HOW TO GET YOUR SITE ONLINE FOR FREE WITH GITHUB PAGES

February 25th, 2015 | Get awesome (and free) stuff here (https://learntocodewith.me/newsletter/)

―――

"I've been doing Codecademy and other tutorials for some time now. **But how do I get my site online?**"

This is a question I have been asked multiple times these past few months.

I know how tough "getting a site online" can be for beginners. Especially for those who just finished a basic HTML/CSS course and are dying to get their work online.

That's why I created a quick step-by-step guide showing how to get your site online using Github Pages.

HOW TO GET YOUR SITE ON GITHUB PAGES

Step-by-step guide for getting your site on Github pages.

Enter your email address

GET THE GUIDE

Don't worry. I'll never, ever spam you!

WHY YOU SHOULD USE GITHUB PAGES

Like anything, there are many ways to get your HTML, CSS (https://learntocodewith.me/code-school-html-css), and JavaScript (JS) files online.

However, I think Github Pages is one of the best ways to do so.

- Github is important for any aspiring developer/designer to learn and use

- Developers and programmers of all kinds use Github

- Plus, setting up a site on Github Pages is free

This walk-through was created with complete beginners in mind, **including those who have never used Github.**

I will show how to take your static website files, already on your computer, and put them online as a website.

> Here's how to get your website online (for free) using Github pages. (https://twitter.com/share?text=Here%27s+how+to+get+your+website+online+%28for+free%29+using+Github+pages.&via=learncodewithme&related=learncodewithme&url=https://learntocodewith.me/tutorials/github-pages/)

E%29+USING+GITHUB+PAGES.&VIA=LEARNCODEWITHME&RELATED=LEARNCODEWITHME&URL=HTTPS://LEARNTOCODEWITH.ME/TUTORIALS/GITHUB-

REQUIRED TOOLS TO GET YOUR SITE ON GITHUB PAGES

To get started, this is what you'll need:

- **Computer/laptop:** I will be using my Macbook Pro (https://learntocodewith.me/my-macbook)

- **Text editor (https://learntocodewith.me/programming/basics/text-editors/):** I use Sublime Text 2, but there are other options

- **Basic HTML (https://learntocodewith.me/html5-cheat-sheet/) and CSS (https://learntocodewith.me/getting-started/topics/css/) know-how:** the demo files I will be using are very simple

- **Github (https://github.com/) account:** which I'll show you how to make if you don't already use Github—it's like creating any other account online

- **Command line (https://learntocodewith.me/getting-started/topics/command-line/):** AKA CLI, AKA Terminal, AKA Console. Or willingness to use it in a very easy way :)

To go along with this tutorial, **I created a video to follow (http://youtu.be/XBBDCffHSAE).**

The video starts from the absolute beginning. It is perfect to watch if you are especially new to HTML and CSS.

8 STEPS TO GETTING YOUR SITE LIVE ON GITHUB PAGES

First, here is the link with more information about Github Pages and setting one up: https://pages.github.com/ (https://pages.github.com/)

STEP 1: SIGN UP FOR / LOGIN TO GITHUB ACCOUNT

Do this at Github.com (https://github.com/).

Creating a Github account is like signing up for Facebook or Twitter. You'll give your name, username, email address, and set a password.

If you already have an account, then log in.

STEP 2: UNDERSTAND THE DIFFERENCE BETWEEN USER SITES AND PROJECT SITES

When it comes to making a Github Page, there are two options: a **project site** or a **user/organization** site.

The difference is this: you have unlimited project sites, but only one user site. My user site is **laurencebradford.github.io**. Because **laurencebradford** is my username.

An example project site would be: laurencebradford.github.io/wp-quiz/ (https://laurencebradford.github.io/wp-quiz/)

Here, **wp-quiz** is tied to the wp-quiz repository (https://github.com/laurencebradford/wp-quiz).

GITHUB PAGES

PROJECT VS. USER/ORG

unlimited in number	only **one** per account
attached to a repository	attached to username
ex: **username.github.io/repo-name**	ex: **username.github.io**

>LEARNTOCODEWITH.ME

If your head is spinning: don't worry.

Basically a repository is like a (complex) folder with a bunch of files. Different people can even edit the files within a repository at the same time. And you can create as many repositories as you'd like on Github, each with a corresponding Github page.
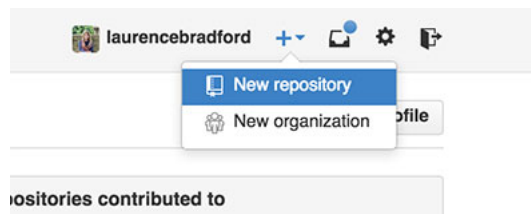
On the other hand, you're only allowed one user site. So if your username is chunkymonkey—your Github user site will be **chunkymonkey.github.io.**

For this run-through, I will be setting up a user site. Meaning the steps I follow are a bit different than those for a project site.
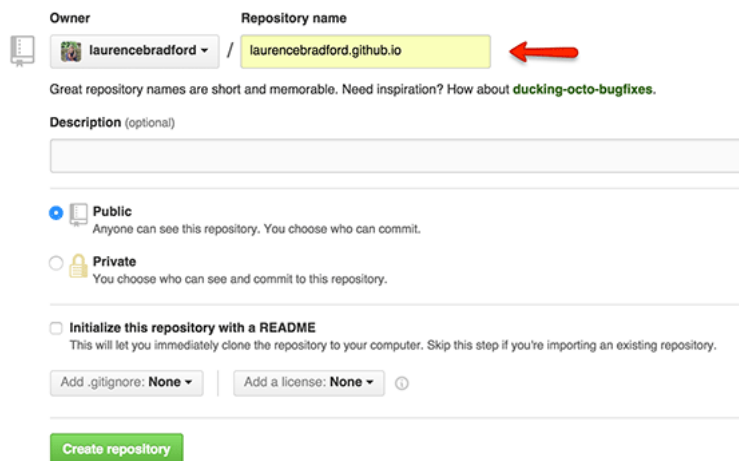
When deciding which to create, consider your goals. And remember you only have one user site. (But you can always change the contents of the site you setup now, if you follow along, in the future.)

STEP 3: CREATE A NEW REPOSITORY

In your Github dashboard, click "**new repository**" as shown below.

Then, and this is very important, make sure the repository name is your *username.github.io.* As pictured below.

Of course, **make sure to input your username**—not mine. So, if your username is chunkymonkey—it'd be **chunkymonkey.github.io**
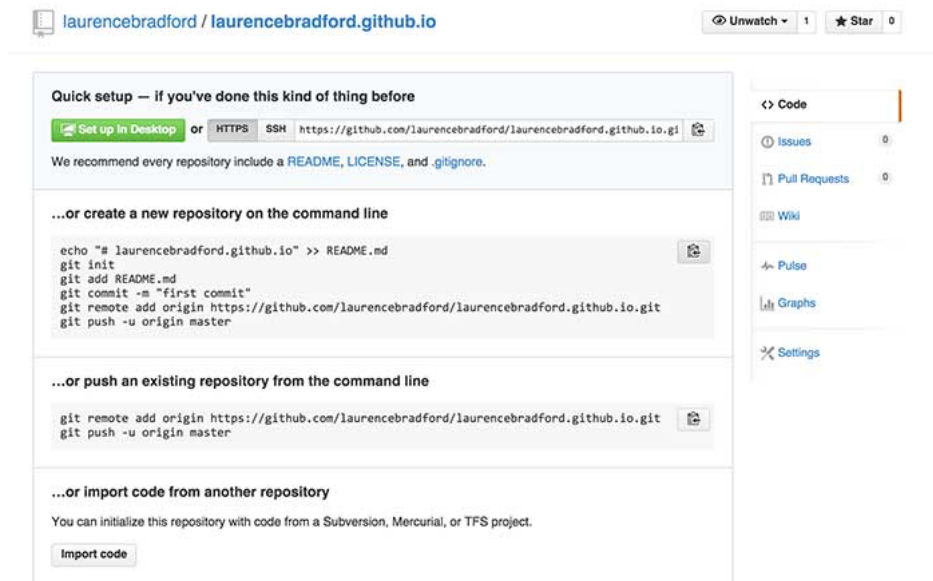
Feel free to add a description or not. (Basically, it's just a description of your repository.) And, unless you want to pay monthly, the repository must be "**public**".

I'm **not initializing** with a README. So if you're following along—**don't either.**

Finally, click the green "**create repository**" button.

STEP 4: NEW REPOSITORY PAGE EXPLANATION

Immediately after clicking the"**create repository**" button, you will see a page that looks like this.

This page is explaining how to get your files (your HTML (https://learntocodewith.me/html5-cheat-sheet/) and CSS) into the repository. (Again, think about it like a folder attached to your Github account.)

You are basically putting the files that are on your local computer into a folder on Github using git.

Now, if you're brand new, all of this may look scary. But fear not.

The directions above are showing how to get files on your computer onto Github using your terminal. AKA the command line. (If you're on a Mac, click here to see how to open your terminal (https://learntocodewith.me/command-line/open-command-line/).)

Don't worry about learning the command line right now if you're just starting out. However, I highly recommend getting comfortable in it. And sooner is better than later.

As you'll notice in the official direction set (https://pages.github.com/), there are a few ways to get your files onto Github. Meaning you don't have to use the CLI.

But, in my opinion, **using the terminal is the fastest and easiest way.** Plus, it's good practice. Because using the terminal is an important part of being a developer.

So that's what I'll be doing. And because I am on a Mac, I will be following instructions for the Mac terminal.

NOTE TO WINDOWS USERS:

The terminal is different from that on a Mac. I am not familiar with the Windows terminal, meaning I am not 100% certain the following steps will work if you are on a Windows machine. But, according to what I've seen online and the official Github documentation, I *think* it should work. At the bottom of this article I have additional resources for Windows users.
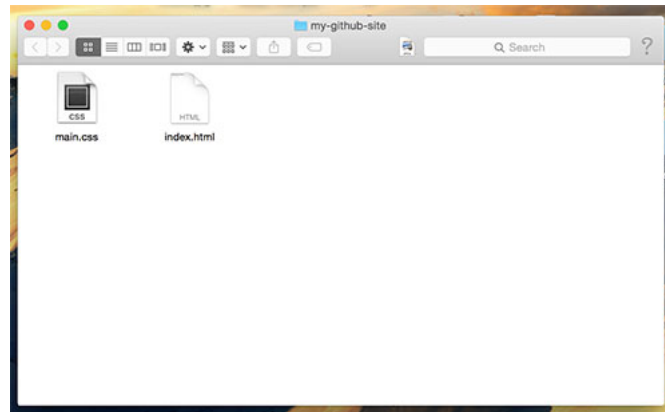
STEP 5: HAVE YOUR SITE FOLDER EASY TO ACCESS

Have your folder with your website contents ready to go. (It can be named anything.)

I have mine saved on the desktop of my computer.

The contents of my folder are very simple. Just an **index.html** file and a **main.css**.

IMPORTANT: In order for your website to work on Github pages, it is very important that your primary HTML file be named **index.html**. So, if you have it saved as something else, rename it to **index.html**. If you have other pages—like an about.html—that's fine. As long as the main is named **index.html**.

STEP 6: OPEN YOUR TERMINAL AND GO TO PROJECT LOCATION

Open your terminal (https://learntocodewith.me/command-line/open-command-line/). (AKA the command line.)

Go to the folder where the project is stored on your computer.

(**Pro tip:** in a new terminal window, type " `cd` " and then simply drag your folder into the terminal. Press enter.)

If you are having trouble opening your command line on your Mac computer, or getting to the folder location, please turn to the video I created (http://youtu.be/XBBDCffHSAE?t=7m25s).

STEP 7: GIT INIT AND PUSH TO GITHUB

These are the steps I followed once getting into the appropriate folder, where my website files are located.

1. `git init` — initializes git in your folder on your computer

2. `git add .` — adds all the files in the folder, the period is important

3. `git commit -m "first commit"` — committing to Github with a

message of "first commit"

4. `git remote add origin`
   `https://github.com/laurencebradford/laurencebradford.github.io.git`
   (https://github.com/laurencebradford/laurencebradford.github.io.git)
   — this command is only done when adding a new repository to
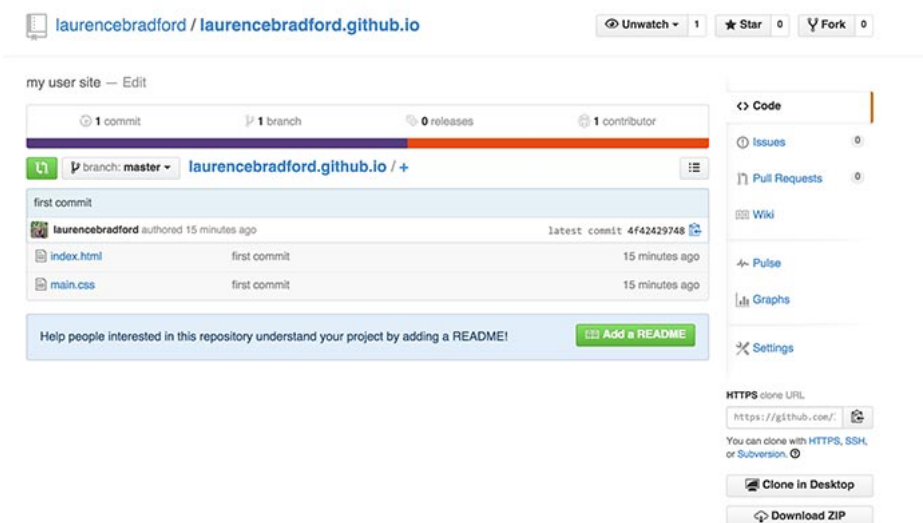   your folder. So this is only done this first time, not when making edits
   in the future.

5. `git push -u origin master` — pushes changes to your Github
   repository, in the "master" branch.

As always, there are multiple ways to do one thing.

Nonetheless, after you push your changes, **refresh your page,** please.
(The last we were on in step four.)
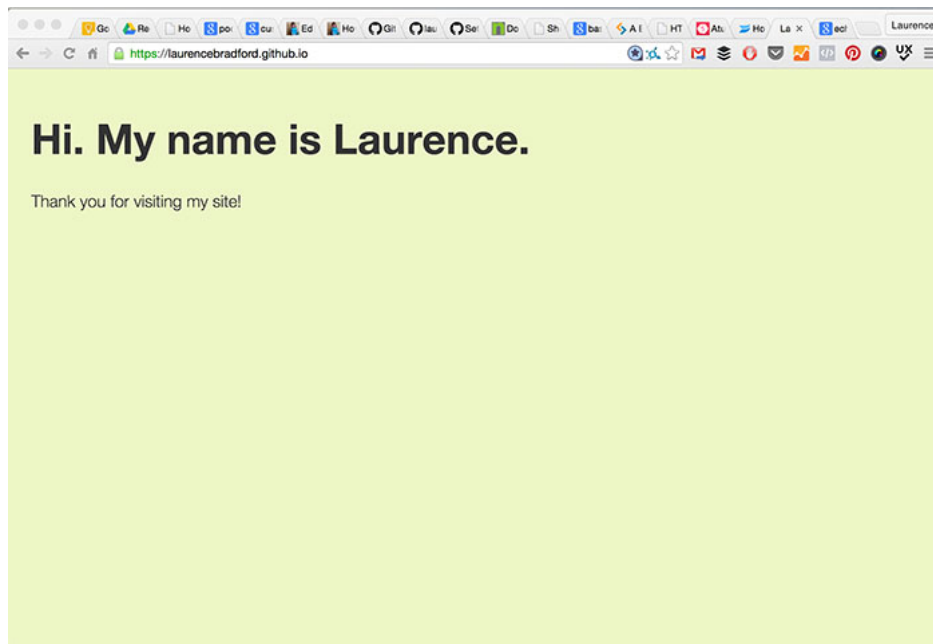
STEP 8: MAKE SURE IT WORKED

After refreshing the page, something like this should appear.

As you will see, the contents of your project folder will be there. For me,
only **index.html** and **main.css**.

Now, head over to the URL **username.github.io**. (Where username =
your Github username.)

This is what my site looks like. (Exactly how it looked locally on my
computer.)

And, that's it folks. You officially have a site online.

HOW TO GET YOUR SITE ON GITHUB PAGES

Step-by-step guide for getting your site on Github pages.

| Enter your email address | GET THE GUIDE |

Don't worry. I'll never, ever spam you!

QUESTIONS YOU MAY ENCOUNTER WHEN USING GITHUB PAGES

WHAT ABOUT MAKING CHANGES IN THE FUTURE TO GITHUB PAGES?

With Github Pages, it's easy to make changes to your site.

Simply,

1. Edit the files locally on your computer, in your favorite text editor

2. `git add` the files on your terminal, those you want to stage to be

committed

3. `git commit` your changes with a message

4. And `git push` to your master branch

I made another video (separate from the first) that shows **how** to make changes to your website files and then bring them onto your live Github Page.

Click here to watch the video. (http://youtu.be/xmjXxJww9QU)

WHAT ABOUT USING A CUSTOM DOMAIN NAME WITH GITHUB PAGES?

You can attach a custom domain name (yourdomain.com (http://yourdomain.com)) to your Github page. It is possible to do this for both project sites and user sites.

Here are directions from Github on how to set this up. (https://help.github.com/articles/setting-up-a-custom-domain-with-github-pages/) Of course, you must buy a domain name first. (My favorite place to buy domain names is name.com (https://www.name.com/).)

WHAT ARE MY OTHER OPTIONS, BEYOND GITHUB PAGES?

Don't want to use Github Pages? Got it.

Here are a few other ways to get your site (namely HTML, CSS and JS files) online.

1. AMAZON WEB SERVICES (AWS) (HTTP://AWS.AMAZON.COM/) S3

Here's a tutorial: http://davidwalsh.name/hosting-website-amazon-s3 (http://davidwalsh.name/hosting-website-amazon-s3). Here's information from AWS (http://aws.amazon.com/s3/), too.

2. BIT BALLOON (HTTPS://WWW.BITBALLOON.COM/)

Here's a video explanation thanks to General Assembly (https://generalassembly.wistia.com/medias/qzig8mp4mv) on getting a site up on Bit Balloon.

3. NEOCITIES (HTTPS://NEOCITIES.ORG/)

It was tough finding a video tutorial, but here is an article about the service (http://www.creativebloq.com/design/neocities-launches-widen-website-creation-7135540). And this post (http://web.appstorm.net/reviews/web-dev/neocities-the-free-place-to-code-your-own-site-from-scratch/) shows you how to get going.

WHERE CAN I LEARN MORE ABOUT GIT, GITHUB, AND RELATED?

Here are some online courses that cover topics mentioned above, so you can learn more.

- Git Basics course (https://learntocodewith.me/git-basics-treehouse) on Team Treehouse

- Console Foundations (https://learntocodewith.me/console-foundations-treehouse) on Team Treehouse

- Github for Web Designers (https://learntocodewith.me/github-web-desginers-lynda) on Lynda.com (http://Lynda.com)

CONCLUSION

There you have it: an easy eight-step process to getting your site live on Github Pages.

Whether you're a more experienced web designer/developer or brand new, I encourage you to give Github Pages a try.

And then report back here, in the comments section below, with what you think.

UPDATE: FEBRUARY 27, 2015

After receiving some questions via email, I wanted to clarify a few things and add some additional resources.

THIS TUTORIAL DEMONSTRATES SETTING UP GITHUB PAGES ON A MAC COMPUTER.

I wish I had a Windows computer I could access (with screencast equipment) so I could do another demo.

However, based on research that I did, I believe the above steps *should* work.

Anyways, here are some helpful resources if you are a Windows user:

1) Setting up Git on Windows (https://help.github.com/articles/set-up-git/#platform-windows) (official Github documentation)

2) Github for Windows (https://windows.github.com/) (official Github Client for Windows)

3) "How to Setup Github Pages – Website on Github – Complete Beginner Tutorial" (https://www.youtube.com/watch?v=FiOgz3nKpgk) (video on YouTube where a Windows computer is being used in demonstration)

"IF MY REPOSITORY IS PUBLIC, CAN PEOPLE EDIT MY FILES?"

This is another question I received.

There's a lot that goes into this. And if you learn more about Github by taking a course (https://training.github.com/) or simply by experimenting, it'll make more sense.

But, basically: **no**. A random person cannot edit your files so that they will appear differently in your repository, and hence on your **username.github.io** site.

But, when you make your files public on Github, **anyone can view them**. (Even without a Github Account.) And other Github users can "fork (https://help.github.com/articles/fork-a-repo/)" your public repositories. Which is like a copy and paste.

People typically fork repositories if they want to either (a) build off your files for their own project or (b) to suggest changes.

I feel like I am guilty of "forking and forgetting". Meaning I fork a repository, plan on building off of it or something…and then never do because I get distracted.

Anyways—the point is: **DO NOT PUT ANYTHING UP ON GITHUB IN A PUBLIC REPOSITORY THAT YOU DO NOT WANT OTHERS TO SEE**.

Sorry to yell. But the last thing I want is someone coming back to me because they put ____ in their repository and didn't want anyone seeing…

Examples of this include but are not limited to:

- Sensitive information (addresses, passwords, credit card numbers)

- Images (that you do not want anyone seeing)

- Some idea you are working on (that you don't want anyone to see)

- And so on…

"WHAT CAN I USE GITHUB PAGES FOR?"

AKA—what kinds of sites.

With Github pages, you can only serve static assets (HTML, CSS, JS and images). You **cannot** set up a WordPress (https://learntocodewith.me/getting-started/topics/wordpress/) site, a RoR application, a Meteor.js app, and so forth.

But you *can* use Github's Jekyll (http://jekyllrb.com/) (static site/blog generator), which a lot of people use and love.

For more on Github Pages uses, etc.—look to the comment thread (https://learntocodewith.me/tutorials/github-pages/#comment-1874941530) below.

And if you have any more questions, or thoughts, on using Github Pages, I'd love to hear them. Leave a comment below!

Note: there are affiliate links (https://learntocodewith.me/affiliate-disclaimer) in this post.

COMMAND LINE (HTTPS://LEARNTOCODEWITH.ME/TAG/COMMAND-LINE-2/) GITHUB (HTTPS://LEARNTOCODEWITH.ME/TAG/GITHUB/)

Tweet

LAURENCE BRADFORD

Hey, my name is Laurence. I am the creator of Learn to Code With Me, a blog for beginners teaching themselves how to code. Once upon a time, I was digitally unsavvy. *But not anymore.* Find out how I changed my life by teaching myself digital skills here (https://learntocodewith.me/about/).

Read Next ...

BECOME A WEB DEVELOPMENT NINJA WITH CODE SCHOOL (HTTPS://LEARNTOCODEWITH.ME/REVIEWS/WEB-DEVELOPMENT-CODE-SCHOOL/)

READ MORE » (HTTPS://LEARNTOCODEWITH.ME/REVIEWS/WEB-DEVELOPMENT-CODE-SCHOOL/)

Most Talked About Posts

1 Do You Have What it Takes to be a Front End Developer? (https://learntocodewith.me/webdev/front-end-developer-skills/)

2 70 of The Best Places to Learn to Code For Free (https://learntocodewith.me/post for-free/)

3 How to Develop an Android App (5 Things to Know Before Starting) (https://learntocodewith.me/prog app-development/)

4 The Most Epic Guide to Online Coding Bootcamps, Ever (https://learntocodewith.me/post coding-bootcamps/)

5   What Should I Learn as a Beginner: Python 2 or Python 3? (https://learntocodewith.me/prog 2-vs-python-3/)

THE SIMPLE 4-PART FRAMEWORK FOR LANDING TECHNICAL SIDE GIGS

You can **make money while learning to code,** even if you're just starting out.

WATCH IT NOW » (HTTPS://LEARNTOCODEWITH.ME/SIDEGIGTRAINING/?UTM_SOURCE=BLOG-POST-I

20 Comments   **Learn to Code With Me**   🔒 **Disqus' Privacy Policy**   1 **Login** ⌄

♡ **Recommend** 4   🐦 **Tweet**   f **Share**   Sort by Best ⌄

Join the discussion…

LOG IN WITH   OR SIGN UP WITH DISQUS ⑦

Name

This comment was deleted.

Laurence Bradford **Mod** ➜ Guest • 5 years ago

Awesome thanks!

⌃ | ⌄ • **Reply** • **Share ›**

gunti samyuktha • 4 months ago

I can't ignore the added security that a static site offers. Also, appreciate the Ink for All program's Markdown export capability. I used to rely on WordPress for my company site, just I switched to a static site. I always felt that wp was

1 ⌃ | ⌄ • **Reply** • **Share ›**

Alam • 2 years ago

Really helpful
Thank you.

1 ⌃ | ⌄ • **Reply** • **Share ›**

Laurence Bradford **Mod** ➜ Alam • 2 years ago

So glad you liked it!

⌃ | ⌄ • **Reply** • **Share ›**

absamin diacatra • a year ago • edited

I created one site(WordPress with simply Static plugin) and push it to gh-pages too. after few days, i notice that there's some images lost. I lost my backup "wordpress local site" for this. so I searched how to edit a static site, but the codes are different from the video tutorial. mind is more on twitter code but the tutorial used a simple themes. is there anyway i can return it to my WordPress again so that i can edit it.
Tnx…
my contact won't work as well. what plugin i can use.

H E L P

∧ | ∨ • Reply • Share ›

Prakash • 2 years ago

I don't want my site to be ending with github.io..i want to have the site named mysitename.com..is this possible here?

∧ | ∨ • Reply • Share ›

absamin diacatra → Prakash • a year ago

YES.

∧ | ∨ • Reply • Share ›

Aditya Sharma • 2 years ago

I have created my github user site. It works like a charm. I am able to access the webpage. But the only problem I am facing is that my website is not indexed in google search. I cannot find it on google even when I search with the exact name. I thought it would take time but it's still not visible after 3 weeks. Any specific reason or method to get the website live?

∧ | ∨ • Reply • Share ›

absamin diacatra → Aditya Sharma • a year ago

Google Search Console, and Bing too. I can do that...

∧ | ∨ • Reply • Share ›

Just Dee • 2 years ago

wow..
beautiful post, it was really helpful and thank you for providing such a catalyst.

∧ | ∨ • Reply • Share ›

A Webpage Online • 2 years ago • edited

Great instruction! :)

If Github (or even HTML) are intimidating, though, be sure to check us out at www.awebpage.online.

It's free, you control your webpage's link (`/whatever-you-want`), we offer Microsoft Word-like (WYSIWYG) editors, and your page goes live instantly!

∧ | ∨ • Reply • Share ›

Patrick Seiuli • 3 years ago

Just joined the blogging community! This helped a lot as I am new to github. Thanks for the great information!

∧ | ∨ • Reply • Share ›

Laurence Bradford Mod → Patrick Seiuli • 3 years ago

You're welcome, Patrick! =)

∧ | ∨ • Reply • Share ›

ian cikal • 4 years ago

thank's **@Laurence Bradford** , i got new experience from here.. :-) anyway, feel comfort reading here, simple style but not simpler.. thank's.. :-)

∧ | ∨ • Reply • Share ›

Dave • 5 years ago

Lots of great info. Thanks!

∧ | ∨ • Reply • Share ›

Laurence Bradford Mod → Dave • 5 years ago

Thank you! Glad you found it helpful :)

∧ | ∨ • Reply • Share ›

Marisa Hendrickson • 5 years ago

I've been meaning to get used to using and learning how to use Github for a while now so I really appreciate this post. I do have a question, would this be a good way to test websites before actually launching them? And a further question, could you do this with WordPress?

The reason I ask is that the company I work for and the way I've been taught to do sites recently is to have a subdomain that would be test.example.com and I'd build the site there and then move it over. But if I could do this, and not have to use up server space, it just makes me wonder if this would actually be easier to do.

-Marisa | http://marisahendrickson.com/

∧ | ∨ · Reply · Share ›

Laurence Bradford Mod → Marisa Hendrickson · 5 years ago

Hi Marisa,

As far as testing websites before launching, I think that is a perfect use case for Github Pages.

However, a big catch: it depends on the type of site you are running. If you are just doing static assets (HTML, CSS, Images and JS) -- then you can run on Github Pages.

However, you cannot host a WP site, RoR application or similar on Github Pages. (But you can use Github's http://jekyllrb.com/ -- which is kind of like CMS .. but it's built with Ruby. I've only used Jekyll a few times so I can't offer much thoughts on it.)

Another issue with Github Pages that I should mention is ... well, it's free! Meaning you only get so much server speed, etc. And if you have a site getting tons of traffic and it's hosted on Github pages -- I'd imagine it'd be quite slow.

I've only ever used a subdomain once when building a new site in the way you mentioned. By problem subdomains is that someone could stumble across the test site (unless it's PW protected, which is easy) and also it can make Google Analytics funky. (It starts to log traffic twice and messes up some of the metrics.) There's a way to add a filer to block the subdomain .. but I never figured it out. (Admittedly, didn't spend all that much time trying to!)

You could also set up a test site, or "staging url", that is unattached from the main url? So develop locally, push changes to the staging url and then once it's ok'ed over to the live site. Just an idea.

... This has become a much longer response than I thought haha but I hope some of this helped!

1 ∧ | ∨ · Reply · Share ›

Marisa Hendrickson → Laurence Bradford · 5 years ago

That actually helped a ton, thank you! I would like to really start developing static websites as well so I'm glad to know that I can use Github Pages for that.

The reason I don't develop the WP sites locally is because I have to have a marketing director or my boss give me feedback on the site and they're usually remote. I make sure that the site isn't indexed by the Google bots so I'm sure that helps somewhat but I'll admit that it's not an ideal way of doing things. One day I'm sure I'll find a better way.

1 ∧ | ∨ · Reply · Share ›

Laurence Bradford Mod → Marisa Hendrickson · 5 years ago

Okay, cool!

If you're building on WP, I have used this password plugin before: https://wordpress.org/plugi...

It's really easy to use. Visitors to the site must enter a password before seeing anything. (Once entered, they don't have to enter it again for a few days it seems.)

You can even set up certain passwords for certain people. Or just one for everyone.

1 ∧ | ∨ · Reply · Share ›

✉ Subscribe   🅓 Add Disqus to your siteAdd DisqusAdd   ⚠ Do Not Sell My Data

Still reading? Awesome. Now why don't we make our friendship official? ☺

🅕 (https://www.facebook.com/learntocodewithme) 🐦
(https://twitter.com/learncodewithme) 🅘🅝 (https://www.linkedin.com/in/laurencebradford)

Learning digital skills has changed my life for the better. **And it can do the same for you.**

LEARN TO CODE, TOO (HTTPS://LEARNTOCOD

Blog (https://learntocodewith.me/blog/) Podcast
(https://learntocodewith.me/podcast/) Top tech skills
(https://learntocodewith.me/posts/tech-skills-in-demand/) Tools
(https://learntocodewith.me/tools) About (https://learntocodewith.me/about) FAQ
(https://learntocodewith.me/faq/) Contact (https://learntocodewith.me/contact/)

built by Laurence Bradford with ❤ and ☕ | Permissions (https://learntocodewith.me/permissions-policy/), Disclaimer
(https://learntocodewith.me/affiliate-disclaimer/), Privacy Policy (https://learntocodewith.me/privacy-policy/)

learntocodewith.me (https://learntocodewith.me) | © 2014 - 2020 | Get digital skills, be happy.