Making a website for your HTML, JS, CSS or p5.js files

We are going to make a website using something called GitHub pages to make a website. Signing up for a GitHub account and subsequently creating a GitHub pages URL will allow you to upload your HTML, JS & CSS files online, so that you can access your website from anywhere.

Bit of background (Feel free to skip)

GitHub is a website that stores what are called *repositories*. Repositories contain code, very much just like a folder on your computer. *Git* is what is called a *version control system* and to us what it means is that when you save files and *commit them* it will also save the previous version of your files too. This is very popular with a lot of people in the world of tech, because if something goes wrong, you can always go back to when it did work!

Step 1: Signing up for GitHub

Go to **https://github.com** and sign up for an account, the form should be on the homepage. There will be a few extra forms when you signup, you can skip through these.

Very important!!!1!!1!

The way that GitHub pages work is that in the end, your website will have the name of your username in it (.github.io). For instance, if my username was 'jonny' my GitHub pages URL will be https://jonny.github.io. So make sure you choose your username carefully!

Step 2: Creating a repository

Now we have to create a repository. On the left hand side there should be a link that says **Create a repository**. Click this link and it should take you to this page:

Search or jump to	7 Pull requests Issues Marketplace Explore	♣ +•
	Create a new repository A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.	
	Owner Repository name • Image: ctl-test • / Great repository names are short and memorable. Need inspiration? How about sturdy-invention? Description (optional)	
	Public Anyone can see this repository. You choose who can commit. Private You choose who can see and commit to this repository.	
	Initialize this repository with a README This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository. Add .gitignore: None Add a license: None ①	

Very important!!!1!!1!

This is where we need to set our repository name to equal exactly this structure: .github.io. So again if my username is 'jonny' in the **Repository name** field I would insert 'jonny.github.io'.

Keep the button marked as 'Public' and everything else as default and click Create Repository

Step 3: Uploading your code

Now we need to upload our files to the Github repository. Hopefully after you created your repository you should see the below page. What we want to do is click the link that says **uploading an existing file**

C Searc	h or jump to Pull requests Issues Marketplace Explore	▲ +• •
	□ ctl-test / ctl-test.github.io	ge Fork 0
	Quick setup — if you've done this kind of thing before	
	Set up in Desktop or HTTPS SSH https://github.com/ctl-test/ctl-test.github.io.git	đ
	Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .git or create a new repository on the commend echo "# ctl-test.github.io" >> README.md git init git add README.md git commit -m "first commit" git remote add origin https://github.com/ctl-test/ctl-test.github.io.git git push -u origin master	ignore.
	or push an existing repository from the command line	
	git remote add origin https://github.com/ctl-test/ctl-test.github.io.git git push -u origin master	Ē

After that, if you drag your HTML, JS and CSS files (if you have them) onto the upload section, they should upload to your repository. **BUT!** You need to 'commit the changes' for them to be saved.

Step 4: Commiting your changes

This section is also relevant if you want to update your files too. Everytime you make changes to the files, you have to 'commit them'. To the Git repository, that basically means, storing these new files, committing them to memory.

Github makes this part quite easy. If you want to add a message you can do, but if not, all you have to do is click **Commit changes** and your files will be stored in the git repository.

This is me adding	a message
Add an optional e	tended description
	ctly to the master branch.
- Commit dir	

Step 5: Viewing your website

It may take a couple of minutes but, after you have uploaded your code, you should be able to see your website running at .github.io (replacing with your actual username).

Congratulations!

Step 6: Updating files

To update files so that you make your newest files available on the internet, you need to follow the same as **Steps 3 & 4**. The only difference is you now click that says **Upload files** which you can find where, below

jthawme / UI-Tools Private						O Unwatch ▼	1 🖈 Star	0	¥ Fork	0
<> Code	() Issues 0	ឿ Pull requests 0	Projects 0	C Security	Insights	🗘 Settings				
No descrip Manage topic	otion, website, o s	or topics provided.								Edit
	72 commits		រ្រៃ 1 branch		🛇 0 ré esses			∲ MIT		
Branch: ma	ster 👻 New pu	ll request			Create new fil	e Upload files	Find File	Clone	or downlo	ad -

Optional steps

Everything in this section isn't necessary but it might help you out, to work faster and better!

Step 7: Using a Git application

We can download github's application to help speed up commiting files. You can download the program here: <u>https://desktop.github.com/</u>. Once you've downloaded it, you need to log in and set up your name and e-mail address. Then you will see all of your repositories on your account.

Let S get Started!		
Filter Your Repositories	C	Clone a Repository from the Internet
L ctl-test/ctl-test.github.io		Create a New Repository on your Hard Drive
		Add an Existing Repository from your Hard Drive
		ProTip! You can drag & drop an existing repository folder

If you click your .github.io link and then the blue 'clone' button beneath it, it will download the repository to your computer. But the more important part now is that from that folder it downloaded, any changes you make it will watch them so that you can upload them back to GitHub.

By default it will download the repository to this location: Documents/Github/REPO_NAME



When you make changes to your files, if you check Github Desktop again, you should see that its detected changes in your files.

Current Reposito ctl-test.github.	ry 🗸	P Current Branch master	Publish branch Cannot publish unborn HEAD
Changes 1	History	index.html	+
I chan	ged file		@@ -0,0 +1,21 @@
Create index. Description	html	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21	<pre>+<!DOCTYPE html> +<?if lt IE 7?> <html class="no-js lt-ie9 lt-ie8 lt-ie7"> <?endif?> +<?if IE 7?> <html class="no-js lt-ie9 lt-ie8"> <?endif?> +<?if IE 8?> <html class="no-js"> <?endif?> +<?if gt IE 8?><!----> <html class="no-js"> <!--<![endif]--> + <head> * <meta charset="utf-8"/> * <meta charset="utf-8"/> * <meta content="IE=edge" http-equiv="X-UA-Compatible"/> * <title></title> * <meta content="" name="description"/> * <meta content="" name="viewport"/> * </head></html></html></html></html></pre>
Commit	to master		

After you commit them in Github Desktop, in the bottom left of the screen (similar to Step 4) you can then publish the changes to your website

Current Repositor	y -	Current Branch	Publish branch	
cti-test.github.i	0	* master	Publish this branch to GitHub	
Changes	History			
O chang	ged files	Nie le eel ele ere		
		You have no uncommitted char for what to do next.	IGES nges in your repository! Here are some friendly sug	ggestions
		Publish your branch The current branch (maste publishing it to GitHub you with others. Always available in the tool	er) hasn't been published to the remote yet. By can share it, open a pull request, and collaborate bar or XP	Publish branch
		Open the repository in you Select your editor in Prefer Repository menu or 第 쇼	ar external editor ences	Open in Atom
Summary (red	uired)	View the files of your repo Repository menu or 🕷 🚱	sitory in Finder F	Show in Finder
±+		Open the repository page Repository menu or (¥) ♪	on GitHub in your browser G	View on GitHub
Commit t	o master			

Revision #2 Created 6 June 2019 14:54:10 Updated 6 June 2019 15:07:22