Publishing your Personal Project to GitHub Pages

Daily Outline:

I. **Understanding GitHub Pages.**

II. **Publishing your website in GitHub Pages.**

III. **Making changes to your website**

I. Understanding GitHub Pages

Check out the following URL to understand what GitHub Pages does.

https://www.w3schools.com/git/git_github_pages.asp

Here is another web page that gives information about GitHub Pages. Watch the video for more information about GitHub Pages.

https://pages.github.com/

1. What is GitHub Pages?

GitHub allows you to host a webpage from your repository using GitHub Pages.

2. Before we publish a webpage on GitHub Pages, here is a list of the steps involved.
   i. Back up your folder.
   ii. Open Command prompt and change to your project folder.
   iii. Initialize your local folder as a git repository.
   iv. Stage and commit your files.
   v. Log in to GitHub Pages.
   vi. Create a special repository for GitHub Pages on GitHub.
   vii. Push your local repository to the GitHub Pages repository.
   viii. Verify GitHub to see if your files are there.
   ix. Run your webpage in a new browser tab.

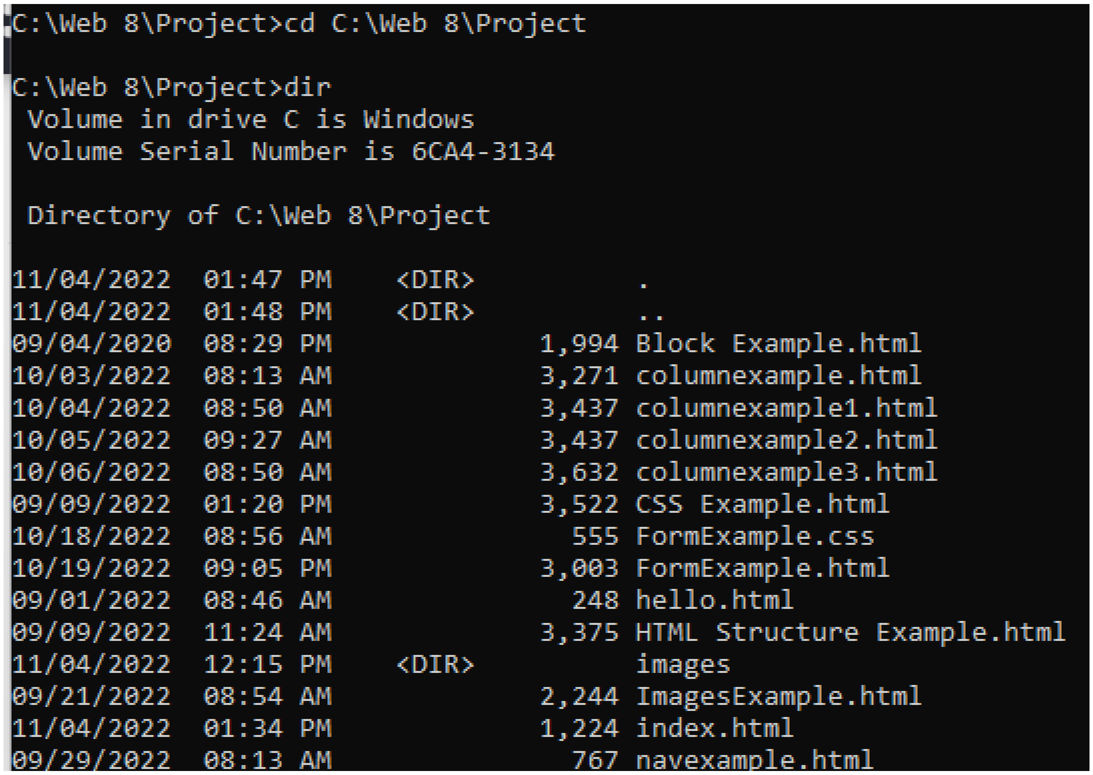
II. Publishing your website GitHub Pages

Now let’s do these steps. Here are more detailed instructions.

1. Back up your folder.
   a. When you are ready to publish your web site, make sure that your home page is named index.html
   b. Just to be safe, it is a good idea to back up your folder. You can do this by creating a zipped folder.

   c. In Windows explorer, right click on the folder name and use Send to Compressed Folder option.

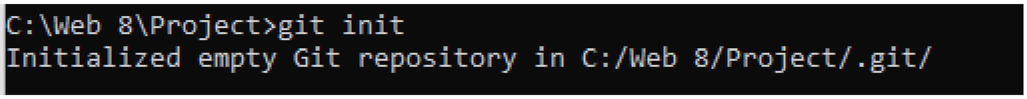
2. Open Command prompt and change to your project folder.
   a. Open the command or terminal window.
   b. Enter the following cd command to change to your project folder. Use your own full absolute path instead of the one in this example
      cd C:\Web\Project
   c. Enter the dir or ls command to see your files. Your work should look similar to the following snip.



```
C:\Web 8\Project>cd C:\Web 8\Project

C:\Web 8\Project>dir
 Volume in drive C is Windows
 Volume Serial Number is 6CA4-3134

 Directory of C:\Web 8\Project

11/04/2022  01:47 PM    <DIR>          .
11/04/2022  01:48 PM    <DIR>          ..
09/04/2020  08:29 PM             1,994 Block Example.html
10/03/2022  08:13 AM             3,271 columnexample.html
10/04/2022  08:50 AM             3,437 columnexample1.html
10/05/2022  09:27 AM             3,437 columnexample2.html
10/06/2022  08:50 AM             3,632 columnexample3.html
09/09/2022  01:20 PM             3,522 CSS Example.html
10/18/2022  08:56 AM               555 FormExample.css
10/19/2022  09:05 PM             3,003 FormExample.html
09/01/2022  08:46 AM               248 hello.html
09/09/2022  11:24 AM             3,375 HTML Structure Example.html
11/04/2022  12:15 PM    <DIR>          images
09/21/2022  08:54 AM             2,244 ImagesExample.html
11/04/2022  01:34 PM             1,224 index.html
09/29/2022  08:13 AM               767 navexample.html
```

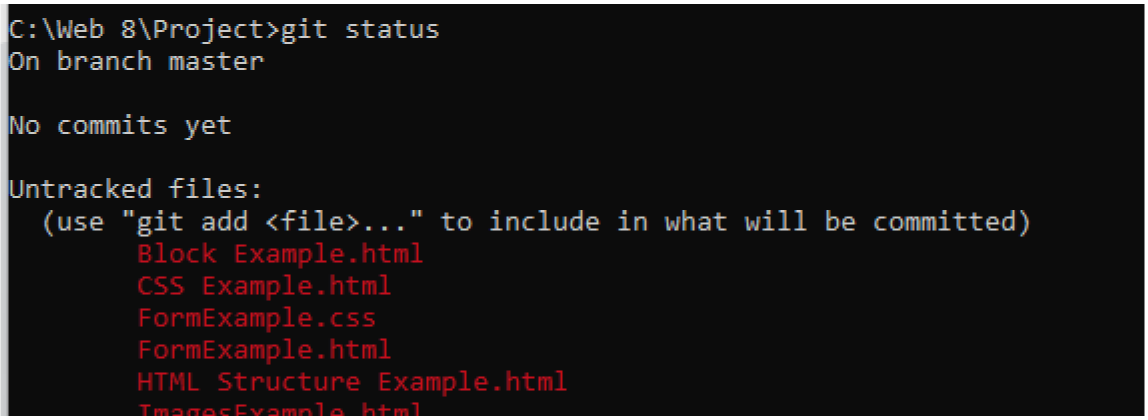
   d. All commands should be done in this directory, that is in the folder of your project. If you leave this window and come back into it later, you must do the cd command to change to your project folder.

3. Initialize your local folder as a git repository. Use the command: git init
   This makes your folder a git repository. However, at this point, your files are not being tracked yet.



```
C:\Web 8\Project>git init
Initialized empty Git repository in C:/Web 8/Project/.git/
```

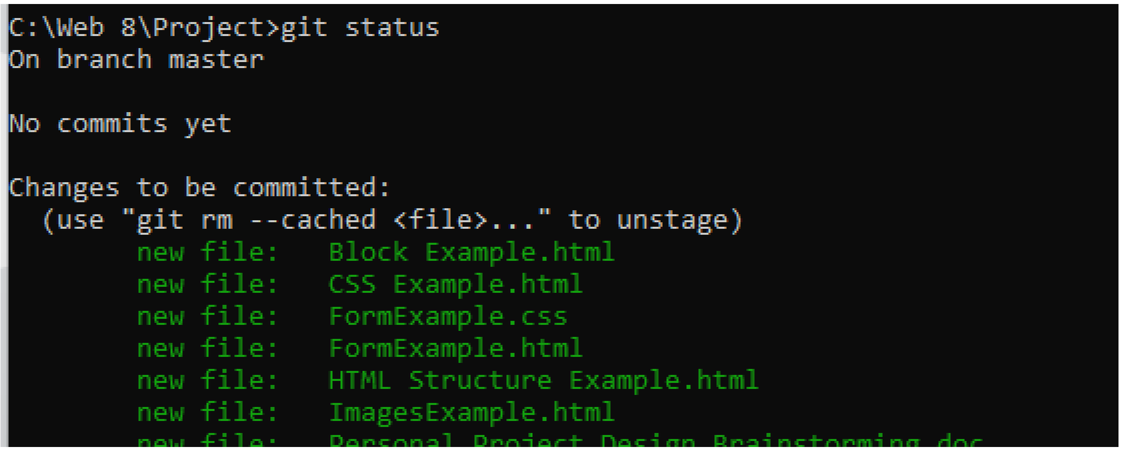
   a. Your repository isn’t tracking any files yet. Run the command git status.
   You will see that nothing is committed, and your files are listed as untracked.



```
C:\Web 8\Project>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Block Example.html
        CSS Example.html
        FormExample.css
        FormExample.html
        HTML Structure Example.html
        ImagesExample.html
```

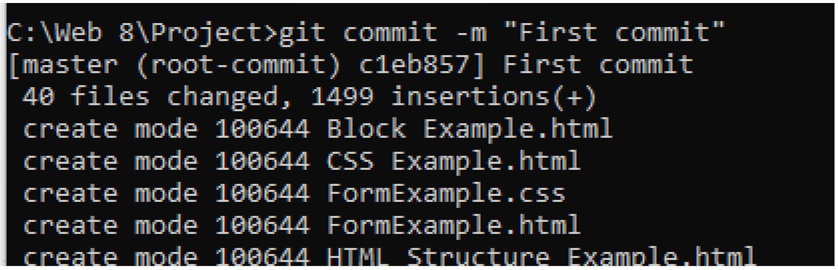
4. Stage and commit your files.
   a. Add your files to the staging environment of your git repository.
   Use the command: git add --all

   b. Check the status again and you will notice that the files are listed as new files in the repository.



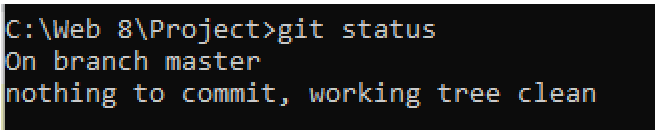
```
C:\Web 8\Project>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   Block Example.html
        new file:   CSS Example.html
        new file:   FormExample.css
        new file:   FormExample.html
        new file:   HTML Structure Example.html
        new file:   ImagesExample.html
        new file:   Personal Project Design Brainstorming.doc
```

   c. Use the commit command to your files to the staging environment of your git repository. Use the -m option to add a message as a comment.
   Use the command: `git commit -m "First commit"`



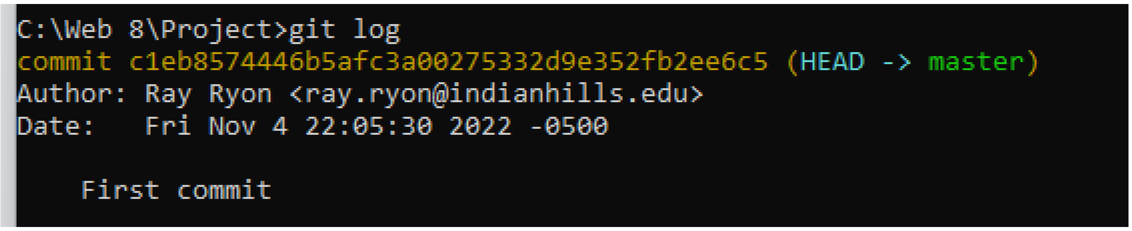
```
C:\Web 8\Project>git commit -m "First commit"
[master (root-commit) c1eb857] First commit
 40 files changed, 1499 insertions(+)
 create mode 100644 Block Example.html
 create mode 100644 CSS Example.html
 create mode 100644 FormExample.css
 create mode 100644 FormExample.html
 create mode 100644 HTML Structure Example.html
```

   d. Check the status and everything looks good.



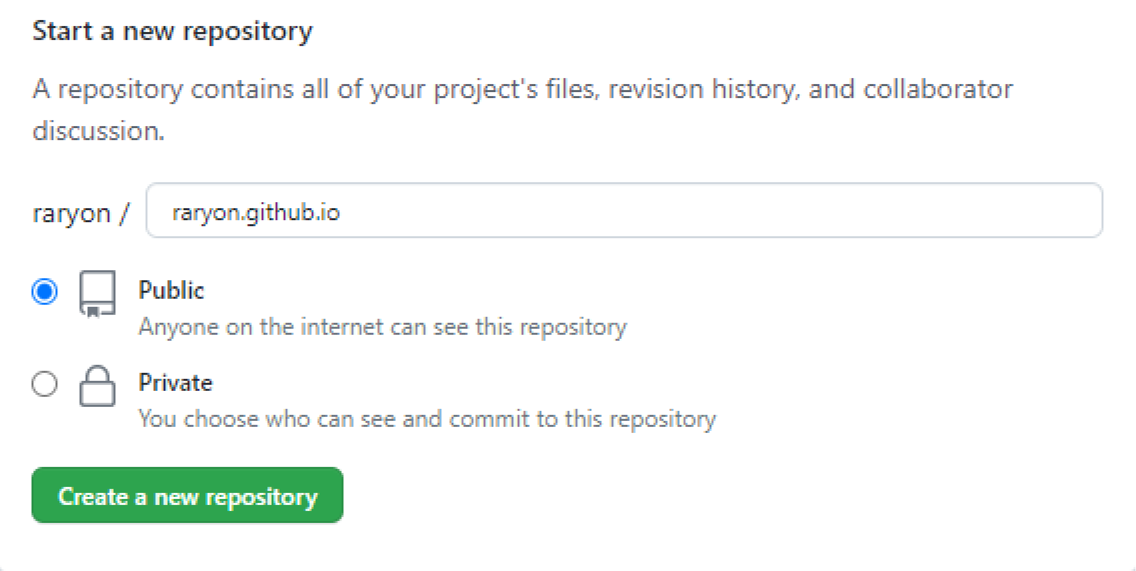
```
C:\Web 8\Project>git status
On branch master
nothing to commit, working tree clean
```

e. Another helpful command at this point is git log.



```
C:\Web 8\Project>git log
commit c1eb8574446b5afc3a00275332d9e352fb2ee6c5 (HEAD -> master)
Author: Ray Ryon <ray.ryon@indianhills.edu>
Date:   Fri Nov 4 22:05:30 2022 -0500

    First commit
```

5. Log in to GitHub. You can use the URL https://github.com/ and use the sign in link at the upper right corner.
6. Create a special repository for GitHub Pages on GitHub.
   a. **Important!** The name of this repo needs to start with your GitHub username and followed by .github.io
   b. It should look something like this: raryon.github.io
   c. Use the public option to allow others to see your website on the Internet.



Start a new repository

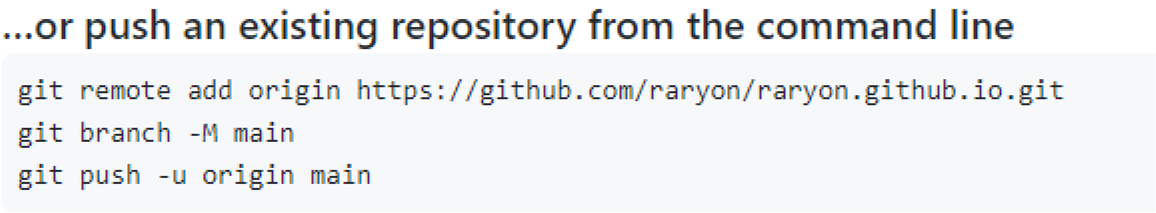
A repository contains all of your project's files, revision history, and collaborator discussion.

raryon / raryon.github.io

● Public
Anyone on the internet can see this repository

○ Private
You choose who can see and commit to this repository

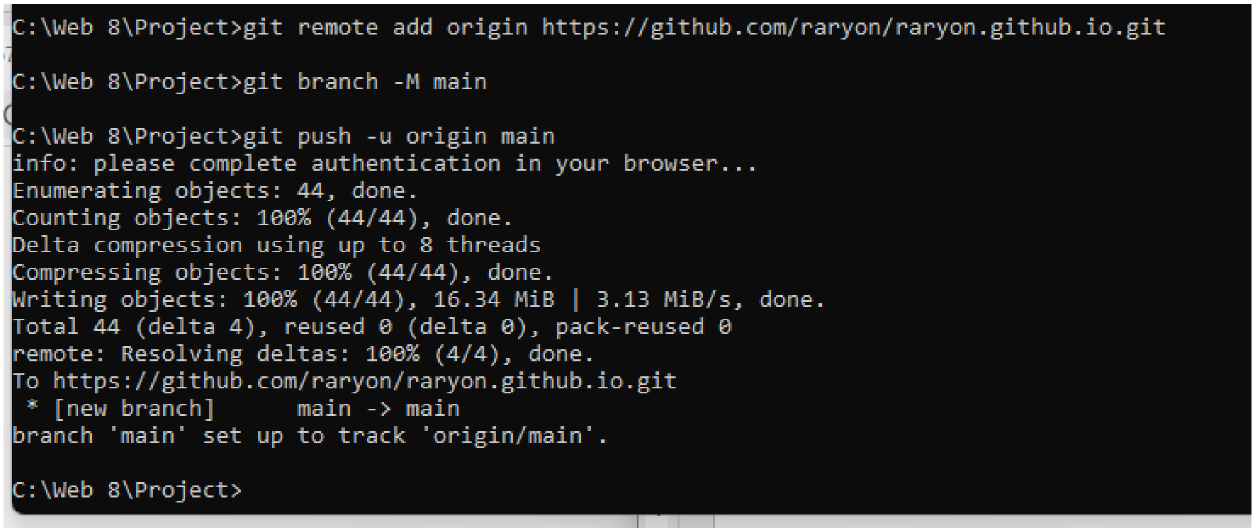
7. Push your local repository to the GitHub Pages repository.
   a. You should see a screen that gives you instructions on how to push your local repo.



…or push an existing repository from the command line

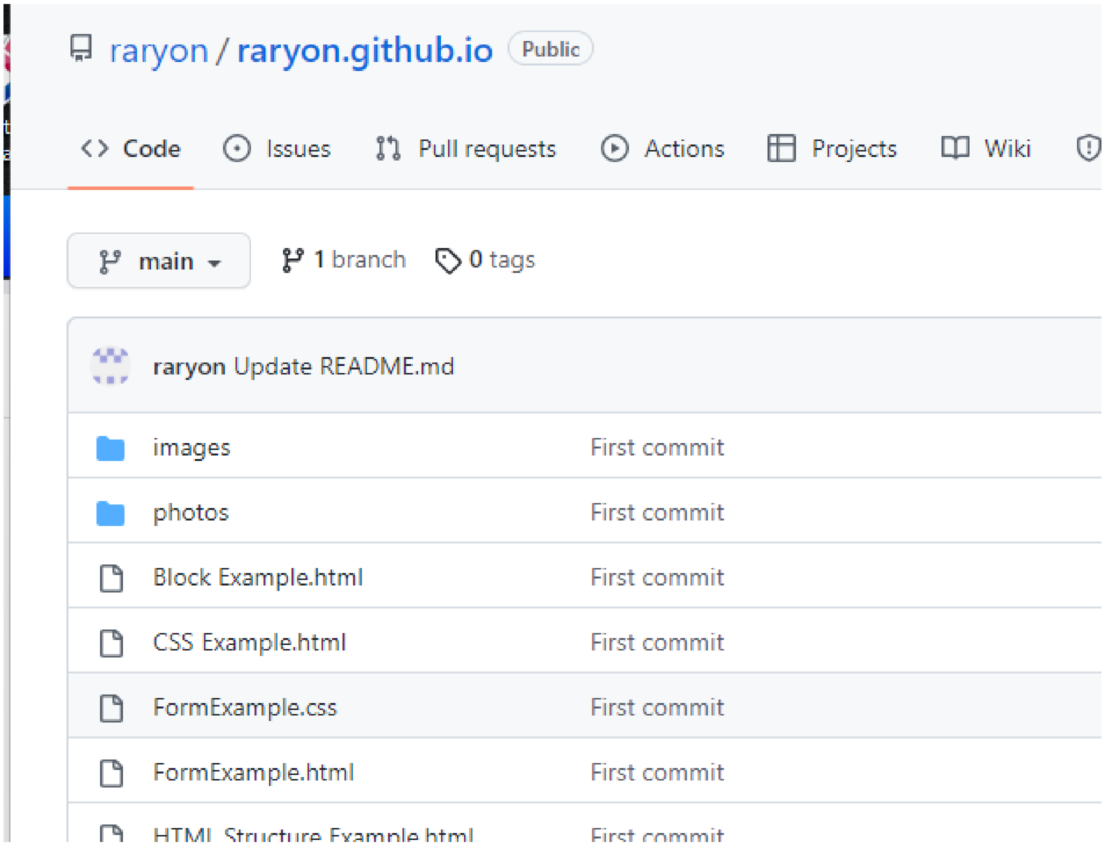
```
git remote add origin https://github.com/raryon/raryon.github.io.git
git branch -M main
git push -u origin main
```

b. You can copy and paste the commands into the command window.
c. You will be asked to authorize with Git Credential Manager.
d. Check your command window to see when it is done pushing.



```
C:\Web 8\Project>git remote add origin https://github.com/raryon/raryon.github.io.git

C:\Web 8\Project>git branch -M main

C:\Web 8\Project>git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 44, done.
Counting objects: 100% (44/44), done.
Delta compression using up to 8 threads
Compressing objects: 100% (44/44), done.
Writing objects: 100% (44/44), 16.34 MiB | 3.13 MiB/s, done.
Total 44 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), done.
To https://github.com/raryon/raryon.github.io.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

C:\Web 8\Project>
```

8. Verify GitHub to see if your files are there.
   a. Click on your repository name to see your repository files and other information.



raryon Update README.md

images	First commit
photos	First commit
Block Example.html	First commit
CSS Example.html	First commit
FormExample.css	First commit
FormExample.html	First commit
HTML Structure Example.html	First commit

9. Run your webpage in a new browser tab.
   a. Your URL will look similar to the following, but with your userid.

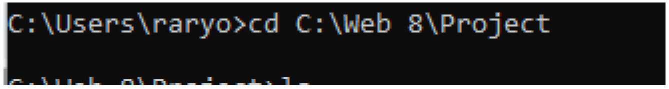
      https://raryon.github.io/

   b. Also, on the GitHub page it will show Environments on the right column. Click on the github-pages link, and then click on the view deploy button on the right side of the page.

III. Making changes to your website

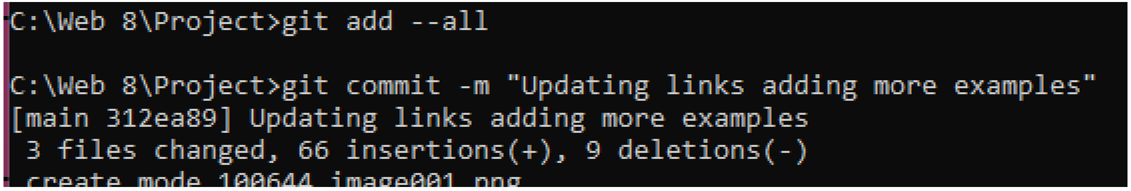
After you publish your web site you will want to make changes. Follow the following steps.

1. Make your changes in your project folder.
2. Open Command prompt and change to your project folder.



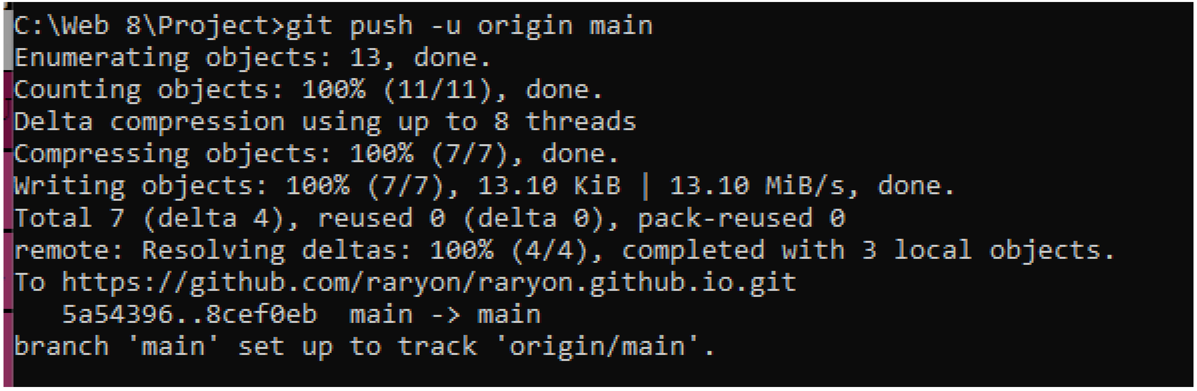
```
C:\Users\raryo>cd C:\Web 8\Project

C:\Web 8\Project>ls
```

3. Stage and commit your files.



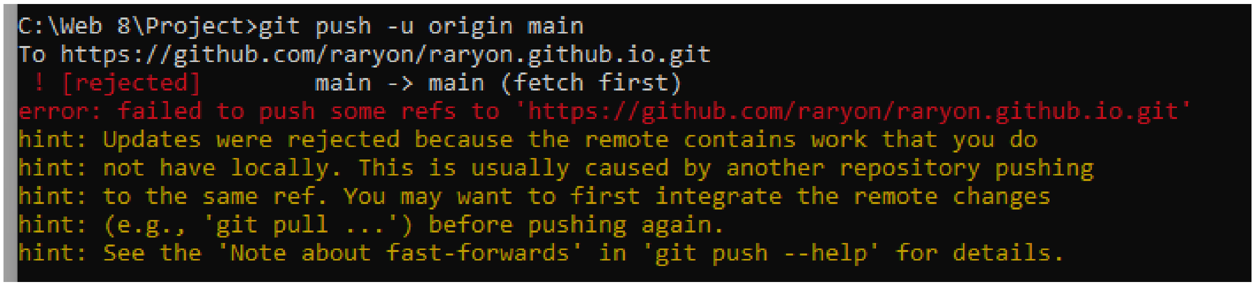
```
C:\Web 8\Project>git add --all

C:\Web 8\Project>git commit -m "Updating links adding more examples"
[main 312ea89] Updating links adding more examples
 3 files changed, 66 insertions(+), 9 deletions(-)
 create mode 100644 image001.png
```

4. Push your local repository to the GitHub Pages repository.



```
C:\Web 8\Project>git push -u origin main
Enumerating objects: 13, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 13.10 KiB | 13.10 MiB/s, done.
Total 7 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 3 local objects.
To https://github.com/raryon/raryon.github.io.git
   5a54396..8cef0eb  main -> main
branch 'main' set up to track 'origin/main'.
```

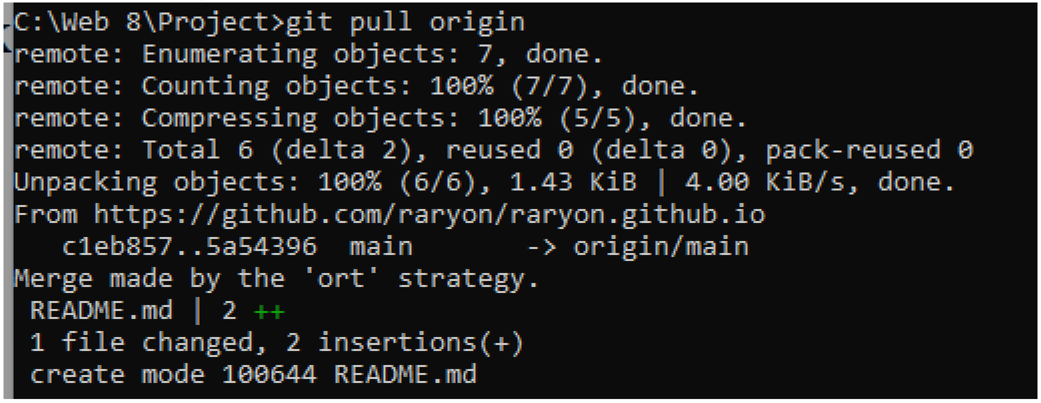
5. Note if you had made any changes through GitHub, you would get a message that the push failed because your remote doesn't match. And it suggests you do a git pull command.



```
C:\Web 8\Project>git push -u origin main
To https://github.com/raryon/raryon.github.io.git
 ! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'https://github.com/raryon/raryon.github.io.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

6. If you get this message do the command git pull origin.

Then retry the git push command.



```
C:\Web 8\Project>git pull origin
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 1.43 KiB | 4.00 KiB/s, done.
From https://github.com/raryon/raryon.github.io
   c1eb857..5a54396  main       -> origin/main
Merge made by the 'ort' strategy.
 README.md | 2 ++
 1 file changed, 2 insertions(+)
 create mode 100644 README.md
```