Presentation Week 1 - Day 2

Introduction to GitHub Pages

Makers Institute

What is GitHub Pages?

GitHub Pages is a static site hosting service.

GitHub Pages is designed to host your personal, organization, or project pages directly from a GitHub repository. To learn more about the different types of GitHub Pages sites, see "User, organization, and project pages."

You can create and publish GitHub Pages online using the Automatic Page Generator. If you prefer to work locally, you can use GitHub Desktop or the command line.

Why You Should Use GitHub Pages

Like anything, there are many ways to get your HTML, CSS and 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

GitHub Pages Examples

microsoft.github.io

twitter.github.io

electron.atom.io

ibm.github.io

etc...

STEP 1: SIGN UP FOR / LOG IN TO GITHUB ACCOUNT

Do this at 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. Example of user site: **username.github.io**

An example project site would be: **username.github.io/repo-name/**

Here, **repo-name** is tied to the repo-name repository.

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*.

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**".

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

STEP 4: HAVE YOUR SITE FOLDER EASY TO ACCESS

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

The content of the folder are very simple. Just an **index.html** file with content "Hello World."

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 5: OPEN YOUR TERMINAL AND GO TO PROJECT LOCATION

Open your terminal. (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.)

(**Another Pro tip:** right-click the folder while pressing shift button, and then choose Open command window here.)

STEP 6: GIT INIT AND PUSH TO GITHUB

1. `git init`
2. `git add .`
3. `git commit -m "first commit"`
4. `git remote add origin https://github.com/`*`username`*`/`*`username`*`.github.io.git`
5. `git push -u origin master`

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

...WHAT ABOUT MAKING CHANGES IN THE FUTURE?

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

…WHAT ABOUT A CUSTOM DOMAIN NAME?

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

Of course, you must buy a domain name first.

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

When you make your files public on Github, anyone can view them. (Even without a Github Account.) And other Github users can "fork" your public repositories. Which is like a copy and paste.

DO NOT PUT ANYTHING UP ON GITHUB IN A PUBLIC REPOSITORY THAT YOU DO NOT WANT OTHERS TO SEE.

- 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?"

With Github pages, you can only serve static assets (HTML, CSS, JS and images). You cannot setup a WordPress site, a RoR application, a Meteor.js app, and so forth.

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

Questions?

Makers Institute