Learning Git, GitHub, and GitHub Desktop

Git and GitHub will make your work as a coder much easier, and they'll look good on your resume as well.

What is Git?

Git is a type of software called a version control system, which tracks changes to files. What this means is that it will detect all your code changes, and you can use it to record different versions of your files at different points in time. This is helpful for developers for a number of reasons. For starters, having Git is kind of like having a giant set of save points. If you make a code change that you realize later on was a huge mistake, you can revert back to the older version of that File. Git also makes working on a team a lot easier.

Each developer can work on a copy of the project on their computer. This copy is stored in a folder connected to Git, which is called a local repository. The developers can then send their code changes up to the main remote repository, which is stored online, and get code changes made by other developers.

Take note of the following terms:

TERMINOLOGIES	DESCRIPTION
1. Git	It is the actual version control software. Originally meant to run on the command line, and a lot of developers run Git this way.
2. GitHub	GitHub is a website where you can host your Git repositories, collaborate with other users or groupmates, and discover other open-source repositories.
3. GitHub Desktop	It is a program that runs Git on your computer and allows you to do most of the Git command line commands, just through a more visual program interface.

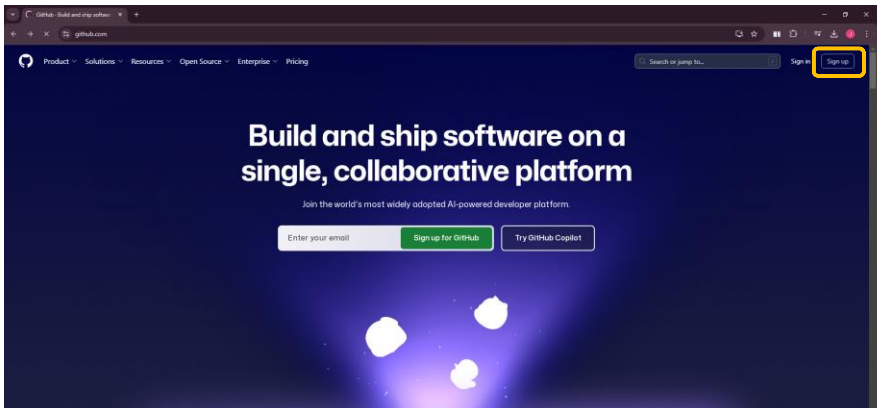
GitHub

It is the most popular place to store your remote repository, which is owned by Microsoft and is free to use. You can create your GitHub account to GitHub for free.

Creating GitHub account

Step-by-step guide:

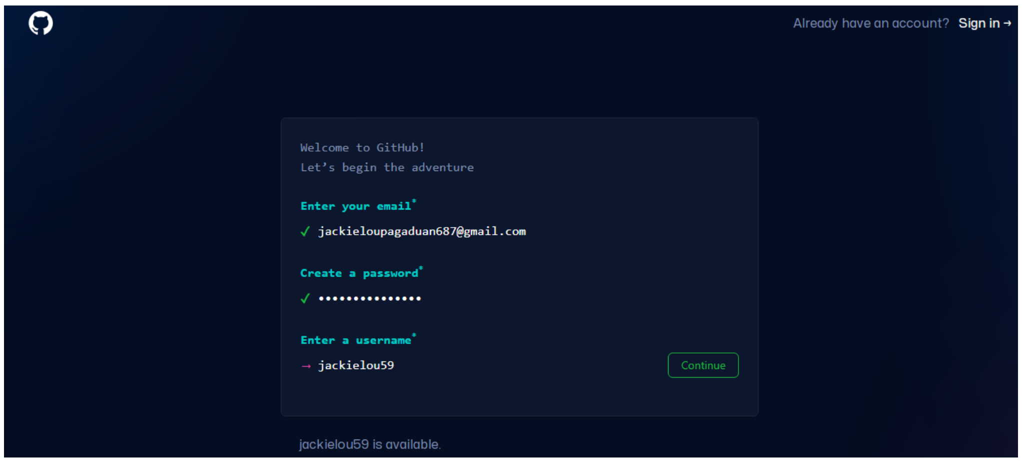
1. Go to GitHub.com, then click Sign Up to create a new account.



Build and ship software on a single, collaborative platform

Join the world's most widely adopted AI-powered developer platform.

2. A page below will appear after clicking Sign up, allowing you to enter your valid email address and create a password and username. Follow on-screen instructions, then hit continue.



Already have an account? Sign in →

Welcome to GitHub!
Let's begin the adventure

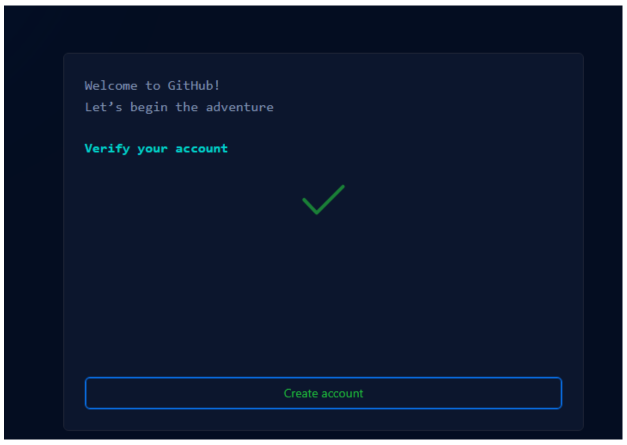
Enter your email*
✓ jackieloupagaduan687@gmail.com

Create a password*
✓ ••••••••••••••

Enter a username*
→ jackielou59

jackielou59 is available.

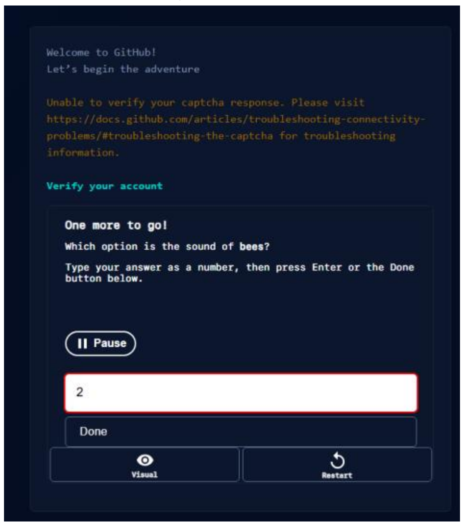
3. Wait for GitHub to verify your account, then hit Create Account.



Welcome to GitHub!
Let's begin the adventure

Verify your account

4. After clicking Create an account, the web page below will appear for you to solve. Follow the on-screen instructions. After complete verification, a PIN code will be sent to your email.



Welcome to GitHub!
Let's begin the adventure

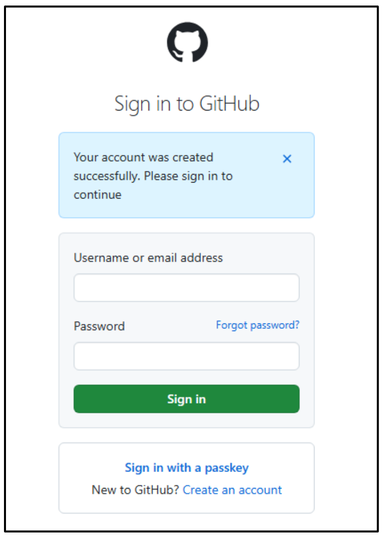
Unable to verify your captcha response. Please visit https://docs.github.com/articles/troubleshooting-connectivity-problems/#troubleshooting-the-captcha for troubleshooting information.

Verify your account

One more to go!
Which option is the sound of bees?
Type your answer as a number, then press Enter or the Done button below.

2

5. After successfully verifying your account, you can now log in to GitHub using your newly created account. Input necessary details. Then, follow on-screen instructions.



Sign in to GitHub

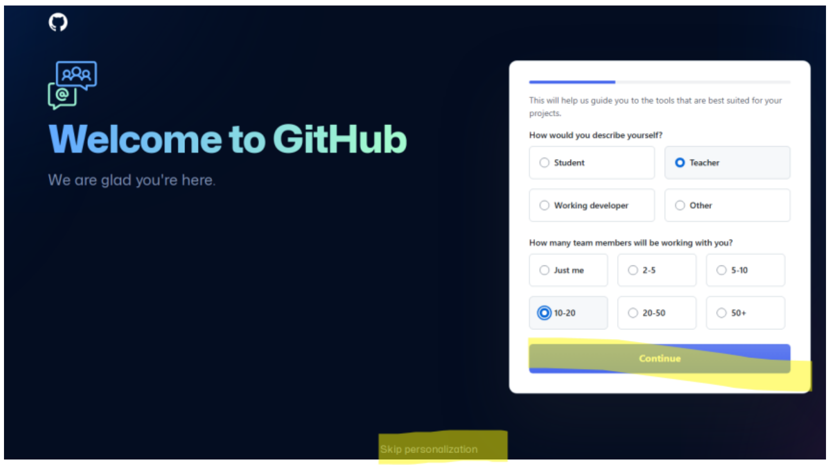
Your account was created successfully. Please sign in to continue

Username or email address

Password

New to GitHub? Create an account

6. Provide the necessary details, or You may skip this personalization.



This will help us guide you to the tools that are best suited for your projects.

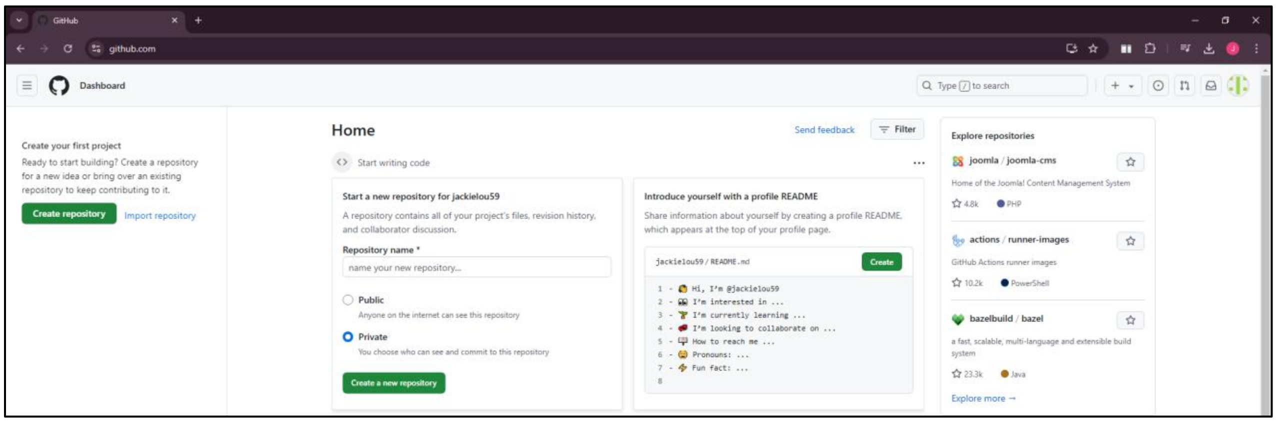
How would you describe yourself?

Student [✓ Teacher] Working developer Other

How many team members will be working with you?

Just me 2-5 5-10 [✓ 10-20] 20-50 50+

7. Hit continue for free. Wait for your dashboard to be initialized. Once done, you will be directed to the following dashboard.



Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Home

Start writing code

Start a new repository for jackielou59

A repository contains all of your project's files, revision history, and collaborator discussion.

Repository name *

name your new repository...

Public — Anyone on the internet can see this repository

[✓ Private] — You choose who can see and commit to this repository

Introduce yourself with a profile README

Share information about yourself by creating a profile README, which appears at the top of your profile page.

jackielou59 / README.md

```
1 - 👋 Hi, I'm @jackielou59
2 - 👀 I'm interested in ...
3 - 🌱 I'm currently learning ...
4 - 💞️ I'm looking to collaborate on ...
5 - 📫 How to reach me ...
6 - 😄 Pronouns: ...
7 - ⚡ Fun fact: ...
8
```

Explore repositories

joomla / joomla-cms — Home of the Joomla! Content Management System — ☆ 4.8k ● PHP

actions / runner-images — GitHub Actions runner images — ☆ 10.2k ● PowerShell

bazelbuild / bazel — a fast, scalable, multi-language and extensible build system — ☆ 23.3k ● Java

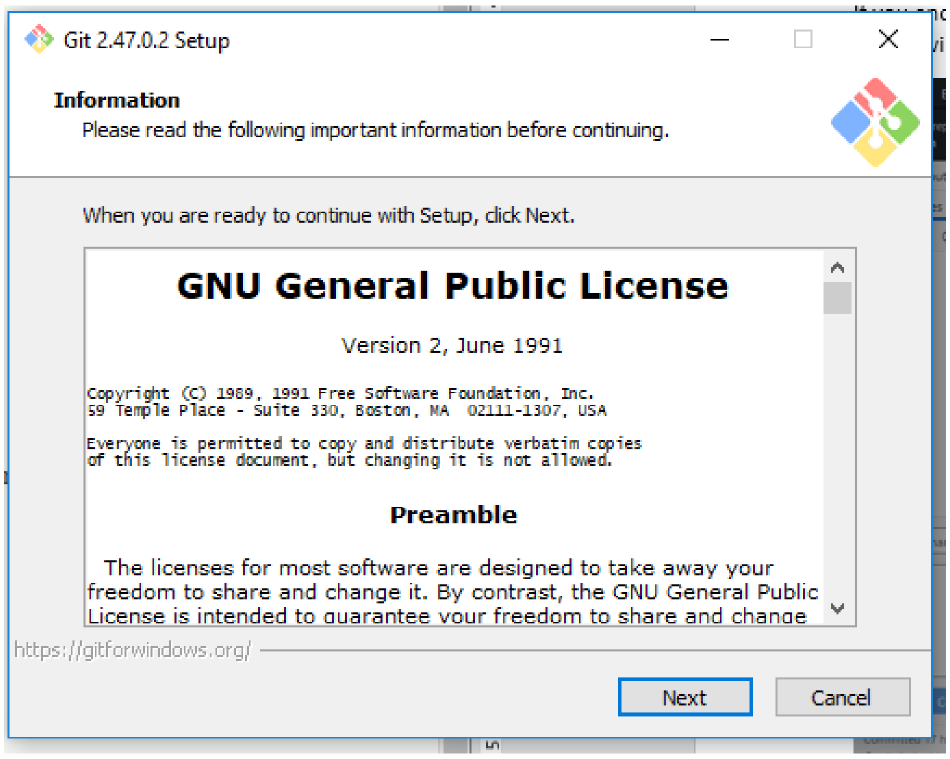
---

Installing necessary tools for Collaboration

Download Git-2.27.0.2-64 and GitHub Desktop, which are uploaded to our Google Classroom, and install them. See details for installation below.

Installing Git for Windows 2.47.0

1. We need to install these tools first before anything else to have a smooth flow of activity. First, install Git 2.47, and then install the GitHub desktop. Follow the given instructions here. Run and install the application.



Information

Please read the following important information before continuing.

When you are ready to continue with Setup, click Next.

GNU General Public License

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

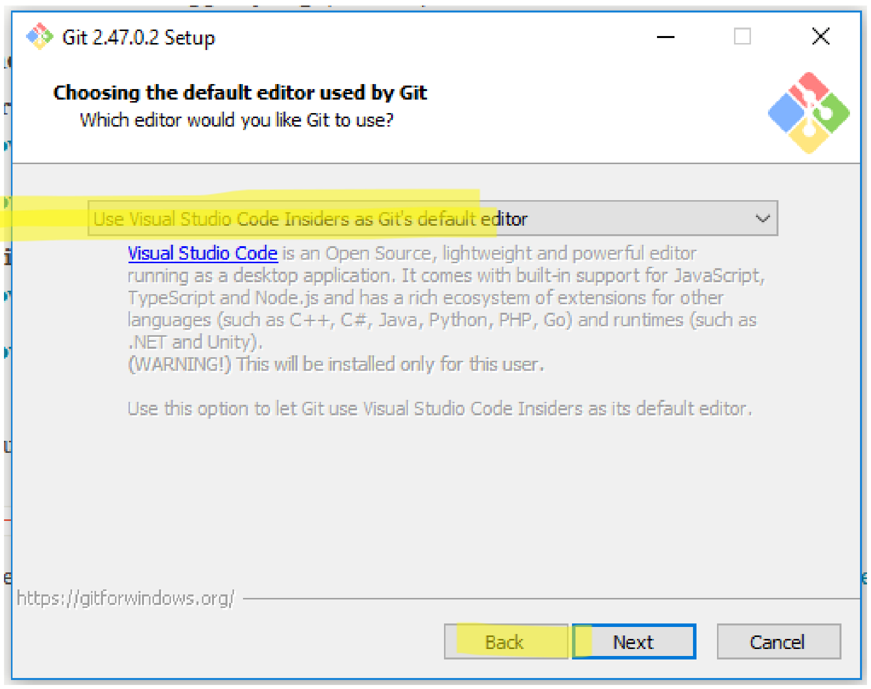
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change

https://gitforwindows.org/

2. Click Next until you reach this page. Choose "Use Visual Studio Code Insiders as Git's default editor," then click Next.



Git 2.47.0.2 Setup

Choosing the default editor used by Git

Which editor would you like Git to use?

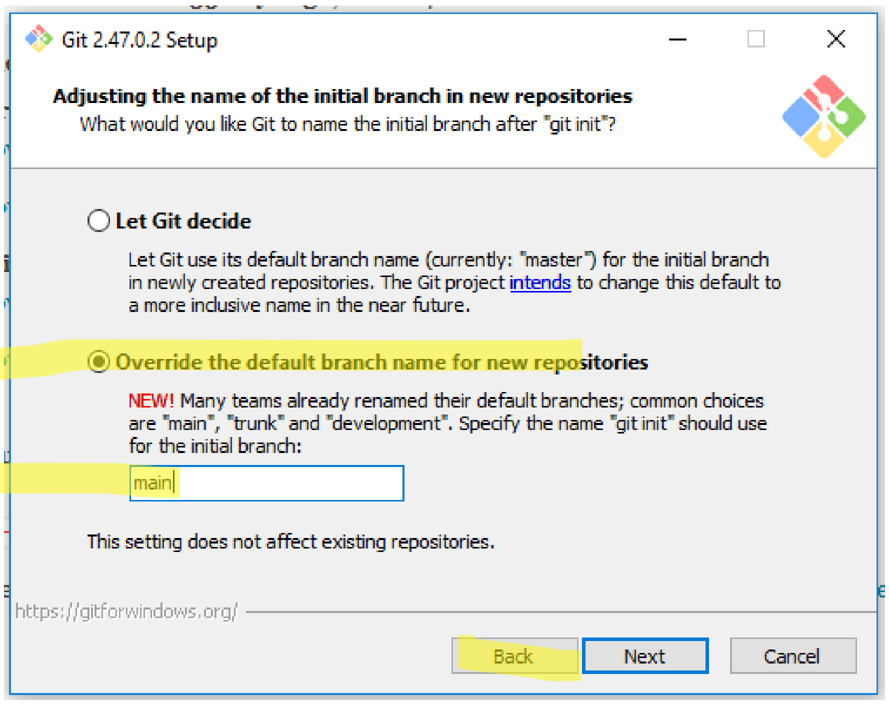
Use Visual Studio Code Insiders as Git's default editor

Visual Studio Code is an Open Source, lightweight and powerful editor running as a desktop application. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).
(WARNING!) This will be installed only for this user.

Use this option to let Git use Visual Studio Code Insiders as its default editor.

https://gitforwindows.org/

3. Choose Override the default branch name for new repositories, then click next until installation is complete.



Git 2.47.0.2 Setup

Adjusting the name of the initial branch in new repositories

What would you like Git to name the initial branch after "git init"?

[ ] **Let Git decide**

Let Git use its default branch name (currently: "master") for the initial branch in newly created repositories. The Git project intends to change this default to a more inclusive name in the near future.

[✓ **Override the default branch name for new repositories**]

NEW! Many teams already renamed their default branches; common choices are "main", "trunk" and "development". Specify the name "git init" should use for the initial branch:

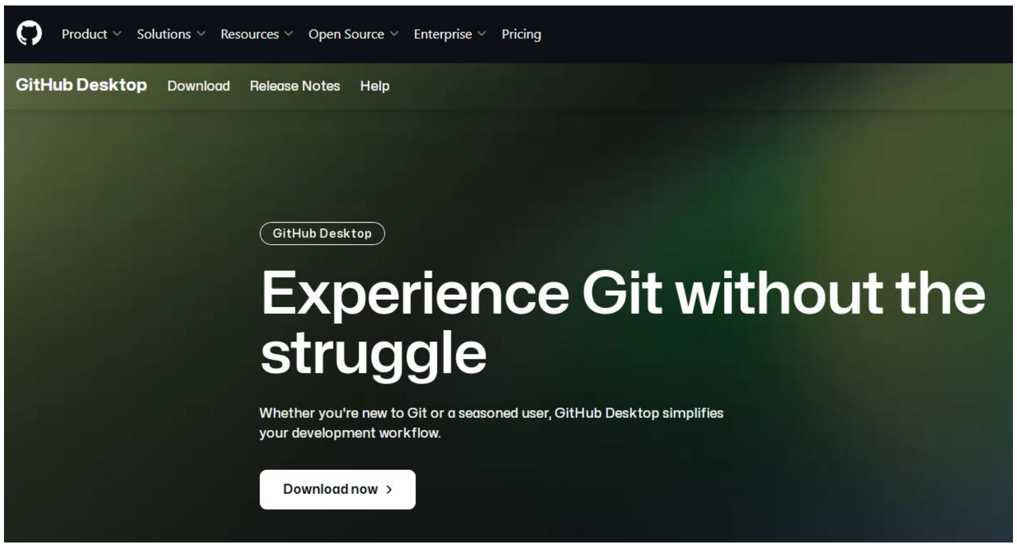
main

This setting does not affect existing repositories.

https://gitforwindows.org/

Installing GitHub Desktop

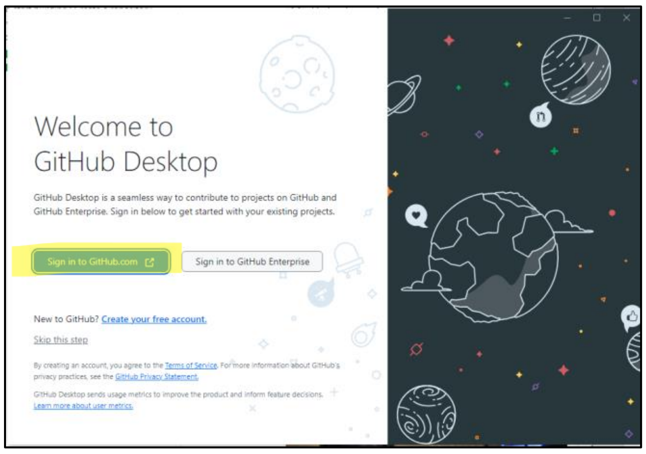
1. Download, run, and install the application.



Experience Git without the struggle

Whether you're new to Git or a seasoned user, GitHub Desktop simplifies your development workflow.

2. Sign in to your GitHub account. Wait for a few seconds.



Welcome to GitHub Desktop

GitHub Desktop is a seamless way to contribute to projects on GitHub and GitHub Enterprise. Sign in below to get started with your existing projects.

Sign in to GitHub.com

Sign in to GitHub Enterprise

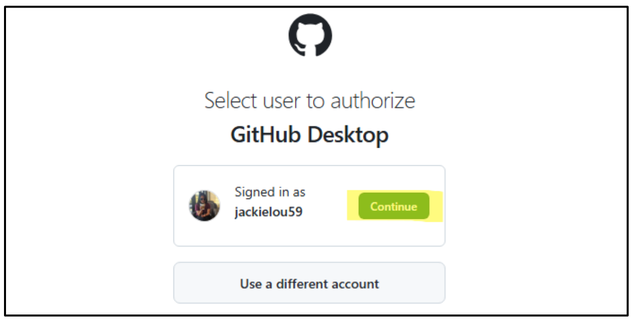
New to GitHub? Create your free account.

Skip this step

By creating an account, you agree to the Terms of Service. For more information about GitHub's privacy practices, see the GitHub Privacy Statement.

GitHub Desktop sends usage metrics to improve the product and inform feature decisions. Learn more about user metrics.

If this does not show during installation, go to File > Options, and in the "Accounts" section, click the "Sign in" button. This will prompt you to sign in using your browser, so click "Continue with browser."



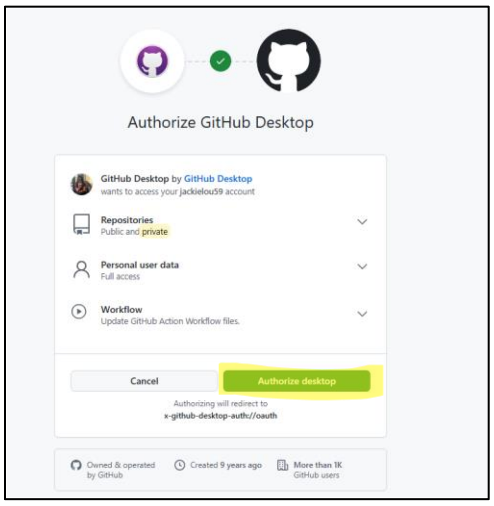
Select user to authorize
GitHub Desktop

Signed in as
jackielou59

Continue

Use a different account

3. Your default browser on your computer will try to log in to your GitHub account. It will prompt you to log in using the account you created earlier.
4. Click "Authorize desktop". This will let you input your password. Once you enter your password, it will open GitHub Desktop. Wait for a few seconds, then click Finish.



Authorize GitHub Desktop

GitHub Desktop by GitHub Desktop
wants to access your jackielou59 account

Repositories
Public and private

Personal user data
Full access

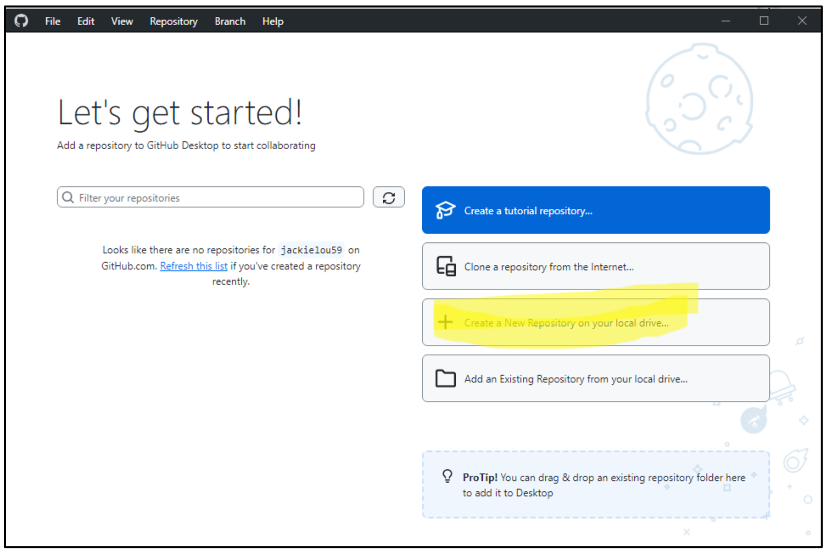
Workflow
Update GitHub Action Workflow files.

Cancel | Authorize desktop

Authorizing will redirect to
x-github-desktop-auth://oauth

Owned & operated by GitHub | Created 9 years ago | More than 1K GitHub users

5. Let's Get Started page will appear. Choose "Create a New Repository on your local drive".



File Edit View Repository Branch Help

Let's get started!

Add a repository to GitHub Desktop to start collaborating

Filter your repositories

Looks like there are no repositories for `jackielou59` on GitHub.com. Refresh this list if you've created a repository recently.

Create a tutorial repository...

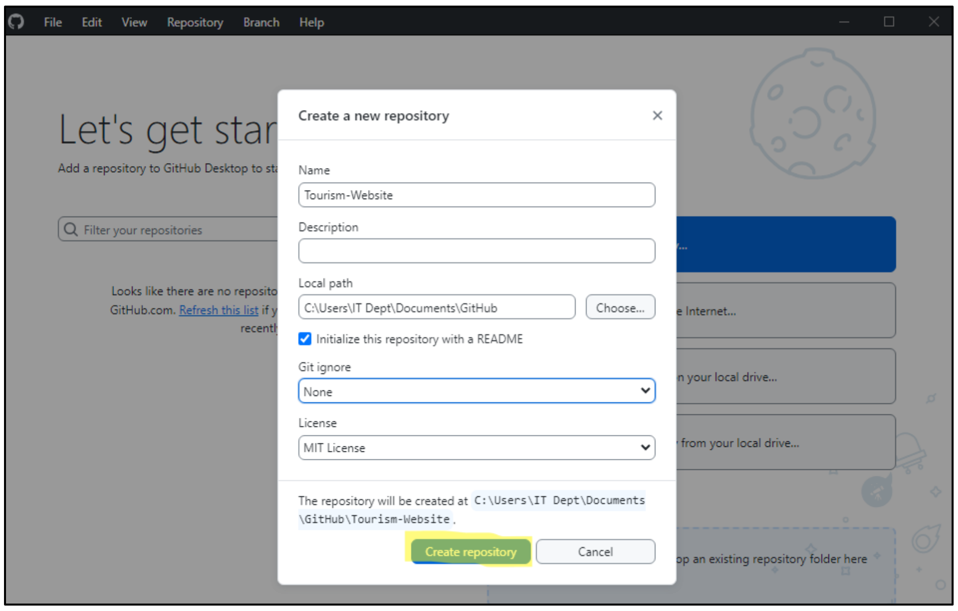
Clone a repository from the Internet...

Create a New Repository on your local drive...

Add an Existing Repository from your local drive...

ProTip! You can drag & drop an existing repository folder here to add it to Desktop

6. Provide the necessary details to create a new repository, e.g., for Name: Portfolio, then hit "Create Repository." See the **Step-by-step guide to creating a new project repository for your Final Project** after this section**.**


File Edit View Repository Branch Help
Let's get star
Add a repository to GitHub Desktop to st
Filter your repositories
Looks like there are no reposito
GitHub.com. Refresh this list if y
recent
e Internet...
n your local drive...
from your local drive...
op an existing repository folder here
Create a new repository
Choose...
Create repository
Cancel

Name: Tourism-Website

Description:

Local path: C:\Users\IT Dept\Documents\GitHub

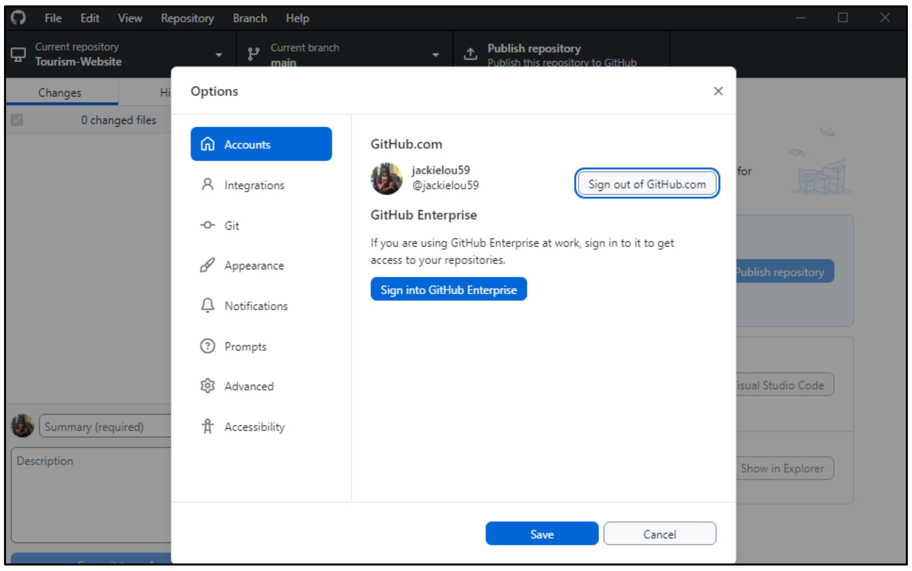
☑ Initialize this repository with a README

Git ignore: None

License: MIT License

The repository will be created at `C:\Users\IT Dept\Documents\GitHub\Tourism-Website`.

7. To make sure everything worked, go back to File> Options> Accounts, and you should see your GitHub username and profile picture on the right side.



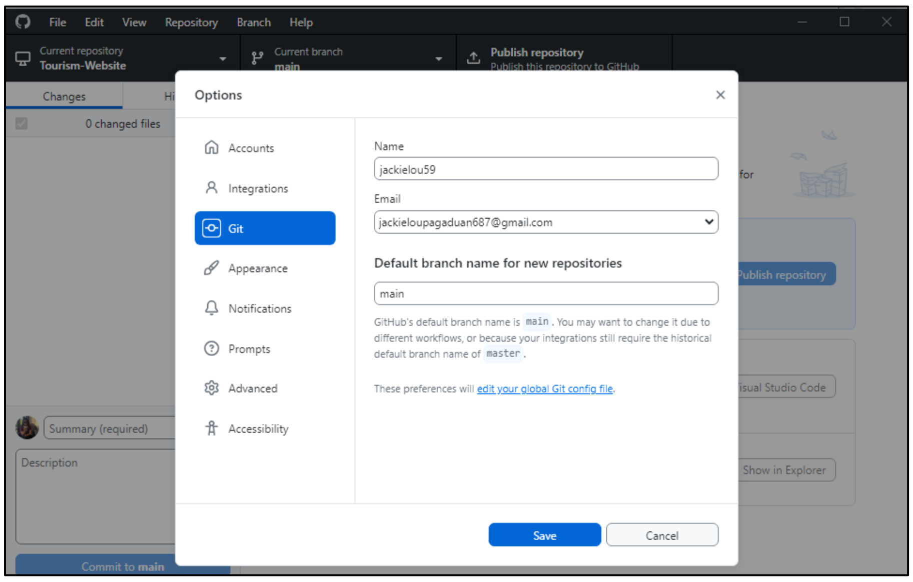
GitHub.com

jackielou59
@jackielou59

GitHub Enterprise

If you are using GitHub Enterprise at work, sign in to it to get access to your repositories.

8. Next, we set our Global Git configuration. In the options, go to the "Git" tab, and you should see your GitHub username listed under Name and your email under "Email." Under "Default branch," select either main. Note that Master used to be the default branch on Git, but currently, GitHub and other places are using "main" as the default, so we will be using it as well. Select "main". Once your settings are set, make sure to click the "Save" button, even if you didn't make any changes on any changes on this screen.



Name: jackielou59

Email: jackieloupagaduan687@gmail.com

Default branch name for new repositories: main

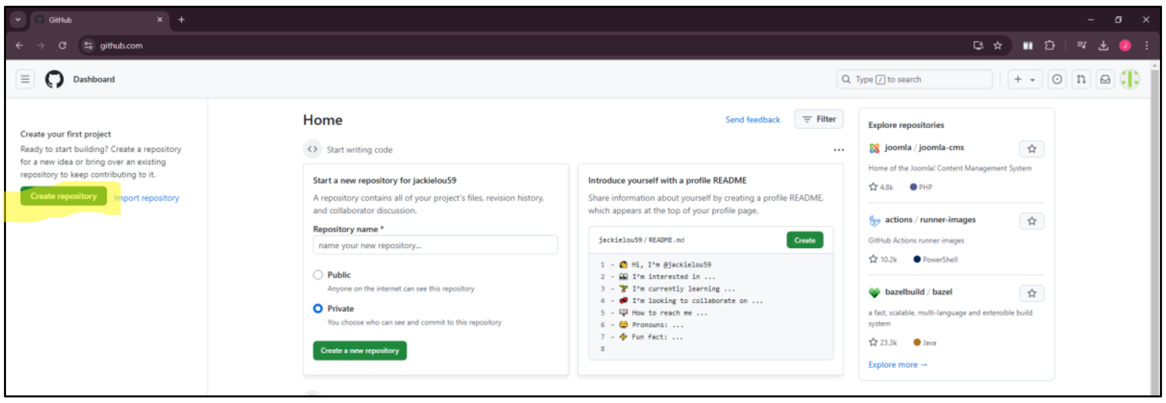
GitHub's default branch name is `main`. You may want to change it due to different workflows, or because your integrations still require the historical default branch name of `master`.

These preferences will edit your global Git config file.

9. Walk through the basics of a Git workflow with a test project.

Uploading your portfolio to GitHub.com

1. Before uploading files to GitHub, you need to create your repository. In your home, click Create New Repository or New. Or use the repository you just created using GitHub Desktop. Just locate it in your repository.

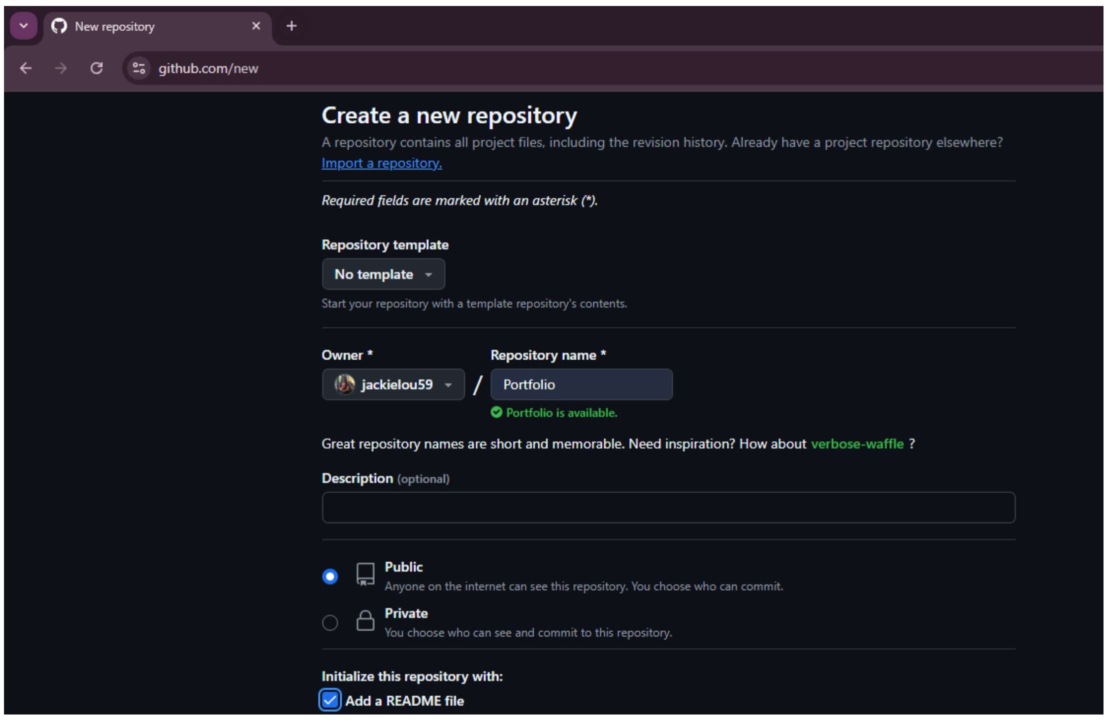


Create your first project
Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Start a new repository for jackielou59
A repository contains all of your project's files, revision history, and collaborator discussion.
Repository name *
Public
Anyone on the internet can see this repository
Private
You choose who can see and commit to this repository

Share information about yourself by creating a profile README, which appears at the top of your profile page.

2. Provide necessary details like Repository Name: Portfolio, set it to public, check Add read me file, choose a License: MIT License, then Hit Create repository. Wait for a few seconds.



Create a new repository
A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Repository template
No template
Start your repository with a template repository's contents.

Owner * jackielou59 / Repository name * Portfolio
Portfolio is available.

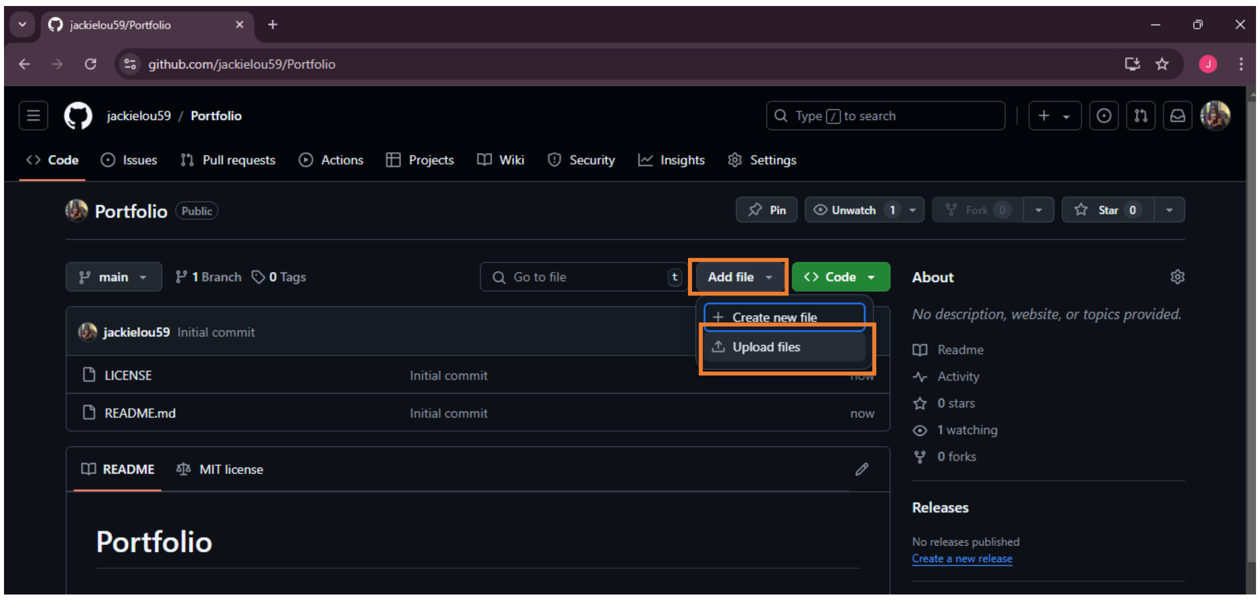
Great repository names are short and memorable. Need inspiration? How about verbose-waffle ?

Description (optional)

[✓ 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:
[✓ Add a README file]

3. In your Portfolio repository, click the dropdown button Add File, and then select Upload files.



jackielou59 Initial commit

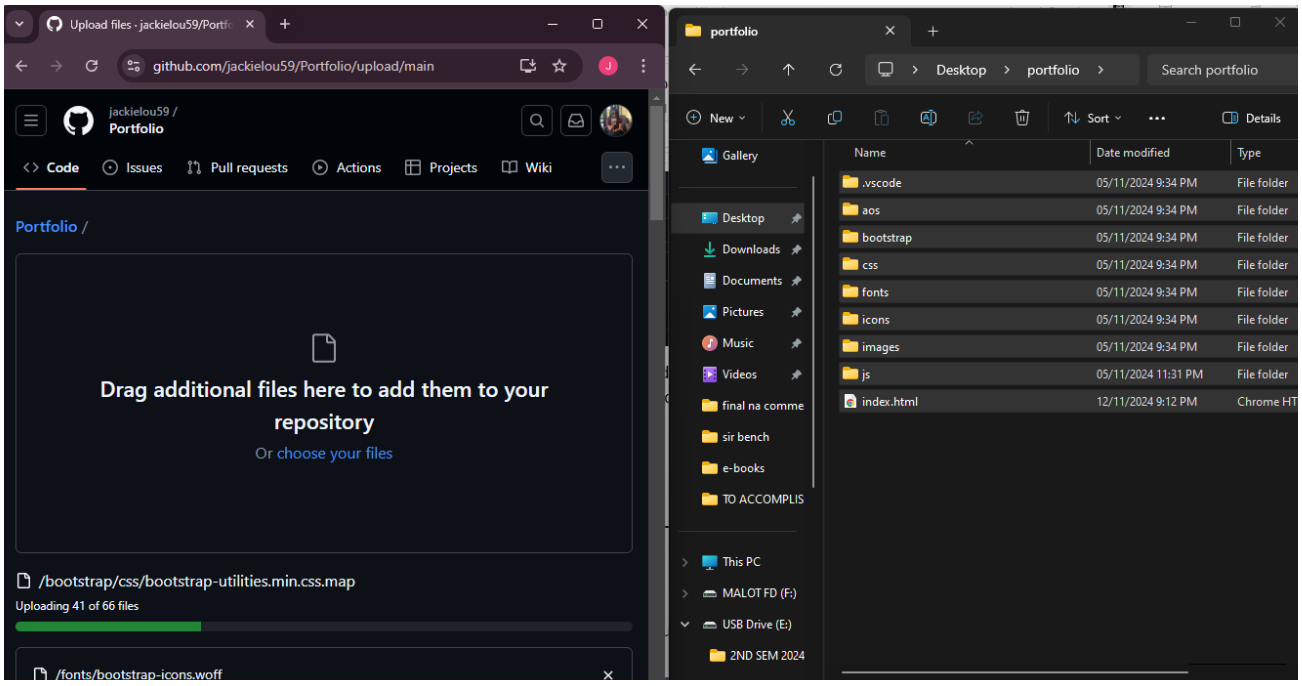
LICENSE	Initial commit	now
README.md	Initial commit	now

No description, website, or topics provided.

No releases published

Portfolio

4. Select all your portfolio files in your local drive, and then drag them to GitHub. If it keeps failing, try to drag your files by two or by folder.



Portfolio /

Drag additional files here to add them to your repository

Or choose your files

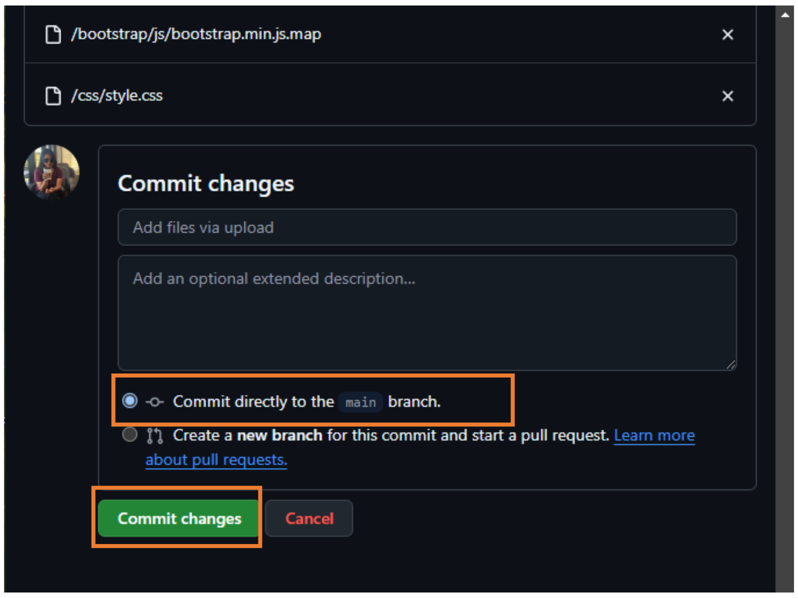
/bootstrap/css/bootstrap-utilities.min.css.map

Uploading 41 of 66 files

/fonts/bootstrap-icons.woff

Name	Date modified	Type
.vscode	05/11/2024 9:34 PM	File folder
aos	05/11/2024 9:34 PM	File folder
bootstrap	05/11/2024 9:34 PM	File folder
css	05/11/2024 9:34 PM	File folder
fonts	05/11/2024 9:34 PM	File folder
icons	05/11/2024 9:34 PM	File folder
images	05/11/2024 9:34 PM	File folder
js	05/11/2024 11:31 PM	File folder
index.html	12/11/2024 9:12 PM	Chrome HT

5. If done uploading, click commit changes. Note: Make sure Commit directly to the main Branch is selected.



/bootstrap/js/bootstrap.min.js.map

/css/style.css

Commit changes

Add files via upload

Add an optional extended description...

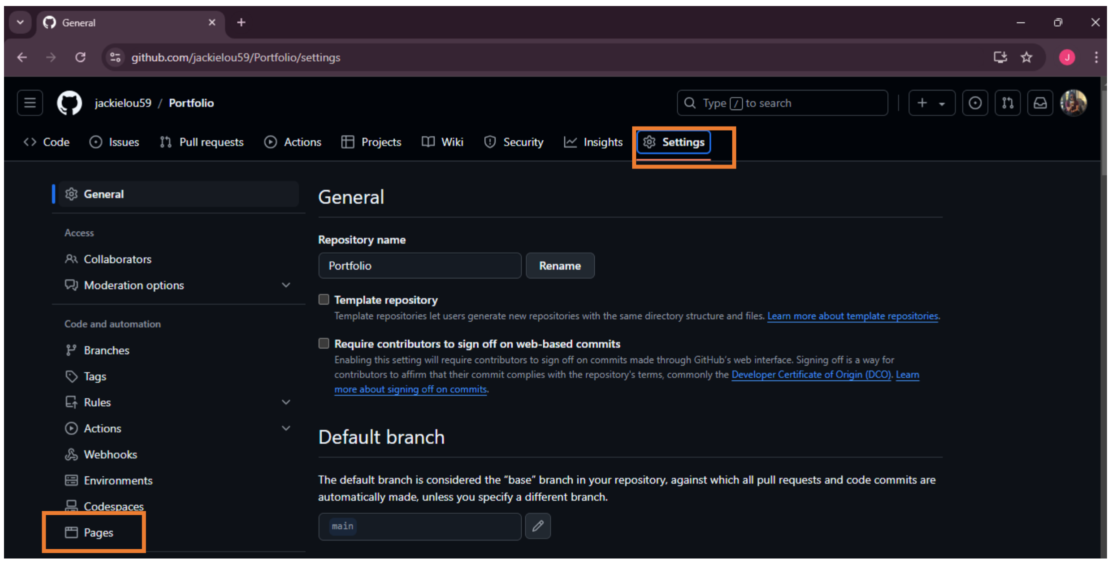
[✓ Commit directly to the `main` branch.]

Create a **new branch** for this commit and start a pull request. Learn more about pull requests.

Hosting your repository using GitHub Pages

To view our repository or portfolio online, we need to host it using GitHub Pages. Follow the following instructions.

1. In your Portfolio repository, select Settings, then Pages.



General

Repository name

Portfolio

Template repository

Template repositories let users generate new repositories with the same directory structure and files. Learn more about template repositories.

Require contributors to sign off on web-based commits

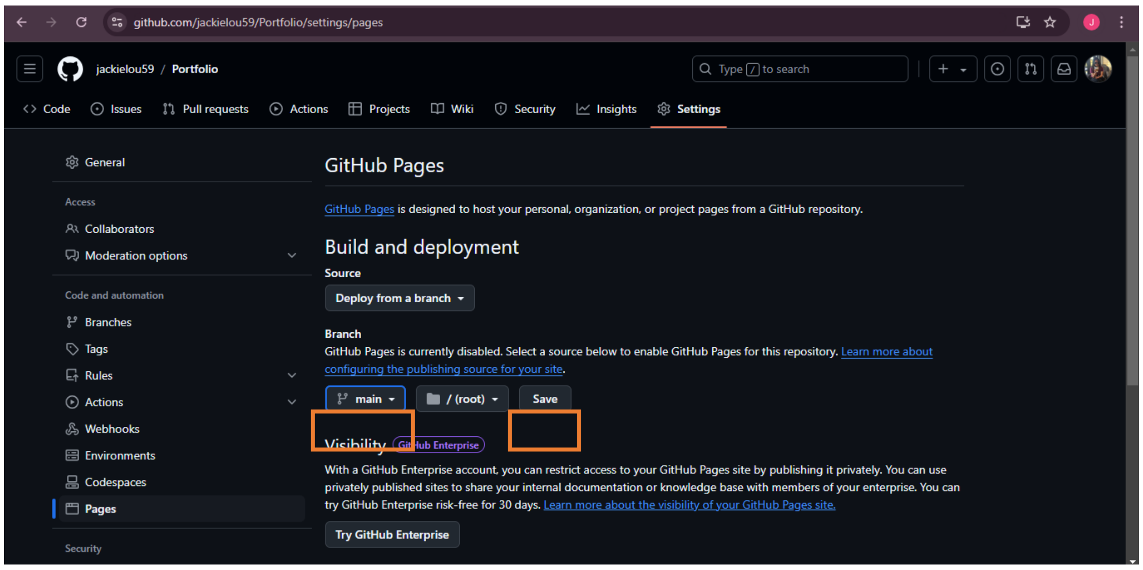
Enabling this setting will require contributors to sign off on commits made through GitHub's web interface. Signing off is a way for contributors to affirm that their commit complies with the repository's terms, commonly the Developer Certificate of Origin (DCO). Learn more about signing off on commits.

Default branch

The default branch is considered the "base" branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.

main

2. In your GitHub Pages page, go to Branch, select Main, then hit save.



GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Build and deployment

Source

Deploy from a branch

Branch

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. Learn more about configuring the publishing source for your site.

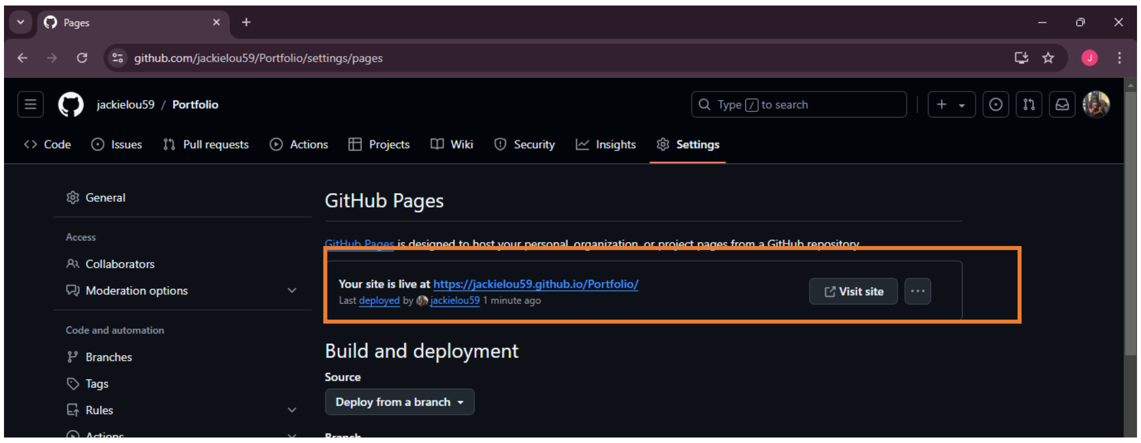
main / (root) Save

Visibility GitHub Enterprise

With a GitHub Enterprise account, you can restrict access to your GitHub Pages site by publishing it privately. You can use privately published sites to share your internal documentation or knowledge base with members of your enterprise. You can try GitHub Enterprise risk-free for 30 days. Learn more about the visibility of your GitHub Pages site.

Try GitHub Enterprise

3. Wait for at least one minute, then refresh. You should be able to see the URL generated by GitHub. Click visit the site to try your very first website hosted using GitHub pages.



GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is live at https://jackielou59.github.io/Portfolio/

Last deployed by jackielou59 1 minute ago

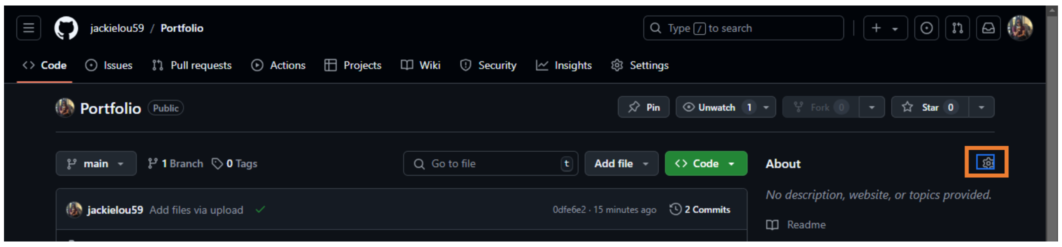
Visit site

Build and deployment

Source

Deploy from a branch

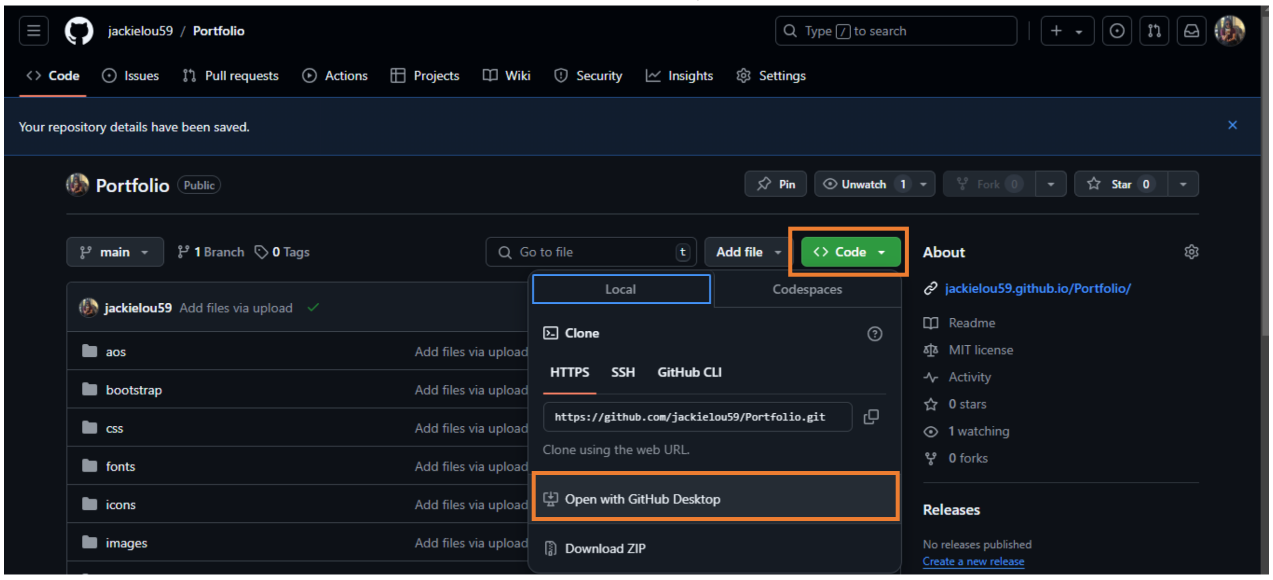
4. Copy the URL, then navigate to Code. Click the About setting, then just click the checkbox "Use your GitHub Pages website" and hit Save Changes. You can see your URL added to your About section. You can now share this with others.



jackielou59 Add files via upload 0dfe6e2 · 15 minutes ago 2 Commits

No description, website, or topics provided.

5. If you want to edit your Code, select GitHub Desktop in your code dropdown. Make sure to have this installed first so you can edit your Code as well as VS Code since this is our default editor. If not yet installed, see installation details in the Installation Necessary Tools section.



Your repository details have been saved.

jackielou59 Add files via upload

aos	Add files via upload
bootstrap	Add files via upload
css	Add files via upload
fonts	Add files via upload
icons	Add files via upload
images	Add files via upload

https://github.com/jackielou59/Portfolio.git

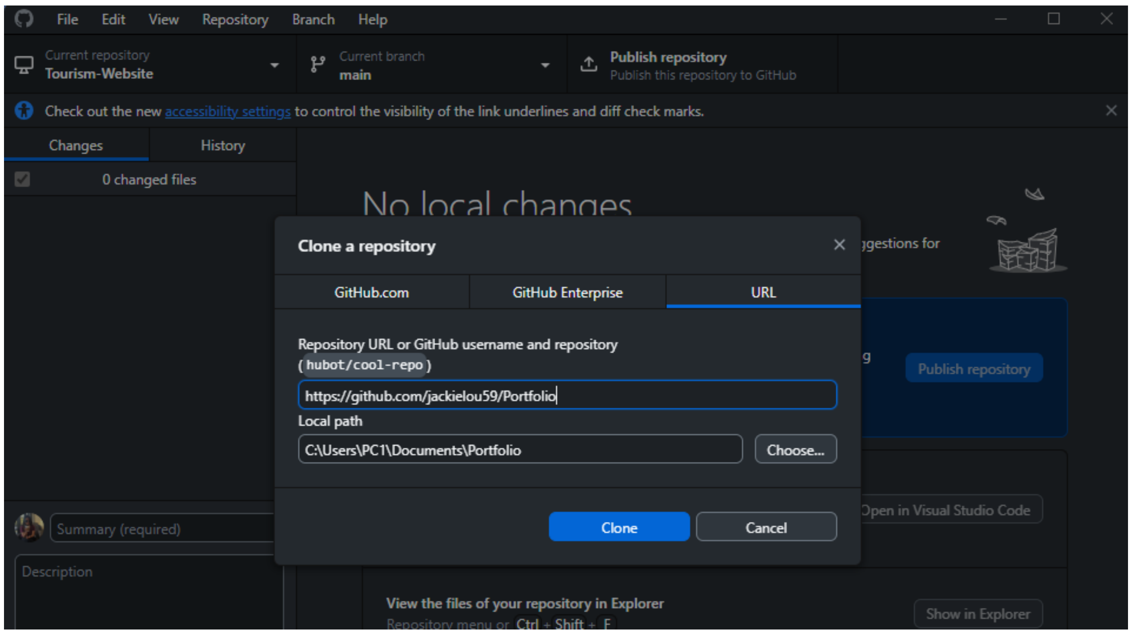
Clone using the web URL.

jackielou59.github.io/Portfolio/

No releases published

Create a new release

6. When you click the Open with GitHub Desktop, a clone repository modal page will appear. This is a necessary step for you to edit your Code locally. Hit clone and wait for it to be cloned.

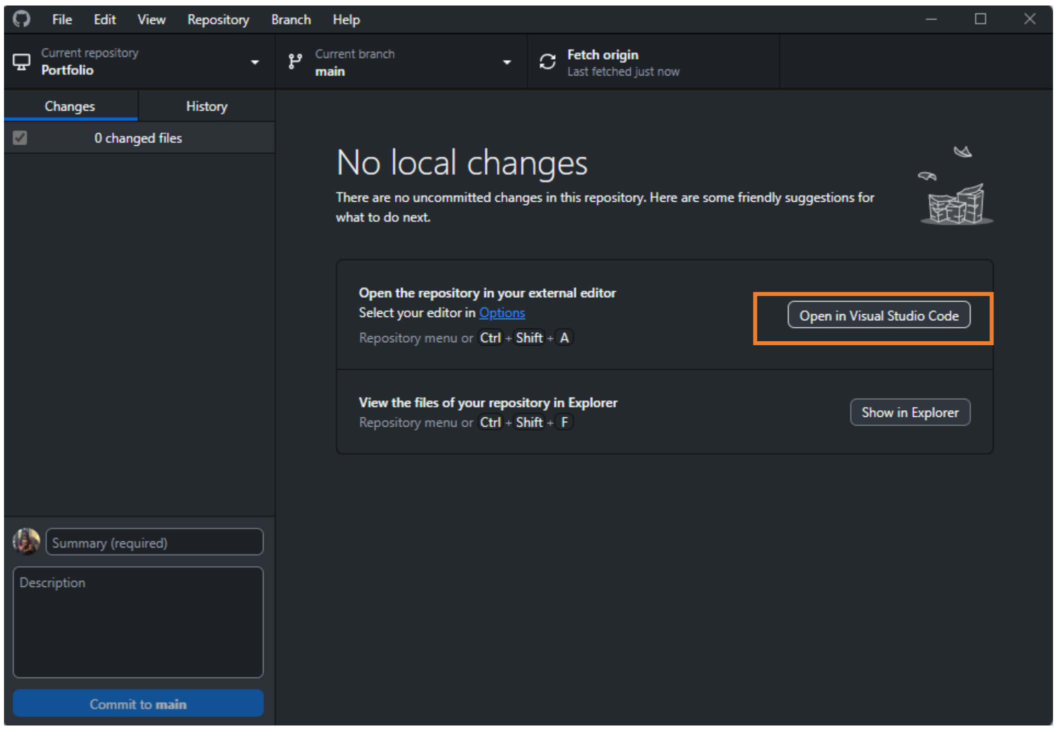


Check out the new accessibility settings to control the visibility of the link underlines and diff check marks.

Repository URL or GitHub username and repository (`hubot/cool-repo`)
https://github.com/jackielou59/Portfolio
Local path
C:\Users\PC1\Documents\Portfolio

View the files of your repository in Explorer
Repository menu or Ctrl + Shift + F

7. After successful cloning, the page below will appear. Hit "Open in Visual Studio code" to continue editing your portfolio code. Do not close this since you are going to use it to save your changes later.



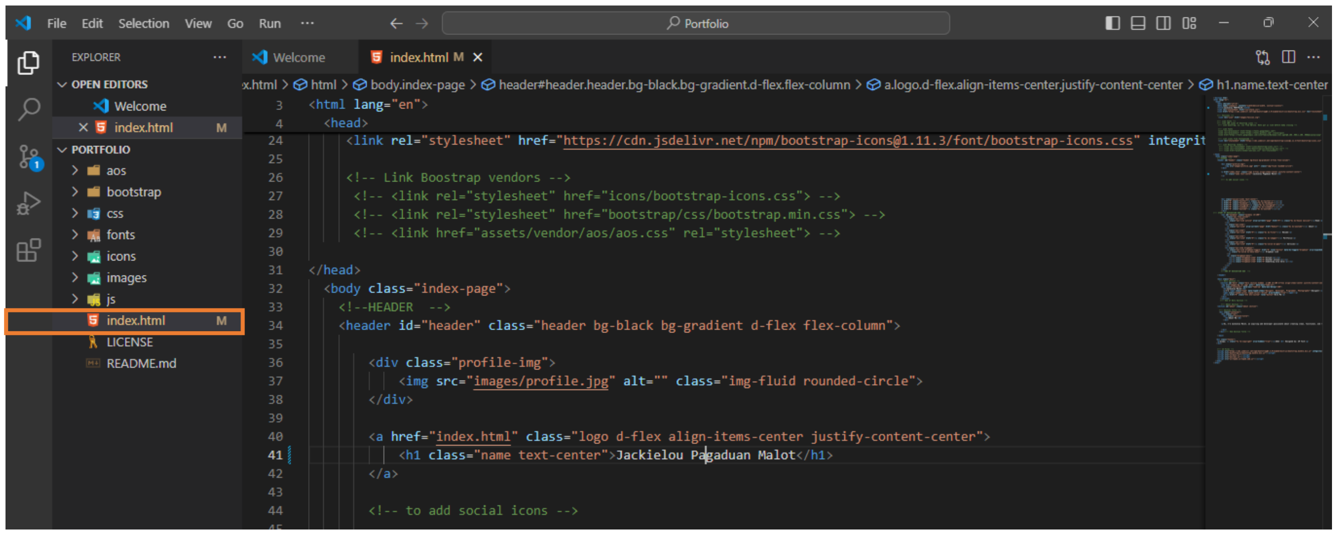
No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Open the repository in your external editor
Select your editor in Options
Repository menu or Ctrl + Shift + A

View the files of your repository in Explorer
Repository menu or Ctrl + Shift + F

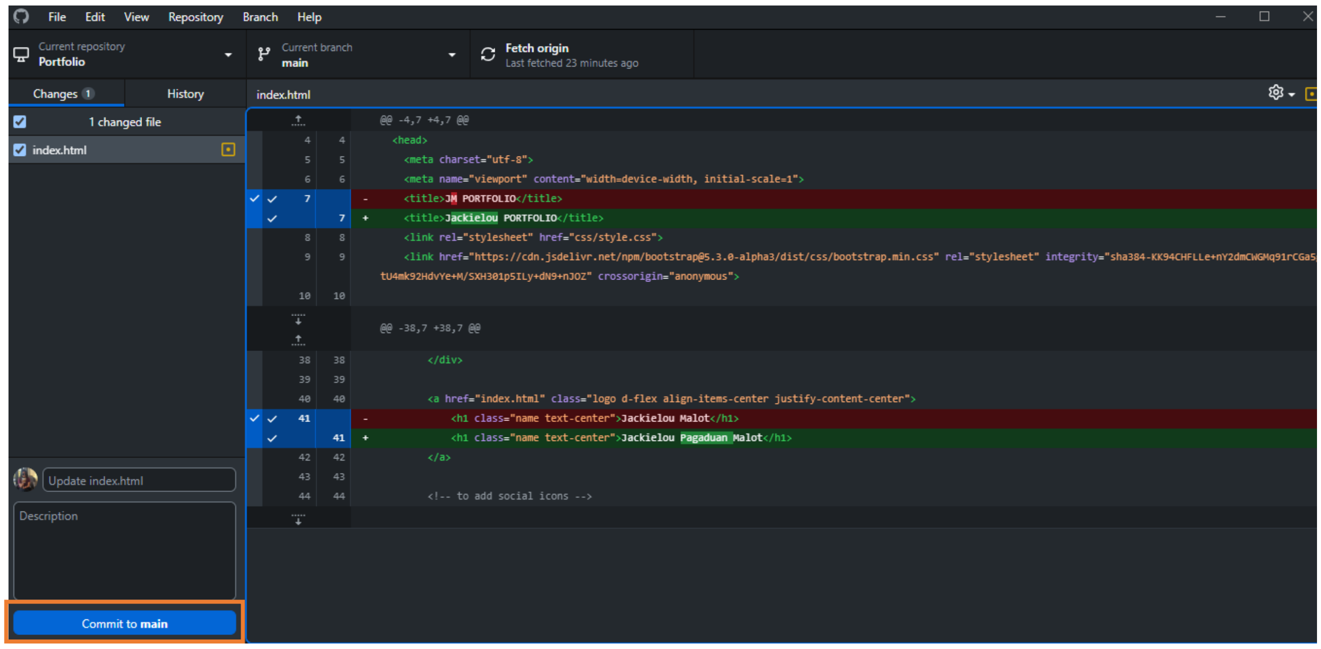
8. In your VS code editor, when you make changes, you can see an indicator that it has been modified. This modification will also be reflected on your GitHub Desktop. Always save when you make changes.



x.html > html > body.index-page > header#header.header.bg-black.bg-gradient.d-flex.flex-column > a.logo.d-flex.align-items-center.justify-content-center > h1.name.text-center

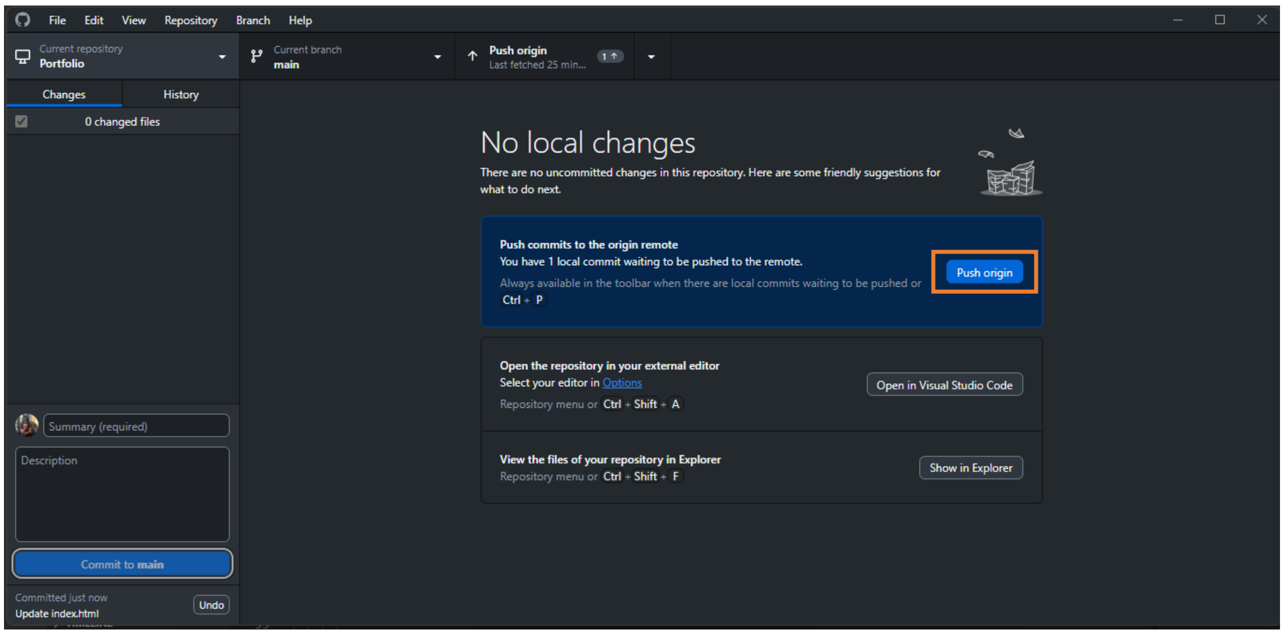
```
<html lang="en">
  <head>
      <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.css" integrit

      <!-- Link Boostrap vendors -->
       <!-- <link rel="stylesheet" href="icons/bootstrap-icons.css"> -->
       <!-- <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css"> -->
       <!-- <link href="assets/vendor/aos/aos.css" rel="stylesheet"> -->

</head>
  <body class="index-page">
    <!--HEADER  -->
    <header id="header" class="header bg-black bg-gradient d-flex flex-column">

        <div class="profile-img">
            <img src="images/profile.jpg" alt="" class="img-fluid rounded-circle">
        </div>

        <a href="index.html" class="logo d-flex align-items-center justify-content-center">
            <h1 class="name text-center">Jackielou Pagaduan Malot</h1>
        </a>

        <!-- to add social icons -->
```

9. To save our changes in our GitHub repository, we need to commit to changes to our main repository. Go back to GitHub Desktop. As noticed, changes made were reflected. To save our changes, click commit to main.



```
@@ -4,7 +4,7 @@
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
-   <title>JM PORTFOLIO</title>
+   <title>Jackielou PORTFOLIO</title>
    <link rel="stylesheet" href="css/style.css">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-KK94CHFLLe+nY2dmCWGMq91rCGa5
tU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ" crossorigin="anonymous">

@@ -38,7 +38,7 @@
        </div>

        <a href="index.html" class="logo d-flex align-items-center justify-content-center">
-         <h1 class="name text-center">Jackielou Malot</h1>
+         <h1 class="name text-center">Jackielou Pagaduan Malot</h1>
        </a>

        <!-- to add social icons -->
```

10. After a successful update, we need to Push this to reflect in our GitHub repository. Click the Push Origin button. Wait for a few seconds.



No local changes

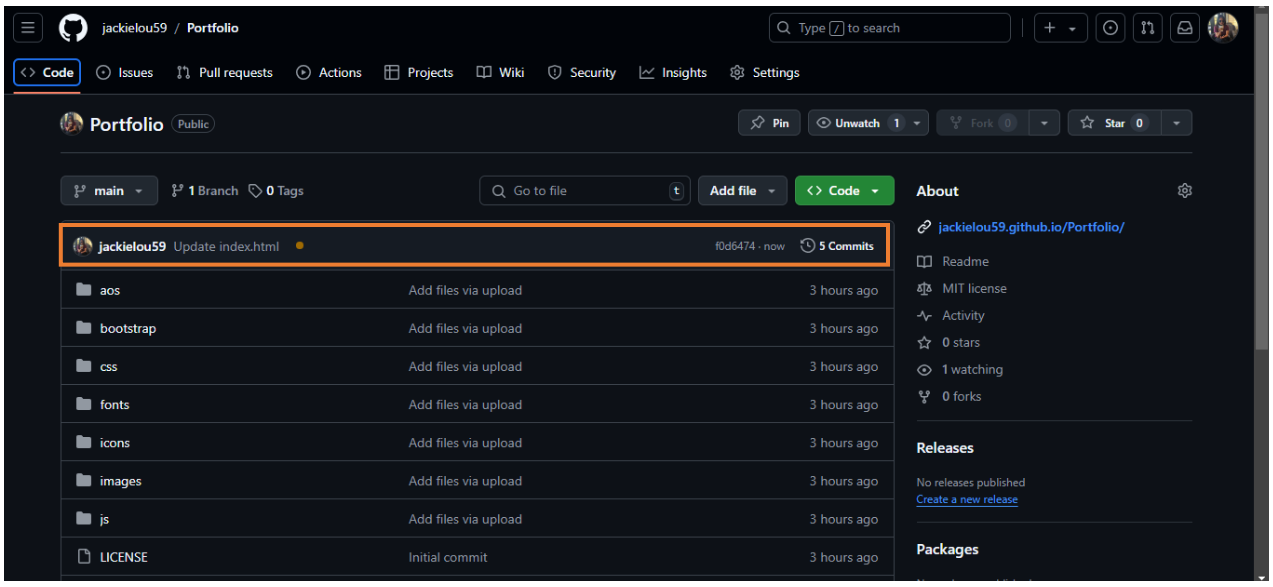
There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Push commits to the origin remote
You have 1 local commit waiting to be pushed to the remote.
Always available in the toolbar when there are local commits waiting to be pushed or Ctrl + P
Push origin

Open the repository in your external editor
Select your editor in Options
Repository menu or Ctrl + Shift + A
Open in Visual Studio Code

View the files of your repository in Explorer
Repository menu or Ctrl + Shift + F
Show in Explorer

11. After push, go back to your repository. Refresh, and you will notice that index.html is just updated. Click the URL to view changes.



jackielou59 Update index.html · f0d6474 · now · 5 Commits

aos	Add files via upload	3 hours ago
bootstrap	Add files via upload	3 hours ago
css	Add files via upload	3 hours ago
fonts	Add files via upload	3 hours ago
icons	Add files via upload	3 hours ago
images	Add files via upload	3 hours ago
js	Add files via upload	3 hours ago
LICENSE	Initial commit	3 hours ago

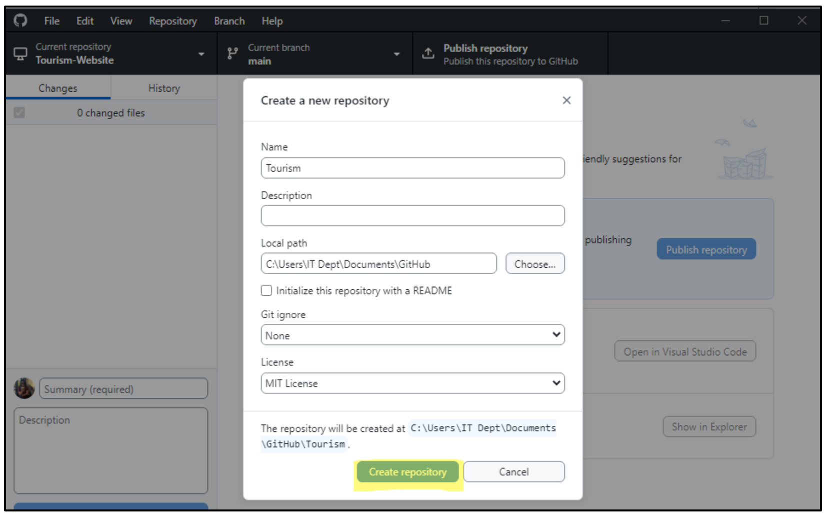
About
jackielou59.github.io/Portfolio/
Readme
MIT license
Activity
0 stars
1 watching
0 forks

Releases
No releases published
Create a new release

Packages

Step-by-step to create a new project repository for your Final Project using GitHub Desktop

1. To create a new repository using GitHub desktop, go to File > New repository, and then in the Name field, write the Name of your test project, for example, your "Tourism" final project.



Create a new repository

Name: Tourism

Description:

Local path: C:\Users\IT Dept\Documents\GitHub

☐ Initialize this repository with a README

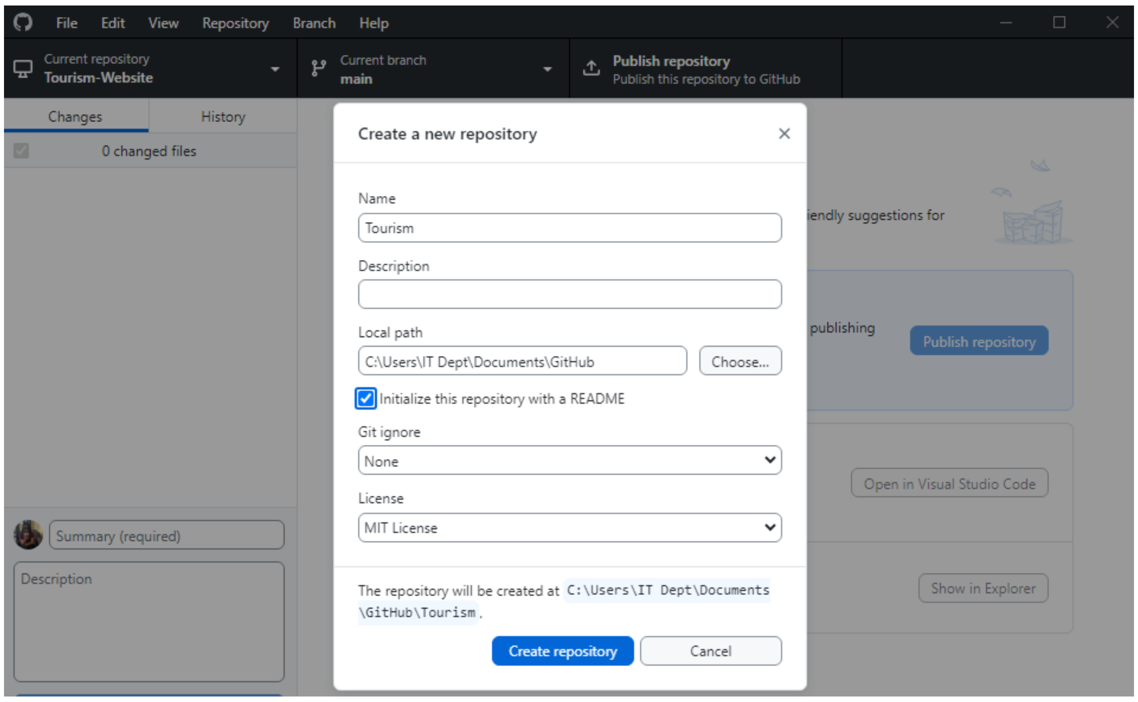
Git ignore: None

License: MIT License

The repository will be created at `C:\Users\IT Dept\Documents\GitHub\Tourism`.

Note: *GitHub repos need to be named a string without spaces. You may use hyphens between words. In the description field, let people who see your repository on GitHub know what it's about. The local path is the folder where you want to create your repository. Let's put our repos in the Documents/GitHub folder. Check the box asking if you want to initialize this repository with a readme file, which lets you write a more detailed description of your project that will show up on GitHub.*

2. The Git ignore fields lets you select types of files that you don't want to add to the repository. Let's just leave it as "None" and then ignore files.
3. The License field tells people what they are allowed to do with your Code in the repository. Since a lot of open sources exist on GitHub nowadays, developers select the MIT License, which allows them to copy, distribute, and modify your Code. If you leave it blank, your project will be licensed with standard copyright. Read further about this at ChooseALicense.com.
4. Now, click "Create repository". This will take a few seconds, but GitHub Desktop should create and automatically load the repository.



Create a new repository

Name: Tourism

Description:

Local path: C:\Users\IT Dept\Documents\GitHub

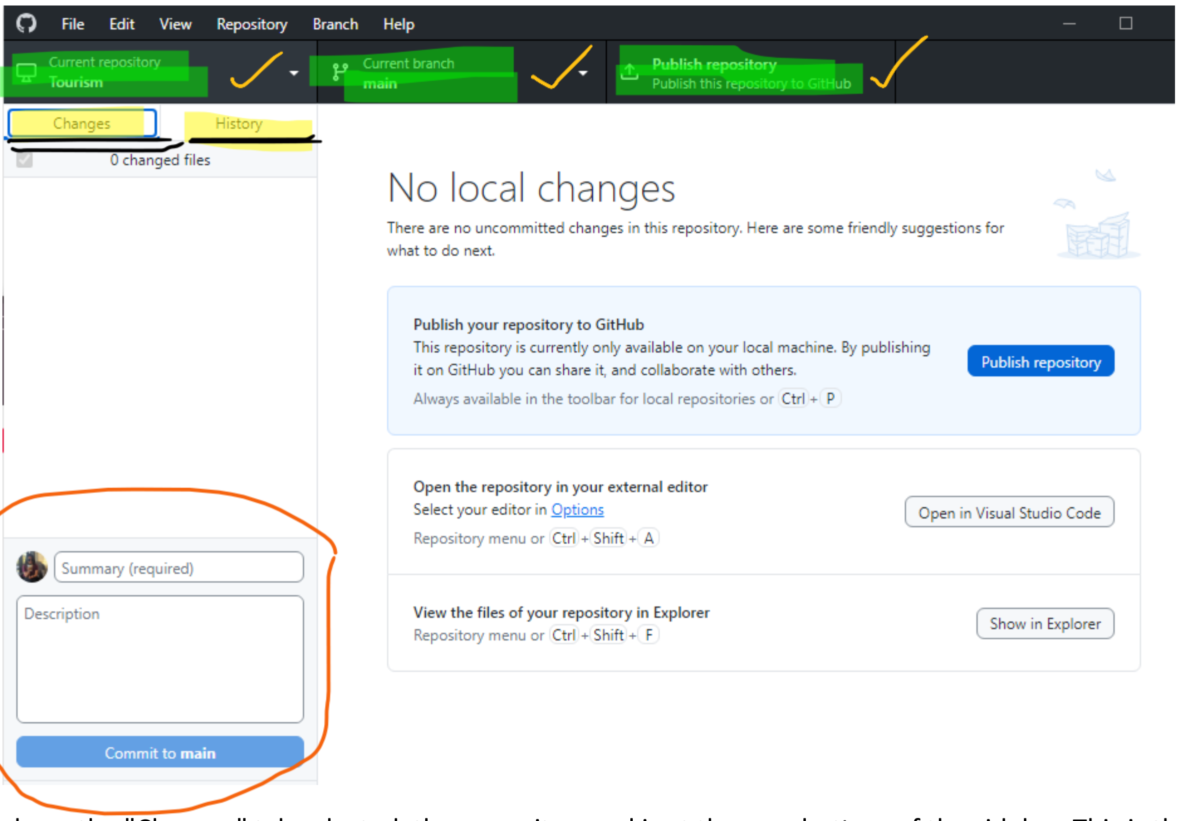
☑ Initialize this repository with a README

Git ignore: None

License: MIT License

The repository will be created at `C:\Users\IT Dept\Documents\GitHub\Tourism`.

5. Under the menu bar, you'll see another bar with information about your repository. On the left is the current open repository. Next are current branches, which we will tackle later. Just take note that this is one version of the codebase, and you can have multiple branches. The third step is to publish the repository on GitHub.
6. In the current repository, you noticed we have two tabs: the changes tab and the history tab. The changes tab is currently blank since we haven't added any files yet or made changes. For the history tab, we already have an "initial commit" with the files generated by GitHub Desktop when we created this repo.



No local changes

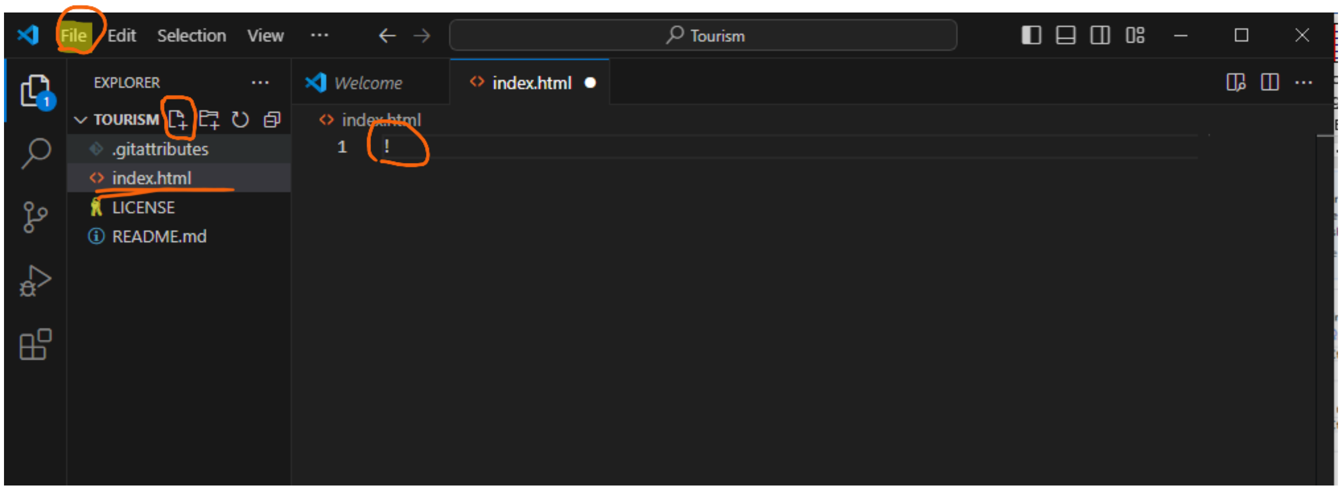
There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Publish your repository to GitHub
This repository is currently only available on your local machine. By publishing it on GitHub you can share it, and collaborate with others.
Always available in the toolbar for local repositories or Ctrl + P

Open the repository in your external editor
Select your editor in Options
Repository menu or Ctrl + Shift + A

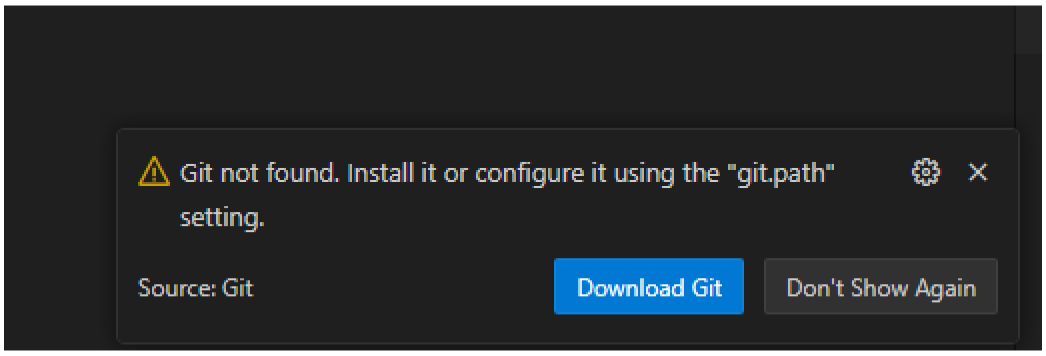
View the files of your repository in Explorer
Repository menu or Ctrl + Shift + F

7. If you have the "Changes" tab selected, the commits panel is at the very bottom of the sidebar. This is the "save point" that you can create during development, which includes a set of changes that you want to track. Let's start making some changes to see what happens in Git.
8. In your VS Code editor, go to File > Open Folder and select the location where you created your GitHub repository (Document)
9. To create a new file, click the "New File" icon in the left sidebar and create an index.html file.
10. In the File, type an exclamation point and hit the tab to create a boilerplate HTML markup, then save it.



```
!
```

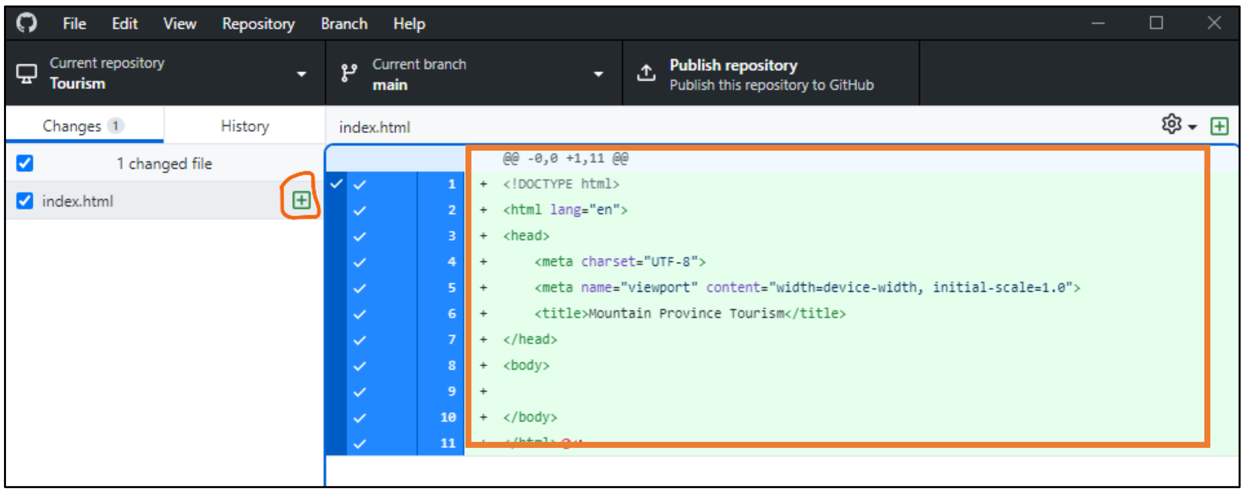
If the following error appears, install the Git 2.47.



Git not found. Install it or configure it using the "git.path" setting.

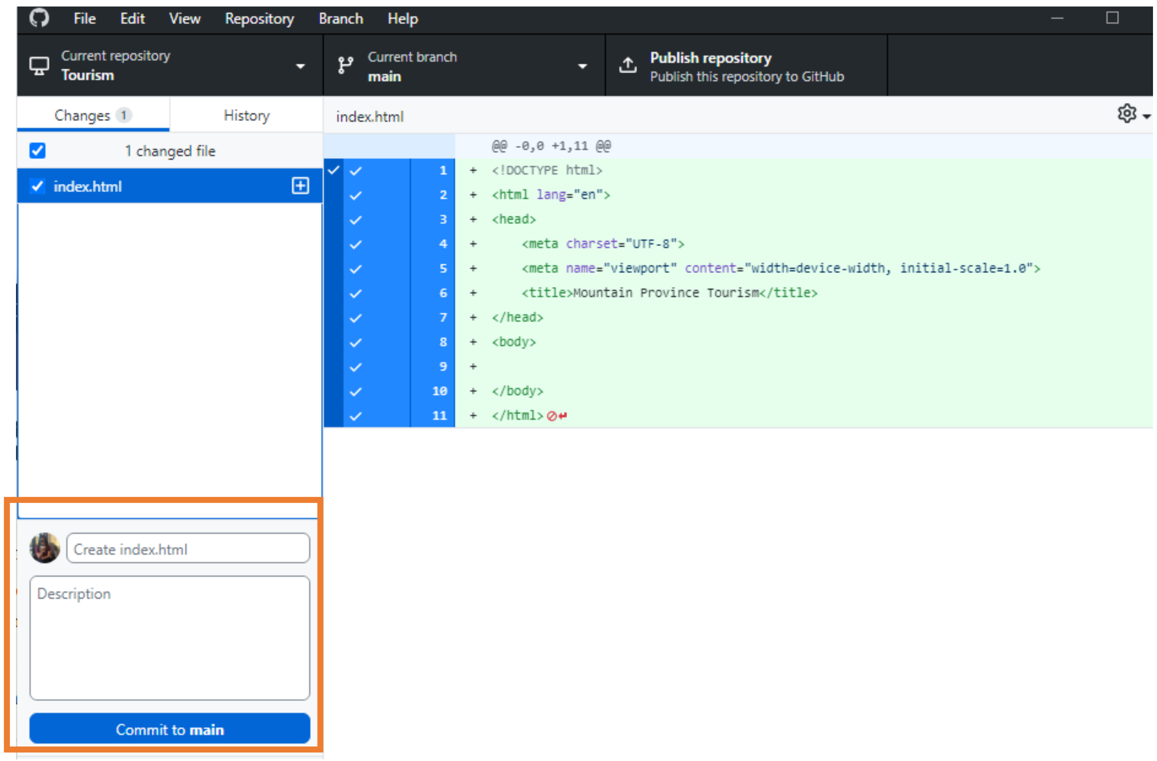
Source: Git

11. Go back to GitHub Desktop, and you'll notice that in the left sidebar under the "Changes" tab will be the index.html File. A green plus icon will be visible beside the filename. This indicates that Git has detected a new file and wants to add it to your repository to track changes. On the right panel, you'll see the actual code changes that you made in the index.html file.



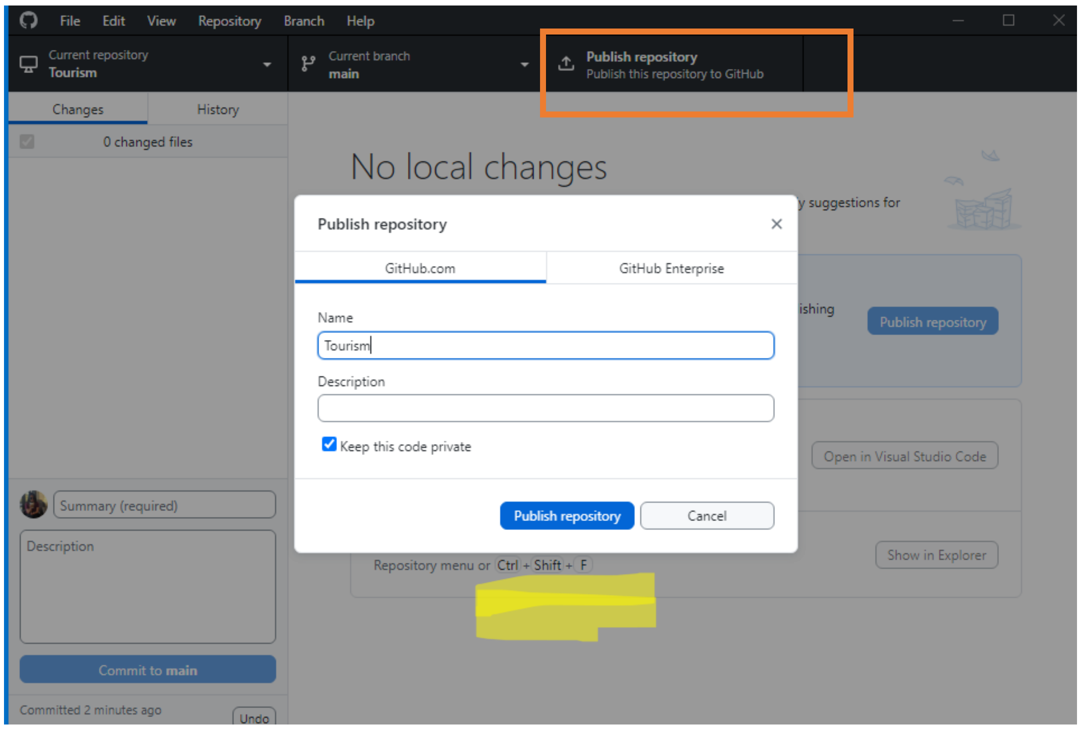
```
@@ -0,0 +1,11 @@
+ <!DOCTYPE html>
+ <html lang="en">
+ <head>
+     <meta charset="UTF-8">
+     <meta name="viewport" content="width=device-width, initial-scale=1.0">
+     <title>Mountain Province Tourism</title>
+ </head>
+ <body>
+
+ </body>
+ </html>
```

12. Let's create our commit to add the index.html file to our repository.
13. In the bottom commit section, there is a field where you can add what we call a commit message. GitHub Desktop provides a text in the input field saying "Create index.html." Just leave it as is. Click "Commit to main" to create a commit on your main Branch.



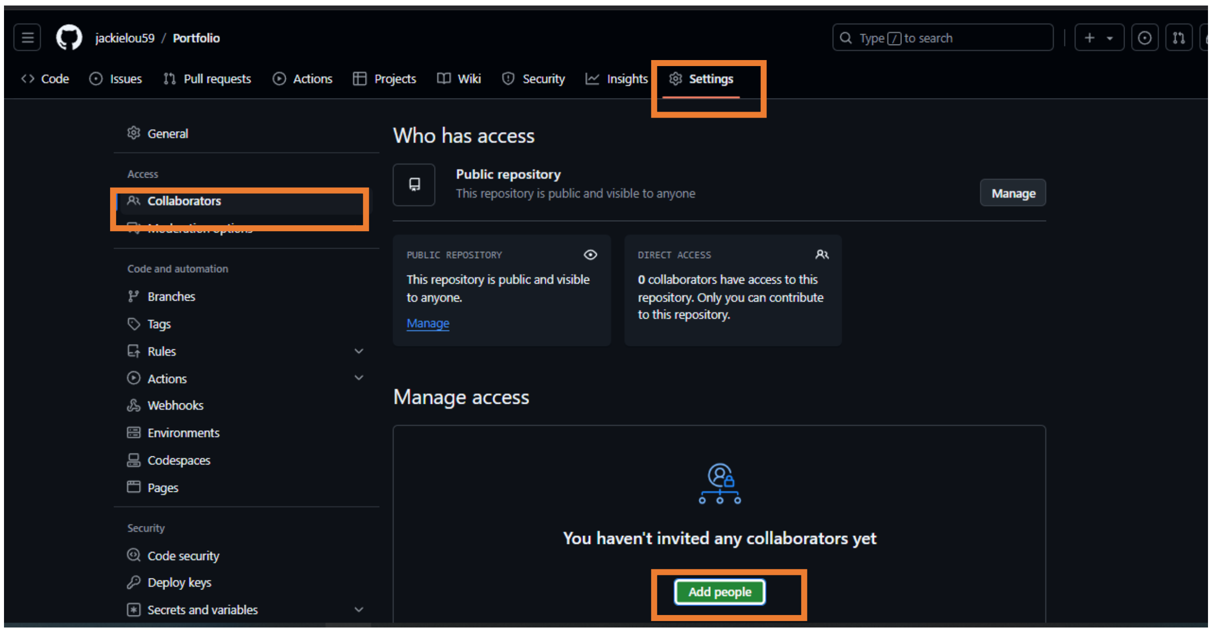
```
@@ -0,0 +1,11 @@
+ <!DOCTYPE html>
+ <html lang="en">
+ <head>
+     <meta charset="UTF-8">
+     <meta name="viewport" content="width=device-width, initial-scale=1.0">
+     <title>Mountain Province Tourism</title>
+ </head>
+ <body>
+
+ </body>
+ </html>
```

14. Since we created a repository locally, it won't exist on GitHub until we publish it. Click Publish repository, then uncheck the Keep this private. Then hit the Publish repository button. Wait for a few seconds.



Adding collaborators (Team members)

1. Assign someone to be the original owner of your project. Then, invite teammates. Do the following to add collaborators or teammates. Make sure your repository is public.
2. Navigate Settings > Collaborators > Click Add people.



Who has access

Public repository
This repository is public and visible to anyone

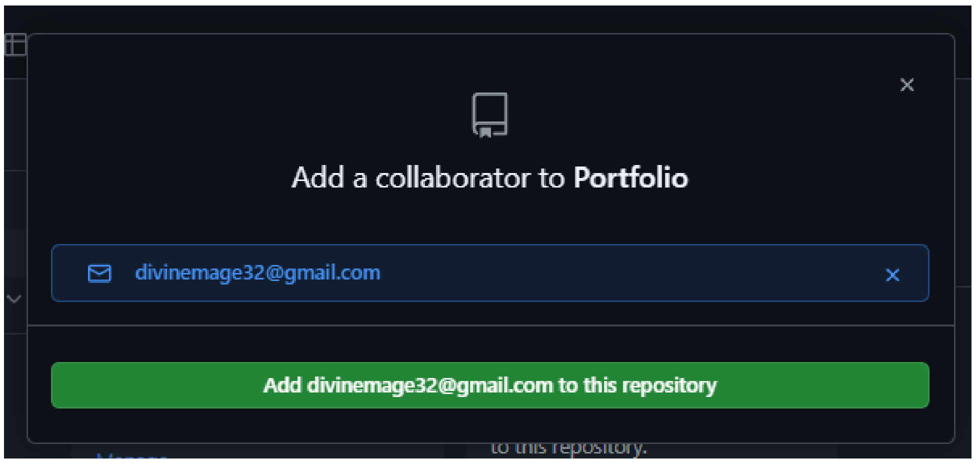
PUBLIC REPOSITORY
This repository is public and visible to anyone.

DIRECT ACCESS
0 collaborators have access to this repository. Only you can contribute to this repository.

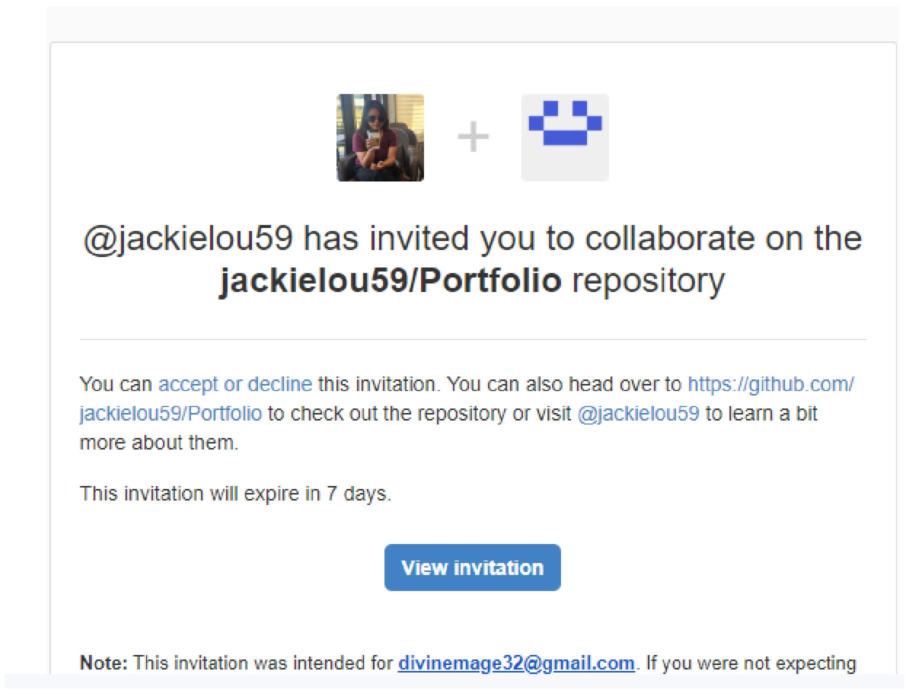
Manage access

You haven't invited any collaborators yet

3. The modal page below will appear. Add all your teammates.


Add a collaborator to **Portfolio**
divinemage32@gmail.com
Add divinemage32@gmail.com to this repository

4. After adding your teammates' accounts, an invitation link will be sent to their email. The sample invitation link is below. Click the invitation to be directed to the shared GitHub repository to edit.



@jackielou59 has invited you to collaborate on the **jackielou59/Portfolio** repository

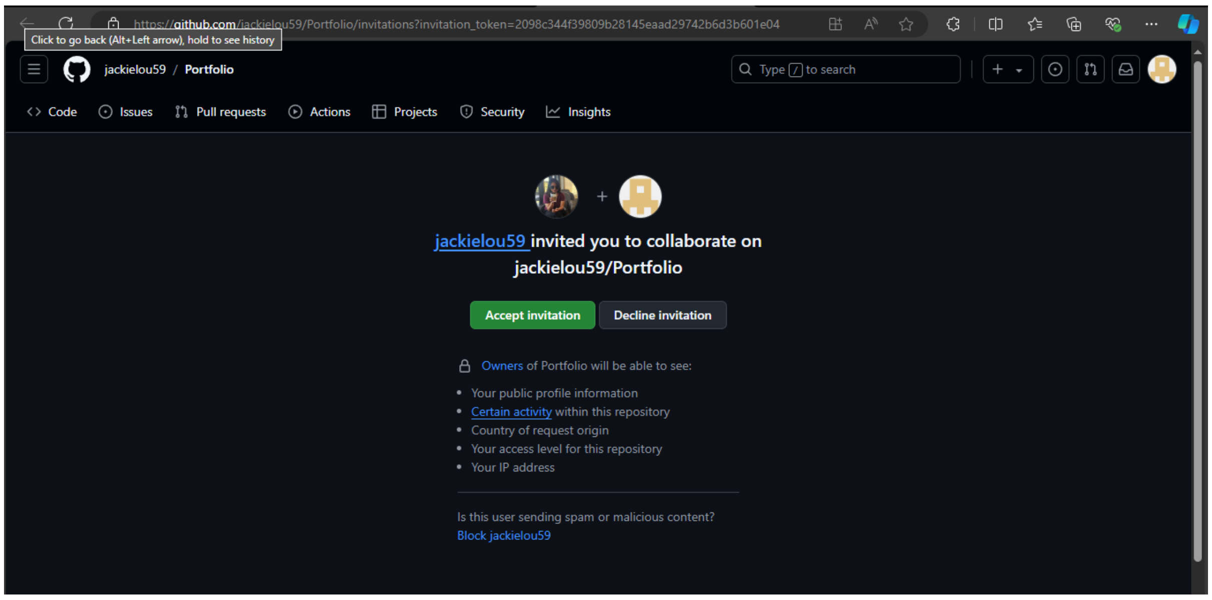
You can accept or decline this invitation. You can also head over to https://github.com/jackielou59/Portfolio to check out the repository or visit @jackielou59 to learn a bit more about them.

This invitation will expire in 7 days.

Note: This invitation was intended for divinemage32@gmail.com. If you were not expecting

Invitation Collaborators side

1. After clicking the View Invitation button, you will be directed to GitHub. Click Accept invitation.



https://github.com/jackielou59/Portfolio/invitations?invitation_token=2098c344f39809b28145eaad29742b6d3b601e04

Click to go back (Alt+Left arrow), hold to see history

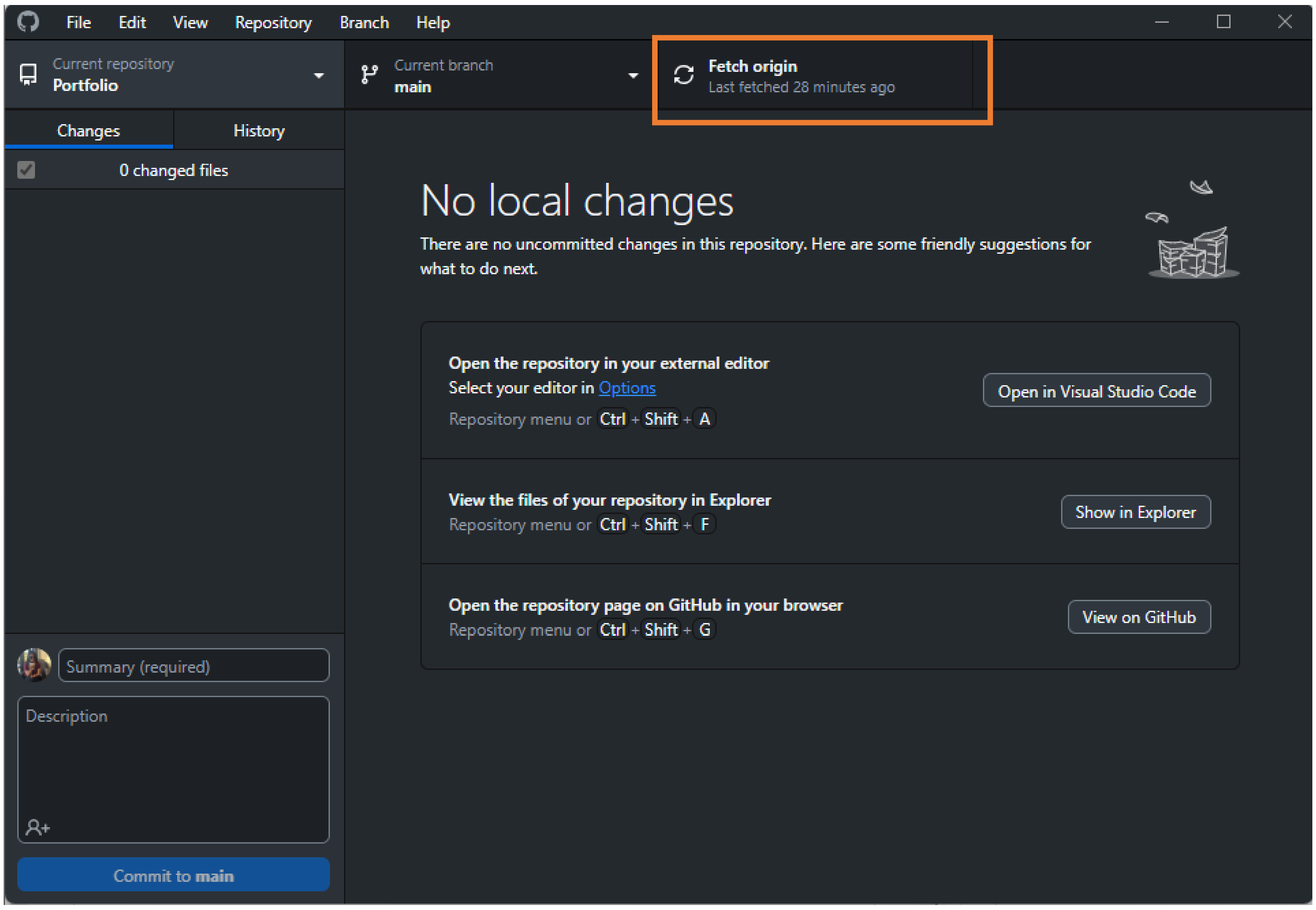
jackielou59 invited you to collaborate on jackielou59/Portfolio

Owners of Portfolio will be able to see:

- Your public profile information
- Certain activity within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

Is this user sending spam or malicious content?
Block jackielou59

2. You can now work as a team. Terms to take note of when working in GitHub are commit, push, pull, and fetch. You will encounter these as you go along with your project.
3. If your groupmates made changes, it will be reflected in Git. So, if you are going to edit the same page in GitHub Desktop, make sure to click Fetch Origin so you can have the latest Code before you open it to VS Code.



Fetch origin
Last fetched 28 minutes ago

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Open the repository in your external editor
Select your editor in Options
Repository menu or Ctrl + Shift + A

View the files of your repository in Explorer
Repository menu or Ctrl + Shift + F

Open the repository page on GitHub in your browser
Repository menu or Ctrl + Shift + G

Or you can assign someone to work on specific page.