.00 KiB/s, listo.
Resolviendo deltas: 100% (15/15), listo.
```

## 11. Hacemos cambios en el fichero y los subimos al repositorio bifurcado

Abrimos uno de los ficheros, en este caso “database Tienda” y lo modificamos

```
DROP DATABASE IF EXISTS tienda;
CREATE DATABASE tienda CHARACTER SET utf8mb4;
USE tienda;

#HOLA, ESTOY MODIFICANDO database TIENDA

CREATE TABLE fabricante (
  id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(100) NOT NULL
);
```

Guardamos los cambios con git add y git commit

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora/Theory_SQL/EJEMPLOS_DB$ git add database\Tienda.sql
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora/Theory_SQL/EJEMPLOS_DB$ git commit -m "commit para subir a bifurcado"
[main 2ac5f8e] commit para subir a bifurcado
Committer: Marichuli <administrador@Ubuntu22.04>
Tu nombre y correo fueron configurados automáticamente basados
en tu usuario y nombre de host. Por favor verifica que sean correctos.
Tú puedes suprimir este mensaje configurándolos de forma explícita:

    git config --global user.name "Tu nombre"
    git config --global user.email you@example.com

Tras hacer esto, puedes arreglar tu identidad para este commit con:

    git commit --amend --reset-author

1 file changed, 3 insertions(+)
```

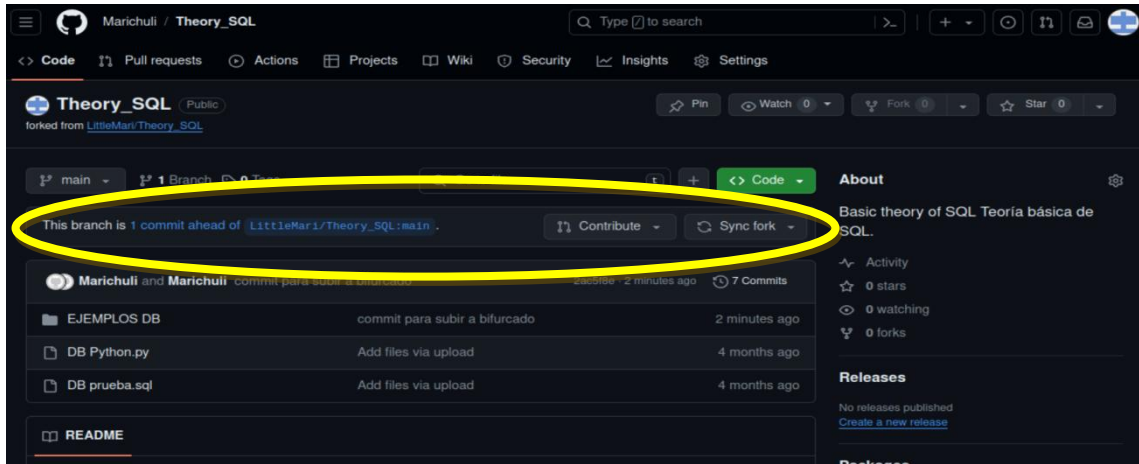
Y hacemos git push para subir los cambios al repositorio bifurcado

```
administrador@Ubuntu22:~/ProyectoEntornos/Calculadora/Theory_SQL/EJEMPLOS_DB$ git push
Enumerando objetos: 7, listo.
Contando objetos: 100% (7/7), listo.
Compresión delta usando hasta 4 hilos
Comprimiendo objetos: 100% (4/4), listo.
Escribiendo objetos: 100% (4/4), 401 bytes | 401.00 KiB/s, listo.
Total 4 (delta 3), reusados 0 (delta 0), pack-reusados 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/Marichuli/Theory_SQL.git
8ca40fc..2ac5f8e main -> main
```

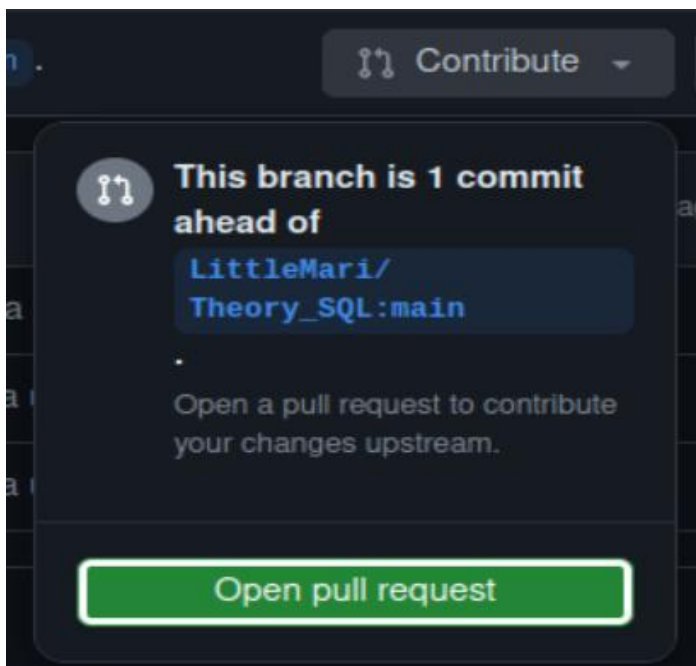


## 12. Solicitamos pull request al repositorio remoto original.

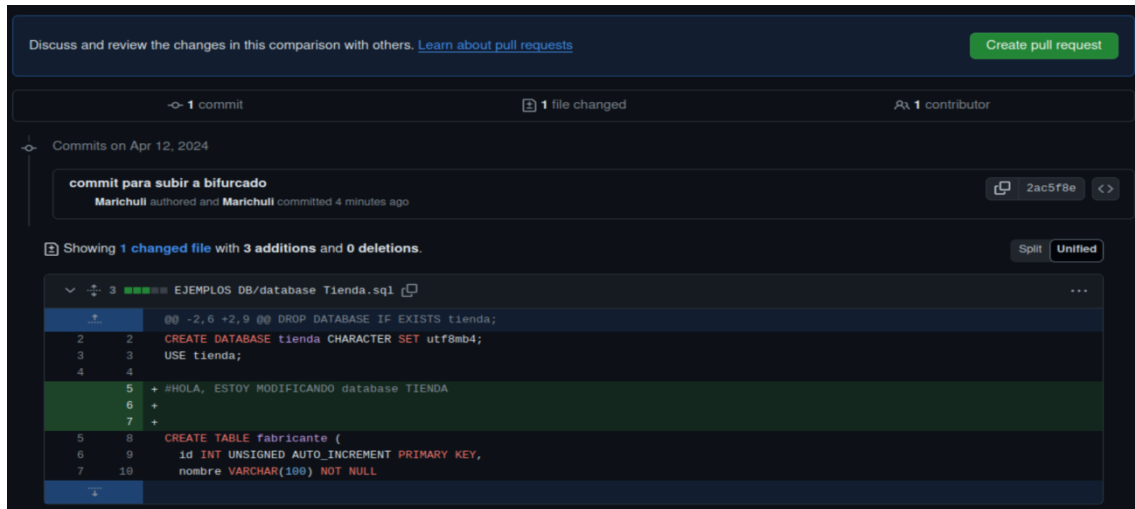
En este momento, cuando nos vamos a nuestro repositorio bifurcado, se han guardado los cambios y nos aparece esto:



Para solicitar subirlo al repositorio principal de nuestro compañero debemos darle a “Contribute” y “Open pull request”



Comprobamos los cambios que hemos realizado y si está todo correcto le damos a crear pull request



Nos sale este texto que nos indica que lo hemos hecho correctamente

