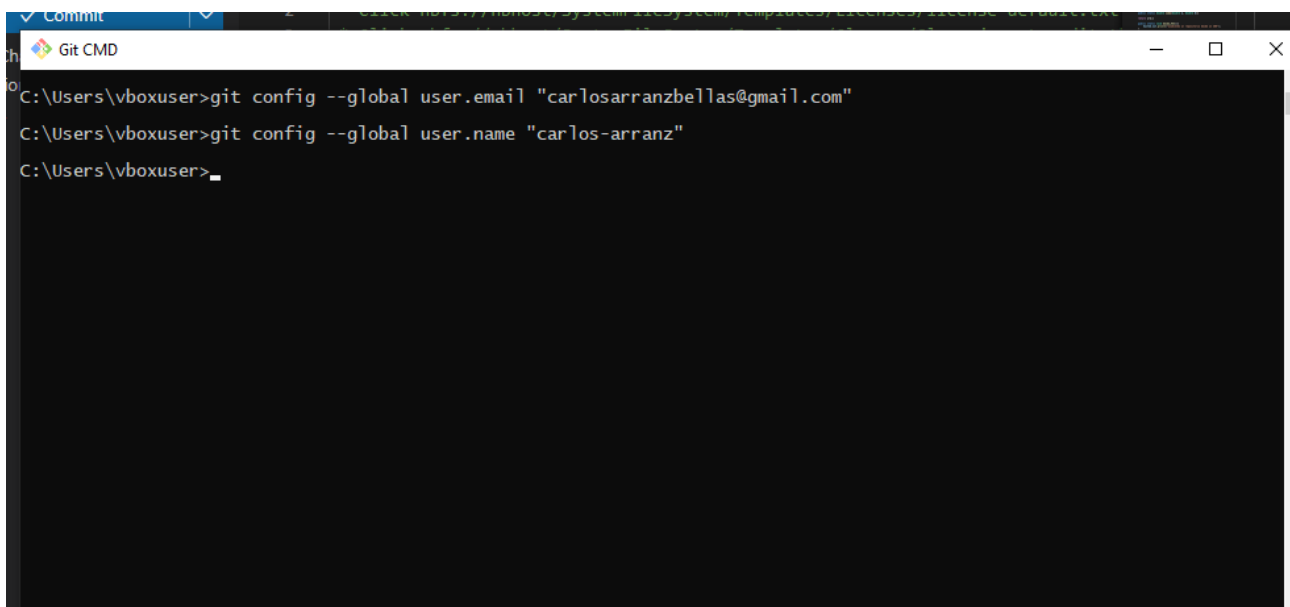
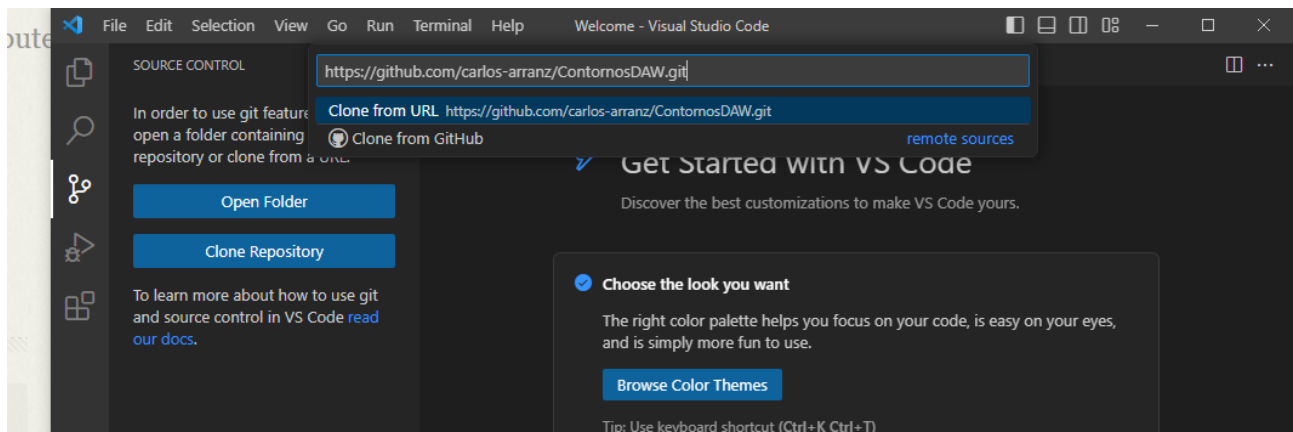


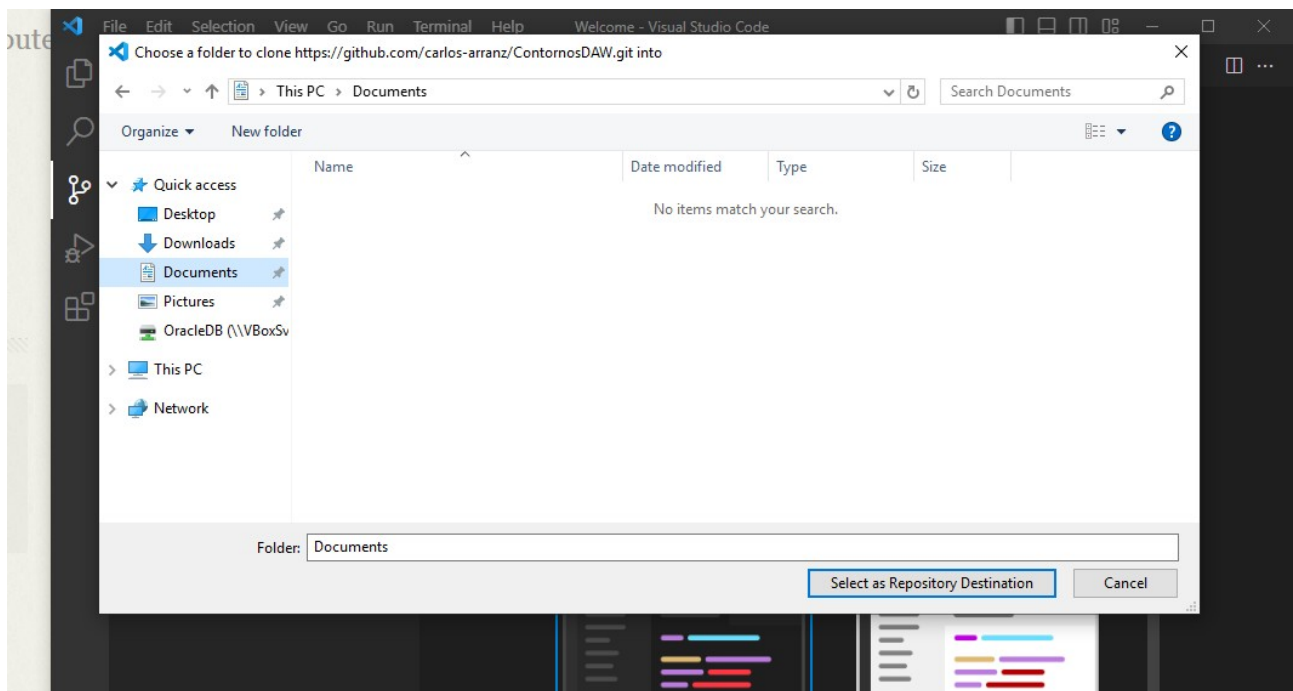
Instalar Git en el equipo



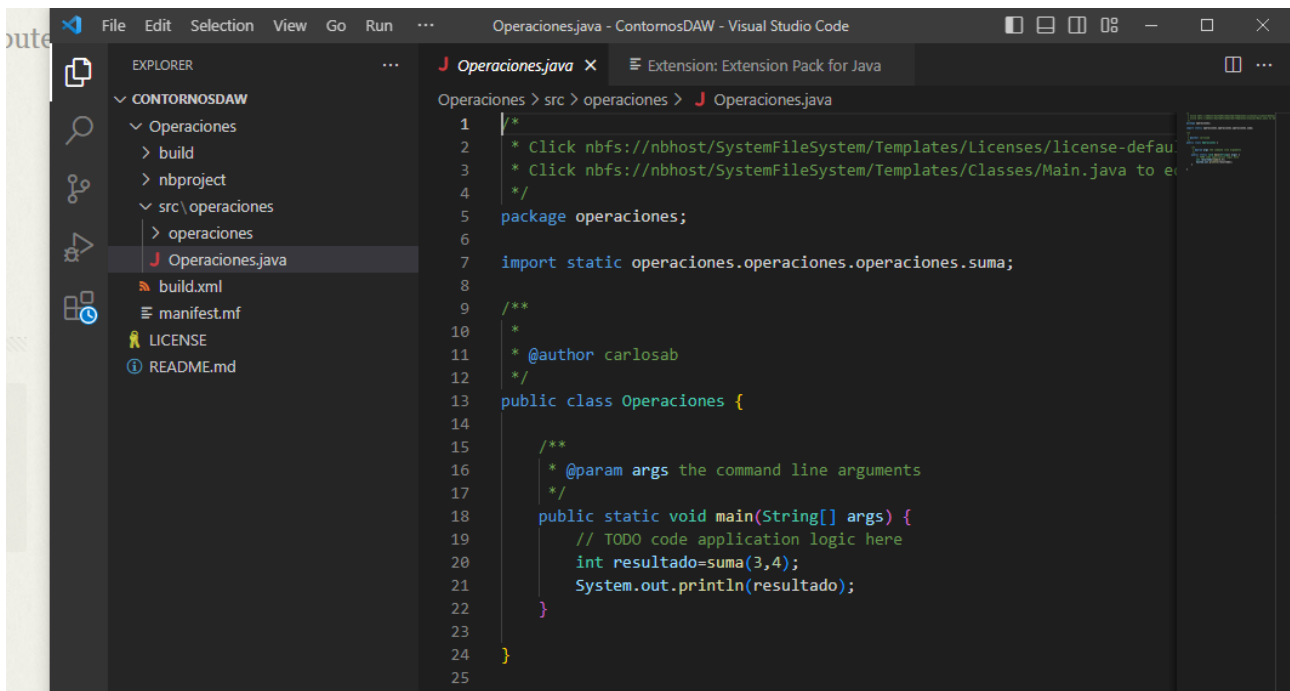
Desde la consola de git, introduce tu email y nombre de usuario de github con “**git config --global user.email/name ...**”



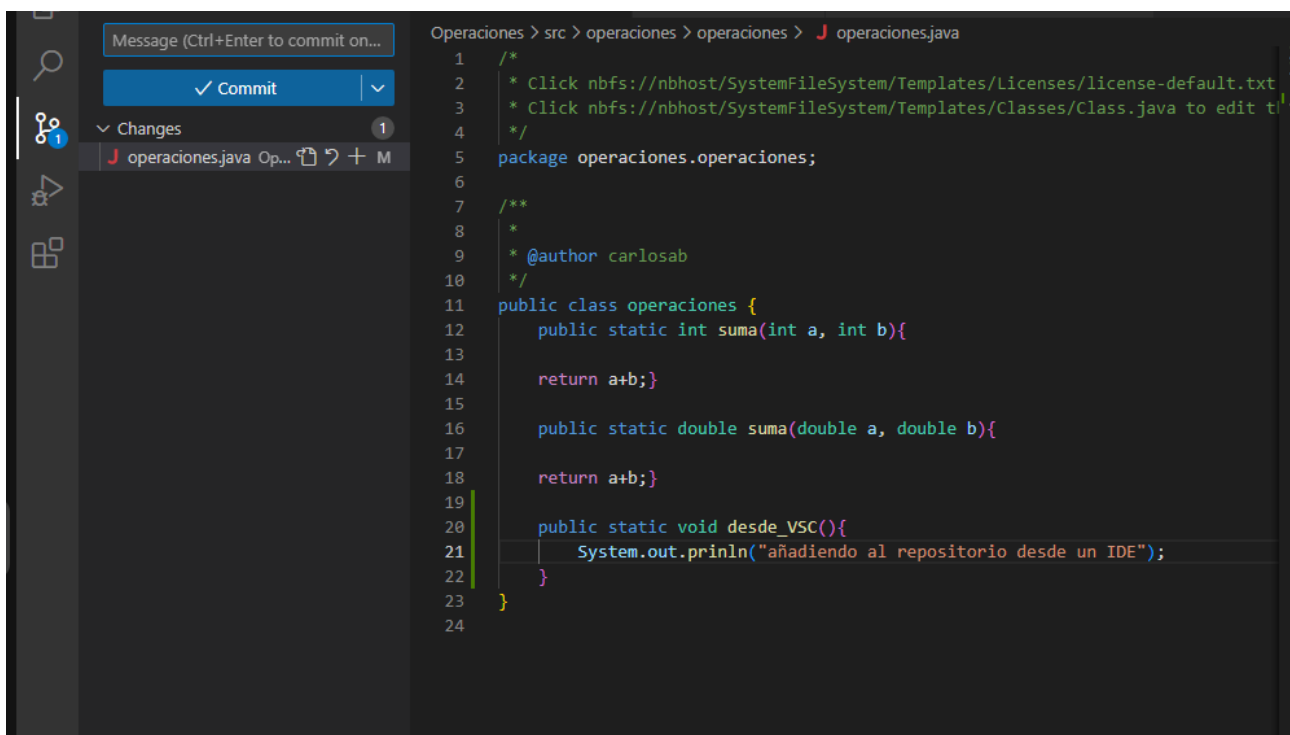
En VSC, ir a la opción “source control” >”clone repository” y pegar el enlace



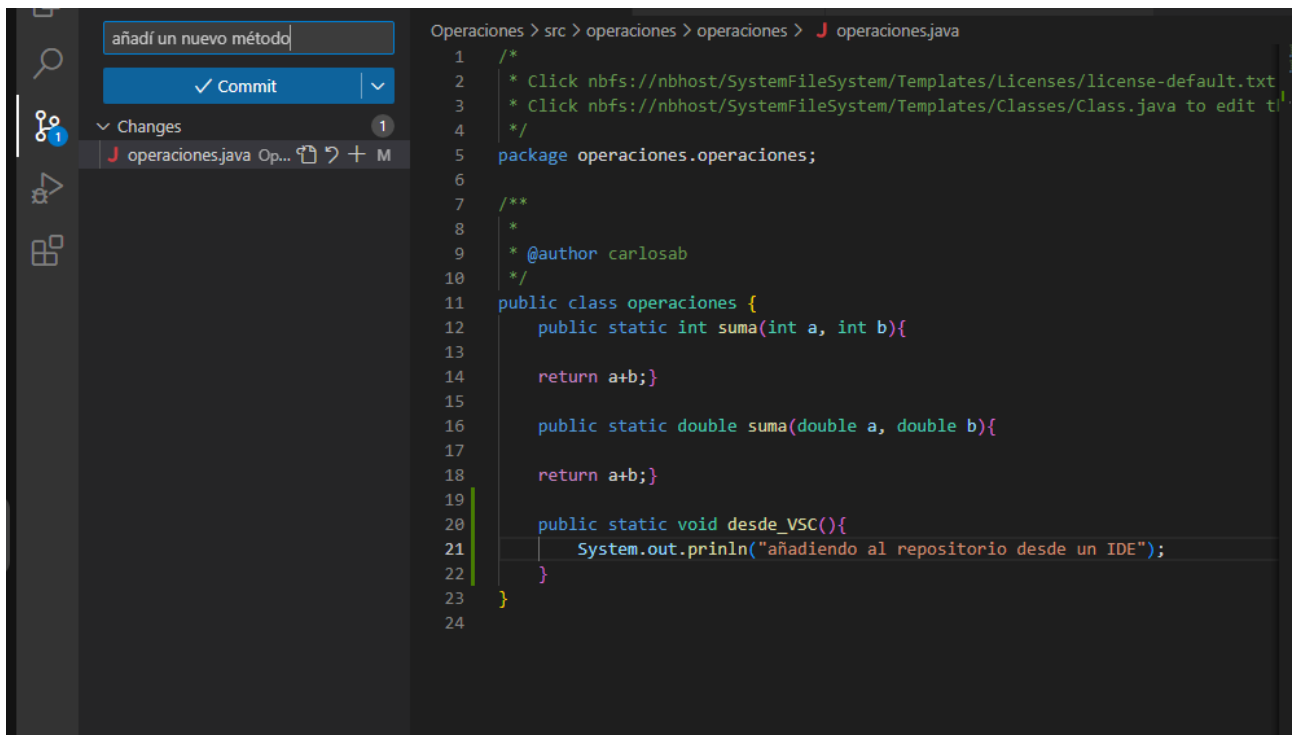
Seleccionar donde quieres tener tu copia del repositorio



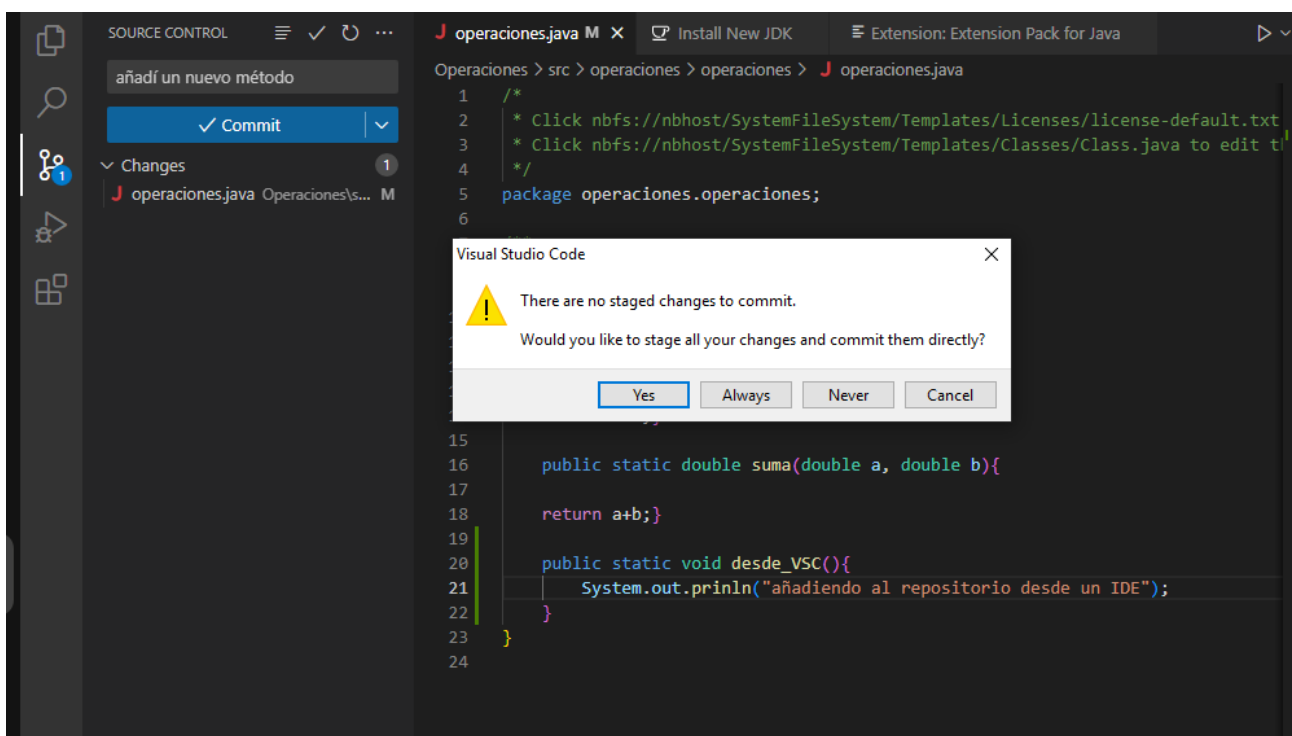
Seleccionamos la copia para trabajar con ella



Una vez hechos cambios sobre la copia, aparecerá una notificación en “source control”, vamos ahí y tenemos varias opciones de qué hacer con esos cambios



Añadimos un mensaje a la actualización que vamos a hacer y le damos a commit



Si no le dimos a “Stage changes”(símbolo +) antes de hacer commit, te preguntará, dale que sí.

- GitHub Apps
- OAuth Apps
- Personal access tokens
 - Fine-grained tokens Beta
 - Tokens (classic)

Fine-grained personal access tokens Beta

Generate new token

Need an API token for scripts or testing? [Generate a personal access token](#) for quick access to the [GitHub API](#).

Si no tienes uno ya, genera un token desde github

- GitHub Apps
- OAuth Apps
- Personal access tokens
 - Fine-grained tokens Beta
 - Tokens (classic)

New fine-grained personal access token Beta

Create a fine-grained, repository-scoped token suitable for personal API use and for using Git over HTTPS.

Token name *

token-DAW ✓

A unique name for this token. May be visible to resource owners.


Expiration *

30 days ⬇ The token will expire on Thu, Apr 13 2023

Description

What is this token for?

Resource owner

 carlos-arranz ▼

Repository access

☐ **Public Repositories (read-only)**

☐ **All repositories**

This applies to all current *and* future repositories owned by the resource owner.
Also includes public repositories (read-only).

☒ **Only select repositories**

Select at least one repository. Max 50 repositories.
Also includes public repositories (read-only).

 Select repositories ▼

Selected 1 repository.

 carlos-arranz/ContornosDAW ×

Configura el token

☐ **All repositories**
This applies to all current *and* future repositories owned by the resource owner.
Also includes public repositories (read-only).

☒ **Only select repositories**
Select at least one repository. Max 50 repositories.
Also includes public repositories (read-only).

Select repositories ▼

Selected 1 repository.

carlos-arranz/ContornosDAW x

Permissions

Read our [permissions documentation](#) for information about specific permissions.

Repository permissions 10 Selected ▼

Repository permissions permit access to repositories and related resources.

Actions ⓘ

Workflows, workflow runs and artifacts.

Access: Read and write ▼

Administration ⓘ

Repository creation, deletion, settings, teams, and collaborators.

Access: Read and write ▼

Code scanning alerts ⓘ

View and manage code scanning alerts.

Access: Read and write ▼

Codespaces ⓘ

Create, edit, delete and list Codespaces.

Access: Read and write ▼

Codespaces lifecycle admin ⓘ

Manage the lifecycle of Codespaces, including starting and stopping.

Access: No access ▼

Codespaces metadata ⓘ

Access Codespaces metadata including the devcontainers and machine type.

Access: No access ▼

Configura los permisos del token para poder hacer cambios sobre el repositorio

[Pull requests](#) [Issues](#) [Codespaces](#) [Marketplace](#) [Explore](#)

[Developer settings](#) / [Personal access tokens](#)

GitHub Apps

GitHub Apps

Personal access tokens

Fine-grained tokens

Classic OAuth apps

Fine-grained tokens

Beta

Fine-grained personal access tokens Beta Generate new token

These are fine-grained, repository-scoped tokens suitable for personal [API](#) use and for using Git over HTTPS.

Make sure to copy your personal access token now as you will not be able to see this again.

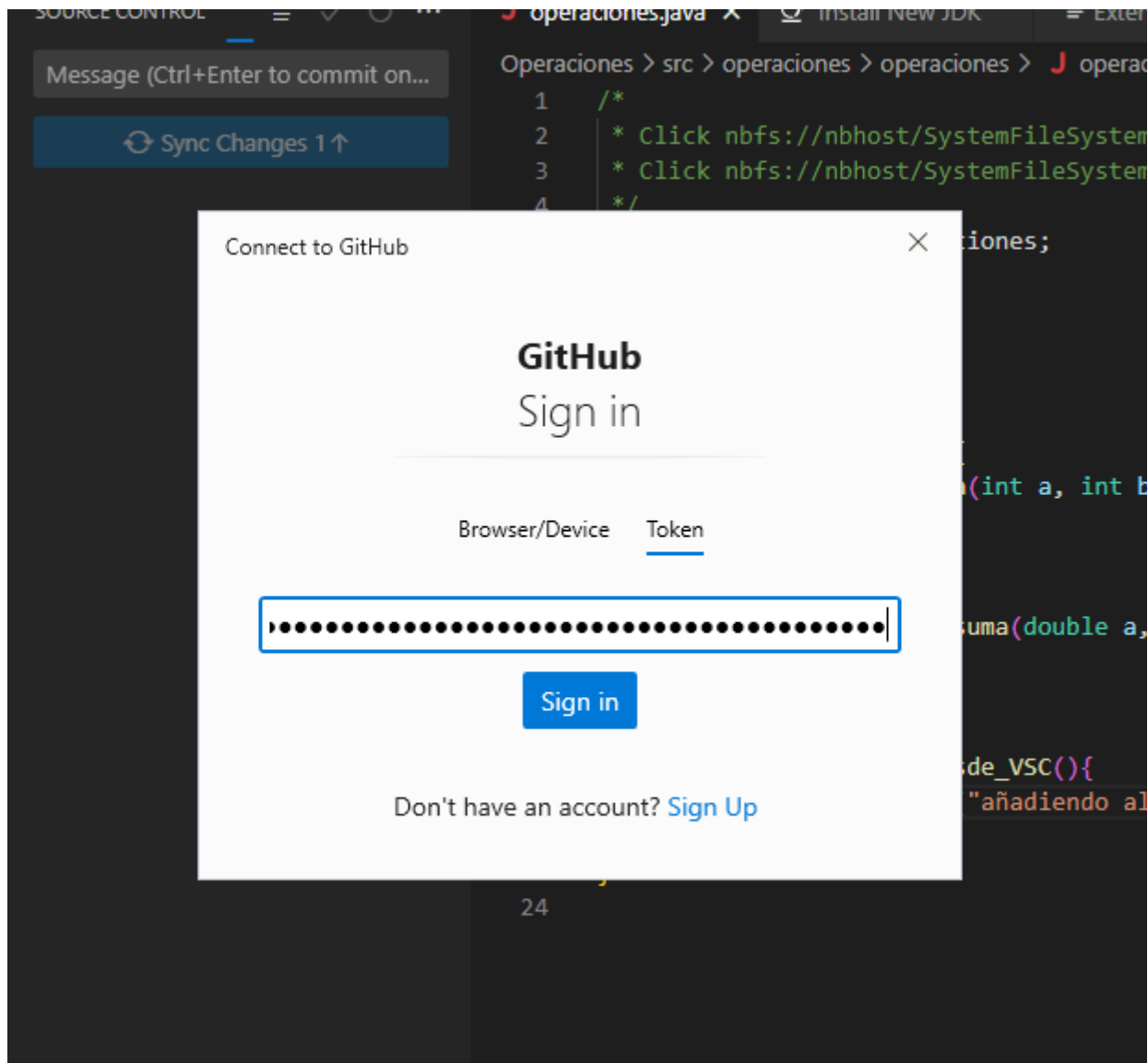
Never used Delete

Expires on Thu, Apr 13 2023.

© 2023 GitHub, Inc.

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

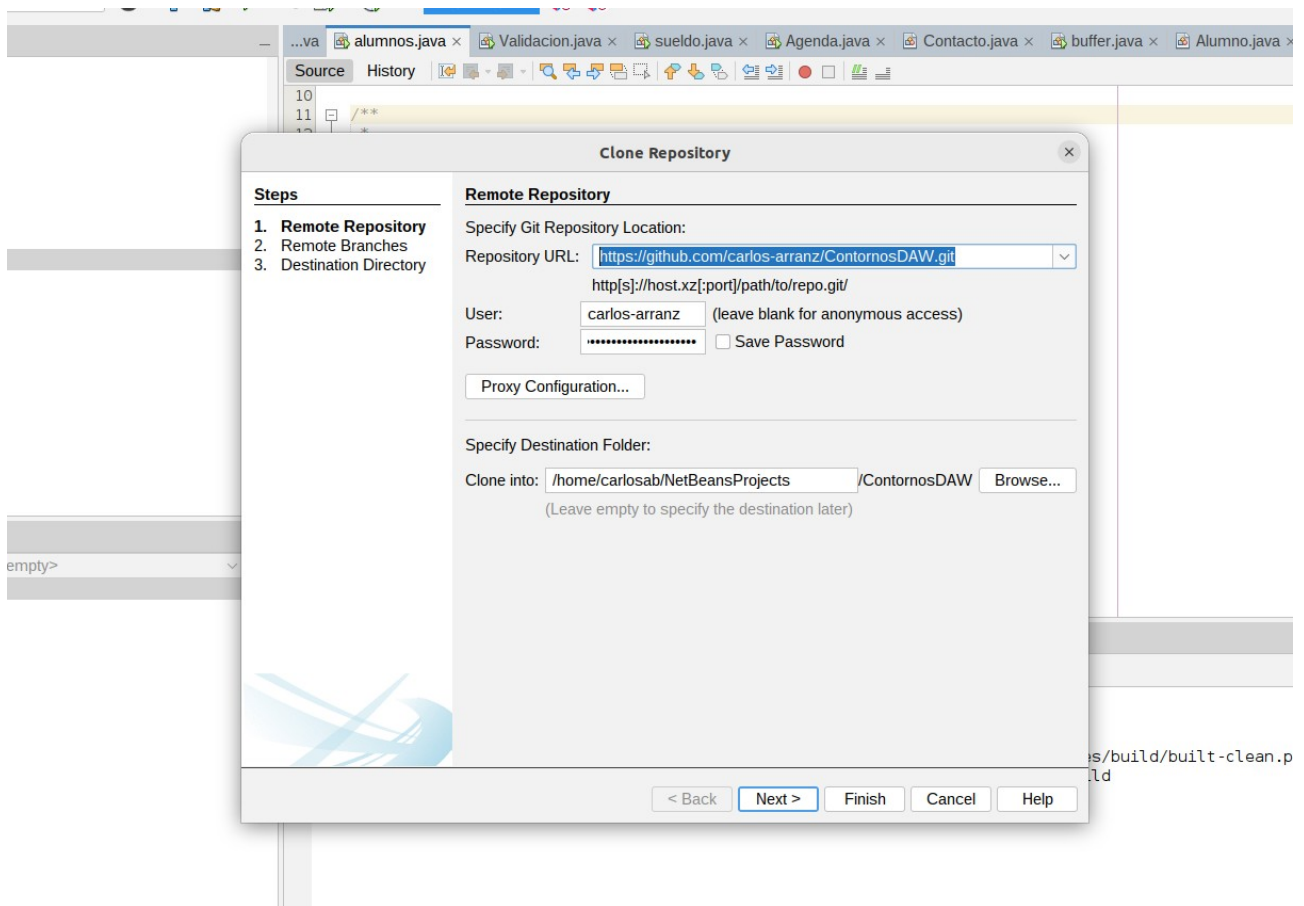
Una vez generado, copialo



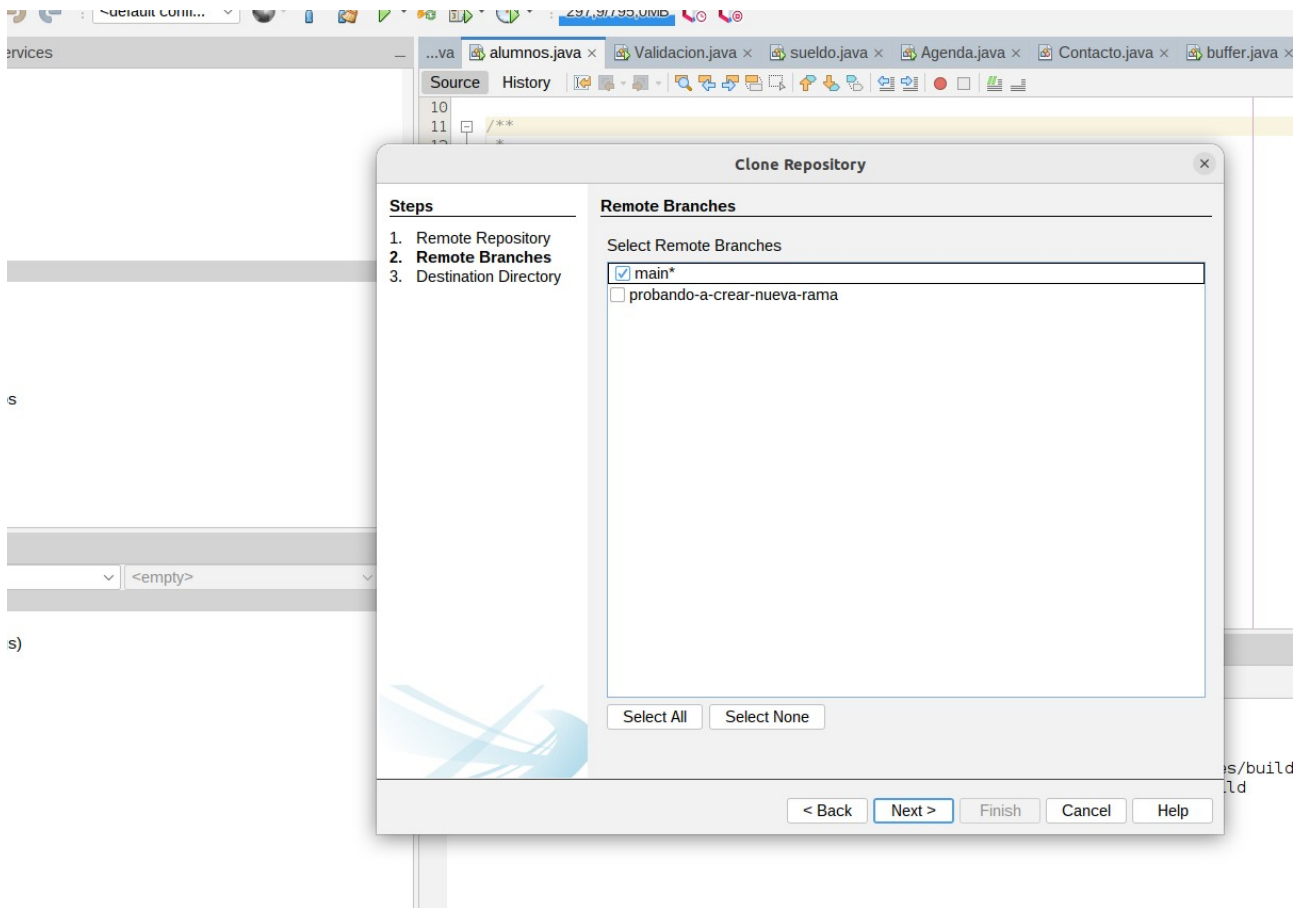
Al hacer commit, se te abrirá una ventana para acceder a github, usa el token para registrarte



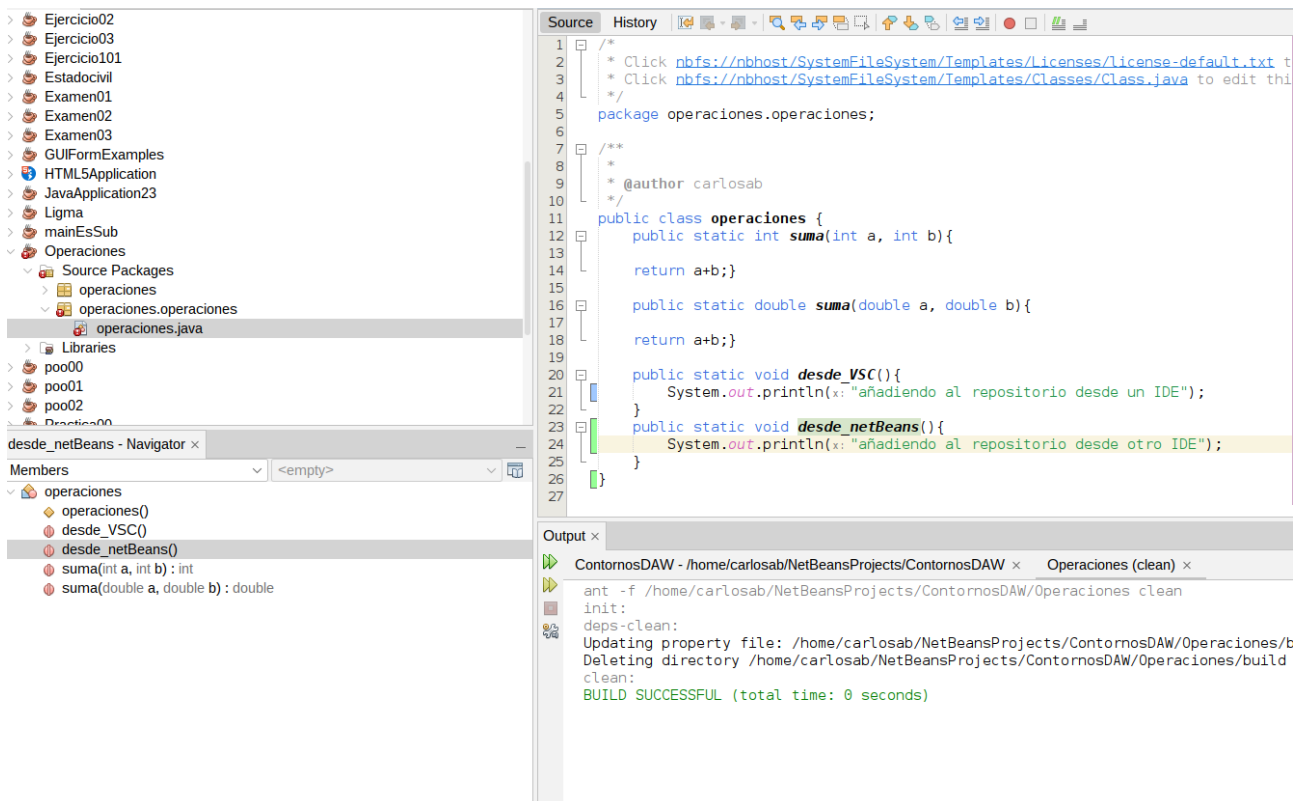
Una vez en el repositorio, verás que el cambio ha sido realizado



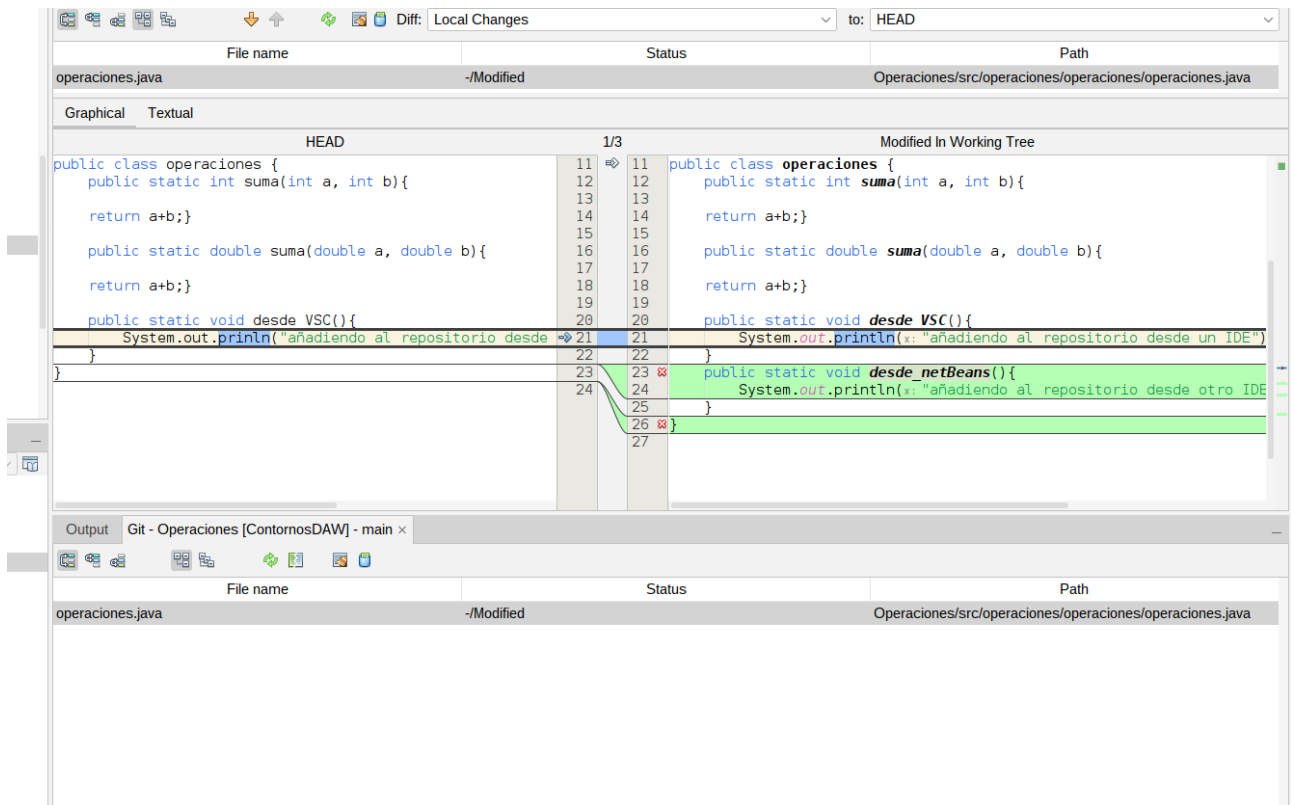
En netBeans, “Team”>”Git”>”Clone Repository”, e introduce el enlace , tu nombre de usuario y el token



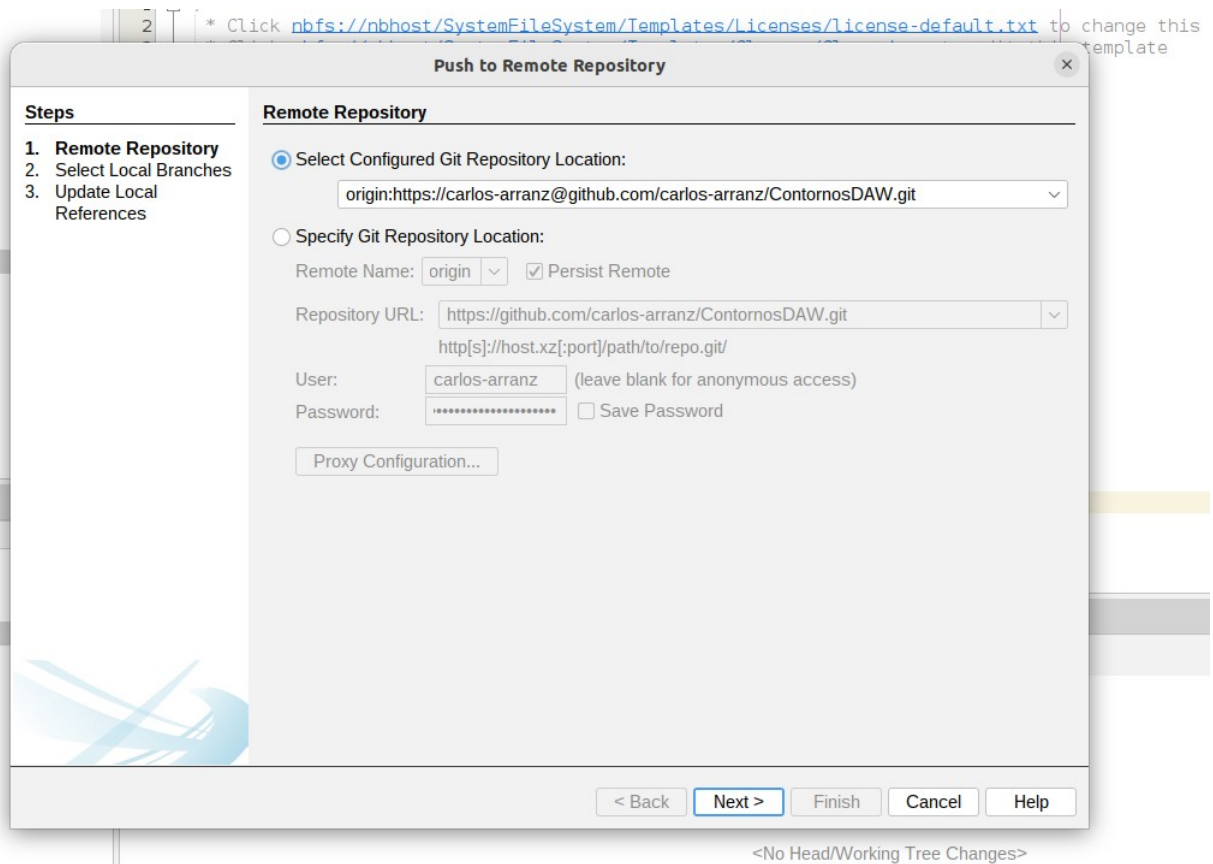
Selecciona la rama o ramas que quieres clonar



Actualizamos el código



click derecho en el proyecto >"Git">"Show changes" nos permite ver lo que se ha cambiado



Click derecho en proyecto>"Git">"Add" y luego "Commit", por último >"Git">"Remote">"Push" para mandar el cambio al repositorio



carlos-arranz solo commit

Latest commit a3224a2 6 minutes ago

History

1 contributor

27 lines (22 sloc) | 677 Bytes

Raw

Blame



```

1  /*
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
4   */
5   package operaciones.operaciones;
6
7   /**
8    *
9    * @author carlosab
10   */
11  public class operaciones {
12      public static int suma(int a, int b){
13
14          return a+b;}
15
16      public static double suma(double a, double b){
17
18          return a+b;}
19
20      public static void desde_VSC(){
21          System.out.println("añadiendo al repositorio desde un IDE");
22      }
23      public static void desde_netBeans(){
24          System.out.println("añadiendo al repositorio desde otro IDE");
25      }
26      //:)
27  }
```