HTML PART 5

Git and GitHub, Wave, Funkify, cPanel, Accessibility – Further Discussions

Last Class

- HTML multimedia
- HTML tables
- Some useful HTML tags
  - Block tags
  - Inline tags
  - Tags that are functional with JS

Agenda

- Web Accessibility and tools like Wave and Funkify
  - Validate your html and check your page for accessibility
- Managing your project folder and files
  - Git and GitHub
- Web hosting options: How to talk
  - cPanel

Discussions on Web Accessibility & Tools like Wave and Funkify

And more on accessibility

Discussions on

- The role of a web accessibility professional
- How disabilities affect web experience
- Four principles of accessible interface design

web accessibility professional - roles

- Reviews policies and decision making for software to purchase that will make the web interface more accessible
- Evaluate web interfaces for accessibility
- Assist people with disabilities to access online content

Disabilities in the Internet

- Visual Issues
  - Blindness, low-vision, color-blindness
  - Review your font size, color contrast, font style
- Hearing Issues
  - Partial or total deafness
  - Videos should use closed captioning; what about audio?
  - Audiences should be able to control the delivery of your multi-media content
- Motor Issues
  - Inability to use mouse or keyboard, slow response time etc
  - Can your website be browsed using the tab key?
  - Is your website steady or uses too much animations and graphics?
- Cognitive Issues
  - Learning disabilities, distractibility, dyslexia, inability to remember or focus large amount of information (ADD/ADHD)
  - Veterans may suffer from Traumatic Brain Injury (TBI) or PTSD – be mindful while developing

More info on disability stats and web accessibility

- https://monsido.com/blog/accessibility-statistics

The Web as an Enabling Technology

- The web enables and provides many opportunities for the disabled
  - Education & News
  - Commerce and social media
- There has been civil rights lawsuits for not making the web accessible in a way that it should be
  - Penn State, NYU, NorthWestern, FSU, Target, Southwest Airlines, Priceline.com, Ramada, Kindle, to name a few
- The DOJ regulates and enforces Title II and Title III of the ADA (Americans with Disabilities Act)
  - DOJ states that ADA compliance requires web accessibility

Defining web accessibility

- Web accessible to widest possible audience
  - Includes temporarily able-bodies users (TABs)
- Current online infrastructure is not the friendliest to people with disabilities
- Remember that aspects such as SEO (search engine optimization), mobile device, and usability are inter-related
  - You work on one aspect and it will improve the others.

Web Accessibility Standards – W3C WCAG 2.0

- Adhere to the standards set by Web Content Accessibility Guidelines
  - It is about principles and not the technology used to develop
- The four principles (POUR)
  - Perceivable
  - Operable
  - Understandable
  - Robust

WAVE (Web accessibility evaluation tool)

- https://wave.webaim.org/
- Available as extensions to chrome and firefox

Funkify

- The disability simulator of the web
- https://www.funkify.org/
- Available on chrome extension

Managing your project

Git and Github

Target Audience – Beginners

GIT

- Version Control System (VCS)
  - Track changes in computer files
  - Distributed in nature
  - Multiple people working on the same project
- Track
  - What changes were made
  - Who made them
  - When were they made [time]
  - Revert any changes
- Can maintain both local and remote repository

GIT

- For coding
  - Tracks code history by taking “snapshots” of the content
  - Snapshots are created by “commit”-ing the files
  - Any snapshot can be accessed at any time
  - Before commit, files can be kept at what is known as the staging area

GIT – Getting Started

Commands	Utility
$git init	Initialize local Git repo, Creates a hidden .git folder
$ git add <filename>	Add files to staging area or Git Index
$ git status	Shows files in staging area
$ git commit	Commit changes in the index or staging area
$ git push	Push local repo to remote repo like GitHub
$ git pull	Pull latest remote repo to local repo
$ git clone	Clone/copy a remote repo to a new local folder

GIT – installing

- Linux
  - Debian : $ sudo apt-get install git
  - Fedora: $ sudo yum install git
- Mac
  - http://git-scm.com/download/mac
  - With **homebrew**
    - **$ brew install git**
- Win
  - http://git-scm.com/download/win

Demo for GIT – download and install

- Download and install GIT
  - You can use the regular git bash in a separate window
  - Or you can use git bash from VS Code Terminal area and select git bash
- Check git version
  - $ git --version
- Create any new folder using mkdir
- To create new files from git bash
  - $ touch index.html
  - $ touch app.js
  - Etc…

Setup

- Initialize git repository
  - $ git init (from within your working directory)
  - Creates a hidden .git folder in your directory
  - The moment you run this you will notice all filename color coding will change – at least from within VS Code interface
- Add your name and email address to git config
  - $ git config --global user.name ‘Your Name’
  - $ git config --global user.email ‘abc@xyz.com’

Add files to git repo

- Add files
  - $ git add filename.extension
  - $ git add enables file tracking
  - $ git add index.html
- Check staging area
  - $ git status
- To remove a file from staging area
  - $ git rm --cached filename.extension

Ways to add files

- Using filename and extensions
  - $ git add index.html
  - Adds the specific file
- Using extensions
  - $ git add *.html
  - Add all files with html extension
- Add everything in the working directory
  - $ git add .
  - Add everything in the working directory to git index

Tracking changes

- After adding files, you can edit some file content
- Then use git status
- After the change you have two options
  - Add modified file(s) to staging area using $ git add OR
  - Discard changes done by :
    - $ git restore <file>

Commit the files in Staging area

- Commit files without any command options
  - $ git commit
  - Note if you use no options with the command, it will open the editor your selected for commit operation. Default git editor is vim
  - So make sure during installation to choose the editor you are familiar with – there are tons of options.

Commit using –m and -a

- Commit using option –m
  - $ git commit –m 'message for commit'
  - This will not open your editor
- If you want to skip using git add and do add and commit in one step
  - $ git commit –a –m 'mesg for commit'
  - The –a option does the add during commit

.gitignore

- What you don’t want specific files or folder in the working directory to not be added to git
- This is the purpose of a .gitignore file
  - Will contain list of all files and folder that git needs to ignore tracking
- Create the file (using the gitbash)
  - $ touch .gitignore
  - You cannot create this using windows explorer since it is a empty filename.

Utilizing .gitignore

- Say you create a log file in the working directory
- This is just for your personal purpose and you don’t want to commit to the project repo.

- Create this log file
  - $ touch log.txt
  - $ git status --- you will see two files the log and gitignore
- Add log.txt to .gitignore
  - Open the file in the editor and write the name of the file or folders
- Do git status again. This time you won’t see log.txt in untracked file list anymore.
- Git add and commit the .gitignore addition

Utilizing .gitignore

- Adding files
  - Specific files – log.txt
  - Adding entire directories - /demoFolder
  - Adding specific extensions - *.txt
- Check documentation for more ways

GIT branches

- Say multiple devs are working on a large project
- You are tasked with working on a specific functionality of the project
- For example, you are developing the login functionality
- You don’t want to push the changes to the main codebase until you have completed and tested your part.
- So you can create a separate branch, say login and keep it separate from the default branch
    - Default branch is called master. Or you can name it main

GIT branches – create your branch

- $ git branch branchName
- Example - $ git branch Login

- Creating a branch does not change the current branch

- See all local branches - $ git branch
- See all remote branches - $ git branch –r
- See all branches - $ git branch -a

GIT branches – switch to a desired branch

- $ git checkout branchName
- Example - $ git checkout Login

- If you do git status now, you'll see the log.txt file will show up in the list of untracked files
  - This is because the .gitignore file was added to the master branch but now we have switched to a new branch which does not have a .gitignore file

GIT branches – demo the differences

- Now within the Login branch do the following using git bash
  - mkdir login
  - cd login
  - touch login.html and then add some code to the file
  - Use git add .
  - Commit the changes
- Switch back to the master branch
  - You’ll not see the login portions

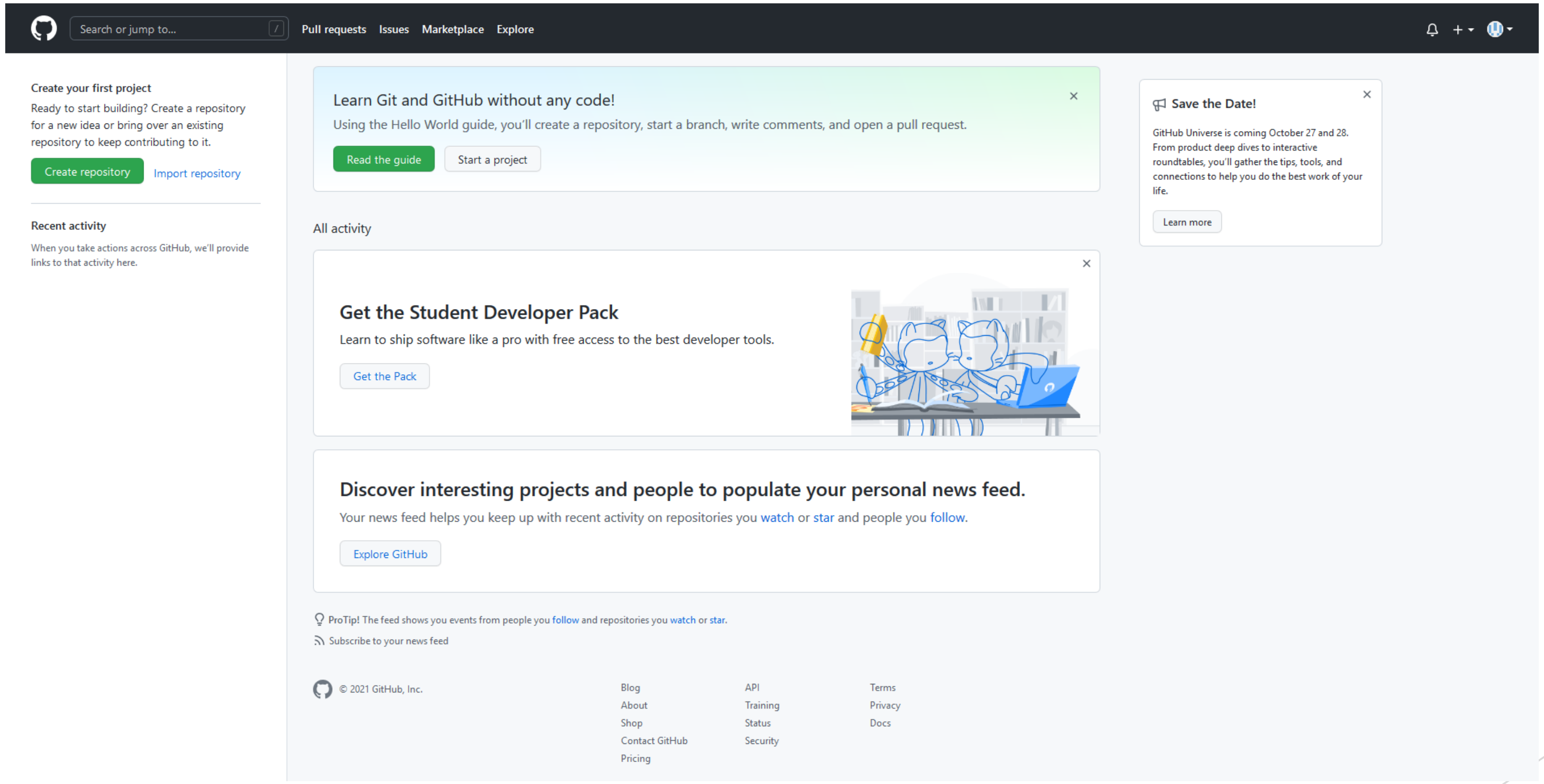
GIT branches – merge branches

- Branch multiple branches as follows
  - $ git merge Login –m ‘added login html page’
- Check out documentation for more on merging

Working with Remote Repo - GitHub

- Go to https://www.github.com
- Sign up or sign in
  - Email, pwd, username

Create a 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.

Create repository Import repository

Recent activity

When you take actions across GitHub, we'll provide links to that activity here.

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Read the guide Start a project

All activity

Get the Student Developer Pack

Learn to ship software like a pro with free access to the best developer tools.

Get the Pack

Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on repositories you watch or star and people you follow.

Explore GitHub

ProTip! The feed shows you events from people you follow and repositories you watch or star.

Subscribe to your news feed

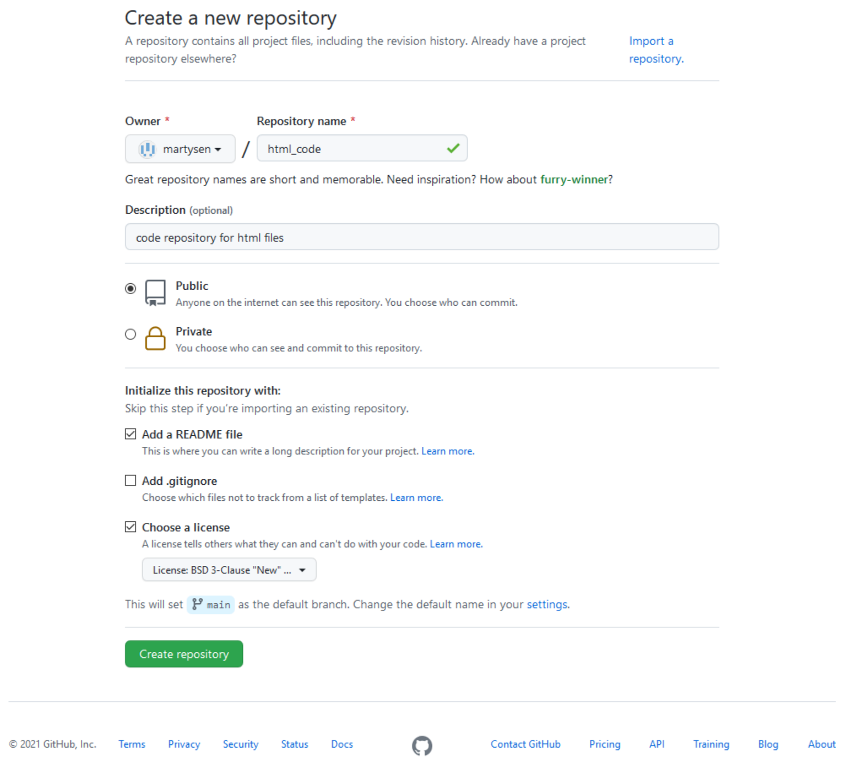
Save the Date!

GitHub Universe is coming October 27 and 28. From product deep dives to interactive roundtables, you'll gather the tips, tools, and connections to help you do the best work of your life.

Learn more

© 2021 GitHub, Inc.

Initialize repo



Create a new repository

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

Owner * martysen / Repository name * html_code ✓

Great repository names are short and memorable. Need inspiration? How about furry-winner?

Description (optional)

code repository for html files

[✓ 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:
Skip this step if you're importing an existing repository.

[✓] Add a README file
This is where you can write a long description for your project. Learn more.

[ ] Add .gitignore
Choose which files not to track from a list of templates. Learn more.

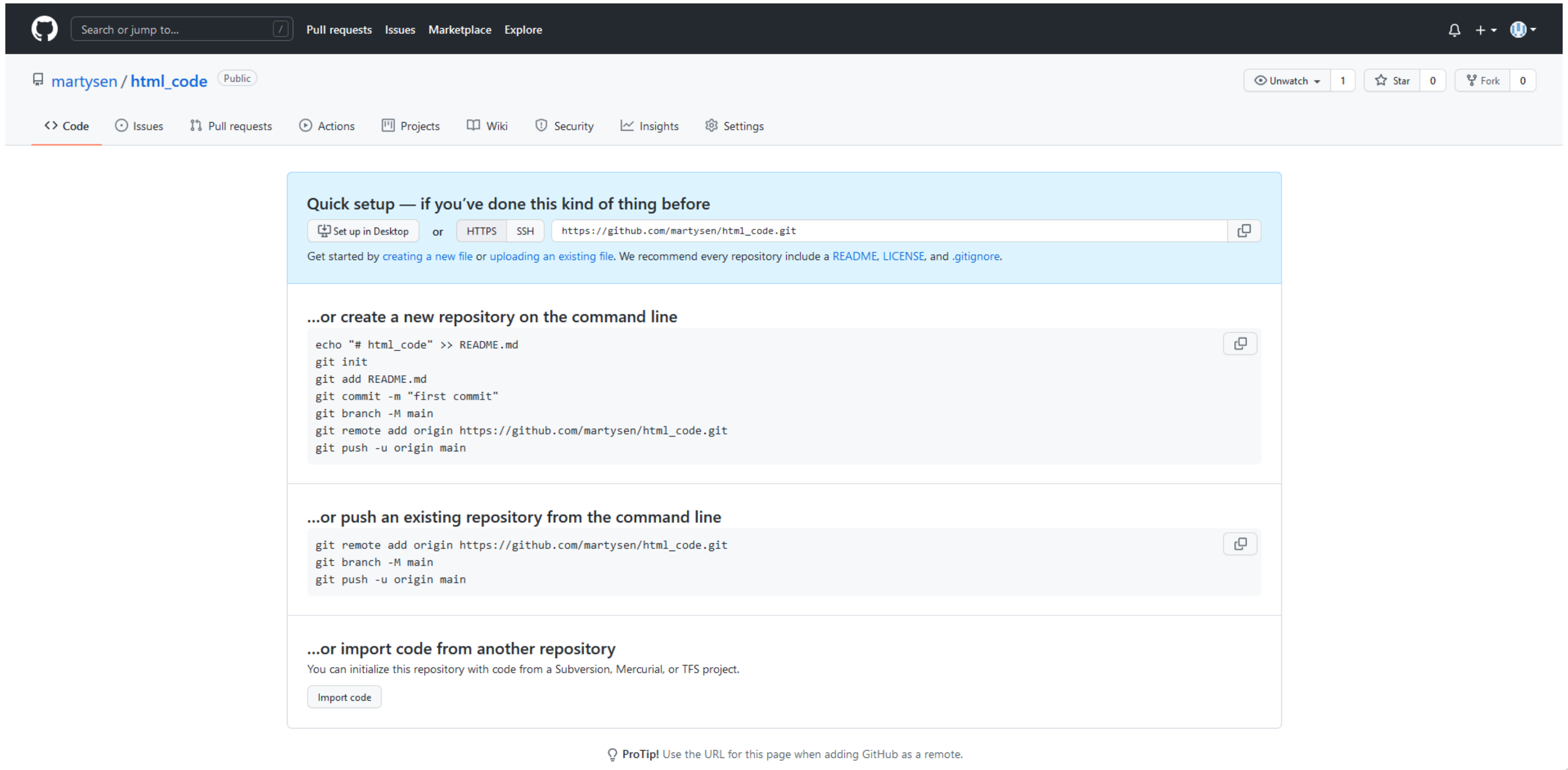
[✓] Choose a license
A license tells others what they can and can't do with your code. Learn more.

License: BSD 3-Clause "New" ...

This will set `main` as the default branch. Change the default name in your settings.

Create repository

Repo main page



Quick setup — if you've done this kind of thing before

https://github.com/martysen/html_code.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# html_code" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/martysen/html_code.git
git push -u origin main
```

...or push an existing repository from the command line

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

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

ProTip! Use the URL for this page when adding GitHub as a remote.

Connecting with GitHub from terminal

- You have two options (you can see it from the main page)
- HTTPS connection url
  - Will require you to log in to github
- SSH connection url
  - Secure shell

Add remote github repo to git

- List your remote repos using git bash
  - $ git remote
- Add the github remote repo to git (HTTPS)
  - git remote add origin https://github.com/martysen/html_code.git
  - Note for HTTPS you will use the HTTPS link
  - Githubremote repo by default is called origin
- Now if you do $git remote, you should see origin

Push git to github

- From git bash
  - $ git push –u origin master
  - Using HTTPS, it will open your browser to finish authentication

- After the last command, you can simply do
  - $ git push
  - As long you are in the same branch and pushing to same remote location

Clone github project

- Open a folder where you want to clone
- Start git in that folder
  - $ git clone https://github.com/martysen/html_code.git

- If multiple devs are working you can pull their work into your local directory using
  - $ git pull

Hosting your Site

How you host your sites?

- You need a domain name
  - Which is going to be your url
- You will need a Hosting service / company
  - Companies like InMotion etc for simple websites
  - If you need more scalable and complex management platform – go to the Cloud (AWS)

Domain names

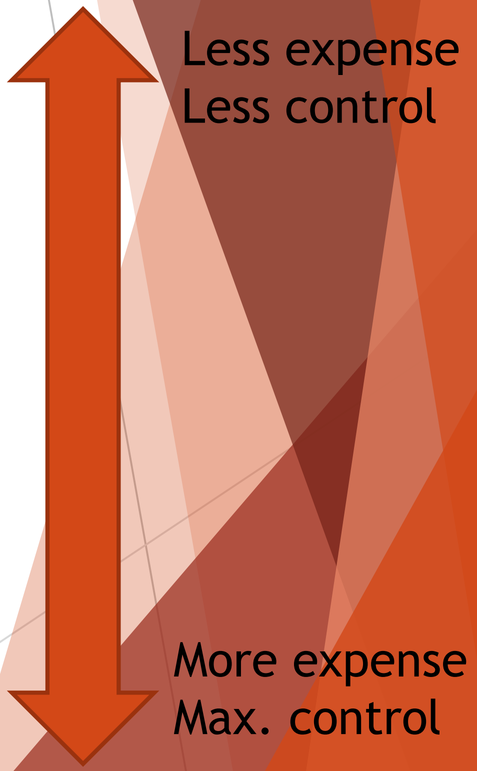
- Purchase domain name
  - Google domains (https://domains.google)
  - Godaddy.com
- When you think of domain names there are two portions to it
  - Second-level domains (SLDs): the name that will be unique to your brand
  - Top-level domains (TLDs): the extension. Think of .com, .net etc
- So example – dogemenot.net
  - sld is dogemenot; tld is .net
- Just having a domain name with no files for it to show is meaningless – you need a hosting service

Hosting

- Need a registered IP address → Map to your domain name
  - Recall FQDN

Different types of hosting services

- Shared hosting (entry level)
  - Multiple websites hosted together with all of them sharing the resources (RAM,CPU)
- Virtual Private Server (VPS)
  - Middle ground between shared and dedicated hosting.
  - You get your own virtual envn
- Dedicated Server hosting
    - You get your own physical machine
- Cloud hosting
- Managed hosting
- Colocation
  - Rent out physical location and put your own machine in there <most expensive>



Check out hosting services of 2021

- https://www.techradar.com/web-hosting/best-web-hosting-service-websites
- Note: this is not a promotion of techradar.com or any of the companies listed in the post above.

There are also Free Web Hosting Services

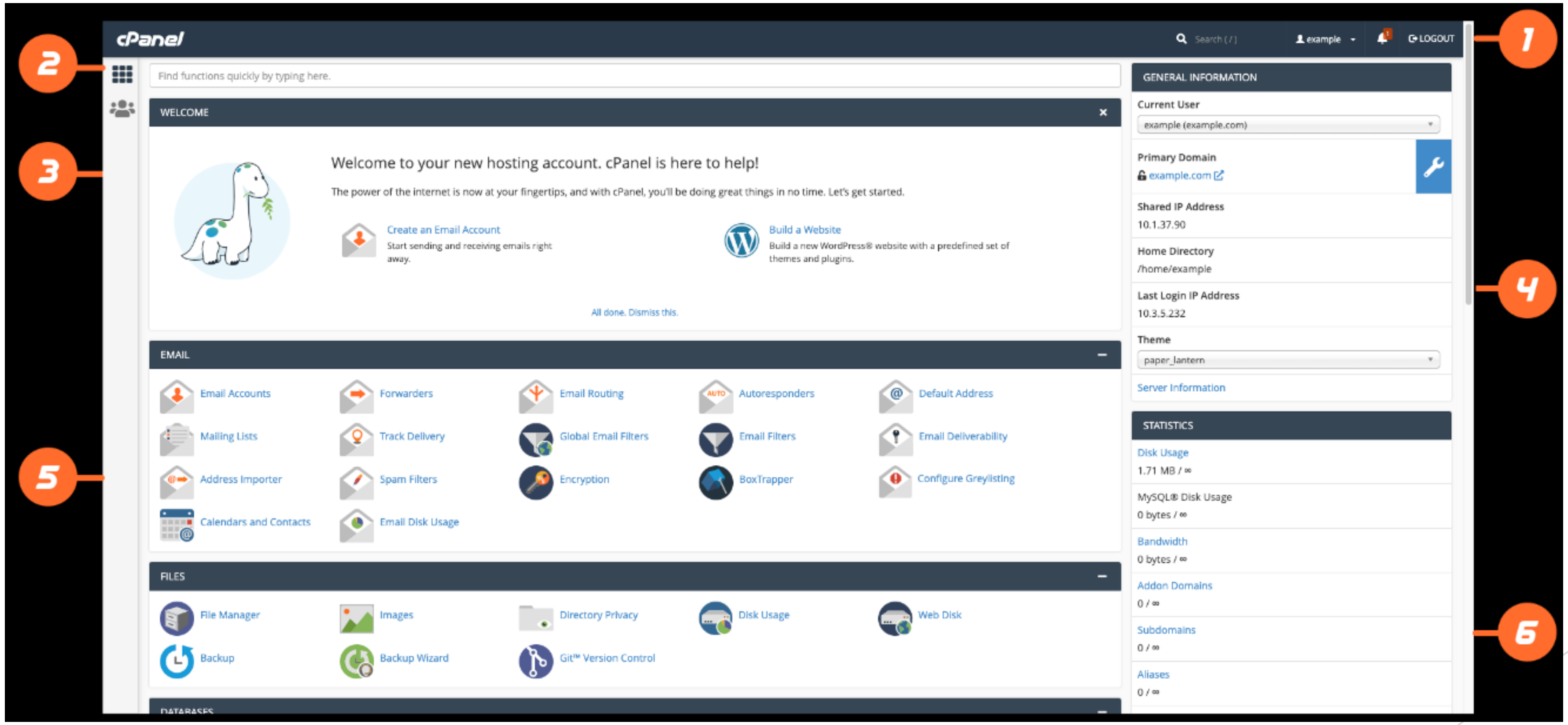
- https://www.techradar.com/web-hosting/best-free-web-hosting
  - Infinityfree, **Byethost**, etc.
- You might not have control of the domain name
- Limited access to add on tools
- Ads and redirects are the price to pay
- Load times might be slow or average; There might be no uptime guarantees

cPanel – interface for managing your site

- cPanel is a File Manager for your website
- Most hosting services will come with a cPanel add-on
- You connect to cPanel using a FTP service software like FileZilla
- Important thing to note for cPanel directory structure
  - public_html directory
  - This is where you will drop your website content
  - There should be a index.html file

- More on these when we learn to style our websites and add some functionality using JavaScript

cPanel Interface



cPanel Search (/) example LOGOUT

1 2 3 4 5 6

Find functions quickly by typing here.

WELCOME

Welcome to your new hosting account. cPanel is here to help!

The power of the internet is now at your fingertips, and with cPanel, you'll be doing great things in no time. Let's get started.

Create an Email Account
Start sending and receiving emails right away.

Build a Website
Build a new WordPress® website with a predefined set of themes and plugins.

All done. Dismiss this.

EMAIL

Email Accounts, Forwarders, Email Routing, Autoresponders, Default Address, Mailing Lists, Track Delivery, Global Email Filters, Email Filters, Email Deliverability, Address Importer, Spam Filters, Encryption, BoxTrapper, Configure Greylisting, Calendars and Contacts, Email Disk Usage

FILES

File Manager, Images, Directory Privacy, Disk Usage, Web Disk, Backup, Backup Wizard, Git™ Version Control

DATABASES

GENERAL INFORMATION

Current User
example (example.com)

Primary Domain
example.com

Shared IP Address
10.1.37.90

Home Directory
/home/example

Last Login IP Address
10.3.5.232

Theme
paper_lantern

Server Information

STATISTICS

Disk Usage
1.71 MB / ∞

MySQL® Disk Usage
0 bytes / ∞

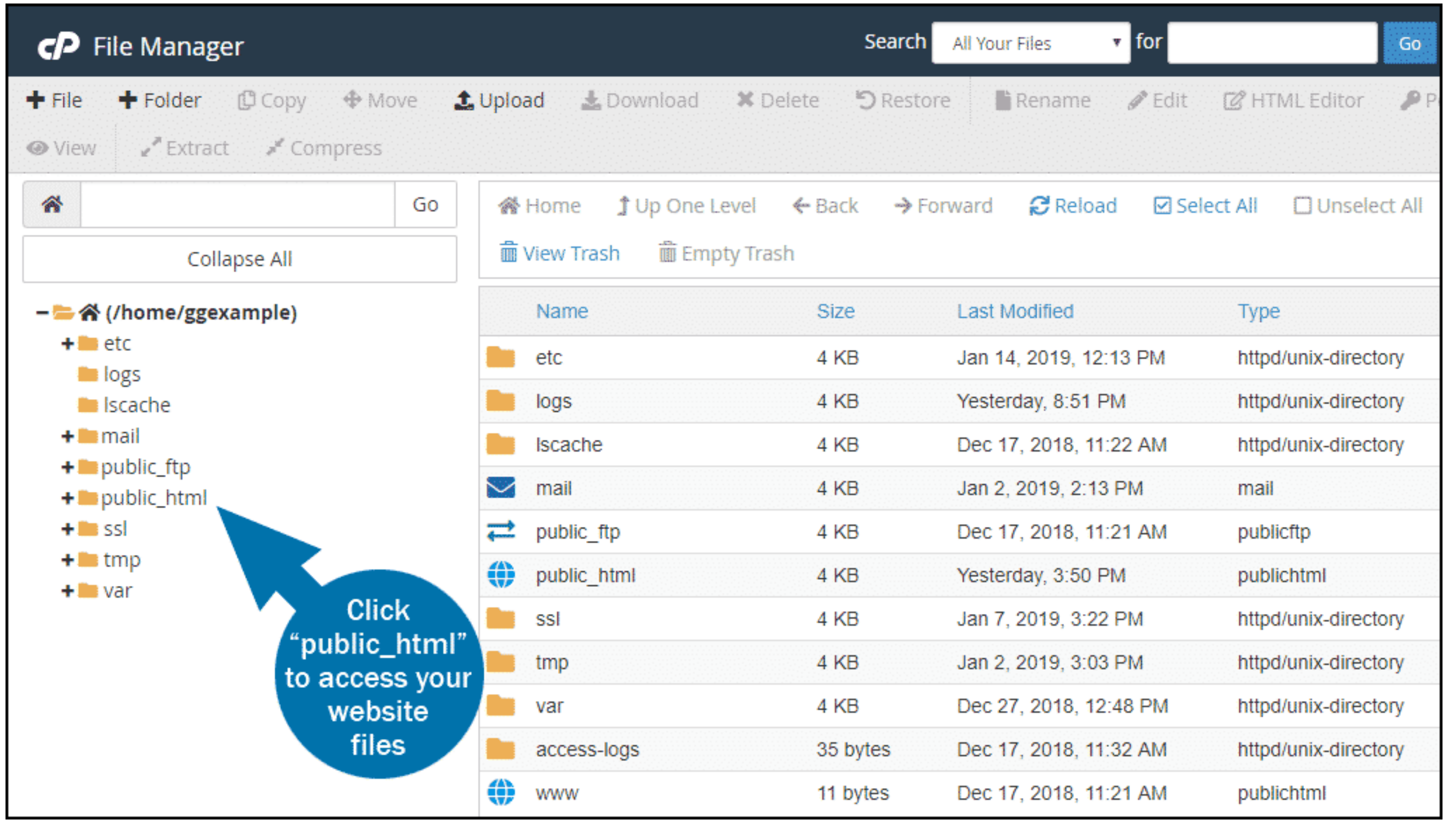
Bandwidth
0 bytes / ∞

Addon Domains
0 / ∞

Subdomains
0 / ∞

Aliases
0 / ∞

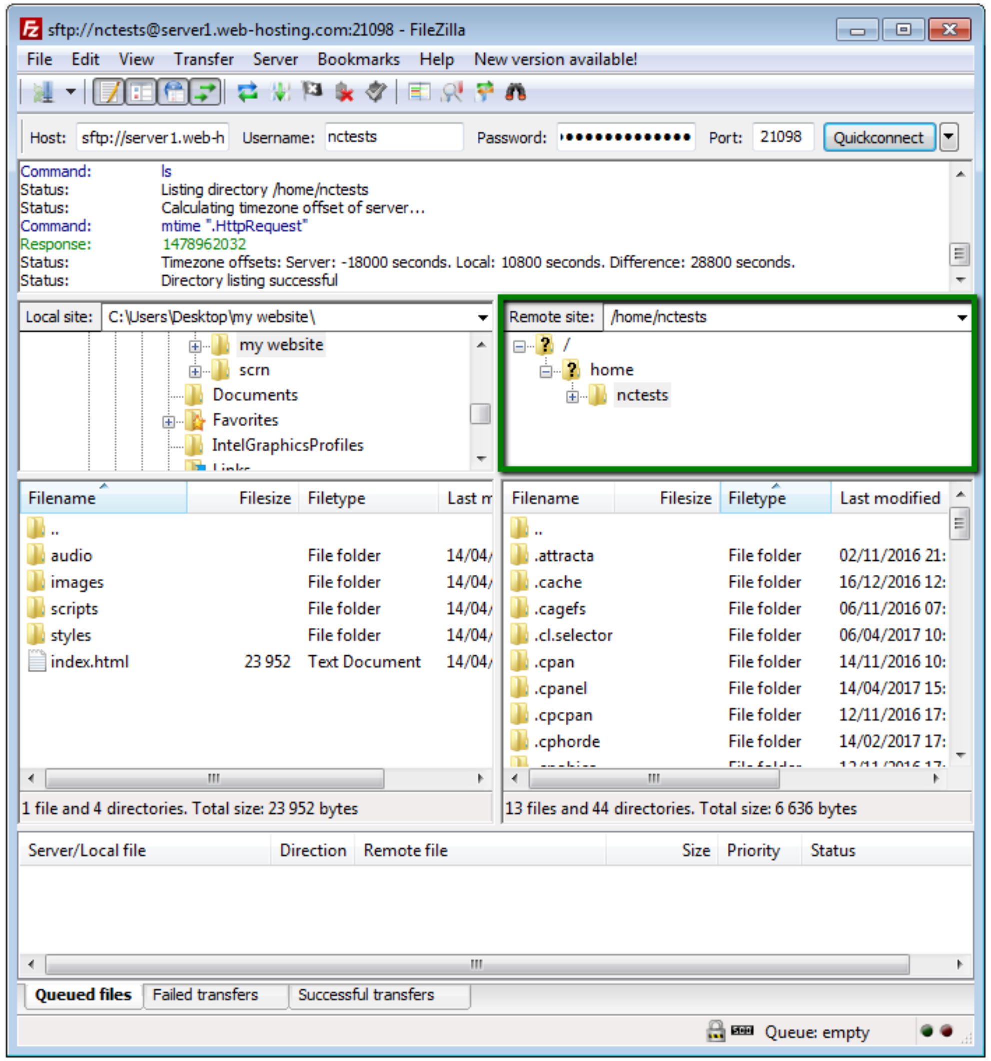
cPanel File Manager



Click "public_html" to access your website files

Name	Size	Last Modified	Type
etc	4 KB	Jan 14, 2019, 12:13 PM	httpd/unix-directory
logs	4 KB	Yesterday, 8:51 PM	httpd/unix-directory
lscache	4 KB	Dec 17, 2018, 11:22 AM	httpd/unix-directory
mail	4 KB	Jan 2, 2019, 2:13 PM	mail
public_ftp	4 KB	Dec 17, 2018, 11:21 AM	publicftp
public_html	4 KB	Yesterday, 3:50 PM	publichtml
ssl	4 KB	Jan 7, 2019, 3:22 PM	httpd/unix-directory
tmp	4 KB	Jan 2, 2019, 3:03 PM	httpd/unix-directory
var	4 KB	Dec 27, 2018, 12:48 PM	httpd/unix-directory
access-logs	35 bytes	Dec 17, 2018, 11:32 AM	httpd/unix-directory
www	11 bytes	Dec 17, 2018, 11:21 AM	publichtml

Option: Use FileZilla instead of cPanel interface



sftp://nctests@server1.web-hosting.com:21098 - FileZilla

Host: sftp://server1.web-h Username: nctests Password: •••••••••••••• Port: 21098

```
Command:    ls
Status:     Listing directory /home/nctests
Status:     Calculating timezone offset of server...
Command:    mtime ".HttpRequest"
Response:   1478962032
Status:     Timezone offsets: Server: -18000 seconds. Local: 10800 seconds. Difference: 28800 seconds.
Status:     Directory listing successful
```

Local site: C:\Users\Desktop\my website\

- my website
- scrn
- Documents
- Favorites
- IntelGraphicsProfiles
- Links

Remote site: /home/nctests

- /
  - home
    - nctests

Filename	Filesize	Filetype	Last m
..			
audio		File folder	14/04/
images		File folder	14/04/
scripts		File folder	14/04/
styles		File folder	14/04/
index.html	23 952	Text Document	14/04/

1 file and 4 directories. Total size: 23 952 bytes

Filename	Filesize	Filetype	Last modified
..			
.attracta		File folder	02/11/2016 21:
.cache		File folder	16/12/2016 12:
.cagefs		File folder	06/11/2016 07:
.cl.selector		File folder	06/04/2017 10:
.cpan		File folder	14/11/2016 10:
.cpanel		File folder	14/04/2017 15:
.cpcpan		File folder	12/11/2016 17:
.cphorde		File folder	14/02/2017 17:

13 files and 44 directories. Total size: 6 636 bytes

Server/Local file	Direction	Remote file	Size	Priority	Status

Queue: empty

Options with Amazon AWS

- There are two options to host using Amazon
- Amazon AWS with S3 bucket used for static web hosting
- Amazon Amplify
  - For code and build and hosting
  - Has GitHub integration
- No control over your domain name / url
- You can map your custom domain to amplify
  - https://docs.aws.amazon.com/amplify/latest/userguide/to-add-a-custom-domain-managed-by-google-domains.html

You can also publish your work with GitHub

- If you don’t want any add ons or management services offered by hosting services
  - You can publish your sites through github