Learning HTML & CSS

A collection of info and resources for Paloma Kop's HTML/CSS classes.

- Course Info
- References & Resources
- Useful Tools
- Inspiring Websites
- Web Hosting Services
- Domains
- Beyond Handwritten HTML & CSS

□□ Course Info

About the course: *Making Fun, Simple Websites with Handwritten HTML/CSS*

https://www.youtube-nocookie.com/embed/0S4E3d_LJLU?si=xgqM_TffdFGAI-di

In an age where most websites are both boring to look at, and unnecessarily complex under the hood, we'll be reclaiming the fun and simplicity of the early web by creating websites the old-fashioned way: writing pure HTML and CSS, by hand.

In this course, we will learn how to create simple websites by writing HTML code. We will learn to write CSS to style our webpages. This is an introductory course for beginners, or those wishing to refresh their skills. At the end of the course, everyone will have created their own site.

To get our pages online easily, we will use the Neocities platform which offers both hosting and a code editor within the browser. (If you've created HTML pages before, and you already have a preferred method for editing your code or a preferred hosting provider, you're free to use that.)

We will cover:

- Basic structure of an HTML document
- Essential HTML tags
- IDs and Classes
- Styling the page and specific classes or elements with CSS
- CSS transitions and animations
- Internal and external links
- Embedding media (Images, GIFs, and MP3 audio files)
- Embedding external media with iframes
- Using developer tools (inspect element, view source, etc.) to debug our own sites and "borrow" styles from other sites
- Examples of websites from across the web that have been inspired by the "small web" movement

We will ***NOT*** cover:

- Javascript or any other executable scripts or code
- Bloated web frameworks such as React
- Anything outside of pure, simple HTML and CSS

We'll each create a site during the first class, and we'll work on expanding it each week, with an instruction period as well as free work time with opportunities for Q&A. It's great to come to this class with a vague idea for your site — or just have fun with it and see what happens □□

Materials

For this course, you will need:

- A computer with a web browser installed
- A microphone and a webcam so we can hear/see you in the virtual classroom

Optional but helpful:

- An extra monitor for your computer so you can look at the virtual classroom and your website-in-progress at the same time

About the Teacher

This course is led by Paloma Kop, a multimedia artist and computer programmer. They've created many websites over the years, both professionally and for fun. Last winter, they completely redid my own artist website in plain HTML/CSS, and wrote a little about how and why. They enjoyed the process and wanted to help others do the same — thus the idea for this class was born.

□□ References & Resources

Knowing how to find the info you need

The main skill of creating websites with HTML & CSS isn't memorizing all of the names of every tag and property. It's knowing how to search for and find the info you need to do what you want to do in a given moment. Searching on your preferred search engine, and knowing the search terms to use, is a good starting point. There are also many more specific collections of reliable reference materials, and online guides and tutorials. Below are some of the ones we've found most helpful; we'll keep adding to them as we find more.

References

W3Schools

The number one point of reference I'd recommend for both HTML and CSS is W3Schools. If you are looking to find info on a specific HTML tag or CSS property, there is a search function; or just visit the HTML or CSS sections and browse the lists of pages in the left sidebar.

There is also a pretty cool CSS demo page here.

https://www.w3schools.com/

Guides and Tutorials

General HTML & CSS Tutorials

- W3Schools also has some excellent introductory tutorials for HTML and CSS.
- Codecademy has free interactive courses on HTML and CSS.
- HTML Dog has tutorials for HTML and CSS for 3 different levels of experience. (They also have tutorials for JavaScript.)
- Mozilla has more in-depth HTML and CSS tutorials.

Guides for Handmade/DIY Websites

- Byhand.website - an approach to handmade websites by Chelsea Thompto
- DIY Web for Beginners - info for a workshop by Honor Ash
- CSS & HTML Tips - a collection of useful info by Thomasorus
- HTML & CSS - guide pages by sadgrl.online

Specific How-Tos

- Center an Image
- Add a Favicon
- Set Backgrounds in CSS
- HTML Div Tutorial
- Align Images Side by Side
- Set a Border (Shorthand - just one line of CSS)
- List of CSS Units (px, em, %, etc.)
- CSS Transitions
- CSS Links
- CSS3 Gradients
- Style Selected Text with ::selection
- Create Responsive iframes
- HTML "Pre" Tag (For ascii art)
- Tinkerbell Mouse Effect (javascript)
- CSS ::before and ::after
- HTML symbol codes
- Open Link in New Tab (example)
- Note to self: try to find a good tutorial for linking to an #ID

Advanced CSS Topics

CSS Fonts

Using web fonts (providing your own font file — this allows for more customization):

- Add a Custom Font

Using system fonts (using fonts already installed on whatever device the page is being viewed on — this allows for faster page load):

- modernfontstacks.com

- systemfontstack.com

CSS Flex & Grid

Flex and Grid can be very powerful tools for creating layouts, but potentially confusing. Even after using them many times, I still always refer to these two pages:

- CSS Tricks - A Complete Guide to Flexbox
- CSS Tricks - A Complete Guide to CSS Grid

These are some cute games to help you get the hang of using them:

- Flexbox Froggy □□
- CSS Grid Garden □□

Embedding Audio and Video

Many 3rd party websites, such as Youtube, Vimeo, and Bandcamp, offer an embed code that you can copy and paste into your HTML in order to include a video or audio embed on your site. However, since these are basically embedded frames of their website inside your website, there are a lot of limitations on how much control you have over the appearance and behavior of this type of embed. Additionally, since visitors to your website will also be loading the 3rd party's website, this can allow the 3rd party to track and monitor your visitors' activity.

It is also possible to embed video and audio files directly on your site with the HTML "audio" and "video" tags. This allows you to have more control over the embed, though the default appearance of the controls will be determined by the browser. This is also a more advanced option since it requires a deeper understanding of files and encoding types, and you will also need to host your own media files which can be a lot larger than plain HTML files.

To get started embedding your own media files using HTML tags, check out these links:

- HTML Video Tag
- HTML Audio Tag

If you want to customize the appearance of the player, you'll need to use javascript, which is outside the scope of our class. But, here's a tutorial just in case anyone wants to dig into that:

- https://css-tricks.com/lets-create-a-custom-audio-player/

Media Queries

Media queries are used to change how your webpage looks based on the size of the screen it's being viewed on. They can come in really handy, but if your layout is simple enough it should work

well on all devices without needing a lot of them.

- CSS Tricks - A Complete Guide to CSS Media Queries

CSS :target Selector

A fancy CSS feature that lets you highlight or affect an element with a target ID in the current URL. If this doesn't make any sense to you please ignore it.

- https://developer.mozilla.org/en-US/docs/Web/CSS/:target

□□ Useful Tools

Favicon generator

- Favicon.io - can generate from text, images, or emoji

HTML Color Picker

- Hue/Saturation/Lightness color picker on W3Schools - easily find the color you want in the html hex format

W3C Validator

- validator.w3.org - make sure your website code is valid (note - use this as a guide, not the law)

Inspiring Websites

Artist Portfolios & Digital Gardens

- hnr.fyi - simple image-focused html/css site with great alt text
- chelsea.technology - nice simple portfolio with looping video on homepage
- aloeloopy.neocities.org - personal website styled after a text adventure game
- palomakop.tv - yes, i'm putting my own website in here :o
- pixouls.xyz - simple but extensive site, including a blog post about handcrafted websites
- ritualdust.com - aesthetic website of a folk & fantasy artist

Web-based Art

- the candy crystal rainbow codex - alien text adventure with cool audiovisual elements
- BlackWings - click expand to enter an interactive map you can explore

Retro-style Sites

- trilobites.info - a guide to the orders of trilobites
- www.cameronsworld.net - a web-collage of text and images excavated from archived GeoCities pages
- odditycommoddity.neocities.org - lots of gifs

Brutally Simple Sites

- brutalist-web.design
- motherfuckingwebsite.com
- bettermotherfuckingwebsite.com
- thebestmotherfucking.website
- perfectmotherfuckingwebsite.com

Webrings and Directories

- webring.xxiivv.com - xxiivv webring
- byhand.website/inspirations - list of neat websites, mostly plain html/css

- neocities.org/browse - browse sites created on the neocities platform

Example Sites

- linkfae.netlify.app - an example webpage meant to be copied and edited

□□ Web Hosting Services

Web Hosting

A web host is a server that hosts the files of your website, and serves them when someone visits your site. Because we are creating sites that are just simple HTML & CSS files, and maybe some images or other assets, our websites are **static**, so the type of web hosting service we need is called **static hosting**. Below are a few options for static hosting services, along with some pros and cons of each one; this will be expanded as we gather more information.

Neocities.org

This is the web host that we are using in class. It is a great service to start out on because they provide a web interface for editing your HTML and CSS files, but you can also easily download your whole site and move it to another service if you'd like.

https://neocities.org/

Pros

- Provides a basic code editing interface through their website so you can edit your site without having to install anything on your computer
- Offers multiple ways to download or upload files
- Free option with 1GB storage limit and neocities domain

Cons

- If you need more storage or want to use your own domain (i.e. http://yourcustomname.com) you will need to upgrade for $5/month (which adds up to $60/year) — not a terrible deal but there are cheaper methods out there

Github Pages

With a bit of technical know-how, it is possible to publish a static website through Microsoft's GitHub.

https://docs.github.com/en/pages/quickstart

Pros

- You can publish a static website with storage limit of 1GB, and a custom domain, for free
- Because websites are a git repository, this gives you version control, letting you see the history of your edits

Cons

- GitHub is owned by Microsoft, so in addition to being run by a dubious company that endorses irresponsible AI tech, the perks of the free edition may go away at any point; be ready to move your site to a new host when this happens
- Can be confusing to set up, especially for those who don't already use GitHub
- Your website repository must be public unless you have a paid plan

□□ Domains

About Custom Domains

Let's say you want your website URL to be something that doesn't end in '.neocities.com'. This is called a custom domain. Generally, to get a custom domain, you need to buy one from a **registrar**, and keep paying for it annually in order to keep it active. The last part of the domain — for example, '.com', '.net', or '.xyz' — is called a **TLD** (top-level domain). Often, different TLDs will be priced differently; some are very expensive, while many common ones can be purchased for less than $20 USD per year.

Here are some common steps you'll need to follow in order to set up a custom domain for your site:

- Choose a domain name (i.e. myexamplename.biz) and find a registrar that offers this name for sale.
  - You will need to create an account and submit your payment to purchase the domain.
- Edit the DNS settings for your domain in order to point it to your site.
  - The exact steps can vary based on your web host and domain registrar, so check the instructions on their websites, but generally, you will need to add one or more **DNS records** so that when someone visits your domain, they will see your website.
  - You can have a site on your **root domain** (i.e. myexamplename.biz), and you can also have as many sites as you want on **subdomains** of your domain (i.e. wow.myexamplename.biz, blog.myexamplename.biz, wiki.myexamplename.biz). You will want to choose which one to use before setting up DNS since the steps are different for root domains and subdomains.
- Wait for it to propagate. If the DNS changes worked properly, you should be able to go to your domain and see your site within a few minutes; however, it can take up to 72 hours for the domain to properly point to your site everywhere in the world.

Domain Registrars

Here are a few registrars you can purchase a domain from. You might want to compare their prices for your chosen domain on a few different registrars before purchasing one.

Registrars we have heard are OK

- Porkbun https://porkbun.com/
- Gandi https://www.gandi.net/en

- Hover https://www.hover.com/
- Web Hosting Canada https://whc.ca/en

Registrars to avoid

- GoDaddy

□□ Beyond Handwritten HTML & CSS

Potential Growing Pains

While creating websites by creating HTML and CSS files by hand can be fun, educational, and sufficient for many use cases, there are certain situations where using this method can result in an excessive amount of tedious copy-pasting.

Here are some situations you may run into that may suggest your site is outgrowing the handcrafted HTML/CSS medium:

- Your site has a lot of pages (a rough number would be over 20), and you want them to all have a consistent layout. If you have a handwritten site with that many pages, and you want to update the layout on all your pages, you will probably either need to do a lot of copying-and-pasting or use some delicate find-and-replace wizardry.
- You have a lot of items of content (portfolio projects, blog posts, etc.) and want to be able to generate pages with a certain number of items for page.

There are many website building tools out there — so many, in fact, that trying to find or pick one can be extremely overwhelming. However, I'd like to focus SSGs as a type of tool that you might want to look into if you've already started your site and run into the above limitations.

Static Site Generators

One type of tool that seems like a natural progression from plain HTML/CSS is a Static Site Generator, or SSG. These are pieces of software that take in source data from some files (formats can vary), and output a static website (usually simply a bunch of HTML and CSS files). When you want to update your site, you would usually update the source data, and then run the SSG to generate a new version of your site. You can then upload your site to a static web host in the exact same way you would with a handcrafted site.

The most simple version of an SSG would be a script that takes the following pieces of information and combines them together into an HTML file for each page:

- The top part of your HTML file, including the entire <head> section, the beginning of the <body> section, and any header content that you want to appear at the top of every page

- The middle part of your HTML file — this is the part that you'd want to be unique to the page
- The bottom part of the HTML file, including any footer content you want to appear at the bottom of every page, and the closing tags for </body> and </html>

If you enjoy coding your own tools, it can be a fun project to roll your own static site generator. If you want to use an existing tool, see below for a list of SSGs that you can check out.

Even if your site may outgrow plain HTML/CSS, or you plan to use an SSG for your site, it can be useful to create a few pages by hand anyway, so that you can experiment with the design and layout for your site. This is called **prototyping**. Then, once you are ready to implement your site using an SSG or similar tool, you'll already have much of the code for your template.

Additionally, basic knowledge of CSS and HTML will go a long way in helping you build a site with many SSGs.

List of SSGs & Site-Builder Resources

- Eleventy a.k.a. 11ty https://www.11ty.dev/ - an SSG
  - Tutorials:
    - https://dev.to/psypher1/lets-learn-eleventy-1a67
- Awesome Small Web Publishing - a list of SSGs, dynamic site generators, and other publishing tools for the small/independent web, as well as some helpful articles about the pros and cons of different approaches
- Lichen - a barebones CMS for making simple sites that don't require any coding to edit