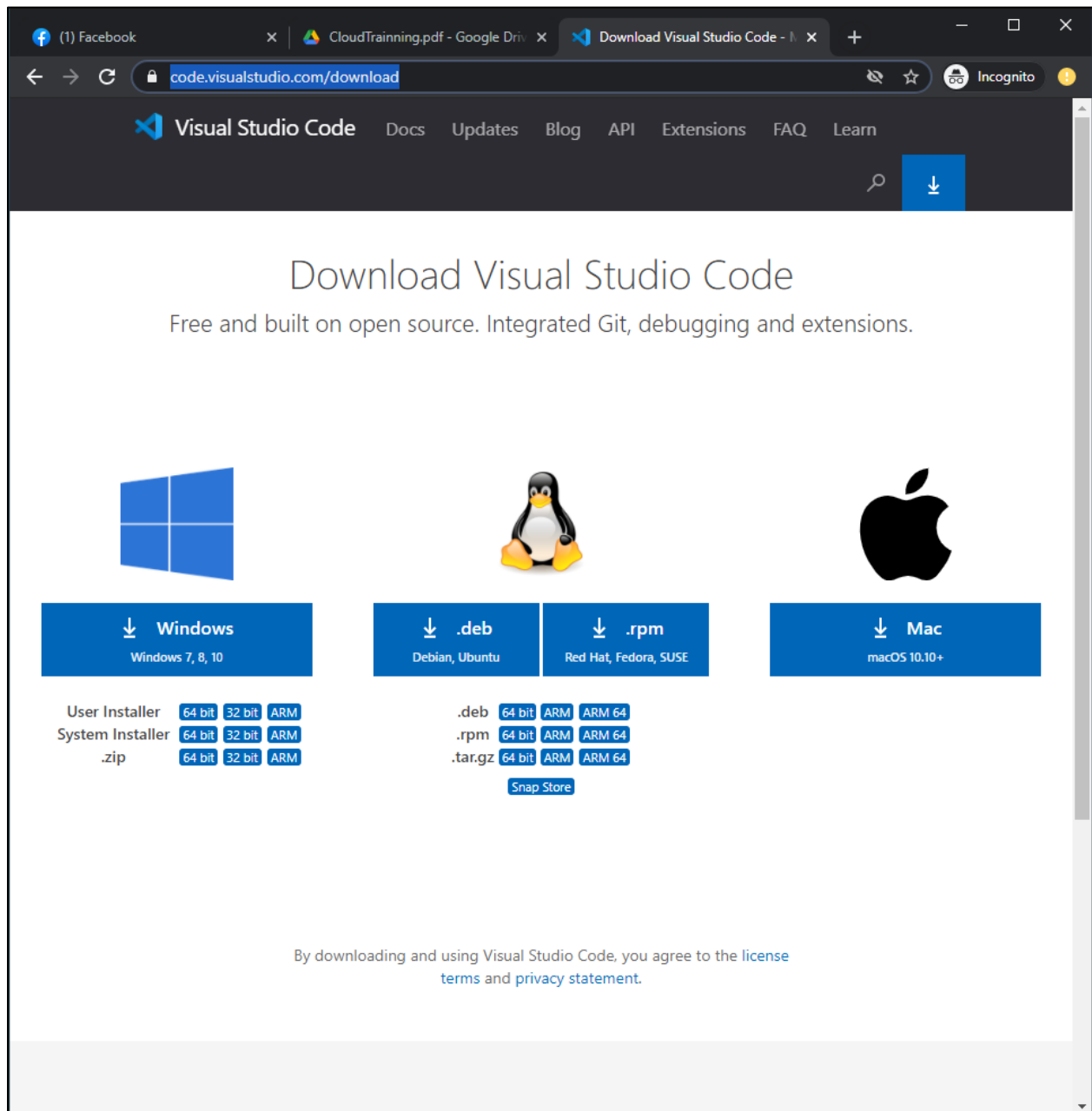
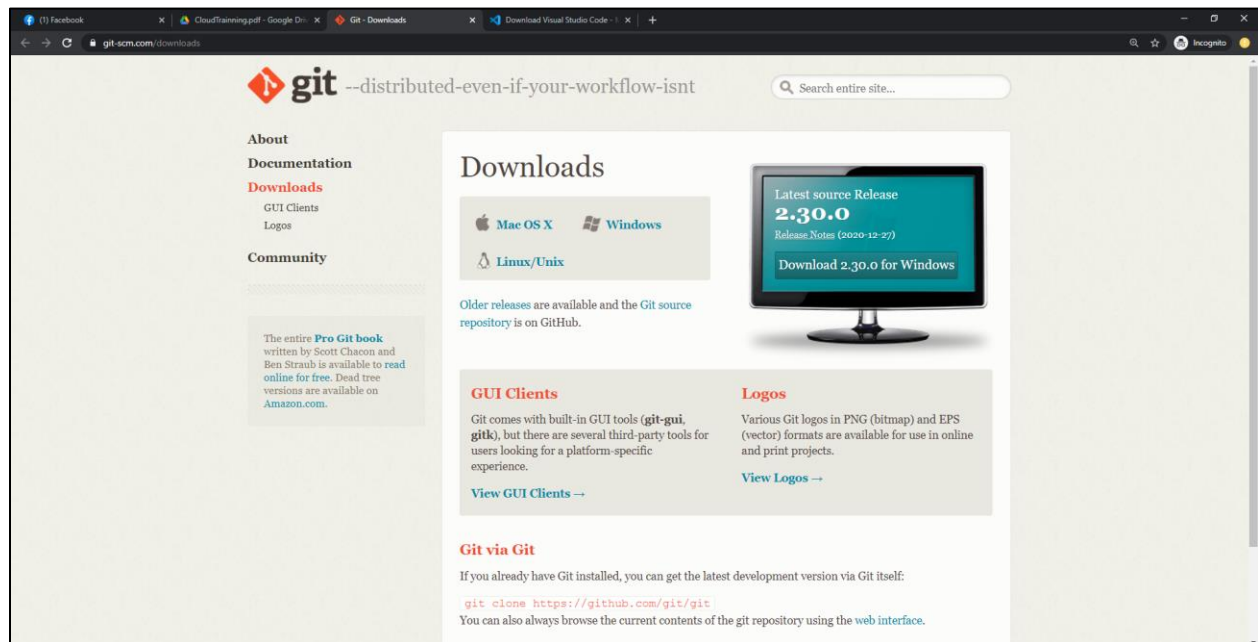


# Set Github on Visual Studio Code



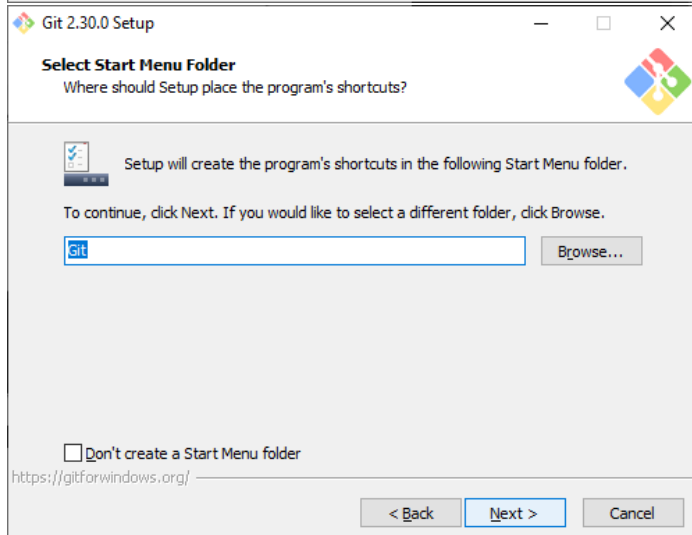
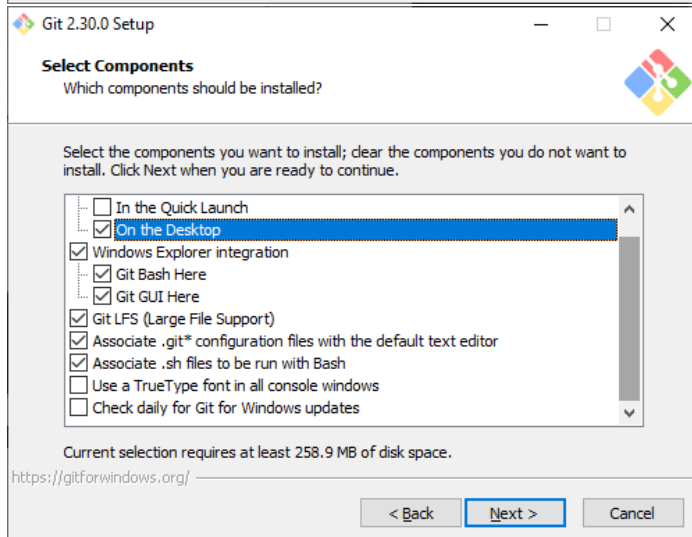
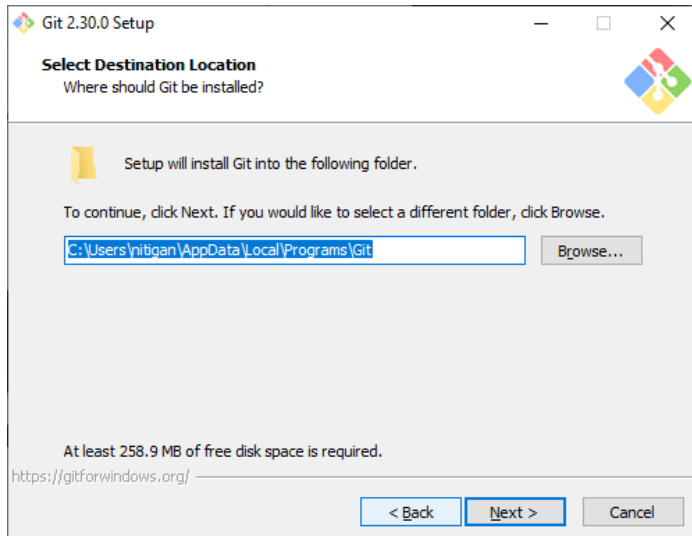
<https://code.visualstudio.com/download>

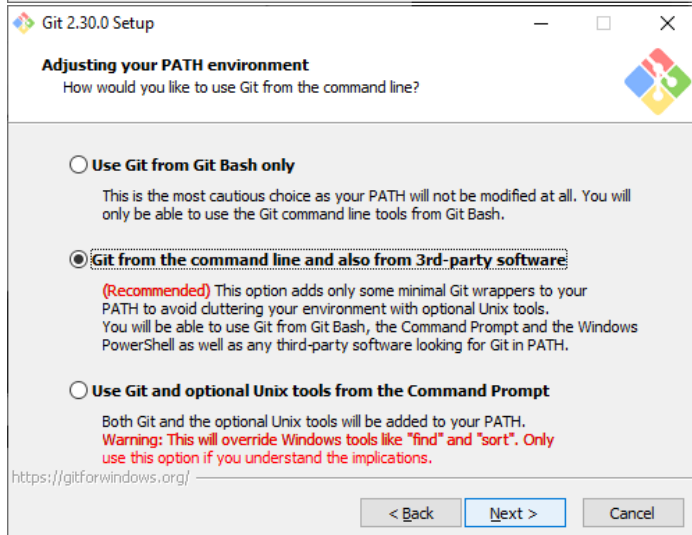
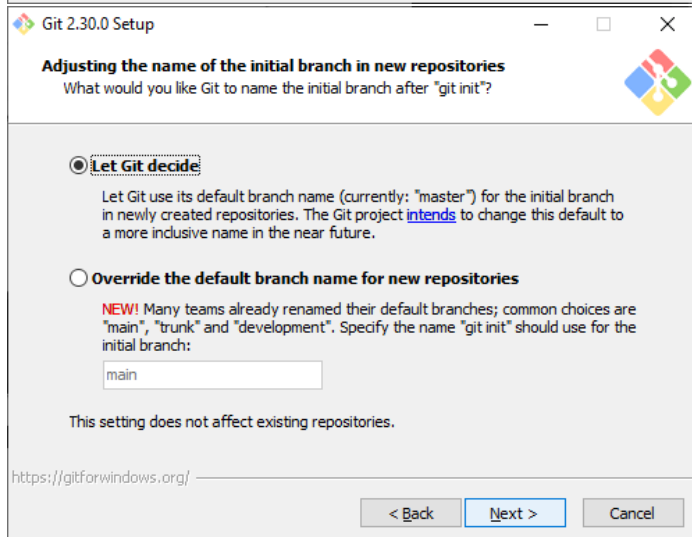
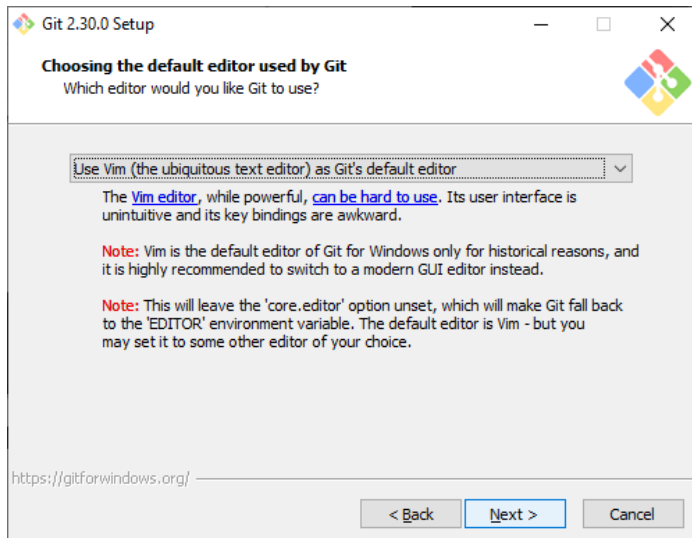
# Download and Install Git



<https://git-scm.com/downloads>







Git 2.30.0 Setup

**Choosing HTTPS transport backend**  
Which SSL/TLS library would you like Git to use for HTTPS connections?

☒ **Use the OpenSSL library**  
Server certificates will be validated using the ca-bundle.crt file.

☐ **Use the native Windows Secure Channel library**  
Server certificates will be validated using Windows Certificate Stores. This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.

<https://gitforwindows.org/>

< Back Next > Cancel

Git 2.30.0 Setup

**Configuring the line ending conversions**  
How should Git treat line endings in text files?

☒ **Checkout Windows-style, commit Unix-style line endings**  
Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").

☐ **Checkout as-is, commit Unix-style line endings**  
Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").

☐ **Checkout as-is, commit as-is**  
Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").

<https://gitforwindows.org/>

< Back Next > Cancel

Git 2.30.0 Setup

**Configuring the terminal emulator to use with Git Bash**  
Which terminal emulator do you want to use with your Git Bash?

☒ **Use MinTTY (the default terminal of MSYS2)**  
Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via 'winpty' to work in MinTTY.

☐ **Use Windows' default console window**  
Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

<https://gitforwindows.org/>

< Back Next > Cancel

Git 2.30.0 Setup

**Choose the default behavior of `git pull`**  
What should `git pull` do by default?

☒ **Default (fast-forward or merge)**  
This is the standard behavior of `git pull`: fast-forward the current branch to the fetched branch when possible, otherwise create a merge commit.

☐ **Rebase**  
Rebase the current branch onto the fetched branch. If there are no local commits to rebase, this is equivalent to a fast-forward.

☐ **Only ever fast-forward**  
Fast-forward to the fetched branch. Fail if that is not possible.

<https://gitforwindows.org/>

< Back Next > Cancel

Git 2.30.0 Setup

**Choose a credential helper**  
Which credential helper should be configured?

☒ **Git Credential Manager Core**  
(NEW!) Use the new, [cross-platform version of the Git Credential Manager](#). See more information about the future of Git Credential Manager [here](#).

☐ **Git Credential Manager**  
(DEPRECATED) The [Git Credential Manager for Windows](#) handles credentials e.g. for Azure DevOps and GitHub (requires .NET framework v4.5.1 or later).

☐ **None**  
Do not use a credential helper.

<https://gitforwindows.org/>

< Back Next > Cancel

Git 2.30.0 Setup

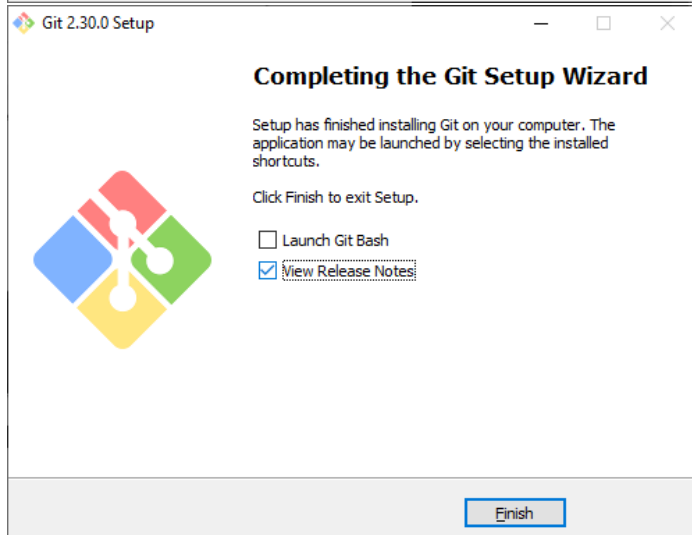
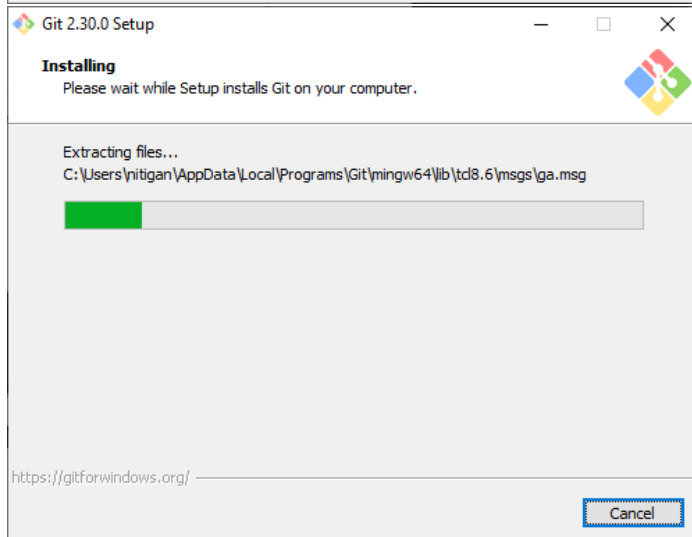
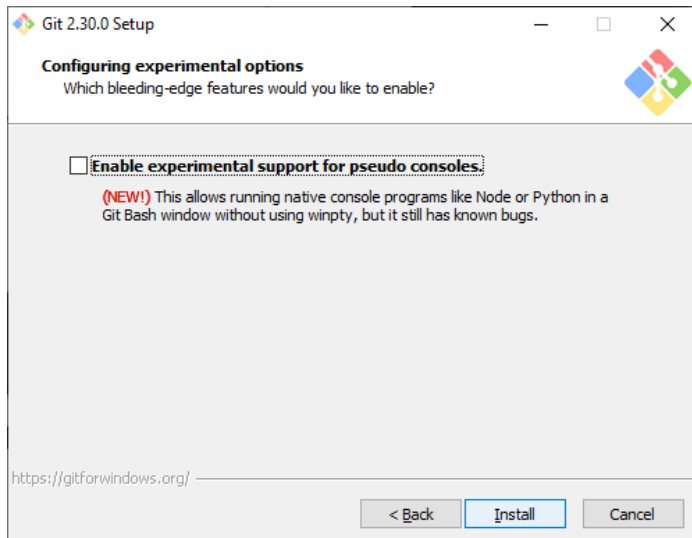
**Configuring extra options**  
Which features would you like to enable?

☒ **Enable file system caching**  
File system data will be read in bulk and cached in memory for certain operations ("core.fscache" is set to "true"). This provides a significant performance boost.

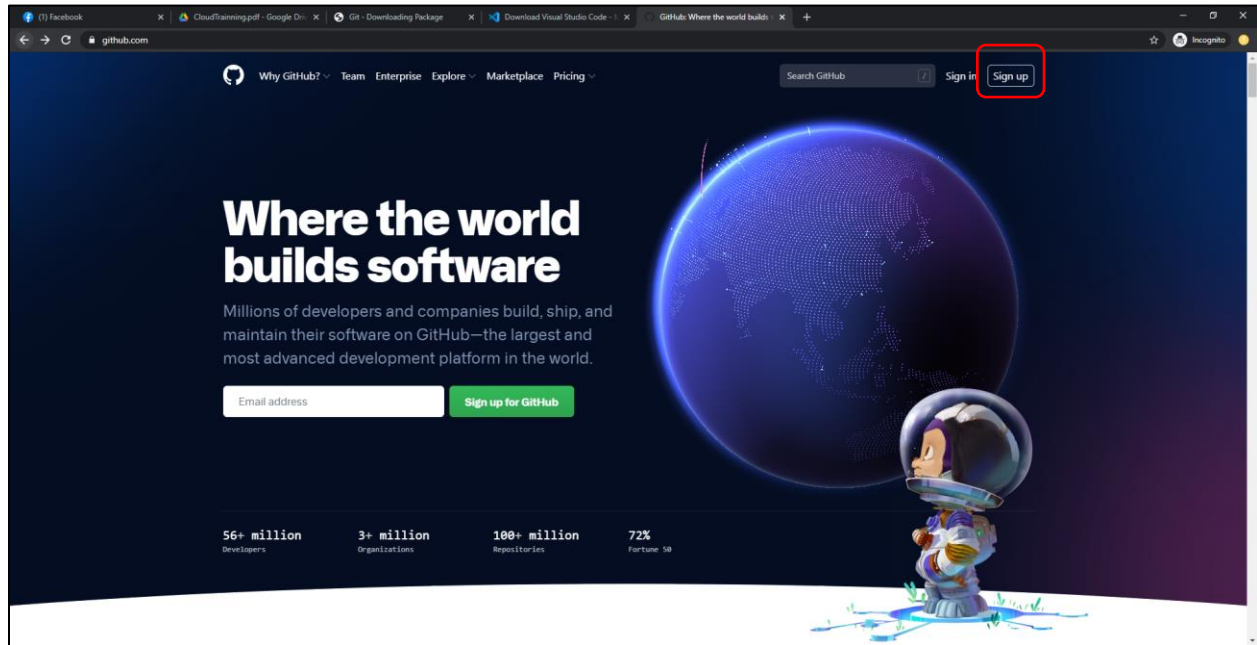
☐ **Enable symbolic links**  
Enable [symbolic links](#) (requires the SeCreateSymbolicLink permission). Please note that existing repositories are unaffected by this setting.

<https://gitforwindows.org/>

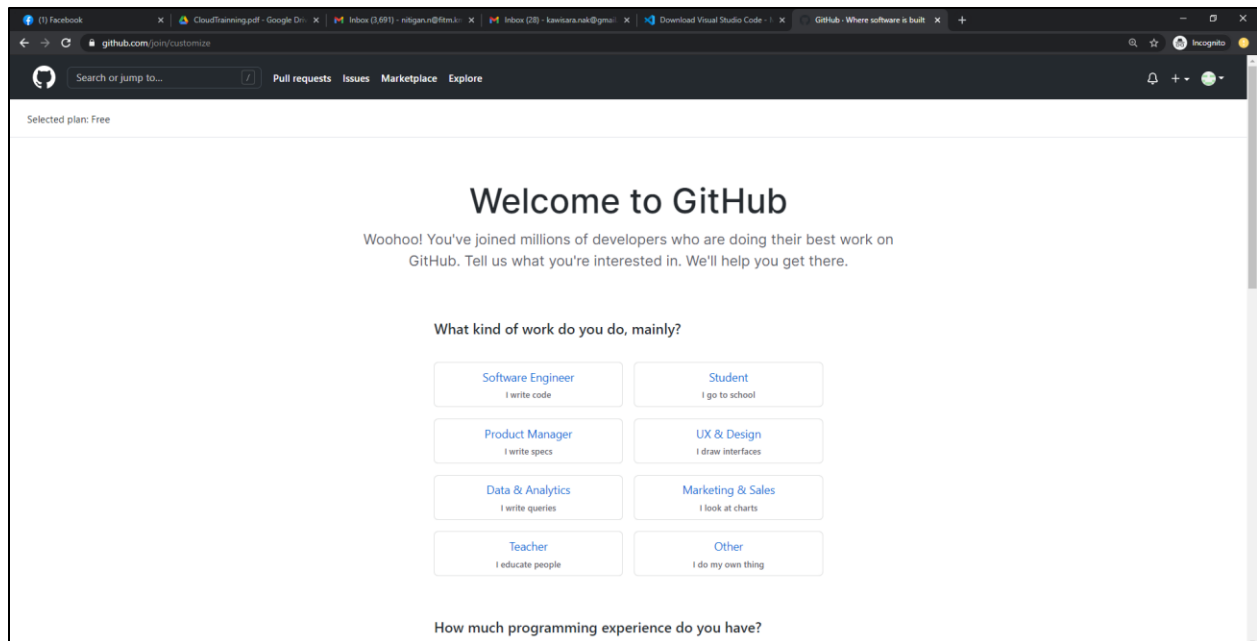
< Back Next > Cancel



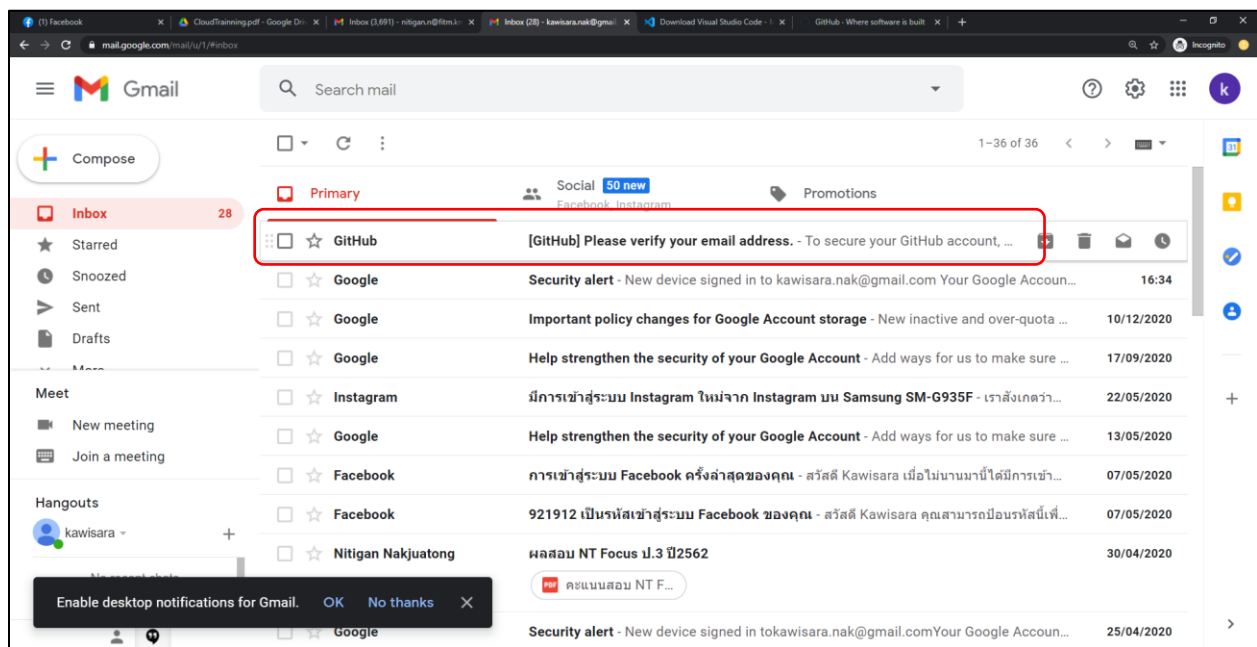
Signup Github with **your-email**@fitm.kmutnb.ac.th

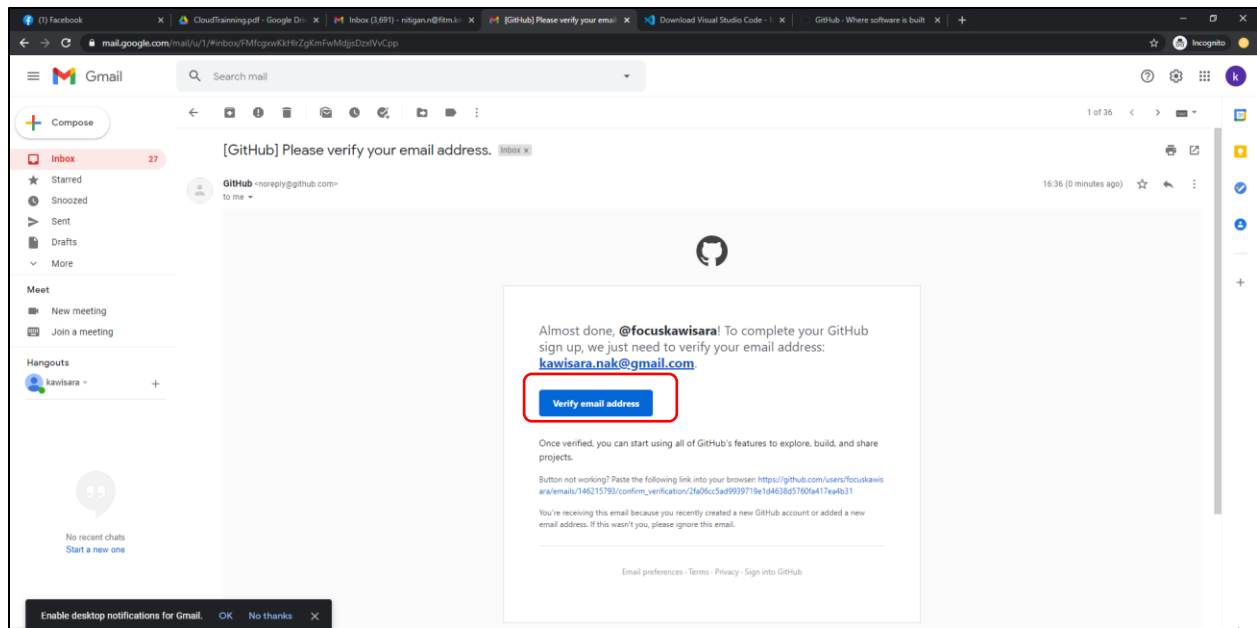
A screenshot of the GitHub account creation page. The title is 'Create your account'. The form includes fields for 'Username \*' (filled with 'focuskawi/sara'), 'Email address \*' (filled with 'kawisara.nak@gmail.com'), and 'Password \*' (filled with '\*\*\*\*\*'). Each field has a green checkmark icon. Below the password field is a note: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. Learn more.' There is an 'Email preferences' section with a checkbox 'Send me occasional product updates, announcements, and offers.' and a 'Verify your account' section with a large green checkmark icon. At the bottom, there is a blue 'Create account' button. The entire sign-up form area and the 'Create account' button are highlighted with red rectangles. Below the button, there is a disclaimer: 'By creating an account, you agree to the Terms of Service. For more information about GitHub's privacy practices, see the GitHub Privacy Statement. We'll occasionally send you account-related emails.'



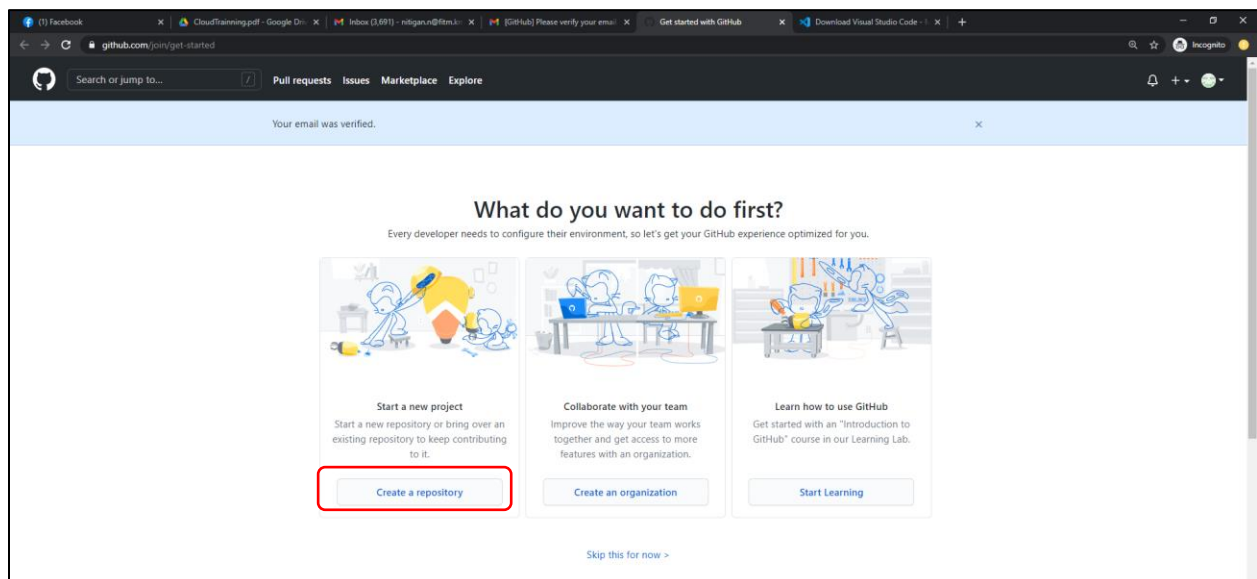


## Verify your mail





Click Create a repository



# Create a new repository

github.com/new

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner \* focuskawisara

Repository name \* testweb ✓

Great repository names are short and memorable. Need inspiration? How about [vigilant-octo-disco?](#)

Description (optional) testweb

☒ 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:  
[Skip this step if you're importing an existing repository.](#)

☒ Add a README file  
This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore  
Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license  
A license tells others what they can and can't do with your code. [Learn more.](#)


This will set `main` as the default branch. Change the default name in your [settings](#).


Create repository

(1) Facebook x CloudTrain x Inbox (3,6 x [GitHub] P x focuskawi x Download x +

github.com/focuskawisara/testweb

Incognito


 Search or jump to... / Pulls Issues Marketplace Explore



# Learn Git and GitHub without any code!

Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

Read the guide

 focuskawisara / testweb

Unwatch 1

Star 0

Fork 0

<> Code

! Issues

🔗 Pull requests

▶ Actions

📁 Projects

📖 Wiki

🔒 Security

📊 Insights


...

main

Go to file


Add file

Code

 focuskawisara Initial commit now 1

README.md Initial commit now

README.md



# testweb

testweb

About

testweb

Readme

Releases

No releases published

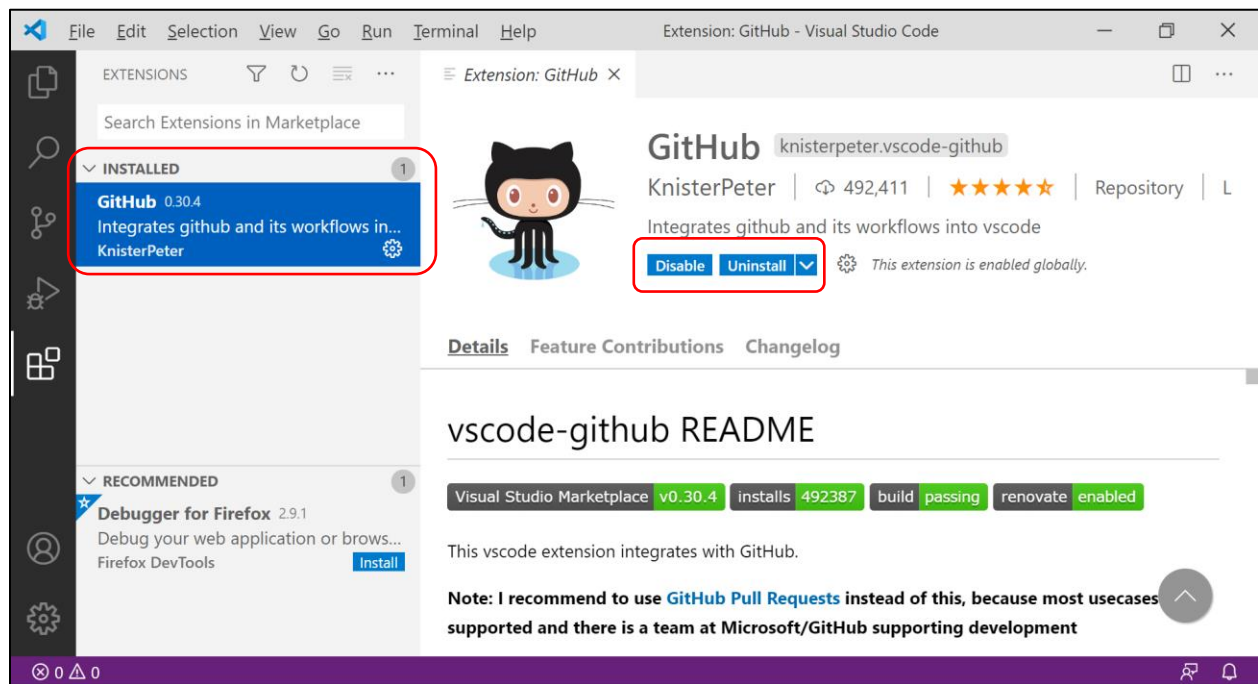
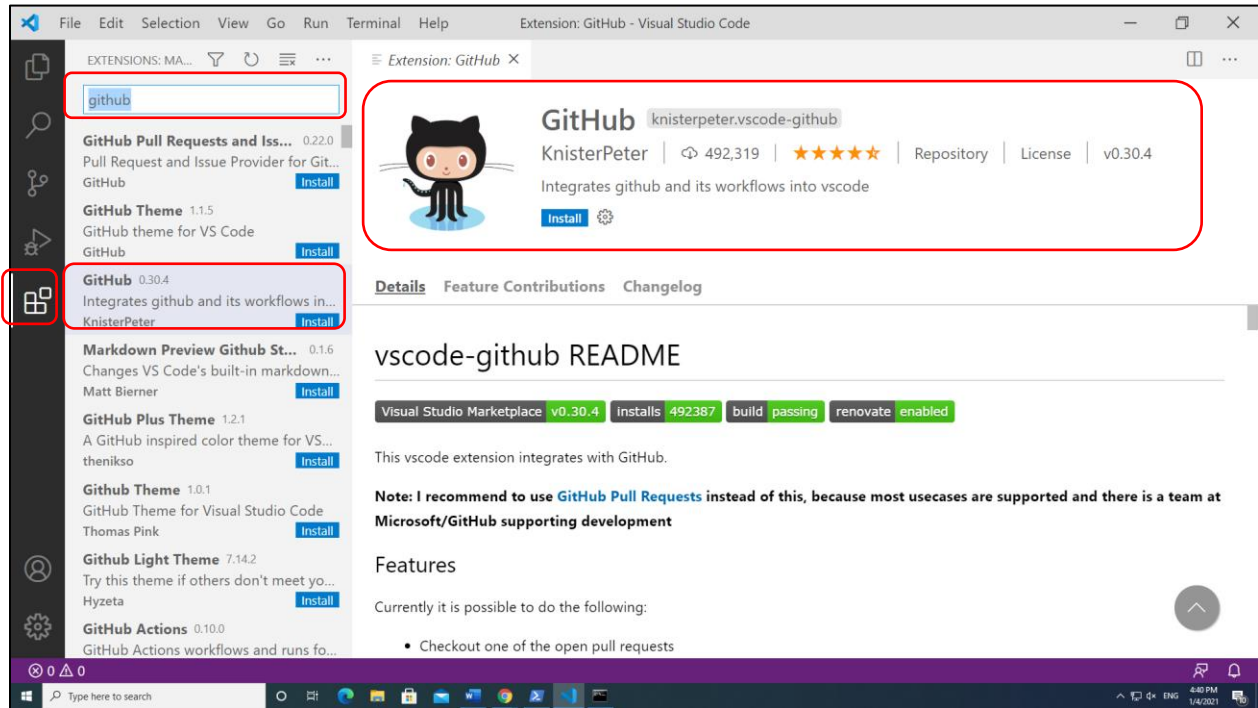
Create a new release

Packages

No packages published

Publish your first package

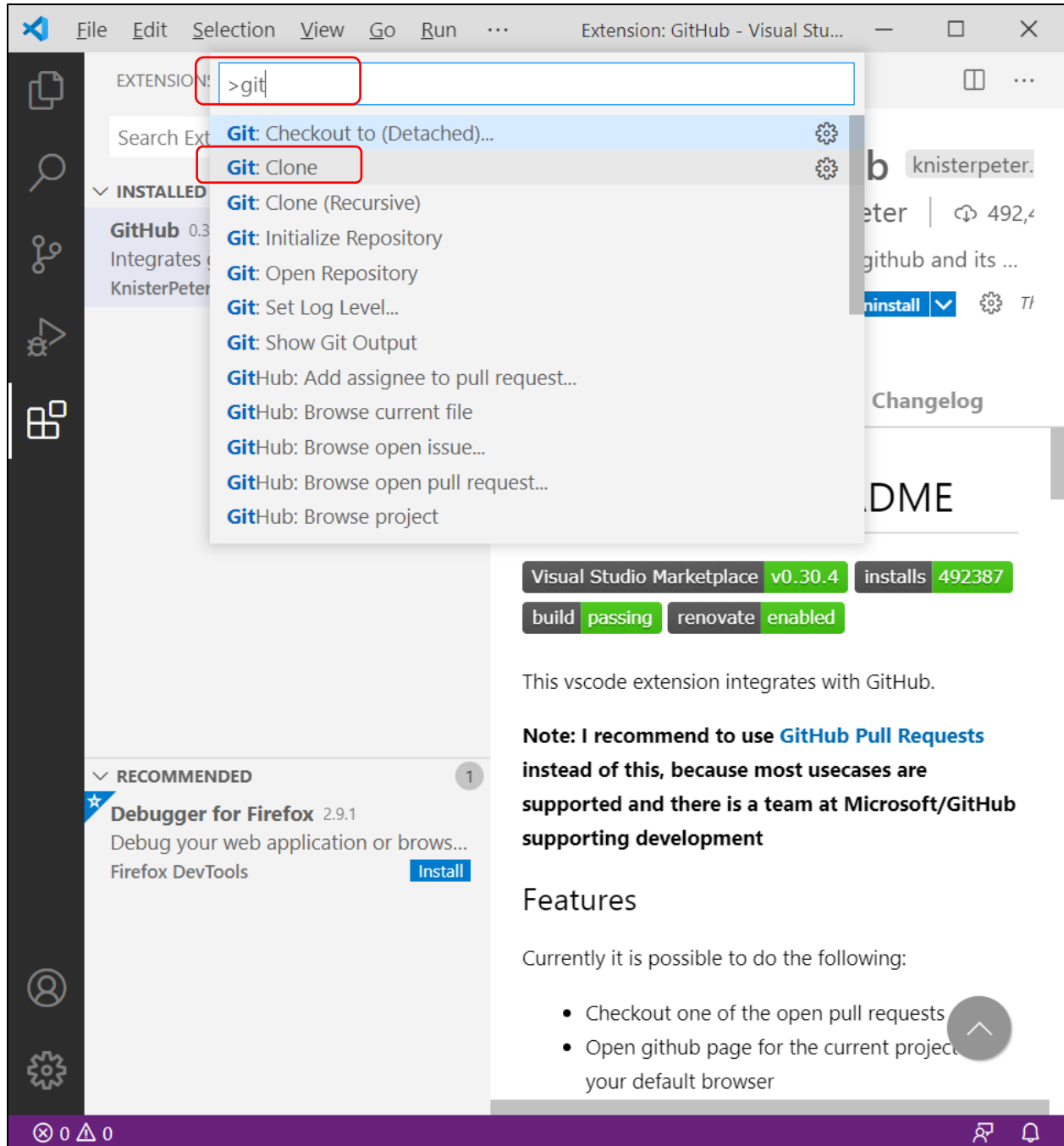
## Set Github with Visual Studio Code



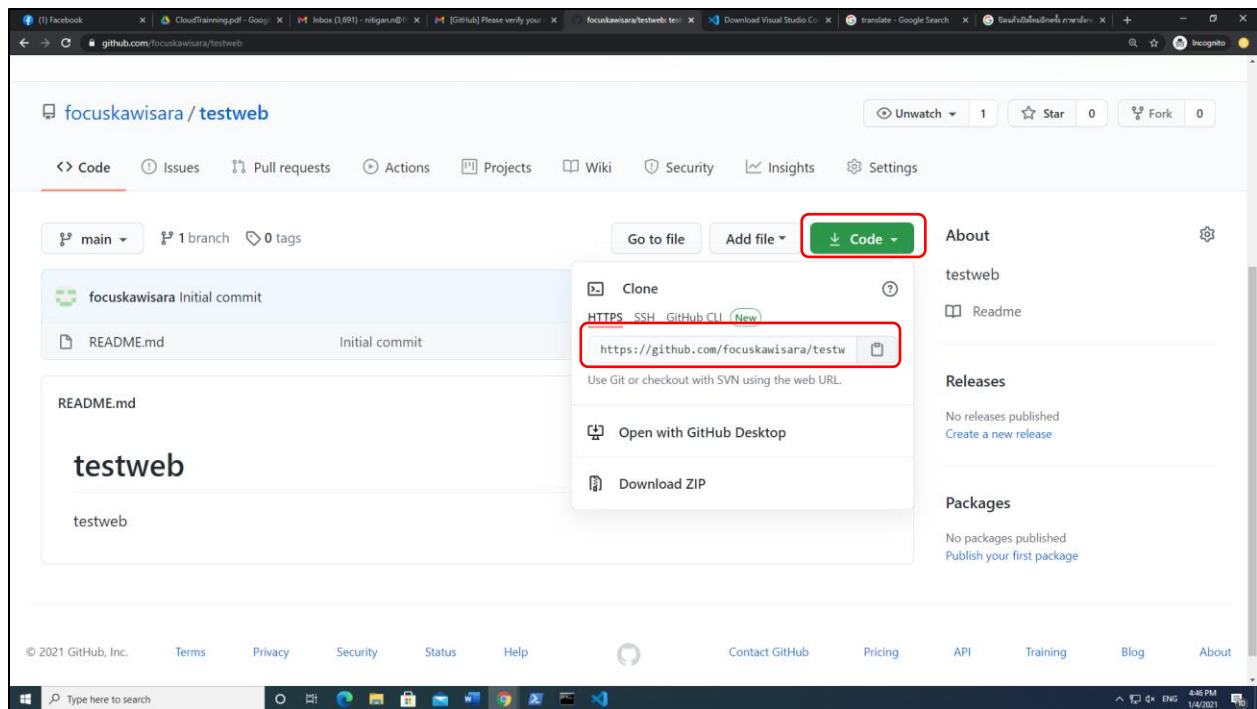
After installed pls Turn it off Visual Studio Code and Open again.

➤ Ctrl+Shift+p

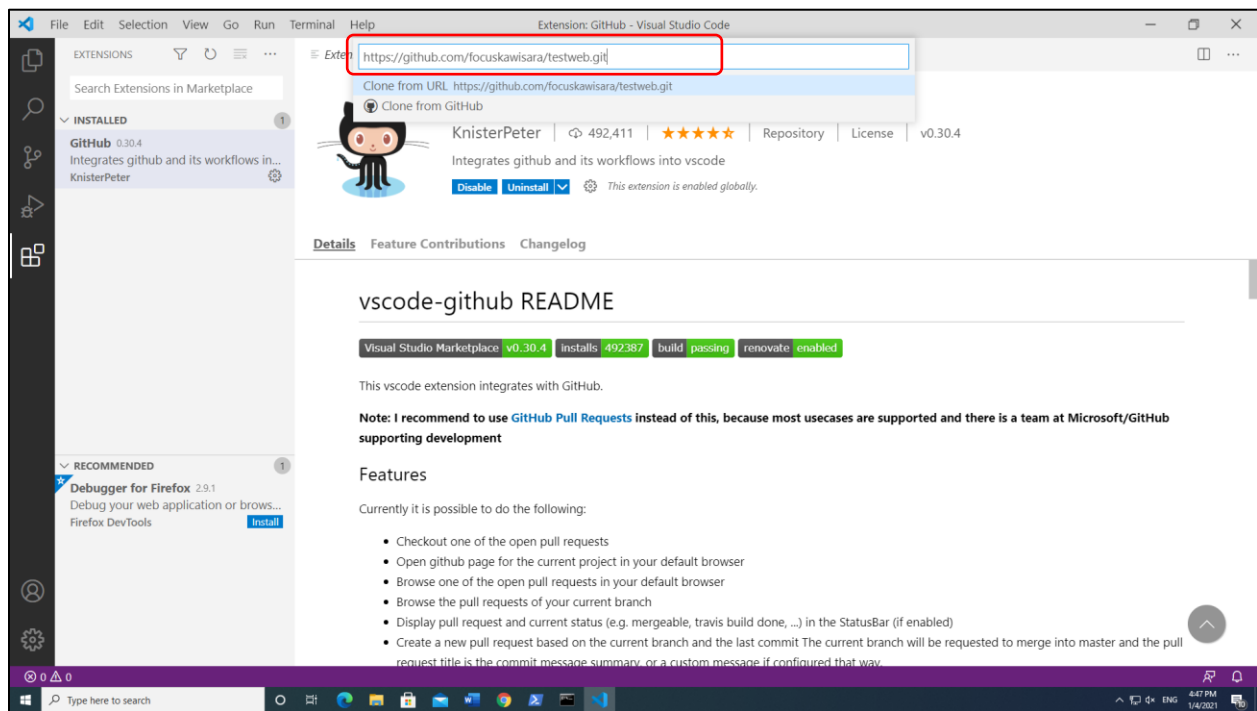
git clone command



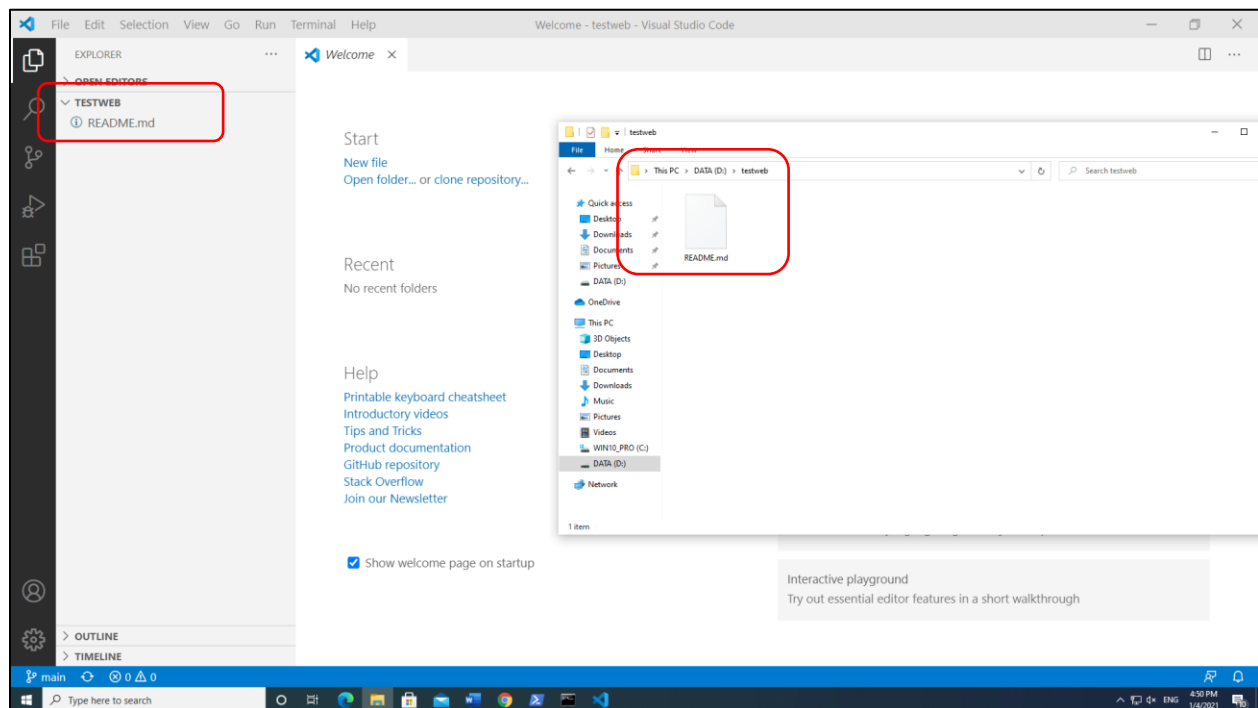
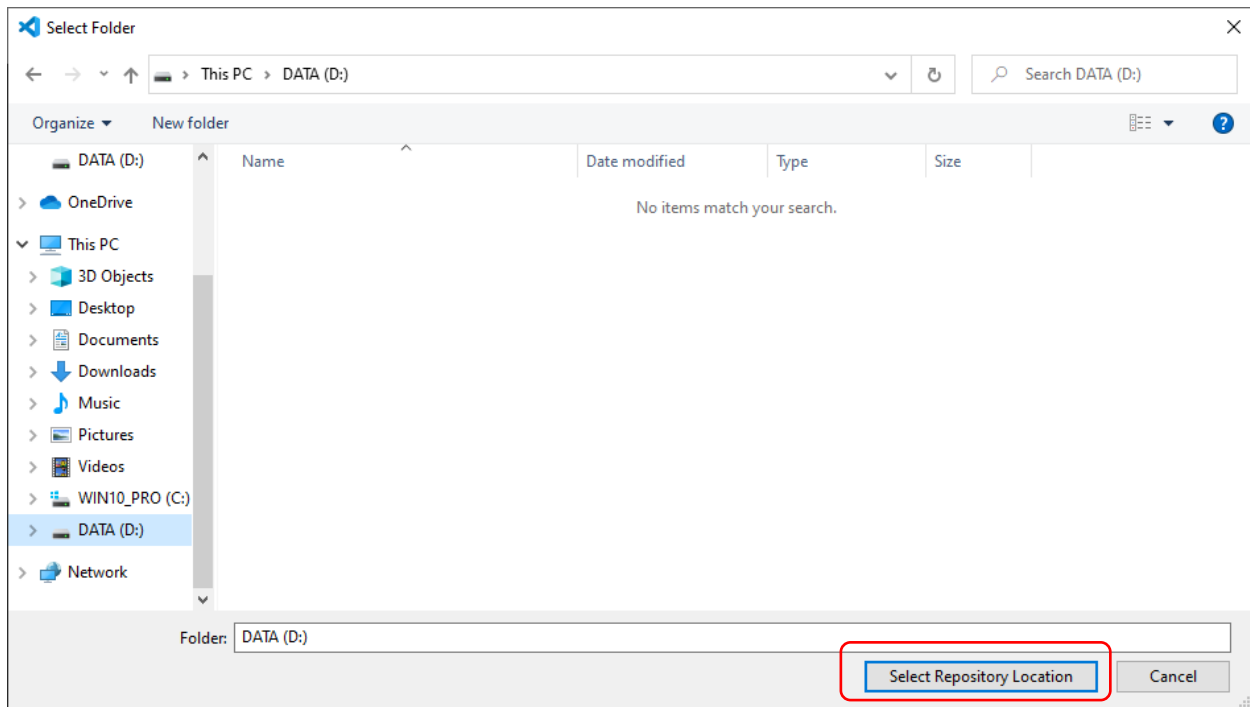
## Copy HTTPS Repository



## Paste url in git clone command

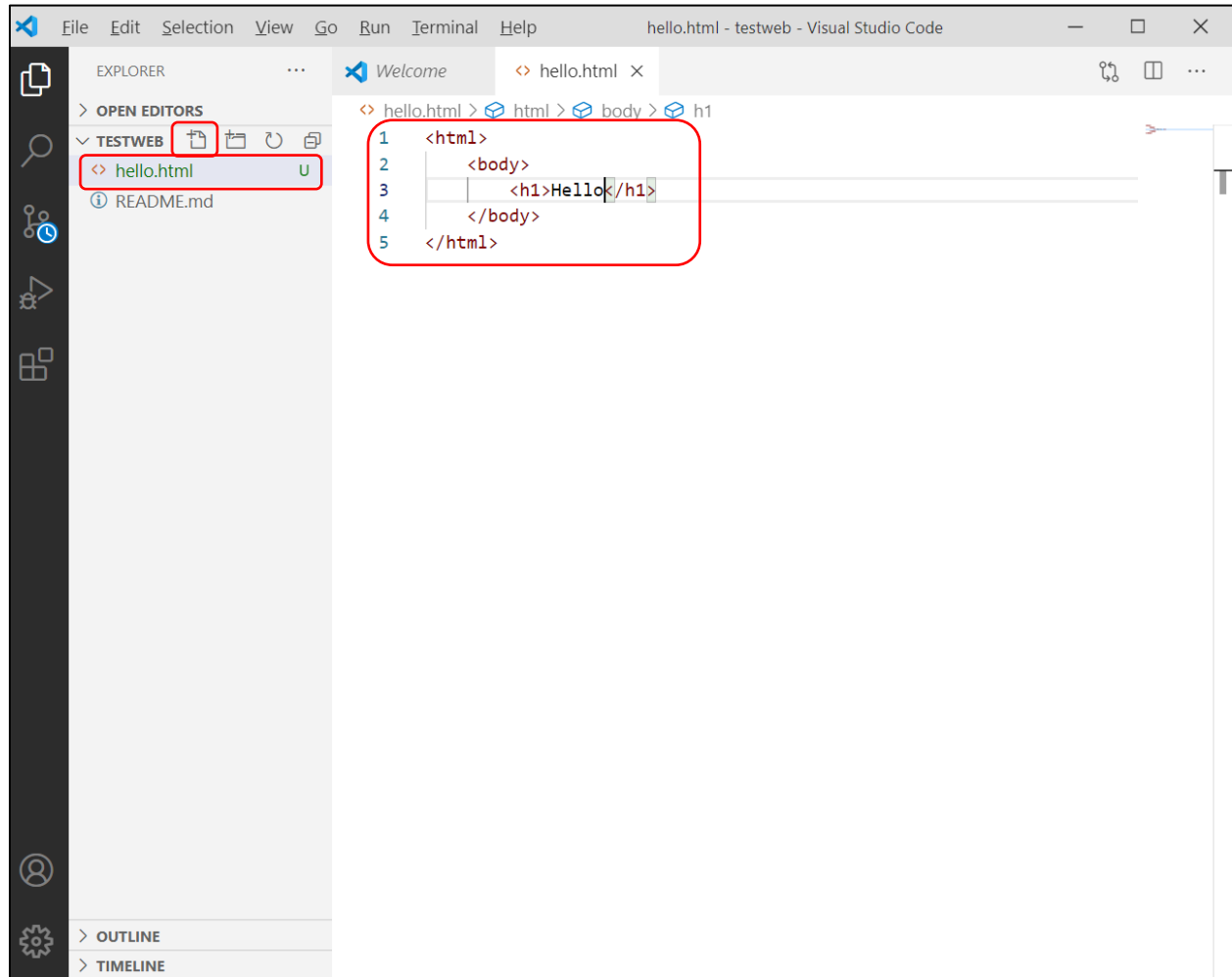


## Select location on your PC

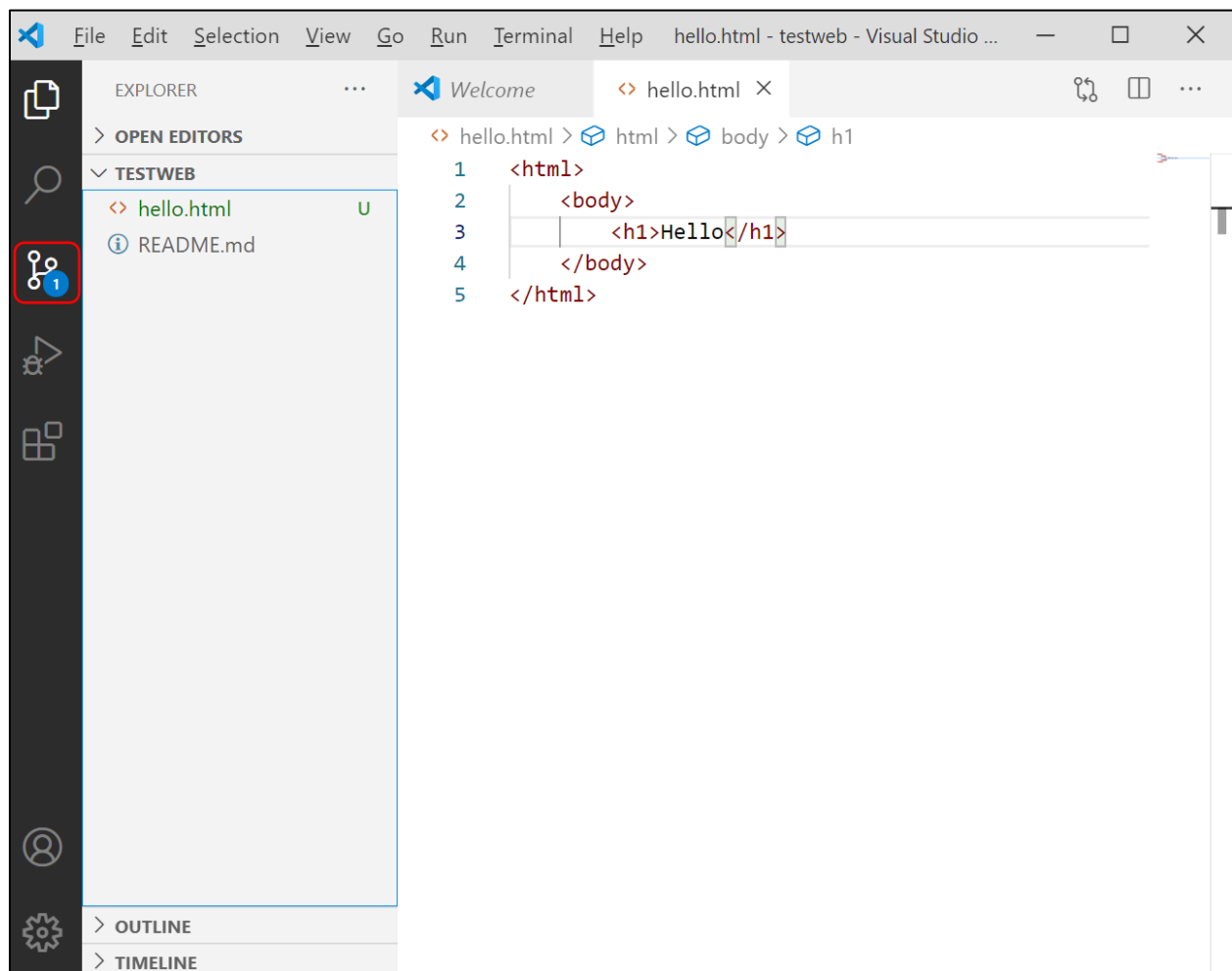




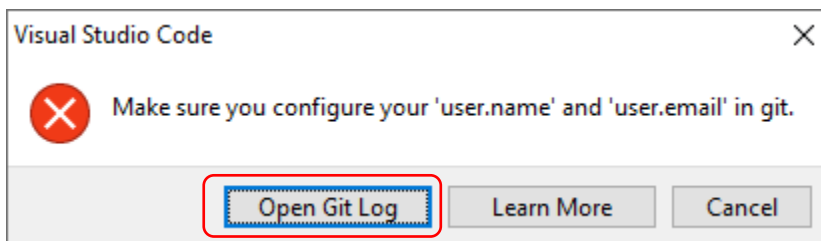
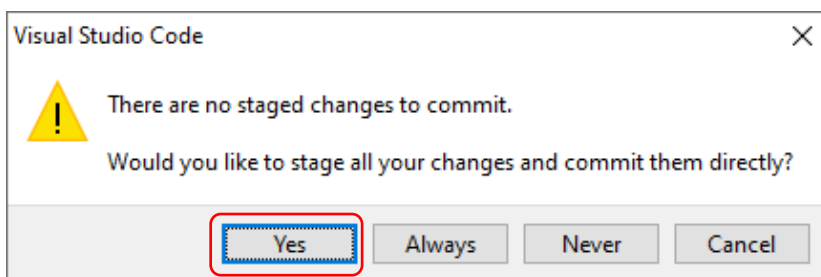
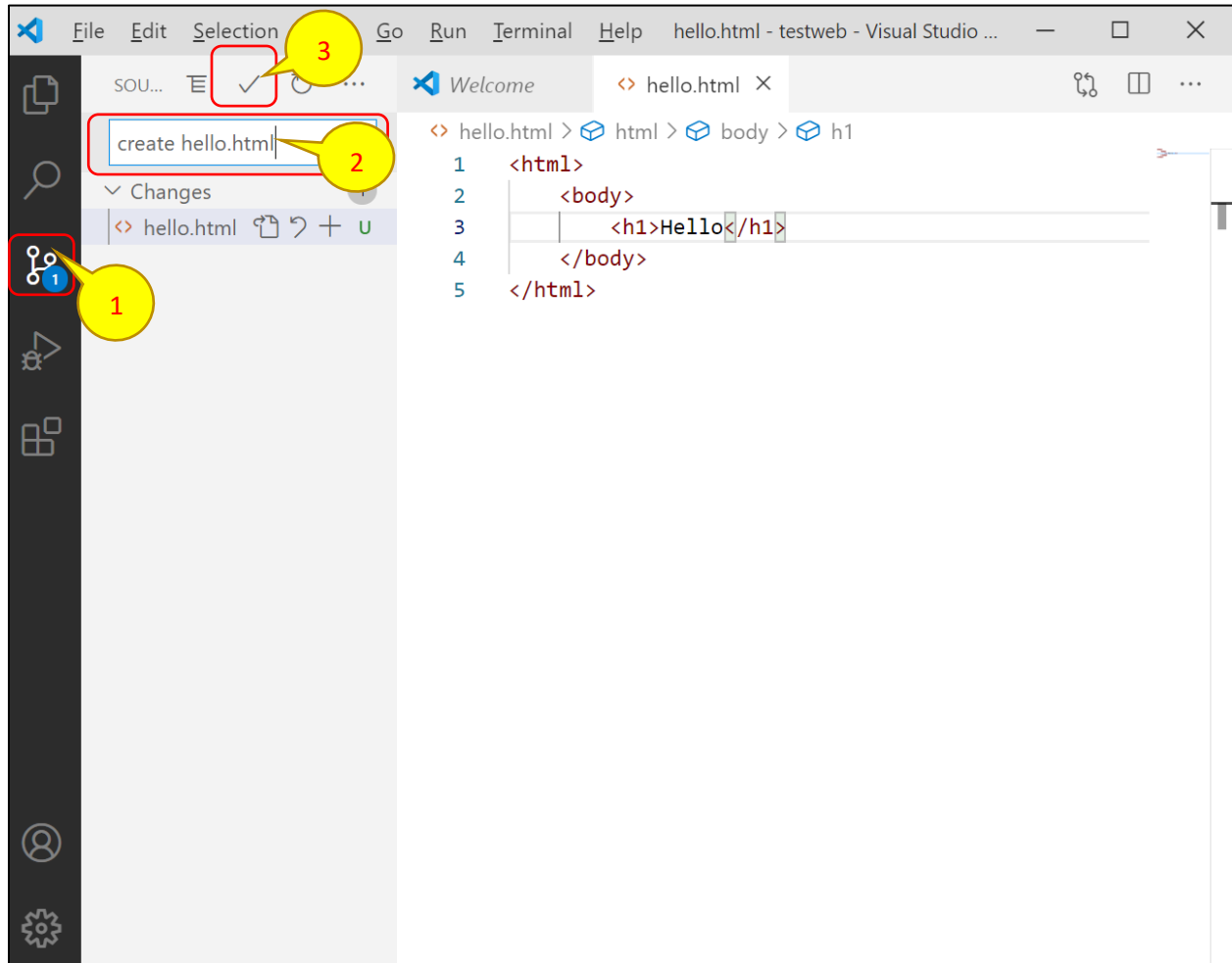
## Create New file



## Save file

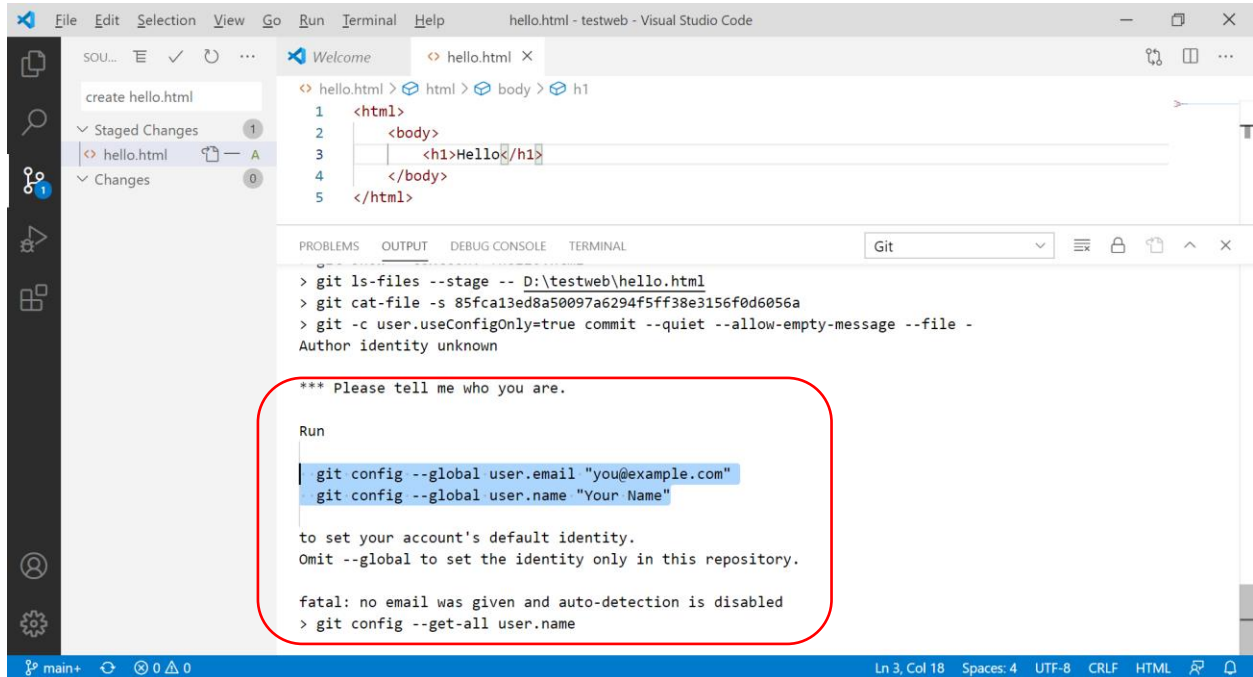


## Commit

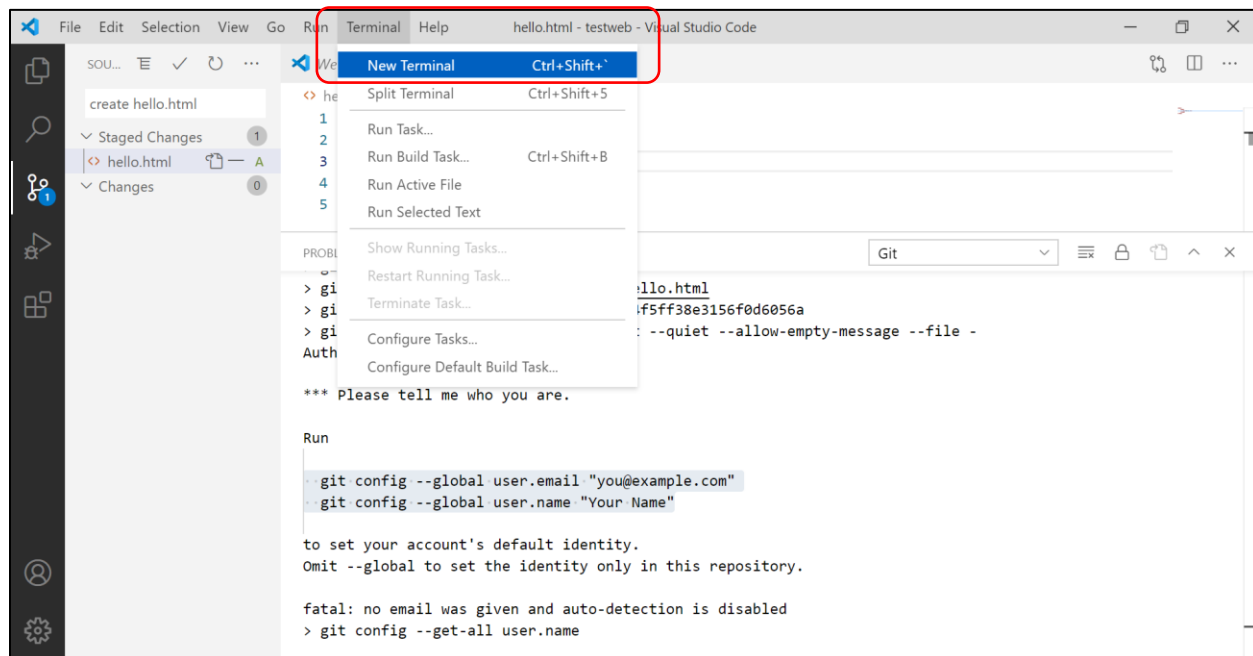


## Run Git Config with Terminal

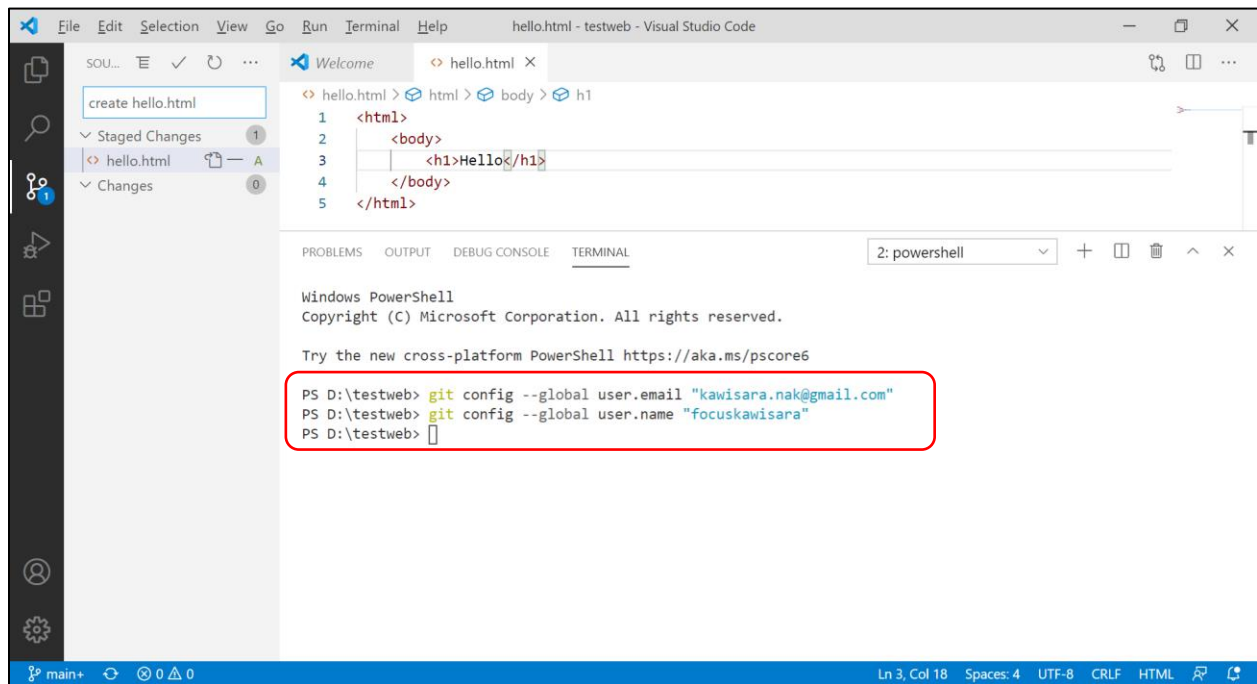
```
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
```



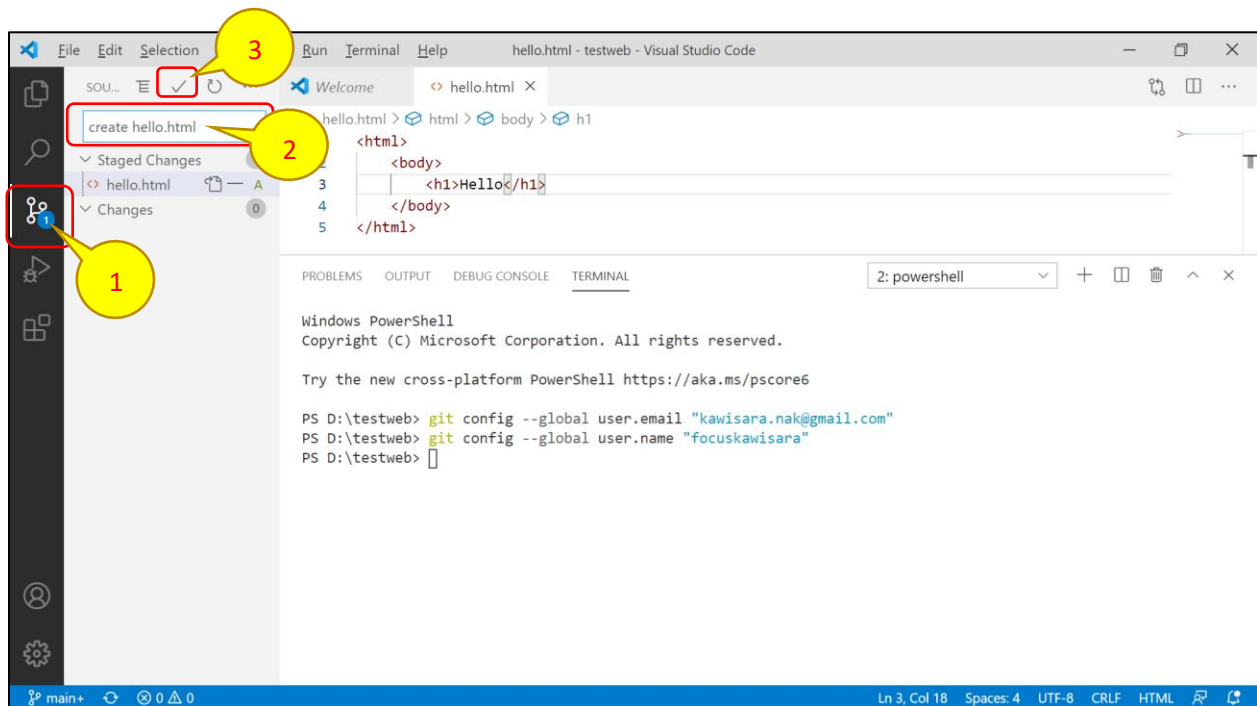
## New Terminal



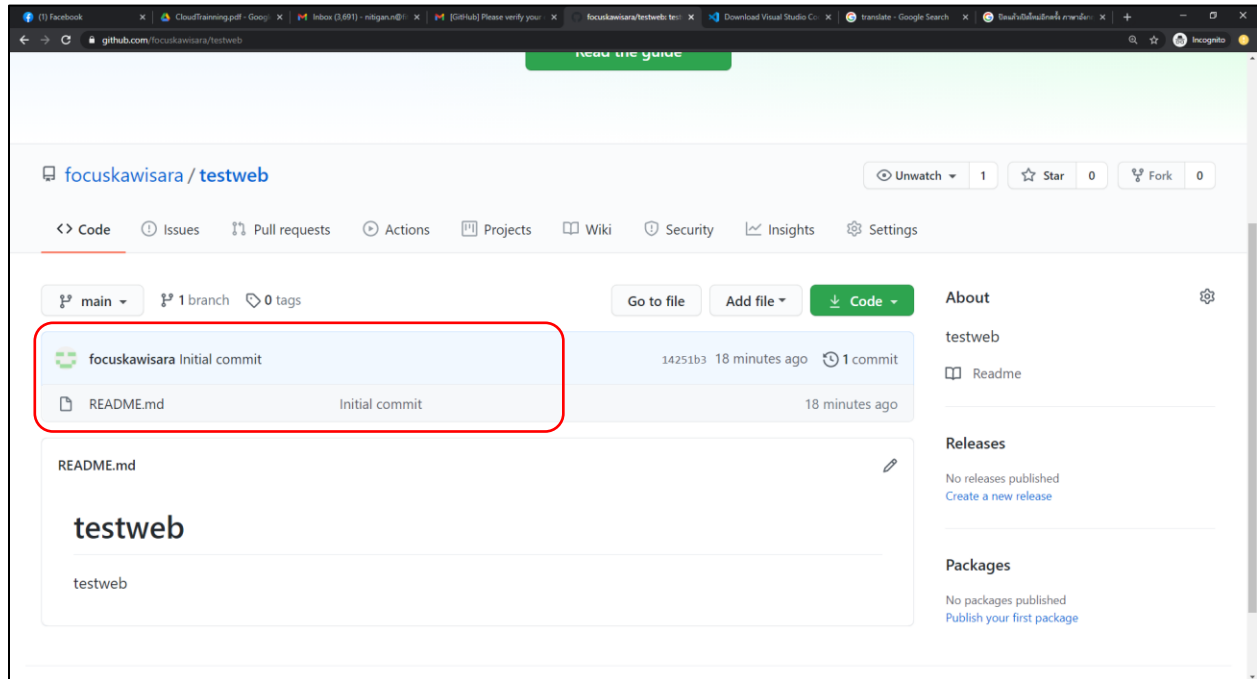
## Config and run git command



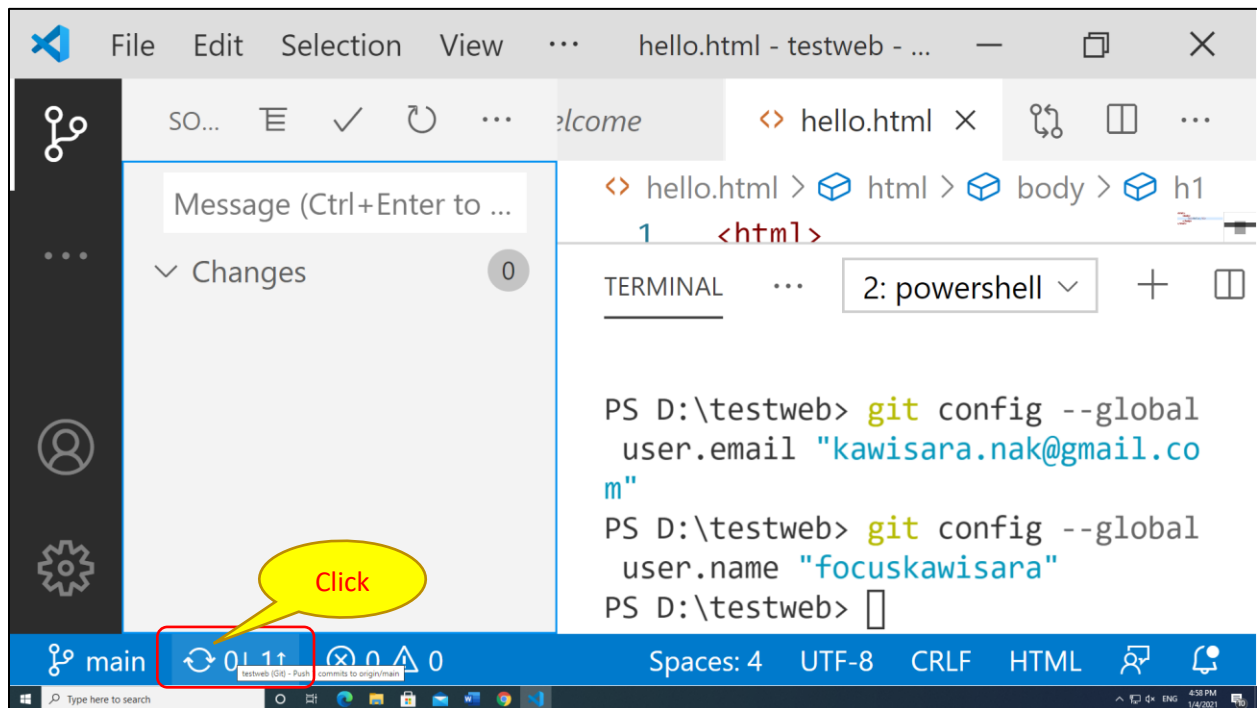
## Commit again

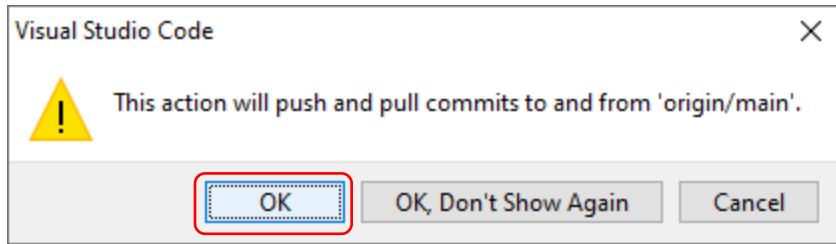


## Github don't update hello.html

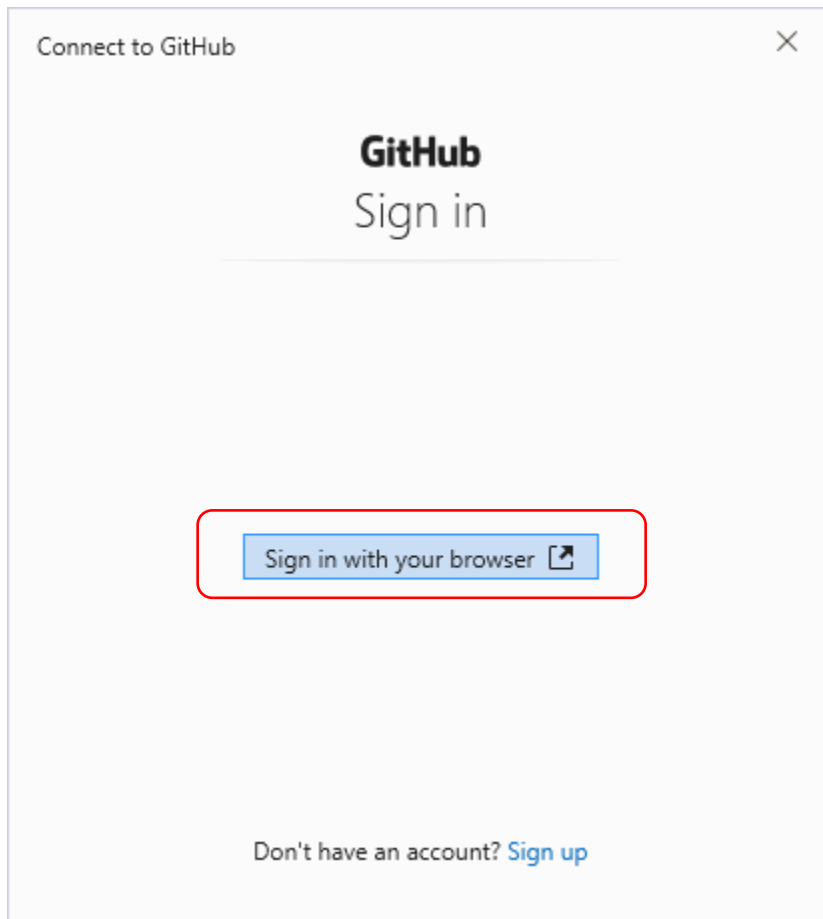


## Push to Github






## Sign in Github




Sign in to GitHub · GitHub


https://github.com/login?client\_id=0120e057bd645470c1e...

 To be most productive with Microsoft Edge, finish setting up your browser.

Complete setup

Maybe later






Sign in to GitHub

to continue to Git Credential Manager

Username or email address

Password

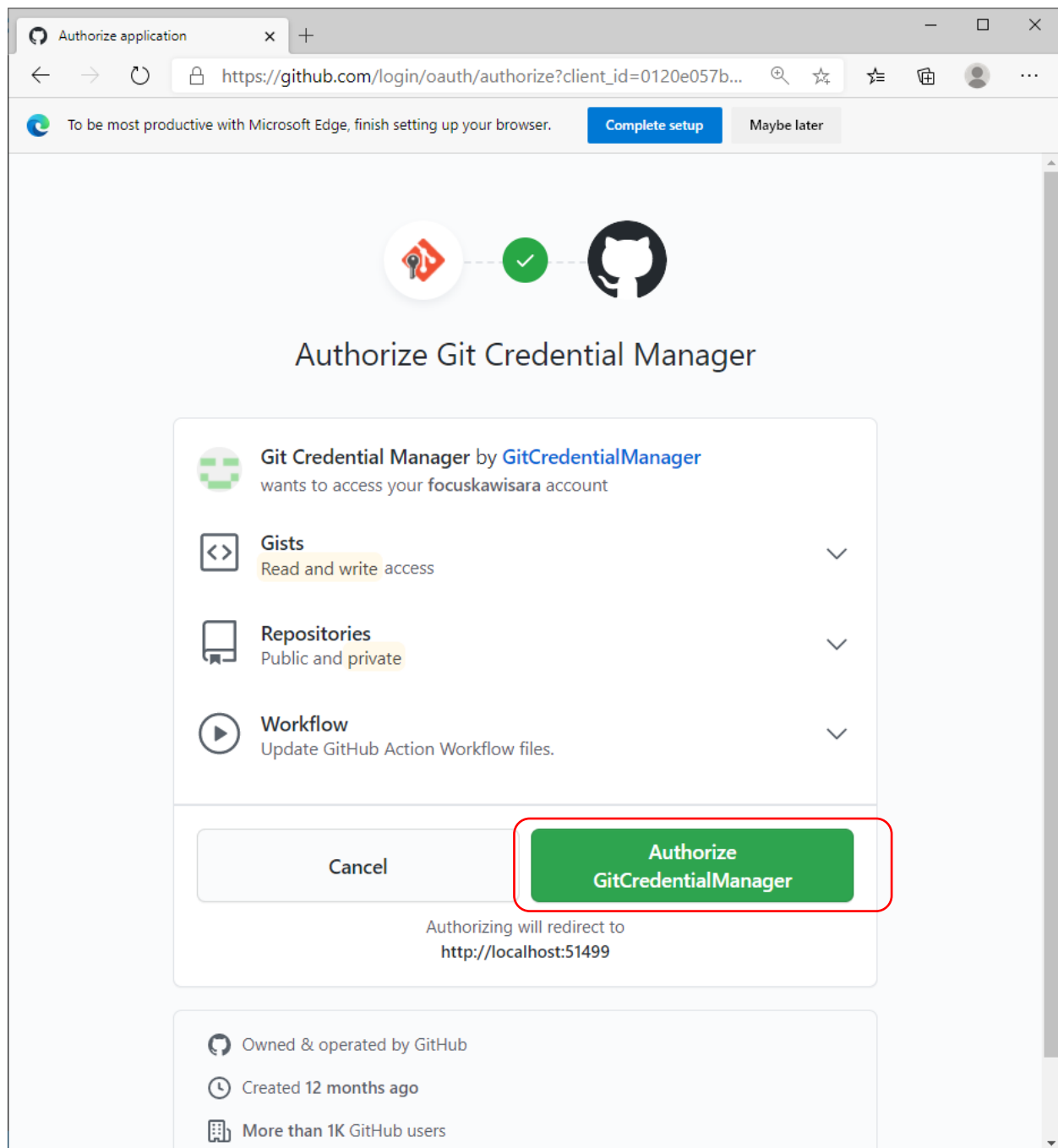
Forgot password?

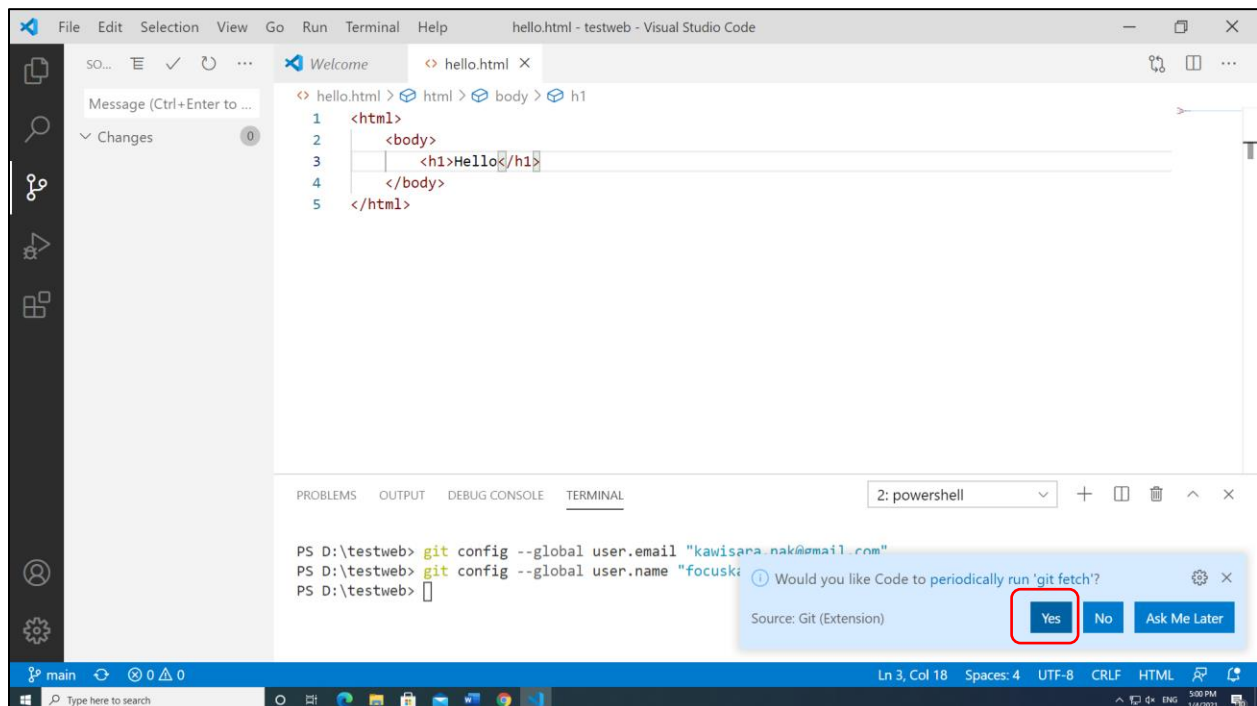
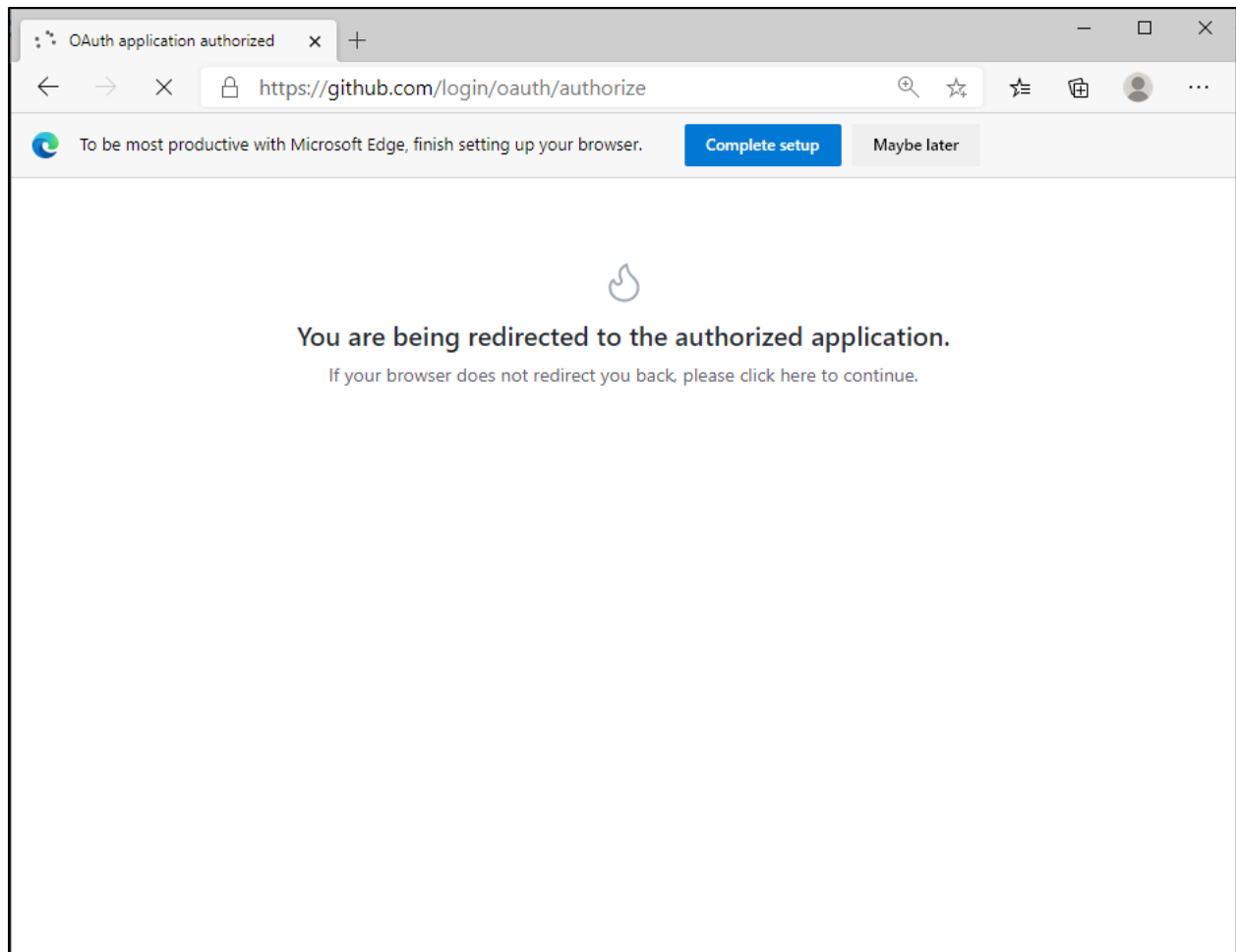


Sign in

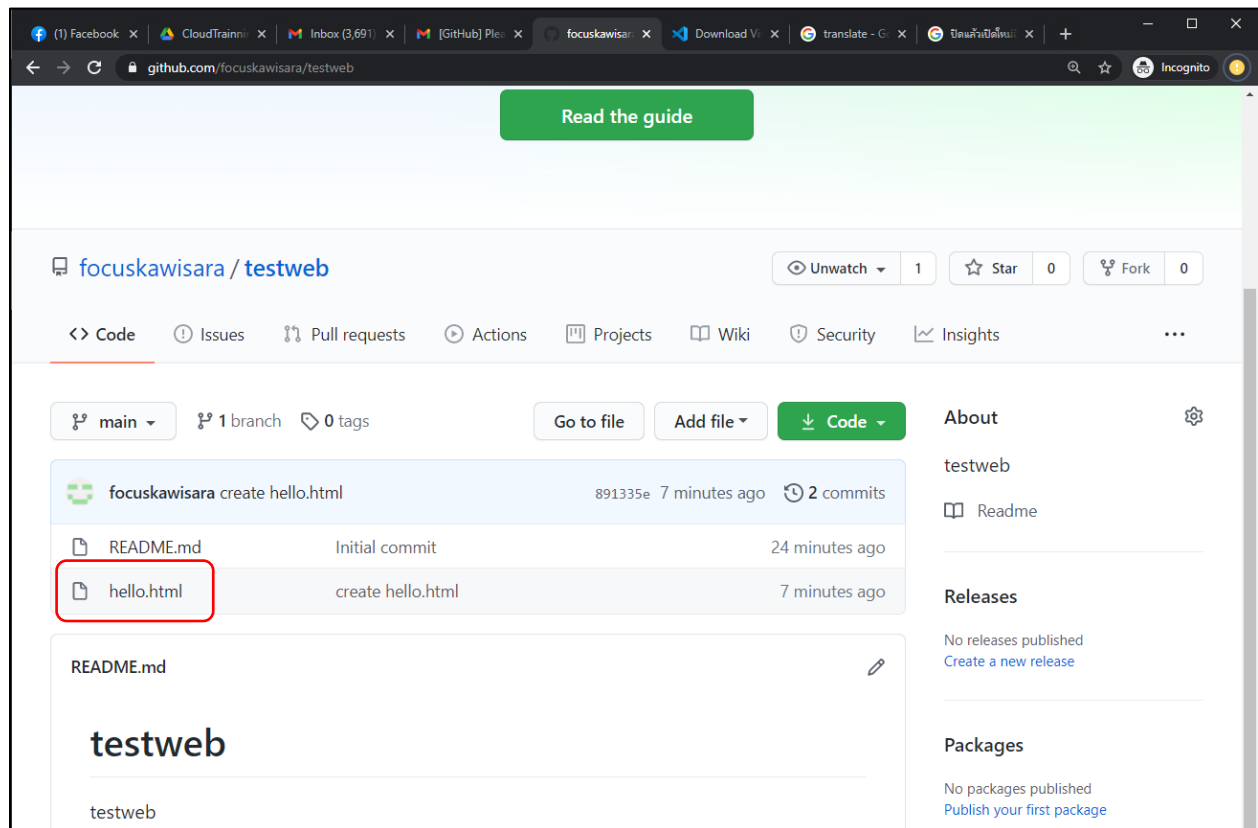
New to GitHub? [Create an account.](#)



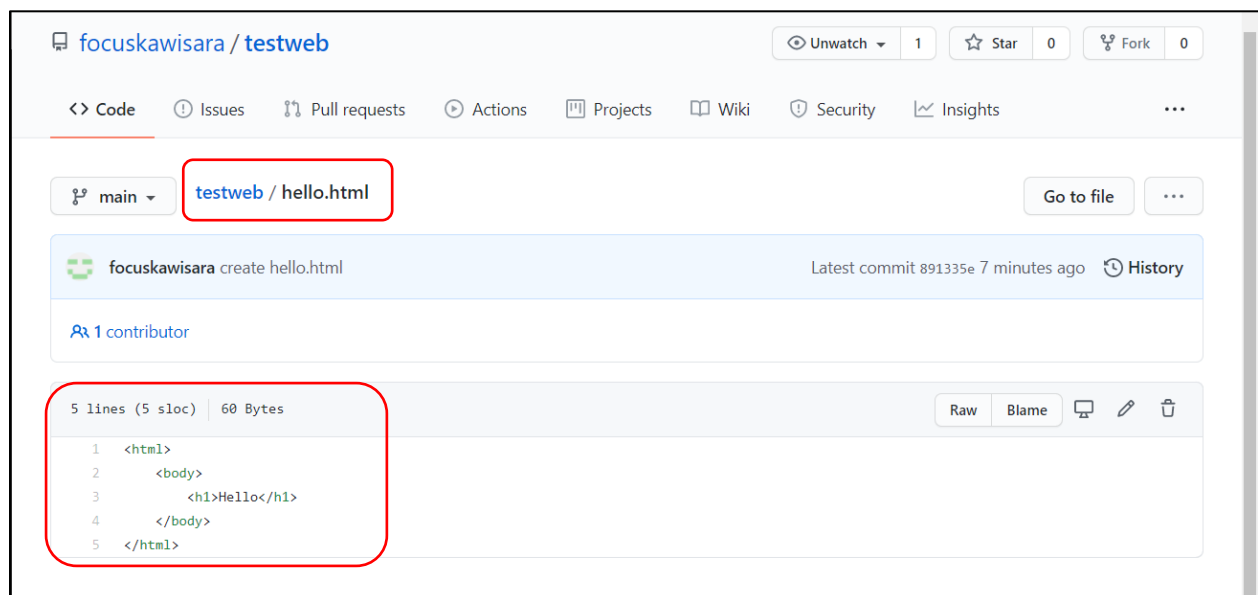




## Hello.html in repository

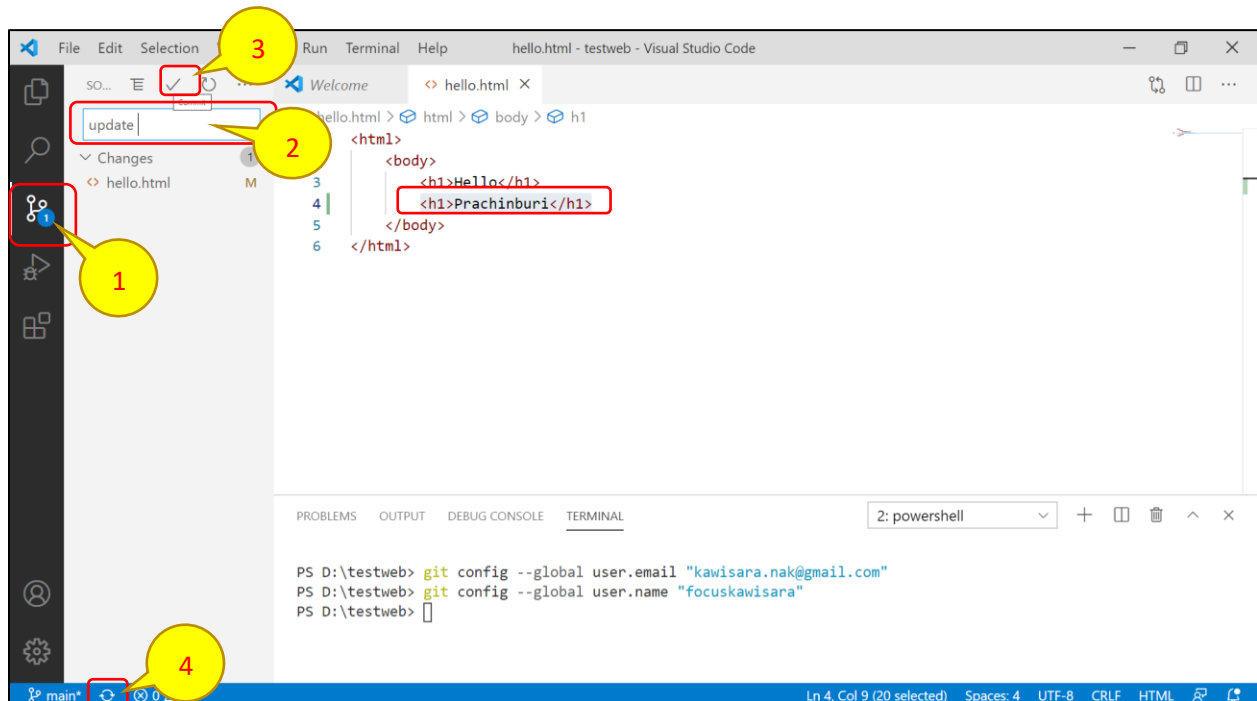


## Click hello.html



## Update code in Visual Studio Code

## Commit and Push to Github



## Check your repository

