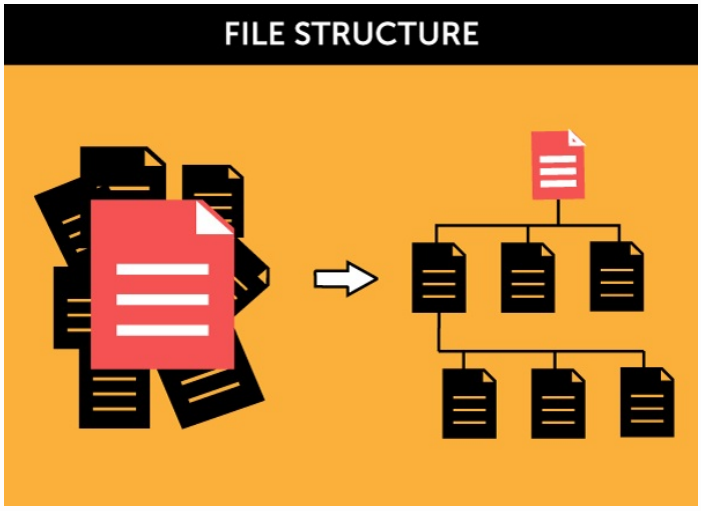
File Structure of a Website



file image

Goals

By the end of this case you will learn:

- What structure your website should have
- Importance of appropriately naming your files
- To create structure trees for all your files to help your visualize and organize
- How parent/child relationship works with files
- To create paths to your files to make them to talk to one another

Introduction

This module is intended to help you understand what the structure of a website looks like in the backend. It is essentially how the website is structured and how the content is organized and accessed. All the files that you build containing text, images, styles and scripts that output a website have to be assembled into a logical structure to ensure they can talk to each other and once it all looks good you can upload them to a server.

Business Context

Once you start in a company or build a website for a client, the first thing you have to understand is how you need to structure your files to not only create a clean system but also follow the conventions so that other

developers that are either working with you or come after you, can quickly and efficiently understand where the files are and how to get the work done. In a company or during a project organization is key to ensure the project is done on time, and file structure is where it starts.

Understanding file structures

Every website is a combination of multiple files working together, and they are organized on the server in a way that they can communicate to each other. All the HTML, CSS, Javascript and image files that you create have to be linked in a sensible manner for the website to load quickly and appropriately.

Naming Conventions

web cloud

Let's start with the basics on how you should name your files. Naming conventions in web development also applies to classes, IDs and variables but for this section we will focus on the files and directories.

Here are a few conventions we have to follow:

1. Always use lower case - no capitalization needed
2. Only use Alphanumerics - this means you can only use the characters a-z and numbers 0-9
3. No spaces - you can substitute spaces with only the following

a. Underscores (_)
b. Hyphens (-)
c. Tildes (~)

4. Appropriate file extensions - depending on the file type ensure that it separated correctly from the file name

a. HTML file - index.html
b. CSS file - style.css
c. Javascript file - script.js

d. Image file PNG or JPEG - image.png or image.jpg

These naming conventions primarily come from the fact that many web servers (where you're uploading your files) are based on UNIX and not Windows (these are both operating systems).

In windows there's no distinction between index.html or iNdeX.html, if you tried to upload one of these to where the other was residing, it simply would overwrite and not treat the files differently but with UNIX the files and folders are case sensitive and it would just add the other in the same folder, treating them differently and potentially causing technical problems.

Exercise 1:

So what does a file structure look like?

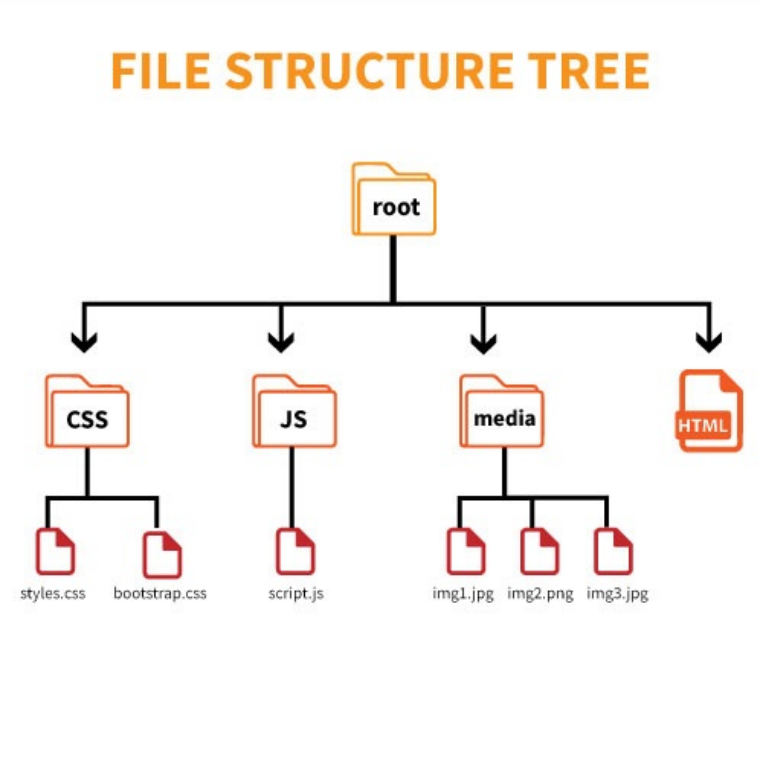
> The folder structure usually always starts with the root folder and it holds all the content of the website. You can name the root folder whatever you'd like (usually it's named after what your website will be called).

Inside that root folder will be:

– `index.html` file
– a folder for your CSS files
– a folder for Javascript files
– a folder for your media files (images, gifs and videos)

We can easily construct a file structure tree to help us visualize this.

See screenshot below.



file structure tree

What is an index.html file?

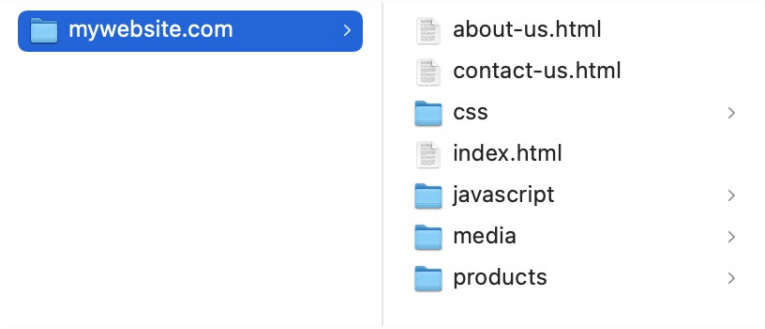
An `index.html` file is the default file that the server looks for if a request is made to display your site (unless specified otherwise). It's basically your homepage. When somebody types in your website URL like (==https://www.google.com==) the server goes to the directory and looks for the default page to display and that default page in the directory is always index.html (unless specified otherwise).

Keep in mind that every page has its own HTML file. You can choose to name the other pages as you please but for organization and naming convention sake we choose to name a page by its ultimate purpose. So a product page would be called `product.html`.

Questions:

Parent Folders and Paths

> Here is an image of what a basic website folder will look like:



website file organization

So the root folder as we can see is 'mywebsite.com' which is our root folder. Within that root folder we have multiple html pages for our website and our homepage called index.html. There are folders like we spoke about earlier for our css, javascript and media files.

When we are creating paths from our root folder to a file within the same folder we would just specify the file name.

Examples:

From index.html to products.html
Path: products.html

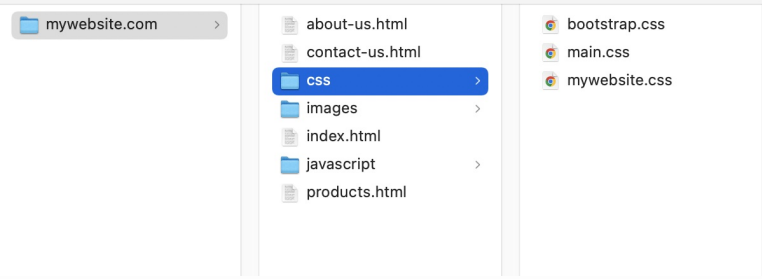
From index.html to contact-us.html
Path: contact-us.html

We can also use `./` and get the same results. This is like the www in the URL, it will be added automatically unless we add it ourselves.

From index.html to products.html
Path: ./products.html

Moving between folders

Typically to move into a folder we identify the folder name and add a forward slash followed by the file name. Here's a screenshot of the inside of a CSS folder:



folder navigation

Here are some examples on how to move between folders:

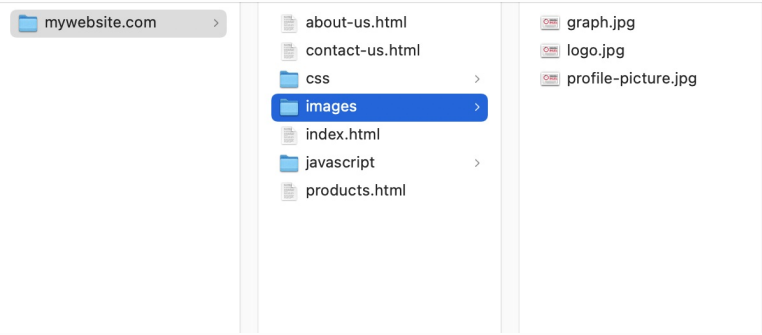
From: index.html to bootstrap.css
Path: css/bootstrap.css

From index.html to mywebsite.css
Path: css/mywebsite.css

This is the same for images and Javascript files.

Moving out of folders:

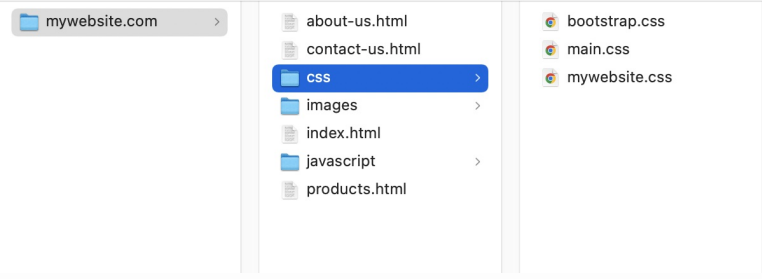
Here's an image showing what the inside of the image folder looks like, it essentially just contains images that live on your website.



folder navigation 2

We use `../` to move to the previous folder. If we are in the image folder and we want to move from logo.jpg to main.css the path will look like this:

Path: ../css/main.css



folder navigation 3

Conclusion & Attribution

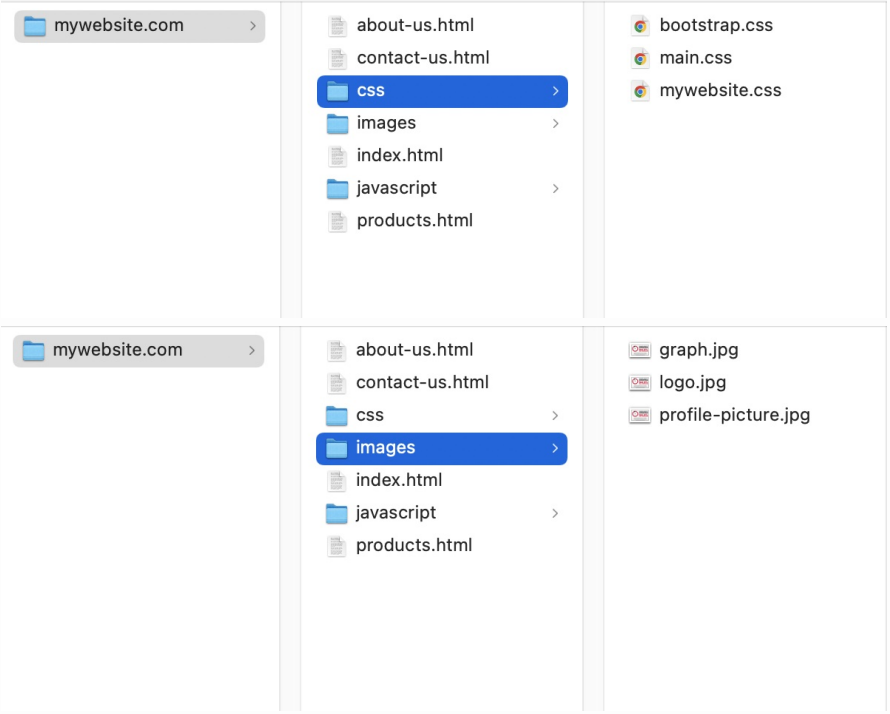
So we are moving out of the folder back into the root folder and from there we are going forward into into the css folder and selecting main.css

Here is a summary of path meanings:

`/` is the root of the current drive
`./` is the current directory (the current folder you're in) - remember this is something you don't have to add but helps in understanding folder structure
`../` is the parent of the current directory (the folder before the current one that you're in)

Exercise:

With the following screenshots please answer the following questions:



Conclusion & Takeaways

- Understanding the file structure is very important to ensuring that your website is built in an organized way and loads appropriately
- Following naming conventions, file paths and understanding what each file means is important to help you build a website in a sustainable and manageable way that allows you to scale efficiently.

- It’s also important to understand these factors since they are universal and it helps other developers takeover where you left off while also allowing developers to help you troubleshoot.

Further Reading

Gledhill, M. (n.d.). 4.2 Getting started - The Website Folder Structure. PracticalSeries. Retrieved November 4, 2022, from http://practicalseries.com/1001-webdevelopment/04-02-starting.html

Attribution

Braunschweig. (2017, June 27). Web foundations word cloud. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Web_foundations_word_cloud.svg

Linking HTML Pages Together

> ## Goals
>
> By the end of this case you will:

- Have the ability to finally link all the pages you create together to create your website experience
- Have an understanding of the limits and capacity of pages linked
- Be able to differentiate between absolute and relative links and which one you need when

Introduction and Business Context

A link is simply a connection from one web page to another. Links come in different forms, you can have hyperlinks, you can have images as links and they can send users to either one of the pages on your website or to a different site.

Business Context

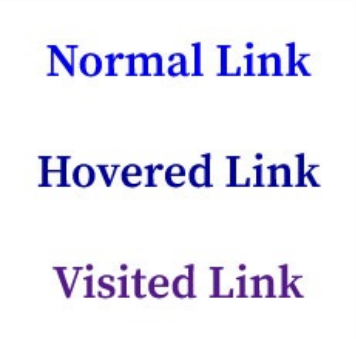
Webpages on a website are typically linked to improve the user experience. This helps provide a clear path for users to follow and can improve engagement and conversions like buy a product, sign up, install an app, share content or create an account. Users come to your site either to browse or for a specific purpose. If pages are not linked properly, it can disrupt the user experience and users may go elsewhere.

Everything about linking pages together

Figure: Duplicate Content - Author: Seobility - License: CC BY-SA 4.0

Hyperlinks are essentially HTML links, they allow you to move from one page to another, these links can be displayed in many different forms. You can choose to style them depending on how your website looks.

Text links have different styles to them allowing you to see their history. Here is an image showing the three different states of text links.



This image shows the standardized color for the normal and visited link and the hovered link is a design choice.

How many pages can we link together?

> Usually there's no limit to how many internal and external pages you can have linked. However, Google recommends 100. This is possibly due to bandwidth issues because internet crawlers that add your page to search engines could only process a certain amount of data on a page.

Anchor Tag & Link Tag

Now comes the fun part, we are going to discuss how you can actually link your pages and other files together. When it comes to connecting different types of files together we refer to different entities: the link `<link>` and the anchor `<a>` tag.

The Link Tag

The `<link>` tag helps link the current document to an external resource, it is typically used to link your stylesheets and site icons like a favicon or icons for your mobile app.

This tag is placed between the header tags `<head>` in your document.

Here is an example of how it looks like:

```
<link href="main.css" rel="stylesheet" />
```

Here is an example for a favicon

```
<link href="favicon.ico" rel="icon" />
```

The href attribute is used to identify filepath and filename.

The rel is short for `'relationship'` and shows the relationship between the current document and the one that it's being linked to.

Exercise:

Below is a basic html code for a page. Correctly use the link tag to link to your external stylesheet called: style.css

```
<!DOCTYPE html>
<html>

<head>
    <title>Link Tags</title>
</head>

<body>
    <h1>
        What’s cooler than link tags? Styled link tags!
    </h1>

<p>
    Link tags are so cool and can be used for so many things.
They make my life
    so much easier! Once I learn this I can link all of my
stylesheets to make sure
    my website looks amazing!
</p>

</body>
</html>
```

Anchor Tags

> The <a> tag is the tag that will link all your pages together by establishing hyperlinks. It can link internal as well as external HTML documents. You can also use this tag for many other purposes, some of them being an email address, telephone number or any other external URL of a website. For this lesson we will only focus on linking your HTML documents.

The href attribute here is used to indicate a link's destination. The <a> tag also has a closing <a> tag that looks like this: . This tag can be placed anywhere between the <body> </body> tags. Here is an example of an anchor tag sending users to Google.

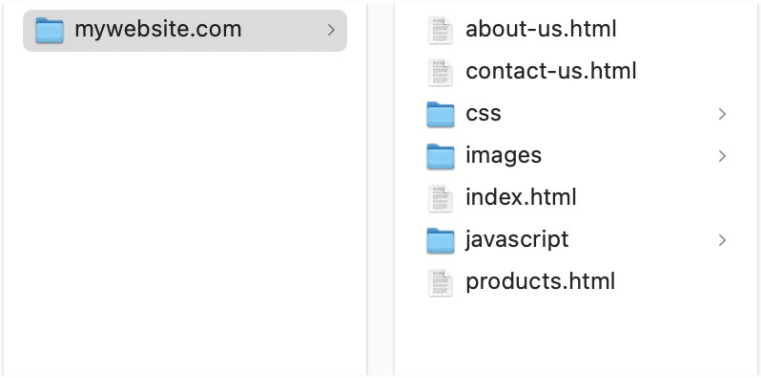
 Go to Google

The URL is case sensitive so ensure that you have the exact URL when placing it in the href. You can put text between the tags. This text will show up like this on the front end:

Go to Google

So let's connect our pages from the previous module. We're going to add a link on our index.html page sending users to our contact-us.html page. On the front-end our users are going from our homepage to our contact us page.

Here's a screenshot of our file structure.



file structure

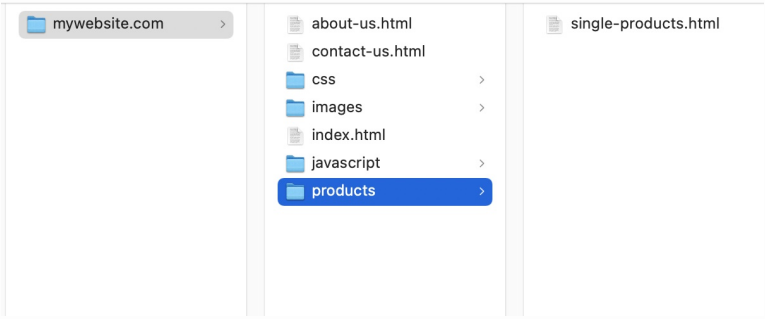
Since the index.html file and contact-us.html file are in the same folder, our file path will be simple and looks like this:

Contact Us Page

If for example the contact us page was in our CSS folder it would look like this:

Contact Us Page

Exercise:



file paths

The Target Attribute

This attribute is used to tell the browser whether to open the link in a new window or not. This is useful in case the link is sending them to a different site but you still want to keep them on your website so you can set the target attribute to _blank which will open a new tab/window. By default the attribute is set to SELF which means that it will just load your page in the same tab/window.

The target attribute is written in the anchor tag right after your href.

Go to Google

The link will look like a regular link on the front-end but will open up a new tab for the user when clicked.

Exercise:

Absolute URLs vs Relative URLs

> It's important to appropriately structure the links on your site so you can ensure that the search engines can crawl and index it. This will help it show up on search result pages, the better structured it is the better chances you have to have it show up as one of the top results.

What are absolute URLs?

An absolute URL is the full URL to the page that you link to. It contains the following:

- Protocol - http or https
- Subdomain - www
- Domain - mywebsite.com
- File Path - mywebsite.com/about-me.html

It looks like this `<a href="https://www.mywebsite.com/about-me.html">About me</a>`

What are relative URLs?

Relative URLs do not use the full address and only the file path that comes after the domain name. Since they do not use the full path, it is assumed when using them that they use the same protocol subdomain and domain that the page is on.

Here is an example of how they look like `<a href="/products/single-product/"></a>`

This relative URL specifies the path to a page called single-product that is located in a directory called products. The folder structure would look something like this.

– www.example.com
– products
– single-product.html
– about.html

In this example, the website has a products directory that contains a page called **single-product.html**, and a root directory that contains a page called about.html. Now let's say that the **about.html** page contains the following HTML: `<a href="/products/single-product/">View Product</a>` In this case, the **about.html page** contains a relative URL in the href attribute of the a tag.

This URL points to **/products/single-product/**, which specifies the path to the **single-product.html** page that is located in the products directory. When this URL is clicked on, the browser will compute the relative URL based on where the about.html page is located in the directory.

Since the about.html page is located in the **root directory** of the website, the relative URL is you are writing it for your website would be `<a href="/products/single-product/"></a>`.

When to use absolute vs relative

> A relative URL is typically used to send users within your site that contains the same protocol, subdomain, and domain name.

An absolute URL is typically used to send users to a website that is outside your server and has a different domain name.

Exercise:

Exercise:

Attribution

Link Limit on a Website | Codehouse. (2021, October 6). https://www.codehousegroup.com/insight-and-inspiration/digital-strategy/is-there-a-web-page-link-limit

The External Resource Link element - HTML: HyperText Markup Language | MDN. (2022, November 1). https://developer.mozilla.org/en-US/docs/Web/HTML/Element/link

Conclusion & Takeaways

Conclusion & Takeaways

> Understanding links and how they connect everything on your website is very important and this is something that you will be using throughout your coding journey.
>
> Links come in different formats and are very powerful to helping you provide the desired user experience and ensure that your users are getting what the need from your website.
>
> Having correct external/internal links and link structure also ensure that your website is crawled and indexed appropriately by search engines such as Google. This is very important to SEO ensuring search engines give your website a high ranking and display it to users as one of the first options. We will go into more detail on this subject in later modules.
>
> **Attribution**
>
> Link Limit on a Website | Codehouse. (2021, October 6). https://www.codehousegroup.com/insight-and-inspiration/digital-strategy/is-there-a-web-page-link-limit

The External Resource Link element - HTML: HyperText Markup Language | MDN. (2022, November 1). https://developer.mozilla.org/en-US/docs/Web/HTML/Element/link

Manage Assets Exercise

> ## Goals

Through this guided exercise you will:
- Create a file structure tree on your local machine using correct naming conventions and parent folders
- Create folders to manage your website assets
- Link your HTML pages together

Introduction

> The makeup of a website contains all different types of files such as code, media, and text content. You need to ensure that these files are assembled into a sensible structure to ensure they can talk to one another. Once it's all in place you can upload it to a server and publish your website to the web.

Business Context

> This is important for when you're building a website for a client, you build out the different pages, stylesheets, code, and media files, you have to ensure it can all communicate before you can publish the website live on the web.
>
> #### Choosing a place for your website on your computer
>
> First let's start with creating a place on your computer where it makes sense for your website to live.

Let's create a folder called mywebsite on your desktop. (you don't have to add '.com' it's not an extension, you can add it if you would like, for simplification purposes)

This folder will contain all of your website files in the correct order so you can upload to a server when you're ready.

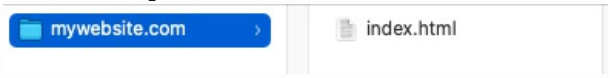
Index.html file

> Now let's create an index.html file (if you already have one you can use that). This is what the most basic index.html file looks like.

```
<!doctype html>
<html>
  <head>
    <title>This is the title of the webpage!</title>
  </head>
  <body>
    <p>This is an example paragraph. Anything in the <strong>body</strong> tag will appear on the page, just like this <strong>p</strong> tag and its contents.</p>
  </body>
</html>
```

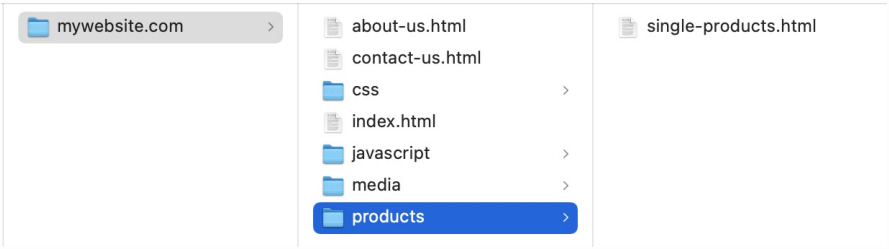
>

You want to place this file in the root folder.



All other pages for your website

> Place all your other pages like contact us, about us and the products page if you are selling products on your website will go in the root folder as well. Doing this beneficial for several reasons such as it can improve the performance of the website by reducing the number of directories that the server needs to search to find the requested page, your links are cleaner, improves organization and helps managing and updating your website easier. If you have more than 5 pages on your website you can create a sub pages folder to place them in.



In the above screenshot when you have more than one product it’s better to separate them in a different folder for cleaner file paths.

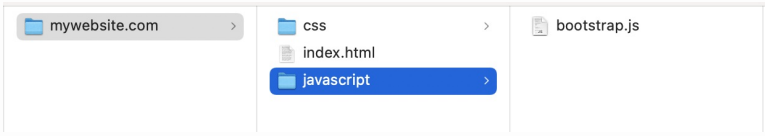
All other files and folders

> ### CSS Folder
>
> This folder will go in the root and will be the parent folder to your stylesheets. From your custom stylesheets to any CSS framework. You can upload stylesheets you may have created here.

css folder

Javascript Folder

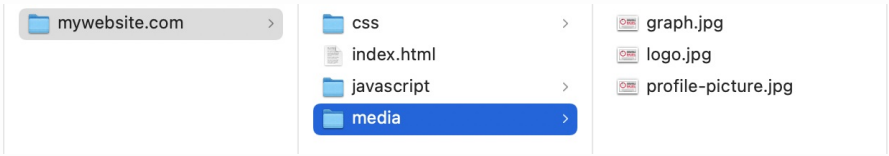
> This folder will also go into the root folder and will contain all your Javascript files.



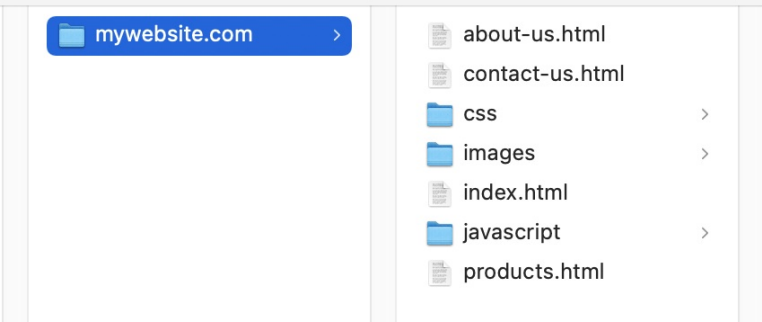
javascript folder

Media Folder

This folder will also be in your root folder. This media folder will contain all of your images, videos and gifs that go on your website.



>
The final structure of your website should look like this:



website structure

Next we want to connect your stylesheets and js files by inputting them in the header

```
<!doctype html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="mywebsite.css">
    <script type="text/javascript" src="bootstrap.js"></script>
    <title>This is the title of the webpage!</title>
  </head>
  <body>
    <p>This is an example paragraph. Anything in the <strong>body</strong> tag will appear on the page, just like this <strong>p</strong> tag and its contents.</p>
  </body>
</html>
```

link stylesheets

Now we want to connect your pages through your index.html file through a navigation menu.

```
<!doctype html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="mywebsite.css">
    <script type="text/javascript" src="bootstrap.js"></script>
    <title>This is the title of the webpage!</title>
  </head>
  <body>
    <div class="nav-menu">
        <a class="active" href="index.html">Home</a>
        <a href="about-us.html">About Us</a>
        <a href="contact-us.html">Contact Us</a>
        <a href="products/single-products.html">Products</a>
    </div>
    <p>This is an example paragraph. Anything in the <strong>body</strong> tag will appear on the page, just like this <strong>p</strong> tag and its contents.</p>
  </body>
</html>
```

connect pages

This gives us a basic index.html webpage that has all the pages, external CSS and Javascript connected.

Conclusion & Takeaways

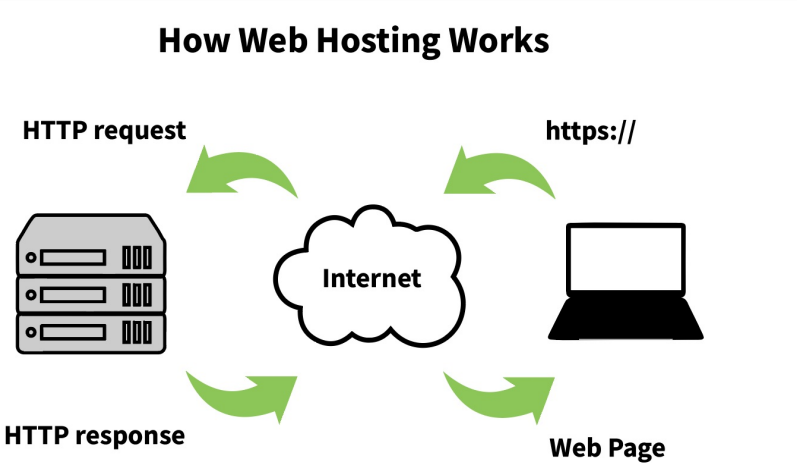
- It's important to remember that when connecting pages and media together through HTML documents that you keep their files paths in mind.
- When organizing your files you want to ensure that you organize them in a sensible way through the use of folders so you maintain clean file paths.

Basic Web Hosting & DNS

> ## Goals
>
> By the end of this case you will:
> - Understand how DNS and hosting works
> - How to choose the right hosting for your website
> - Understand the benefits of the hosting control panel and how to use it
> - How to bring your website to the world wide web
>
> ## Introduction
>
> 
>
> server room

CC0 1.0
>
I'm sure you've seen this image in movies. It looks really cool with all the computers and lights flickering.
>
These are **servers**. Every website on the web lives on a server. As you can see from the image above, those are computers: most servers are computers. When you purchase web hosting, you're renting space on one of these computers from the providers in order for you to be able to build and maintain your website.
>
Business Context
>
It's important to understand how servers work and how much they can handle. When building your website or a client's website, depending on the ultimate goal of the website (ecommerce, blog, portfolio) the hosting space required will vary. You want to make sure that if it is an ecommerce

website that you want more server space because if the hosting space is smaller and you get a large volume of users that the server can't handle, the servers can crash, causing all your customers to not be able to access the site until they are back up and running.
>
What is Web Hosting?
>
Here is a very simple way to visualize the process of web hosting
>



>
You can think of web hosting as a renting storage place for your website. It holds all the files for your website and allows other computers around the world to be able to access your website. Your file is stored on servers that the hosting company owns.
>
In the graphic above you can see that the computer makes a request, the internet delivers that request to the server, and the server finds your website and sends it back through the internet to the computer that requested it.
>
What is DNS and DNS hosting?
>

>

Domain Name System

The Domain Name System (DNS) has been coined several times as being the phonebook of the Internet. All websites have a domain name like www.google.com or www.correlation-one.com. When a request is made from your computer to the servers and the server sends the website to your computer, that's through **Internet Protocol (IP)** which is a set of rules for routing and addressing packets of data so they can travel across networks. (Cloudflare, n.d.)

>

Every device that is connected to the internet is assigned an IP address (for example, 199.117.41.9) and once the server sees your IP address in the request it routes the package to the requested IP so you can load the website.

>

Without DNS you would have to type every website's IP address instead of the name.

>

DNS Hosting

> So let's start with **nameservers**. Nameservers run your DNS software. The DNS information for your website is stored on these servers and they are hosted by a DNS hosting company.
>
> A DNS hosting company builds and takes care of these nameservers. These servers provide common DNS functionality like:
>
> - Domain translation - translates domain names like (www.example.com) into numerical IP addresses that computers can read (192.22.556)
> - Domain lookup
> - DNS forwarding - forwarding queries to specialized server to improve network performance
>
> ### Difference between Web Hosting and DNS hosting
>
> DNS hosting is similar to web hosting with some small quirks, not many web hosting providers also provide DNS hosting as part of the package. As with all website related purchases, you should always consider how much support, uptime and power you'll need.

Exercise:

How to choose Web Hosting Server and Web Domain

As we spoke in the earlier sections, for you to be able to display your website on the web you're going to have to be able to find a web server that can handle your website and your needs.

When picking a hosting provider you first need to ensure you know what you want and how your website is going to look once you start to scale.

Here's a simple list to follow when choosing a web hosting server.

1. Always start with what your website needs - you need to be able to map out what you will be doing with your website once you have it up and running. Will you be building it on Wordpress, or another CMS vs building it on a website builder. All of these options will influence the type of hosting service you decide to go with.

2. Start doing your research - this is the critical part in choosing a provider. Once you have an idea of what you need your website to do, you need to go find a hosting provider that can offer it. Here are some key items that you should focus on depending on your needs:

a. Uptime - this means how long your website stays up within a given period of time. You want to ensure that is 100% or as close to it as possible because even a couple hours of downtime will result in your visitors not being able to access your website.

b. Support - this is important to ensure that if your website gets hacked or goes down, you need to be able to contact someone immediately to get it back up so you don't lose customers. The level of support provided varies by provider with some you can call 24/7 while with others you can only chat, so choose wisely.

c. Pricing - the good thing here is that hosting parties have the cheapest pricing available. You can choose to have a plan ahead of time for the whole year or pay all at once for multiple years.

d. Features - understand the type of features you need and compare them with the other providers. Features like how much storage (how many files/websites you can have), how much bandwidth (be able to handle a lot of traffic, if new site you don't need too much) and RAM (storing data and running programs, the more programs you run the more RAM you need)

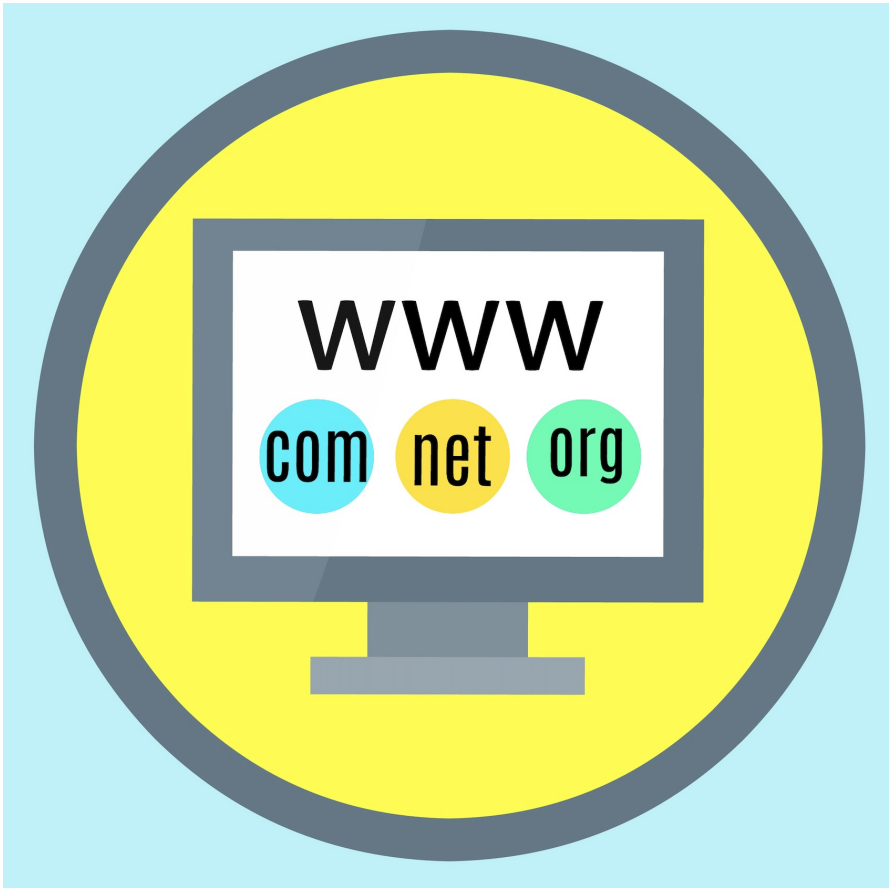
3. Security - You want to ensure that the web host you pick has security measures set up to prevent from getting hacked. Some of these features are DDoS protection, Firewalls, Virus Scanning and Brute force defense.

Overall choosing a web hosting provider is a mix of doing your research and knowing what your website needs. Once you can establish your needs for the present and potential future growth, it will help you decide on the pricing, features and different packages you might need.

Exercise:

How to choose a web domain

How to choose a web domain



>
Choosing a domain name is extremely important to the success of your website, if you choose a domain name that you don’t end up liking, it can be a hassle to switch it in the future.
>
Here are a few tips on what to do and avoid when picking a domain name:
>
Do’s:
- Stick with .com or .org - these are more widely known, more trusted and more credible. They are also more SEO friendly.
- Shorter the better
- Easy and brandable
- Easy to type
- Easy to pronounce
>
Don’t:

- Avoid numbers
- Avoid double letters
- Do not use hard to spell words
- Do not use trademarks
>
If you get stuck you can always use domain name generators which help you come up with creative names depending on your field of business. There's no wrong domain name, just ensure that it's simple and it's one that represents your business appropriately.
>
Exercise:

> **Different types of web hosting**
>
> 
>
> web hosting
>
> Now that you're aware of what a web server is and how hosting works, you need to decide what type of hosting is best suited for your website. Here are the 7 different types of hosting available to you:
>
> **Free Hosting**
>
> Free web hosting is available if you search for it on the web but it comes with a few quirks compared to paid hosting. It is fairly unstable so you can run into downtime, and restrictions are placed on bandwidth limiting your site staying up if visitors increase.

Free web hosting is useful for a personal project, or when you're practicing your web development skills. It also comes in handy if you want to run some plugin tests and run some programs to see what they can do. Free web hosting is useful for a staging site but not ideal for a live one you want to grow.

Shared Hosting

This type of hosting is ideal for entry level websites when starting off. It's cost-effective because websites are sharing resources on the one server, they are essentially 'sharing' the server. This allows for the low price point since most websites don't need their own server.

The only drawback would be if one of the websites has a major increase in traffic it would use up most of the resources of that specific server therefore causing performance issues with the other sites sharing their server, including yours.

The rule of thumb when choosing this web hosting would be if you're not expecting more than 10,000 - 20,000 monthly visits you can save money and select this plan.

Exercise:

Virtual Private Server (VPS) Hosting

> VPS is when a company has a server and they use virtualization technology to split that server into multiple virtual servers. When you purchase VPS hosting you are getting your own virtual server so you're not sharing RAM, CPU or any data with other users.

VPS is simulating a dedicated server experience even though you're still sharing the physical server with others. With this specific type of hosting there is a higher demand of technical knowledge required because unlike with Shared hosting you don't get access to a control panel and might have to use the command line. VPS is still affordable but will require more technical know-how and understanding the technical demands and resources of your website.

Dedicated Web Hosting

This type of hosting is exactly as it sounds like, you get a dedicated server that is reserved for your website only. You don't have to share resources with other website owners. You essentially get 100% control over your resources and can customize your software, but it is much more expensive.

This type is hosting if typically for websites that get a lot of traffic throughout the day and websites that require custom operating systems along with custom software. Compared to shared hosting that usually cost 2-10 USD a month, this would cost you almost 100 USD or more a month.

Cloud Based Web Hosting

Cloud hosting uses the same concept as VPS which is virtualization. It splits a physical server into multiple virtual machines called cloud servers. Then these virtual machines connect together to create a single network for hosting a website.

There are many benefits to this such as your website will always keep running even if a server fails, the others will cover for it. When you get a spike in traffic, it can handle the load because the traffic is spread across the cloud servers and it's easier to scale it because there's no limit on bandwidth, storage or computing power available from a single server.

This is typically the kind of setup that big companies like Netflix use for because you can also customize it to your liking. If you don't know how to configure a server or don't have a massive site to run, this setup will not be

useful for you.

Exercise:

Website Security

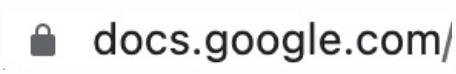
security hacker

According to security magazine a hacker attack occurs every 39 seconds in the US affecting one in three Americans every year. It is extremely important to ensure that your website remains secure because all of the hard work and personal data that your data has can be lost as soon as a hacker gets in.

Some of the ways to do this are:
SSL Certificate
SSL stands for Secure Sockets Layer, it is a digital certificate that authenticates a website. Once it authenticates the website it enables an encrypted connection. It's the difference between a URL having HTTP vs HTTPS in it. If a website has an SSL certificate it establishes a security protocol that creates an encrypted link between the web browser and server.

If you're collecting customer information or doing any type of transactions, you need an SSL certificate to ensure everything is private and secure. With this certificate search engines and browsers don't flag your website and trust it since it creates an encrypted connection. This also reassures your visitors that their data is safe and not at risk of being stolen.

When you see the padlock beside a URL
it means the website is protected by SSL.



padlock image before address

Exercise:

Anti-Malware Software

> If you've ever been anywhere on the web you've probably already heard of anti-malware by now. Anti-Malware is a software program that protects IT systems and personal computers from malicious software. You can run the program and it goes through your computer and website to look for any potential threats and removes them. This is useful to have on your website to add another layer of protection. Anti-malware on a local level is installed on individual computers to protect them versus anti-malware on server level is installed on a server that hosts multiple websites or services and protects the server and all hosted websites.

Extra Security Precautions

There are some extra precautionary steps you can take to ensure your website stays secure:

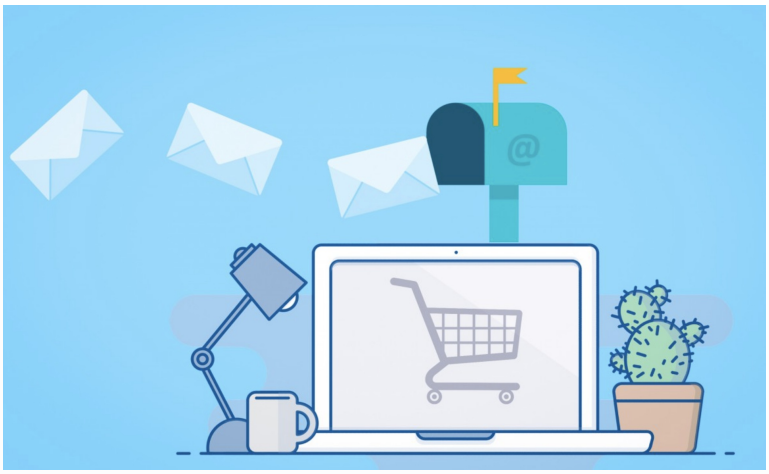
1. Create strong passwords - you always want to ensure your passwords are strong enough to not be guessed or hacked.
2. Update your plugins as often as you can - when you have plugins on your site like Wordpress, the creator of the plugins fix bugs and launch newer versions to prevent breaking or hacking of plugins.
3. Backup your site - this one is crucial in case something does happen you have a backup that you can use to restore your site.
4. Ensure whoever is accessing your backend is trained and understand cyber security best practices.

You can never truly be 100% safe on the web with new technology always coming out, that's why it's always good to ensure that you have covered your bases so that even if somebody tries to hack your website, you're ready!

Exercise:

Exercise Multiple Choice:

Email for your website



website email

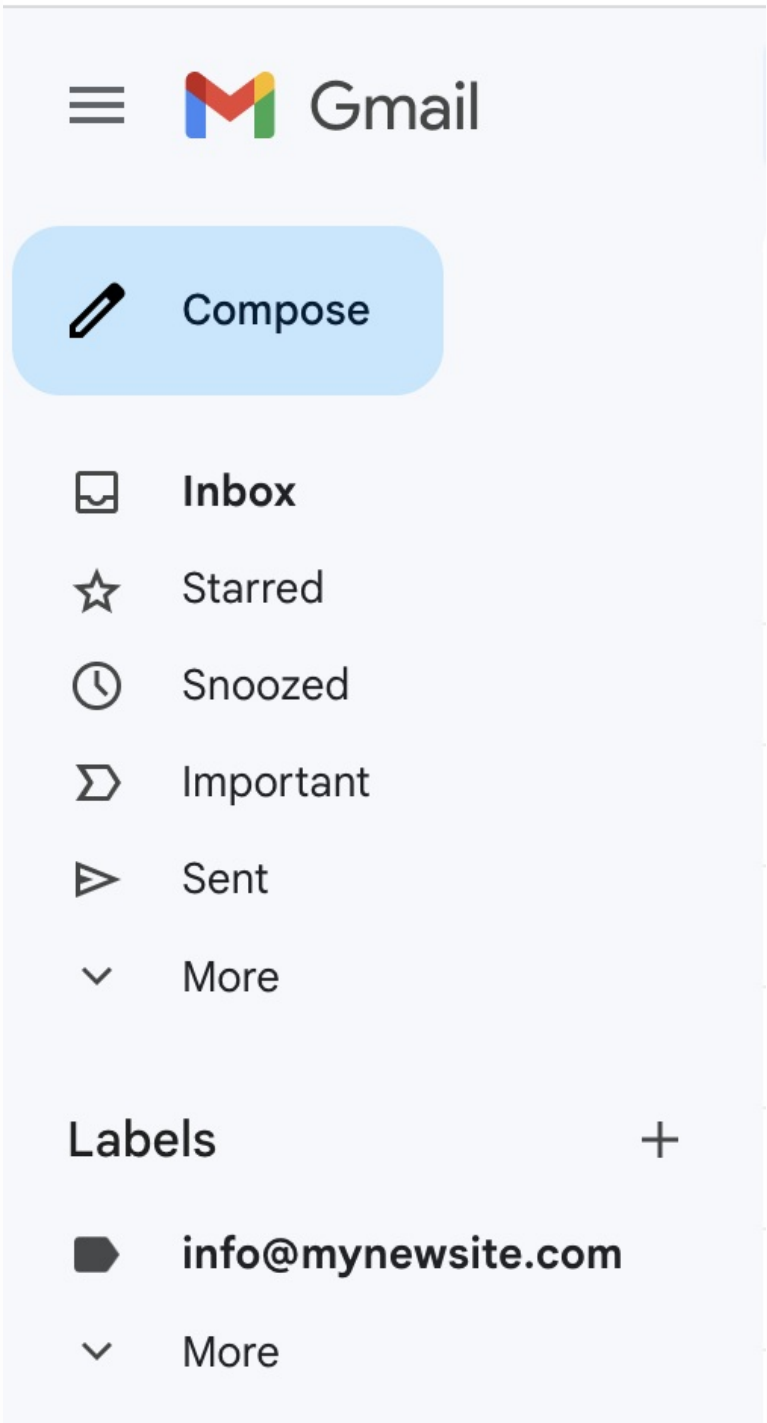
An email for your website is something that's not mandatory but good to have especially if you're going to be running a business and therefore communicating with customers and allowing them to communicate with you.

The way to do this is through email hosting. This is an optional service that is usually provided by your hosting provider when purchasing hosting. You can get it in a bundle so it's cheaper or you can get it later if you change your mind. You can also use separate providers for your email hosting if you choose.

Business email hosting is different from free email hosting like Microsoft Outlook and Google Gmail. With business email hosting the hosting provider gives you server space for your emails and you can use your personalized domain name in the email address. So if your domain is www.newsite.com then you can create an email address like info@newsite.com. This adds professionalism to your business communications.

Once that's decided you get two options for your email client.

1. You can use the hosting provider's webmail online application (this makes it tough for quick mobile or desktop access)



gmail compose

2. You can set up your business email through Outlook or Gmail. The hosting provider gives you the basic setup instructions to follow and once you set that up, you can receive regular emails through your Gmail or Outlook. It will look something like this screenshot on the right one it is setup.

There is also another side to hosting your email and we can look at some disadvantages of taking this route:

1. If the server goes down so does your email, so ensuring that you have a good hosting provider is key for this.
2. The setup takes a little bit of time and basic technical know-how.
3. If you change hosting providers, migrating your email can be stressful and you might experience downtime during this phase.

In the long run even for a personal website it is usually recommended to get the email hosting so you can separate business inquiries with your personal. The added benefit is that it will also look extremely professional adding to your credibility.

Exercise:

Conclusion & Takeaways

> - The importance of understanding web hosting and everything that comes along with it is key to running a successful website that ensures you can deliver an excellent experience to your customers.
> - Choosing the right host and understanding the different features of the hosting packages allows your website to scale as your business grows, without you having to replatform or change packages.
> - One of the most important parts of a website is that it is trusted by visitors and search engines so it can show up in results and browsers don't flag the website when users are visiting, so understanding security and certificates goes a long way in running a successful website.

Further Reading

What Is Web Hosting & How to Choose the Right Hosting Provider. (2022, July 26). Fit Small Business. https://fitsmallbusiness.com/what-is-web-hosting/

Cloudflare. (n.d.). What is the Internet Protocol? | Cloudflare. Cloudflare. Retrieved November 16, 2022, from https://www.cloudflare.com/learning/network-layer/internet-protocol/

Carney, L. (2022, November 9). How to Choose a Web Host: 7 Top Tips for Finding the Best Provider. Website Builder Expert. https://www.websitebuilderexpert.com/web-hosting/how-to-choose-a-web-host/

Lofgren, L. (2021, December 3). The 6 Best Types of Web Hosting to Consider in 2022. Quick Sprout. https://www.quicksprout.com/types-of-web-hosting/

What Is a VPS? A Beginner's Guide to Virtual Private Servers. (2022, June 3). Website Guides, Tips & Knowledge. https://www.dreamhost.com/blog/beginners-guide-vps/

G., D. (2022, November 8). What Is Cloud Hosting? Hostinger Tutorials. https://www.hostinger.com/tutorials/what-is-cloud-hosting

Hackers Attack Every 39 Seconds. (2020, May 24). 2017-02-10 | Security Magazine. https://www.securitymagazine.com/articles/87787-hackers-attack-every-39-seconds

Rosencrance, L. (2021, March 31). antimalware (anti-malware). SearchSecurity. https://www.techtarget.com/searchsecurity/definition/antimalware

SEO

> **Goals**

By the end of this case you will:
- Understand the importance of SEO for your website
- Learn the different techniques for SEO
- How to optimize your site with SEO

Introduction

> 
>
>
> (Ogden, 2018)

Let's start this section with an example, if you go to google.com and type in 'new laptops for sale' you will be greeted with a bunch of search results. For websites and businesses it is important for them to show up on this first page and if they do, that means that they have implemented a successful SEO strategy.

Business Context

A well executed SEO strategy is extremely important in helping you attract customers to your website organically because it will put you on the first page of the search results. As a business you want to execute your SEO strategy through the mind of your customer and how they will be able to find you through a simple google search. Without customers being able to

find you it will be harder for you to grow your business and make sales. A successful SEO strategy is important to the success of your website and your business.

What is SEO & why it matters?

SEO stands for Search Engine Optimization. The meaning of this is optimizing your website to increase its visibility and help it rank at the top of search engine result pages (SERP). The higher your page in SERPs the better.

This is important because it helps users find your website. These users can be your potential customers, investors or employers.

Implementing SEO successfully is an ongoing process due to search engines always improving the way websites are shown in the result pages to ensure customers are finding what they need quickly and efficiently.

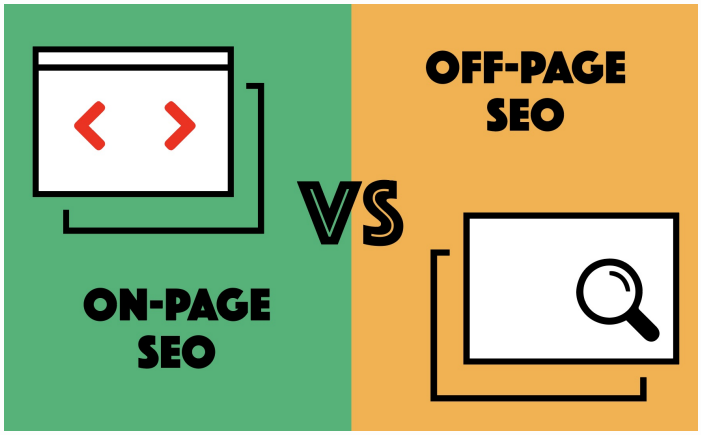
How does SEO work?

There are many different activities that you need to do in order to see successful results. Some of these activities include:

– Doing your keyword research and finding relevant keywords for your business with high search traffic potential
– Creating quality content embedded with those relevant keywords so it is optimized for search engines
– Adding external site links and creating internal site links
– Monitor, track and measure results. Adjust accordingly.

All these little technical things are basically telling search engines what your website is about and why it should rank it higher in the SERPs. We will go more into detail regarding each of these.

On Page vs Off Page SEO



types of SEO

On page and off page are primarily two different types of SEO strategies.
- **On-page SEO** is focused on optimizing your website and different areas of it that are under your control. **On-page** = **within your site**
- **Off-page SEO** is focused on strategies and factors that are off your page like promotion, getting other sites to talk about you and link building. Off-page = outside your site

The best way to look at this is **what you rank for** is determined by on-page factors and **how high you rank** is the search results is determined by off-page factors.

On-page SEO

Now we know that on-page SEO is optimizing areas of your website to positively affect appearance on search engines. This tactic involves not only optimizing your content but also the HTML source code of your page.

Here's a list of the different factors required to do this, we will go into more detail for each:

- Optimizing URLs
- Title tags and meta descriptions
- Header tags
- SEO copywriting (content)
- Keywords and keyword research
- Internal linking and External Linking
- Image optimization

Optimizing URLs

According to Google URLs help their web crawlers better understand everything about the page. So we have to make sure that we optimize our URLs, here are a few ways to do that.

- **Include one of your main keywords in the URL** - if a web page is about baking cupcakes then include the word ‘cupcake’ in the URL, it helps users and search engines understand the content of the page quickly.
- **Avoid numbers** and use descriptive words that focus on your content
- **Keep URL short and simple** - easier and quicker to understand
- **It’s okay to use hyphens** in between words to simplify and make URL more readable. Usually, when an organization’s primary domain name has a hyphen, they will also purchase the domain without a hyphen to redirect to the primary domain name. E.g., `correlationone.com` redirects to `correlation-one.com`.

Exercise:

Title tags and meta description

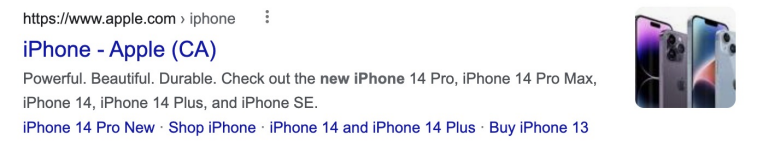
These are called meta tags and they are embedded in your code, and they are the most important factors for on-page SEO.

Here is an example of the first Google search result for `‘red velvet cupcakes’`:

cupcakes search

The title tag is the headline that appears in blue. And the meta description is the short summary below it in a paragraph.

Here is another example if you type in ‘new iphone’ in Google.



iphone search

As you can see the title tag there is in blue ‘iPhone - Apple (CA)’ and they kept their meta description below short, precise and to the point. Just by seeing the title and meta tags you can quickly understand what the page is about.

We can do a quick Google inspect of the Apple page to view the source code and here is how it looks like:

```
<title>iPhone – Apple (CA)</title>
<meta name="Description" content="Powerful. Beautiful. Durable. Check out the new iPhone 14 Pro, iPhone 14 Pro
Max, iPhone 14, iPhone 14 Plus, and iPhone SE.">
```

inspect page

Here are a few good rules to follow when working with meta tags.

- The title should stay within 55 - 60 characters so it doesn’t get cut off in the SERPs.
- The meta description should stay within 155 characters so it also doesn’t get cut off in the SERPs.
- The title should contain your most relevant keywords.
- Avoid making the title or meta description all caps.
- Include your most important keywords in the meta description so Google will bold them in the SERPs.
- Use unique titles per page so Google doesn’t mistake them as duplicates

After creating these, you can monitor performance and adjust accordingly.

Exercise:

Header Tags

> Header tags come from your content. The way your content is structured with different types of headings ranked by importance. You can use multiple headings on your page (H1, H2, H3, etc.) and it helps with SEO since it makes it easier for users to go through your content at a glance and creates a better user experience (UX) which is an important factor for Google).

It is a good rule to use your relevant keywords in your H1 and subheadings H2. If it flows well you can use those keywords in other heads, this will help with Google since they are crawling the website they will be able to quickly understand the content of the page.

Here is an example of how Apple did it on the same iPhone page in the example earlier.

```
<h1 class="pagetitle">iPhone</h1>
```

```
▼<h2>
   <span class="violator-frameless violator-frameless typography-hero-violator">New</span>
   <span class="visuallyhidden">iPhone 14</span>
 </h2>
▼<h2>
   <span class="violator-frameless violator-frameless typography-hero-violator">New</span>
   <span class="visuallyhidden">iPhone 14 Pro</span>
 </h2>
```

You can have multiple H2 tags but you can only have one H1 tag because it correlates with the title of your page, as you see from the example above the H1 tag is just ‘iPhone’ because that’s what the page is about and they have multiple H2 tags to talk about the different versions of those iPhones.

SEO Copywriting (content)

Copywriting is essentially writing content in a specific way to influence users towards your product and increase brand awareness. This is an important factor in on-page SEO because it will better engage users and also rank higher in Google SERPs. Here are some guidelines to follow when creating your content:

- Keep the introductions short and to the point.
- If your section goes over 300 words, break them with subheadings
- Embed the researched keywords into your content as frequently as possible
- Google ranks results based on search intent, so write your copy based on that intent

- Remember to avoid long sentences, break them into smaller easily digestible sentences

Keyword Research

seo keyword

Keyword research is the process of using different tools on the internet to find what people are searching for related to your business. Once you get a list of the most searched keywords for your business you start to incorporate them into your content.

This process helps you understand what your users are searching from long tail to short tail keywords.

Long Tail Keywords

These are longer search terms that your users search for with less queries but more intent behind them since they are more specific. They are typically 3 or more words. For example ‘size 11 red running shoes for men’ in this sentence they are being very specific about the size, color, purpose and who they are for. If a user comes to your site based off long tail keyword search, they have a much higher chance of converting.

Short Tail Keywords

These are the opposite of long in terms of they are 2 or less words and have a much higher volume of search queries. Since they are generic and have a higher volume of search queries, they are more competitive to rank higher for, therefore leading to less conversion. Following the same example as

above, the user would type in ‘red shoes’ and now you would be competing with bigger companies and many more people since the size, purpose for shoes and who they are for is not specified.

Overall for keyword research it is important that you focus on your business and think like your customers would in terms of searching for your website/products. Keyword research is always evolving and is a constant process, this makes it part of your long term content and marketing strategy.

Internal Linking and External Linking

> Internal linking is the process of linking to other pages on your website. This is helpful for your visitors and search engines. Internal linking helps create a structure of your website and creates a web of links from one page to the other while helping you with your on-page SEO.

With internal linking, ensure that the anchor text to the page you are linking to is relevant. Here's an example of internal linking. If our website is about sports and we are talking about basketball and reference basketball shoes for which we have another page, we can link it like this:

```
This year NBA players are all wearing these new [basketball shoes](http://www.nba.com) that help with grip and stability on the court.
```

Which will look like this on the web page:

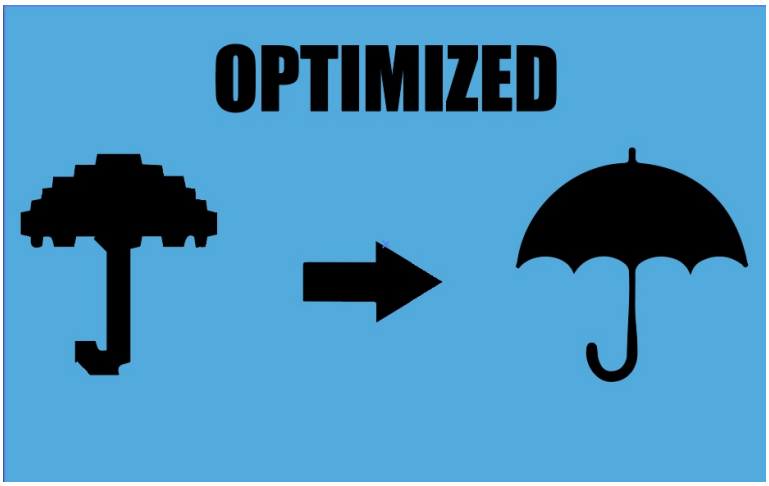
> This year NBA players are all wearing these new basketball shoes that help with grip and stability on the court.

When you click that link it should send users to a section about these new basketball shoes on your site. That's an internal link!
Exercise

External linking is when you are sending people away from your website. This seems counterintuitive but it contains some benefits that can help you rank higher. External links give your website more credibility when citing other websites, and they provide value to your website by giving users references and more high quality content to read. Search engines also associate you linking to high quality websites as a positive and uses those link to know what your website is about.

When you link to other website and they find out, they might even link back to you, helping you create backlinks (which we will discuss in the next section) and the more backlinks the better!

Image Optimization



optimized

This process involves several steps to optimize images. We have to ensure that:
1. They are not large in size since they will cause the page to load slowly.
2. When we do decrease the size, we don't lose resolution
3. They are in the correct format so they load quickly and efficiently on all devices and internet connection speeds - The ideal format is usually JPG format since it compresses well and occurs minimal quality loss. Ideally images should be less than 350kb to not impact page load time. In case of banners and hero images, to retain image quality, a higher size might be required. You can use web software like TinyPNG to help you quickly compress the image.
4. The meta-data or called `'alt-text'` for images is text that you write about the image to give it context for search engine crawlers. This is not visible for users on the front-end. This also helps you be compliant for the WCAG (Web Content Accessibility Guidelines) which is guidelines for screen reader software to help blind internet users.

Off Page SEO

> Now we can discuss off page SEO which like we spoke about earlier is the efforts you make outside of your site.

Here is a list of some of the different factors required for successful off-page SEO.

- Backlinks
- Social Media
- Google My Business (GMB)

Backlinks

These types of links are ranked as one of Google's top 3 ranking factors. They are essentially when other websites link to your website. The amount of backlinks you have and the quality of the backlinks helps Google determine the value of your page. The more websites/web pages that link to your website, the better and of higher value Google sees your website and therefore ranks it higher in SERPs.

The quality of these links is determined by how prominent the website that is linking to your website is. The anchor text is also very important and has to be relevant to the content of the page. The placement of the links on the page affect the authority (the higher in the page and content the better).

Social Media

social media

Social Media contains influencers that have a lot of organic reach with a post depending on the amount of followers they have. The way you can utilize this to your benefit is to find influencers that appeal to your target

market. You can reach out to them and have them make a post, review and talk about your product with their audiences.

You can also create shareable posts, advice and images that users can post on their social media and this also helps you create awareness, generate traffic and possibly create even more backlinks.

Google My Business (GMB)

GMB

Creating and maintaining a GMB listing has become extremely important ever since Google made an update to their search algorithm that prioritized local results depending on the user's location. This doesn't affect your rankings but it definitely helps you be visible to customers in your area if you're offering a product or service.

Exercise:

SEO Tips

Now you know how to implement the different SEO tactics. Since SEO is a constant process needing tracking and adjustments. You can simply use Google Analytics (GA) and Google Search Console (GSC) to help you do this. Once you set up an account and register, GA provides you with a code you

can inject into your website, this allows GA to track numerous metrics. For Google Console you can go here and register and get started. Some of the metrics that you should be looking out for in GA and GSC are:

- Organic traffic - this is when a user gets through you by typing in a search query and clicking your website from SERPs. From the organic data in GA you can go into more detail and look into direct (people directly coming to your website by typing URL or if they have your website saved in favourites), social (social media channels) or paid (paid ads that you sponsor your website with). Looking at this weekly can help judge wether your SEO tactics are working or not.
- Page load speed - Google ranks pages higher the faster they load, longer load times increase bounce rate (people coming to your page and leaving without any interaction) which is not good and decreases the rank of your webpage. To ensure your page is loading past you can simply paste your URL in one of the many free softwares that measure page speed. Here's a good one. GTmetrix
- Keyword ranking - you can do this by just searching for specific keywords and seeing where your website is ranking. For deeper analysis you can check out multiple free keyword rank checkers. Here's a good one. Ahrefs.
- Click Through Rate (CTR) - this number represents the percentage of users that clicked on your website from the SERPs. This helps to understand how good your title tags and meta descriptions are. For example the average CTR for a website that ranks in the first position on Google is almost 30%, so don't worry too much if you see lower numbers.

As your SEO strategy becomes more sound, there will be many other factors that you can track to measure the effectiveness of it and how well your website is ranking. This is a good place to get you started.

Conclusion, Takeaways & Attribution

> - With more individuals and businesses understanding the importance of having an online presence, SEO has become much more extensive and detailed. The competition has risen and businesses everywhere aim to get their website to the top of the search results. The businesses that differentiate themselves are ones that focus on their customers, their target market and aim to build a strategy that reaches them directly instead of having a broad strategy. The more personalized your strategy the better.
> - Implementing SEO is a process and takes time, there are many things that you need to do to ensure your on-page and off-page SEO is sound. There is no wrong way of doing it, just ensure that you are tracking, analyzing and readjusting as needed.
> - There will be newer technology in the future and updates to search engine algorithm that will affect the results they show, always keep an eye on newer tactics and technologies that you can leverage to ensure your website ranks well on the SERPs.

Attribution

Ogden, L. (2018, October 10). SEO Word Cloud. Flickr. https://www.flickr.com/photos/bitsfrombytes/45219987101

Further Reading

Varagouli, E. (2020, October 29). How To Measure SEO Performance and Results. Semrush Blog. https://www.semrush.com/blog/seo-results/
1Zlatin, G. (2022, July 20). On-Page vs Off-Page SEO: What's the Difference? Digital Third Coast | Digital Marketing and Search Agency. https://www.digitalthirdcoast.com/blog/on-page-off-page-seo-difference
Hinds, J. (2022, October 15). On-Page SEO: Big Guide to On-Site Optimization (2022). Terakeet. https://terakeet.com/blog/on-page-seo/

Create your Personal Website

> ## Goals
>
> By the end of this case you will:
> - Have implemented a file structure tree to ensure sensible structure
> - Linked HTML pages to create a website
> - Incorporated SEO learnings

Introduction

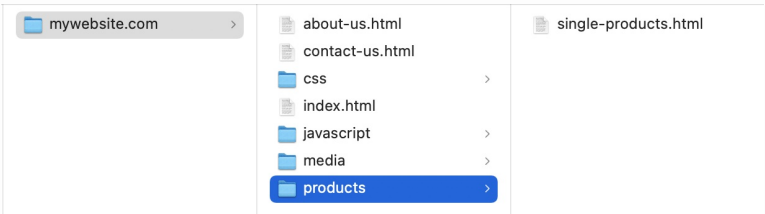
> There are many components that go into building a website and to ensure that it is built in a way where the file structures and HTML pages are linked properly. Implementing SEO is also an imperative part ensuring a successful webpage.

Defining web pages

> Let's start with defining the web pages:
> - **Homepage** - index.html
> - **Contact Us page** - contact-us.html
> - **About Us page** - about-us.html
> - **Products page** - if there are multiple products you place it in a folder with the different products pages - products/single-products.html & products/multiple-products.html

Creating a file structure using folders

> Now that you know where your CSS, JS, media and page files should go, your folder structure should look something like this:



file tree

What pages should be linked together and how?

> After you've got your pages built you can connect them through a simple HTML nav and link them using the correct file paths like this

Adding SEO learnings to webpage

> To finish off your basic personal website - ensure that you add all the relevant SEO tags like H1, H2, title and meta description. Here's how it should look in your index.html page

adding SEO

This is the basic foundation of your website that you can launch. Ensure that the nav and the SEO meta data are also added to the other pages and that file paths are correct. Then you have yourself a basic functioning website.

Conclusion, Takeaways & Attribution

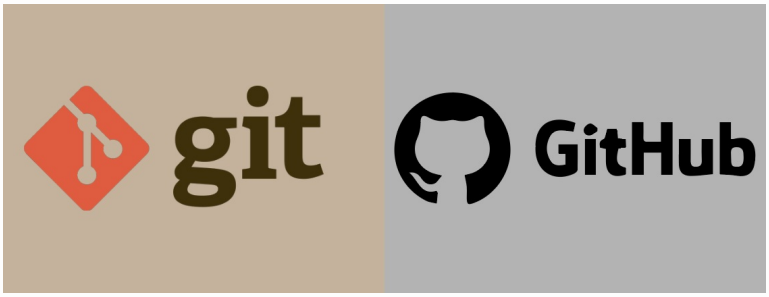
> - It’s important to first define the webpages of your website and have an idea of where they are going to go.
> - Creating a file structure helps you understand how you will link the different files together
> - Linking the web pages through a nav using anchor links or linking them through links in the content
> - Finally applying SEO settings and meta data to pages to ensure they are indexed correctly by search engine crawlers.

Git and GitHub

Goals

By the end of this case you will:
- Understand Version Control Systems and what they are
- Know everything about Git and Gihub
- Be able to install and use Git

Introduction



git and github

When you're designing or working in any type of Microsoft documents from Word to Excel, you always have the undo or redo options, you can even view older versions of your documents prior to making changes. This is really helpful in case something goes wrong or somebody wants to review your work. In this section we will be talking about software tools that help you do just that when it comes to coding. Tracking and managing changes to your code.

Business Context

These tools are extremely helpful when working at a company in a team environment where multiple developers are involved with the source code. Having the ability to see exactly what changes were made, by who and when is imperative to ensure that the website doesn't crash and the code doesn't conflict. As a developer, your knowledge of these software is key to your success.

Everything about Version Control System (VCS)

What is a version control system and why is it important?

Source control is also known as version control and these systems are tools that help you track and manage the changes to your source code. It keeps a log of all the changes that were made along with earlier versions of the source code, this is extremely helpful when working within a fast paced development team and environment.

The software has a specific database that keeps all the changes that were made so this helps developers go back and review the code to potentially find and fix the mistake.

Now imagine that there is a team of 5 developers working on a company's website. Two of them are working on bug fixes and the other three are working on new features all at the same time. The problem arises when they write the code for the feature or find the bug for the fix, the code that they have written might not be compatible with the new features or the fix might affect the website so the new features might not work. The goal of the version control is to ensure that the source code is not impacted. This happens by organizing the code for the software, app or website into a 'file tree' and that tree has branches that are assigned to each developer allowing them to all work on the same project, app or website at the same time.

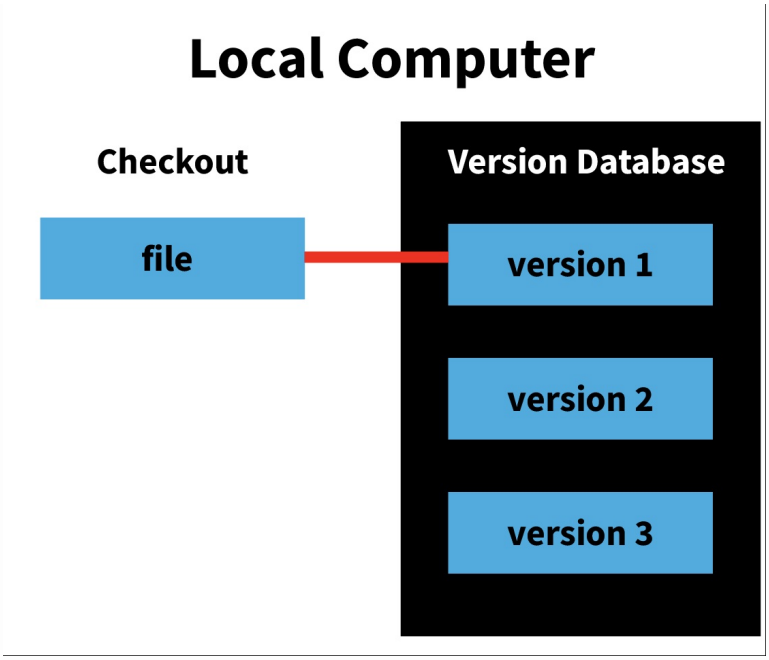
Version control stops work happening at the same time from conflicting. When bugs or conflicting changes are discovered they can be resolved without interfering with the work of other developers.

Exercise

Different types of version control systems

> There are three different types of version control systems:
> - Local VCS
> - Centralized VCS
> - Distributed VCS

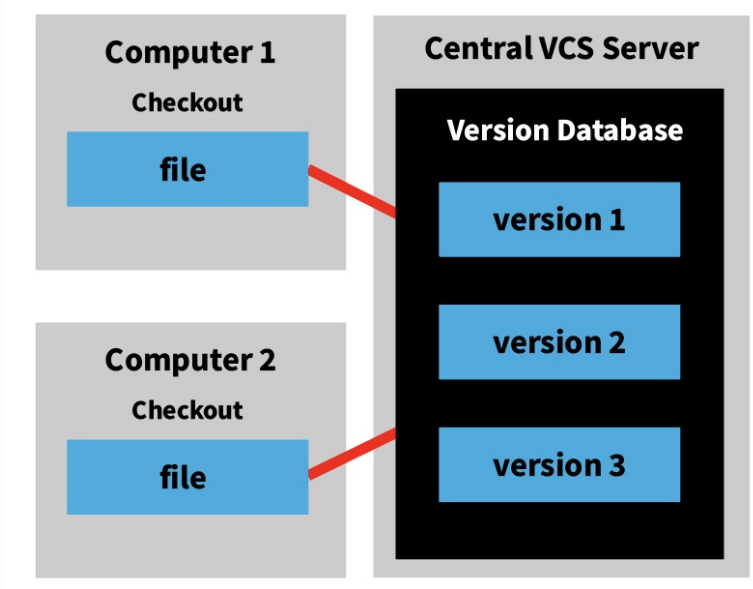
Local Version Control System



file versions

This is when you manage all the files on your own computer system - locally. All the changes you make are stored locally as patches. Every patch contains the changes made to the file since its last version. The problem with this is that it is easy to lose all files and changes if anything were to happen to your local system.

Centralized Version Control System



centralized version control

In this system there is a server that contains all of the files and versions and the developers pull the files onto their local system, make the changes, commit and upload it back onto the server (otherwise also known as repository). The issue with this system is similar to the local VCS, if something happens to the server and there are no backups then the change history can be lost.

Distributed Version Control System

> In this system every developer has a local copy of the whole project on their system. Since its on their local system they can make any change and commit to see how it turns out and it won't affect the remote server. Once they are done, they can push these changes to the remote repo. This is useful if anything happens to the server, any of the developers can send the files back onto the server to get it back up and running. Git is a distributed VCS.

To summarize, in a centralized VCS when you commit the changes everyone can see them but in a distributed VCS you commit the changes and they are still only on your local server until you push them to the remote server.

Exercise:

Difference between Git and Github

What is Git?

> Git is the most popular distributed version control system around today. Just like distributed VCS it is installed on your computer and allows you to see records and versions of different projects you're working on.

Git has a unique feature called 'branching' which allows you to create your own unique branches containing your changes in the code (just like how we discussed file tree earlier). You can revisit older branches, merge and or delete them easily.

What is Github?

> Github is officially known as a Git repository hosting service. This basically means that it is a cloud based online database that allows you to upload, control, track and share your projects outside of your local computer or server. You can share your projects and code with anybody around the world you give access to and they can make revisions and changes to your Git branches. Github offers an intuitive interface and essentially turbo-charges Git functionalities.

So in summary Git is a distributed version control system and Github is a hosting service that is cloud based.

Git lets you manage and track your source code while Github lets you host and manage your Git repositories.

What are repositories?

> Repositories in Git and Github contain all of your project files and revision history. Repositories can be owned by single individuals or they can be shared amongst other developers. You can collaborate or restrict access to anyone if you own the repository.

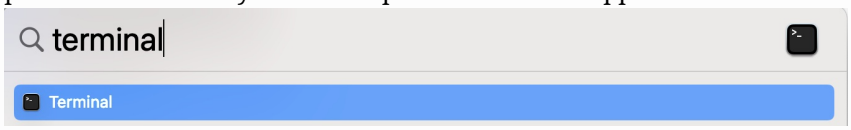
You can use repositories to discuss and share feedback, report bugs and seek out solutions to coding problems.

A repository essentially is tracking all the changes you make to a file and saves the revisions. It saves this data in a repository folder called .git and like we mentioned before Git uses VCS software to track and save all changes made to a project.

How to install Git

> Git can be installed on any operating system. On Mac, Git comes installed by default. If you're on a Mac and want to see if Git is installed do the following:

1. Open up the terminal application on your computer. On a Windows machine, follow the instructions on this page: https://learn.microsoft.com/en-us/windows/terminal/install. On a Mac, press the `"Command"` and `"Space"` keys on your keyboard to open the Spotlight search box, then type `"Terminal"` into the search box and press the "Enter" key. This will open the Terminal application.



2. Type in `git version`. The output will either inform you what version of Git is installed or will alert you that `git` is an unknown command.

Install Git for Windows

1. Go to the latest Git for Windows Installer and download the latest version
2. Once the installer starts, follow instructions in the setup wizard until installation is complete
3. Open the windows command prompt (or Git Bash if you selected not to use the standard Git Windows Command Prompt during the Git installation).
4. Type `git version` to verify Git was installed.

Install Git on Mac

If you don't already have it pre-installed on your Mac you can do so by following the below method from an Installer.

1. Go to the latest macOS Git Installer and download the latest version.
2. Once it has started follow instructions until the installation is complete.
3. Open the command prompt `'Terminal'` and type in `git version` to verify it was installed.

If you want to install git for Linux, Homebrew or any other method you can visit the official Github page here.

We have git installed here in Codio. Open a terminal here typing `shift-alt-T` and type in `git version`.

Git Commands in Github 1

> Here we will go over the basic commands in Git to get you started. These commands stay the same on Github.

`git config` - this command sets up the author name and email address to be used with your commits
How to use it:
`git config –global` user.name `"[name]"`
`git config –global` user.email `"[email address]"`

`git init` - this command is used to start a new project or create a new repository
How to use it: `git init [repository name]`

`git clone` - this command is used to obtain an existing repository - ie. if you're collaborating on a project
How to use it: `git clone [url]`

`git add` - this command adds new or changed files in your working directory to the Git staging area
How to use it: `git add [file]`

`git add *` - this command means add all files in the current directory, except for files whose names begin with a dot
How to use it: `git add *`

`git commit` - this command captures a snapshot of the project's currently staged changes - commit often to ensure a working version of your project
How to use it: `git commit -m "[ Type in commit message]"`

`git commit -a` - this command stages and commits changes to any and all files you've added with the git add command and any files you've changed since then
How to use it: `git commit -a`

`git diff` - this command helps you see all the changes that have happened in your project
How to use it: `git diff`
You can also compare two branches using this but adding the branch names like this:
`git diff [first branch] [second branch]`

`git diff -staged` - this command helps you see the differences between the files in the staging area and the latest version
How to use it: `git diff –staged`

`git reset` - this command resets the branch to match a given commit - it unstages the files but preserves file contents
How to use it: `git reset [file]`

`git reset commit` - this command undoes all the commits after the specific commit
How to use it: `git reset [commit]`

`git reset -hard` - this command resets and removes all changes from everywhere up to the specified commit
How to use it: `git reset –hard [commit]`

`git status` - this command will list all the files that have been committed - it shows the state of the working directory and staging area
How to use it: `git status`

`git rm` - this command helps you remove one or more files from the repository and filesystem
How to use it: `git rm [file]`

`git log` - this command displays a record of the commit history in a repository
How to use it: `git log`

`git log -follow` - this command lists the history of a specific file
How to use it: `git log –follow[file]`

`git show` - this command shows the metadata and content changes of a specific commit
How to use it: `git show [commit]`

`git tag` - this command is used to give tags to specific commits - Git tags are used to highlight a specific point in the project repository (a new version, new feature, etc.)
How to use it: `git tag [commitID]`

Git Commands in Github 2

`git branch` - this command lists, create, rename and delete branches in your repository
How to use it
– For listing: `git branch`
– Creating new branch - `git branch [branch name]`
– Deleting a branch - safe operation where if there are unmerged changes Git prevents you from deleting - `git branch -d [branch name]`
– Deleting a branch - force deleting it even if it has unmerged changes - `git branch -D`
– Rename branch - `git branch -m [new branch name]`
– List all remote branches - `git branch -a`

`git checkout` - this command is to switch from one branch to another
How to use it: `git checkout [branch name to switch to]`
– If you want to create a new branch and also switch to it you can use the following command: `git checkout -b [branch name]`

`git merge` - this command merges the specified branch history into current branch
How to use it: `git merge [branch name]`

`git remote` - this command lets you create, view and delete connections to other repositories
How to use it:
– List remote connections you have to other repositories - `git remote`
– Create a new connection to a remote repository - `git remote add [name] [url]`
– Remove connection to remote repository - `git remote rm [name]`
– Rename a remote connection - `git remote rename [old-name] [new-name]`

`git push` - this command lets you upload local repository content to remote repository
How to use it:
– Push specific branch - to prevent from overwriting commits, Git won't let you push when it results in a non-fast-forward merge in the destination repository - `git push [repository link] [branch]`
– Same as above but force the push even it results in a non-fast-forward merge (overwriting commits) - `git push [repository link] - - force`
– Push all local branches to the specified remote - `git push [repository link] - -all`
– Send all your tags to local repository since they are not automatically pushed when pushing a branch - `git push [repository link] - -tags`
– Delete branch in remote repository - git push origin –delete

`git pull` - this command is used to fetch and merge a specified remote repository's copy of the current branch
How to use it: `git pull [repository link]`

`git stash` - this command takes your uncommitted working changes and saves them for later use, you can reapply them at any time. You can go do other work on different branches, create new commits and switch branches, and when you're ready you can re-apply your stash.
How to use it:
– To stash your changes - `git stash`
– When you're ready to re-apply the changes to working directory - `git stash pop`
– You can also apply to working copy and keep them in your stash - `git stash apply`
– You can create multiple stashed. To view them you use this command - `git stash list`
– You can use a message to label what the stash is about - `git stash save "message"`
– From the list of stash you can choose which to re-apply - `git stash pop stash@(the stash number)`
– Deleting stash with its list number - `git stash drop stash@(number of stash)`
– Deleting all stashes - `git stash clear`

These are the basic commands that will get you started to help you better understand git. Here's a Git Tutorial if you want to learn more.

Practice

Manage your code in GitHub from Codio

Now that we know the importance of having a version control system, let's use it to manage the versions of the files we have worked on during the course.

To accomplish this task you must perform the following steps:
1. Create a free GitHub account
2. Create a repo in your GitHub
3. Connect Codio with your repo
4. Upload files to your remote repo
5. Make changes to your files

1. Create a free GitHub account

If you don't already have a GitHub account, here's how to create one:
- Open https://github.com in your browser, and then select Sign up.
2. Create a repo in your GitHub
- Go to your personal GitHub account on GitHub.
- Click on "New" and name this new repository "me_and_myself"
- Add the required information, give a small description to your new repository, add a .gitignore file to ignore unnecessary files and add a license.
3. Connect Codio with your repo
Follow the steps of this guide:
Connecting Codio to a GitHub repo
4. Upload files to your remote repo
From the terminal in Codio type the following commands:

Add all files contained in the parent and children directories

```
git add .
```

Commit the files with a message

```
git commit -m "initial commit"
```

Send changes to the remote repo

```
git push --set-upstream origin master
```

After executing the commands above, go to your GitHub account and check that you are able to see the files.
5. Make changes to your files
Now select one of the files you have already uploaded and make some changes from Codio. Then go again to your GitHub account and visualize changes there.

Conclusion, Takeaways & Attribution

> - Version controlled systems are important to any development projects from an individual to a whole team. The power they have to allow us to collaborate with others around the world while also managing and maintaining a process that creates structure is imperative.
> - Git and Github are essential to any developer starting out in the field. The skills acquired in learning these softwares and programs help us become better developers along with helping us cooperate at a higher level with other development team members.
> - These softwares and programs are always evolving and have very interactive communities. For more information you can visit the Github Community.

Further Reading

In plain English, what does “git reset” do? (2010, March 27). Stack Overflow. https://stackoverflow.com/questions/2530060/in-plain-english-what-does-git-reset-do
Kappagantula, S. (2022, March 28). Top 20 Git Commands with Example. Edureka. https://www.edureka.co/blog/git-commands-with-example/

Seksaria, H. (2021, December 27). Types of Version Control System - Version Control System, Git and GitHub. Medium. https://medium.com/version-control-system/types-of-version-control-system-766a6b656088

User, D. (2022, July 1). Git vs. GitHub: What’s the Difference? Devmountain. https://devmountain.com/blog/git-vs-github-whats-the-difference/

Atlassian. (n.d.). How to Create a Git Repository | Atlassian Git Tutorial. https://www.atlassian.com/git/tutorials/setting-up-a-repository

Hello, World from GitHub

> ### Goals
>
> By the end of this case you will be able to use GitHub to:
> - Create an account
> - Create a repository
> - Create a branch
> - Commit changes
> - Pull request
> - Merge

Introduction

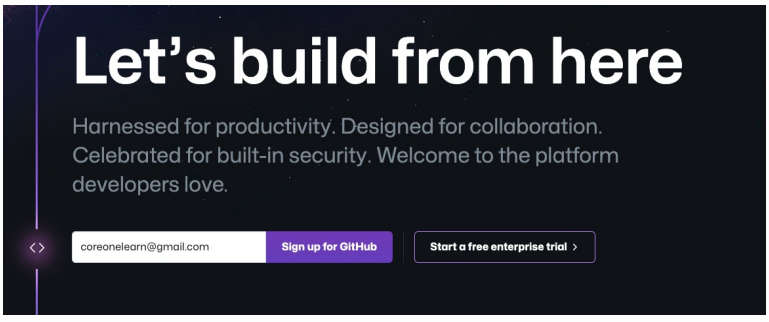
This will be a guided exercise walking you through the process of using GitHub to create your own Hello World repository. For this exercise you don't need the command line or to install Git.

Business Context

This will help you prepare and give you an idea of what your projects will look like and how to use these tools for when you're working at a company or with other developers.
Create an account
First go to this link Github and create a free account.

Create an account

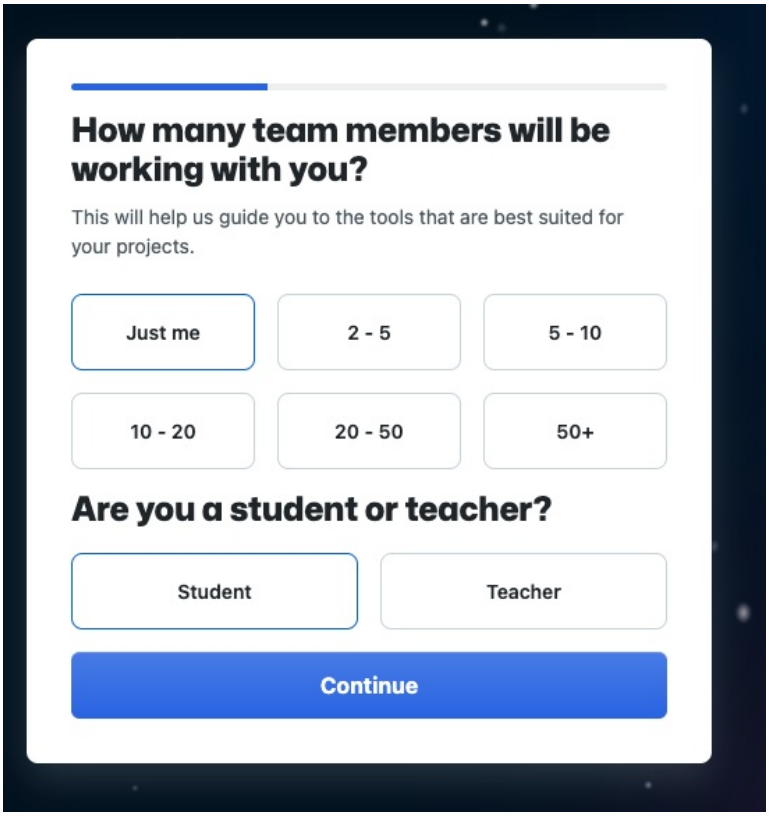
First go to this link Github and create a free account.



github

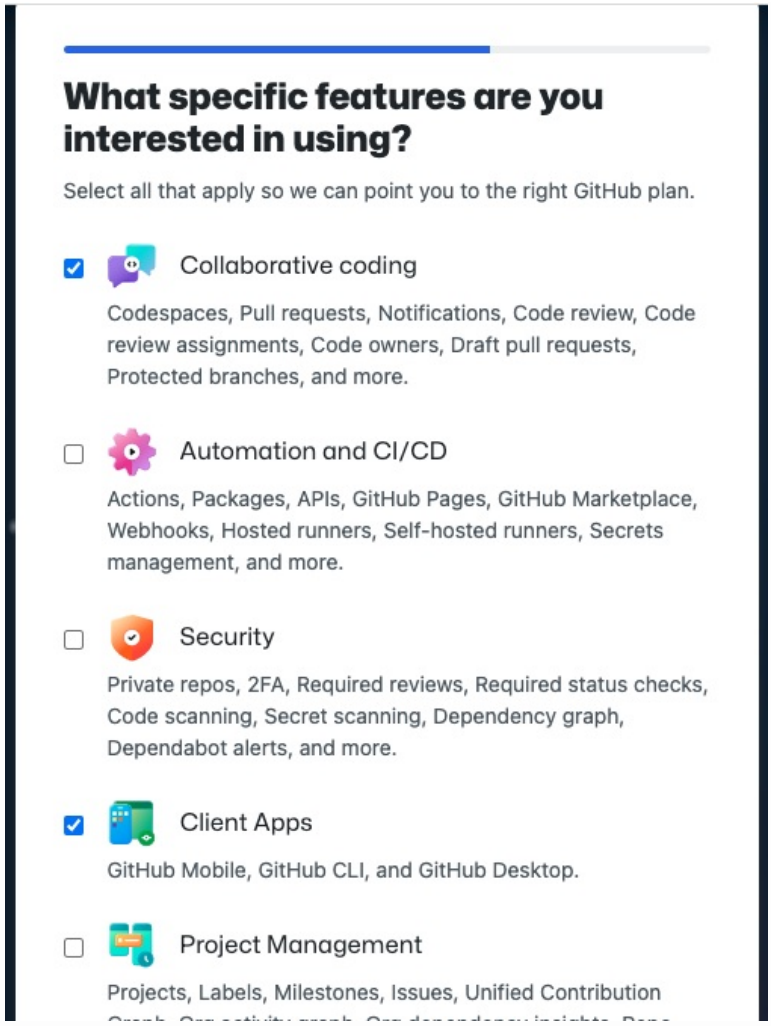
It will prompt you to enter your email, create a password, and username. Ensure that you enter a real email you have access to because it will send you a verification code.

Then select the following for how many team members and if you're a student or teacher.



github teams

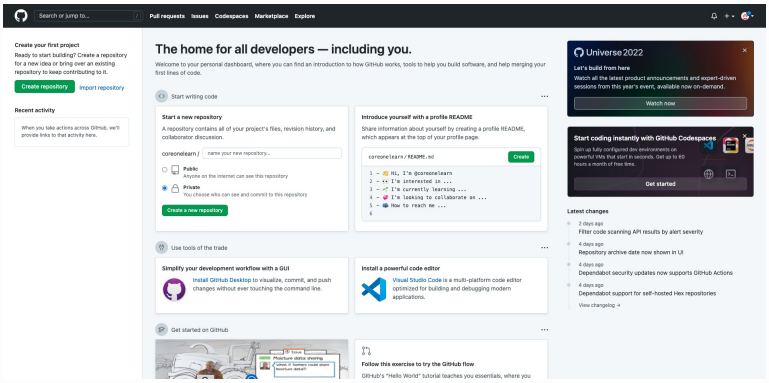
On the next screen it will ask you select specific features - for this exercise we will go with `'Collaborative Coding'` and `'Client Apps'`



github features

Next screen - select first option `'Continue for free'`

And Voila! You've got yourself a GitHub account. Your welcome screen should look like this:



github

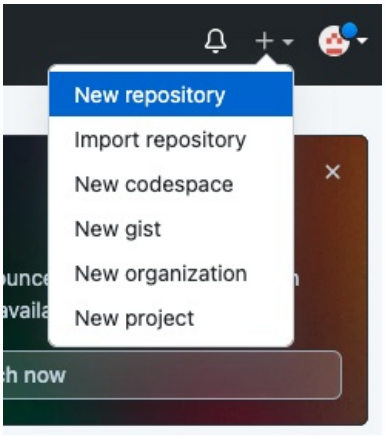
Create a repository

Repositories can contain anything from files, folders, spreadsheets, images and videos for your project. They have a README file that contains information about your project. These files are written in plain text Markdown language. Here's a link to get you restarted with Markdown syntax which is similar to HTML.

Now let’s create a hello-world repository.

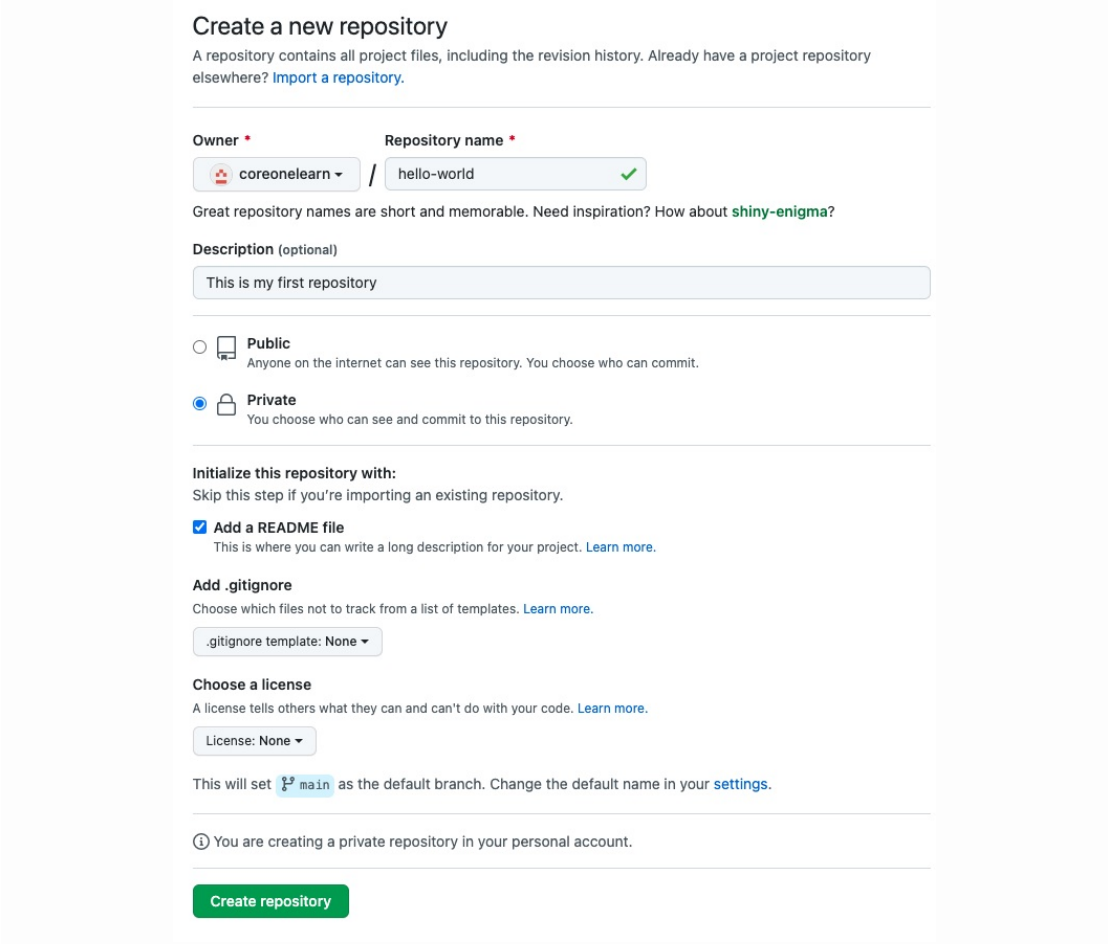
Now let’s create a hello-world repository.

1. On the top right corner of the page click the ‘+’ sign and select `‘New repository’`



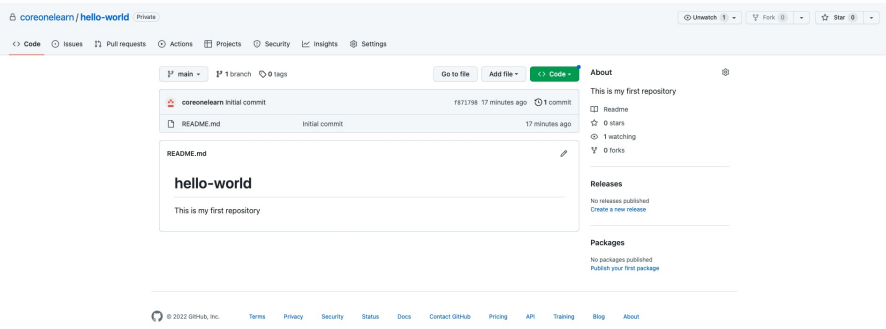
new repository

2. In the **Repository Name** enter `hello-world`
3. Write a short description in the **Description** box
4. Select **Private** so only you choose who can see and commit to the repository
5. Select **Add a README file**
6. Then click **Create repository**



create repository

You've now got your first repository!

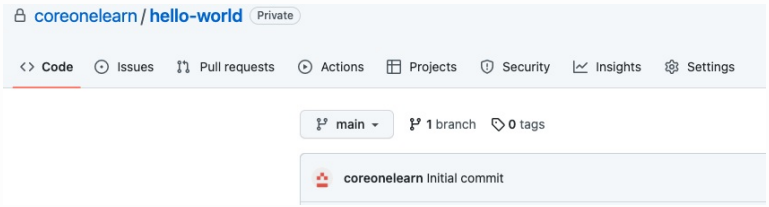


Creating a branch

Your repository has one branch called main . You can create additional branches off of this main branch for different versions of your project (adding new features without affecting the main code) and it will only show on the main branch after you merge. You can use these different branches to edit and experiment before committing the changes to the main branch.

Let’s create a branch in our repository

You’ve got these tabs in your repository - code, issues, pull requests, actions etc.

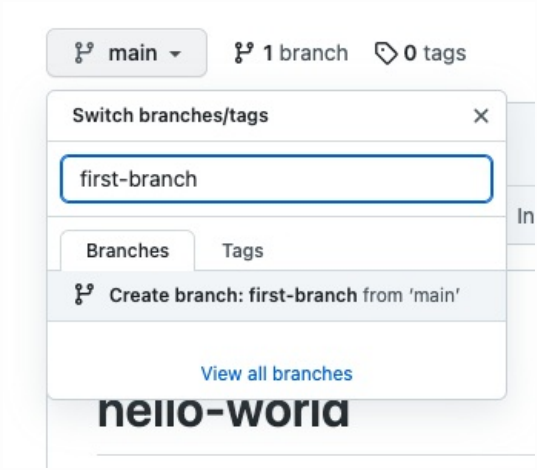


create branch

1. Ensure the **Code** tab is selected and click the **main** dropdown

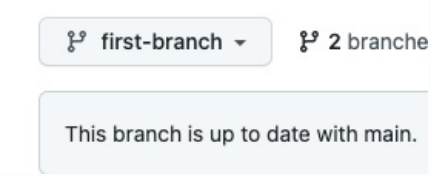
main

2. In the box that appears give your branch a name `first-branch` and click **Create branch: first-branch from ‘main’**



first branch

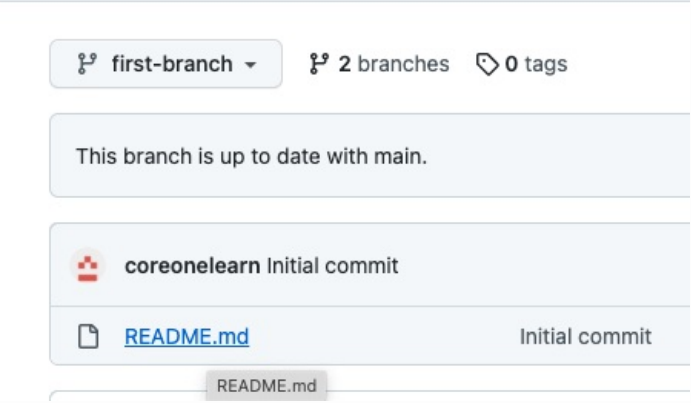
You’ve created your first branch! Now you have two branches: main and first-branch. Now let’s commit some changes.
>
Commiting Changes
After you created your branch Github brought you to the code page for your branch, it looks the same as the previous page which was your `main` branch but it’s not, this is the page for your `first-branch` as you can see from the dropdown, it’s an exact copy.



update branch

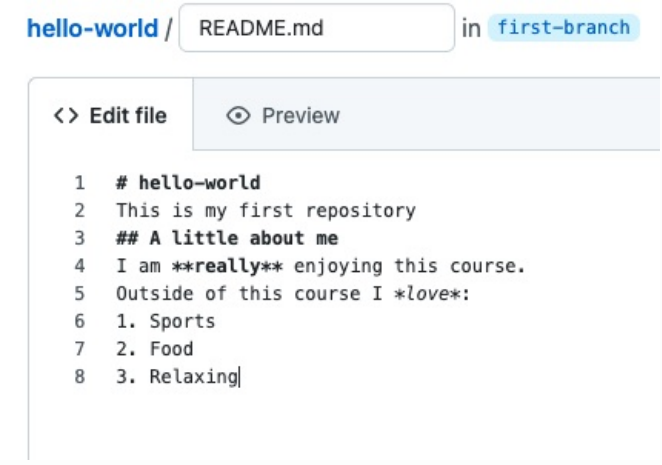
Each commit has a commit message explaining what the change was and why the change was made. This helps other developers understand how and why you got to where you are.

1. Under the `first-branch` you created click on the **README.md** file

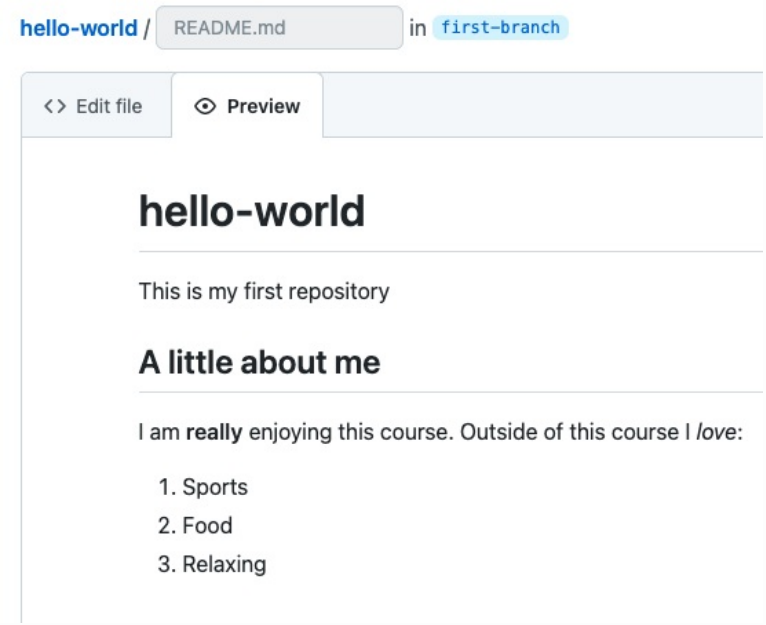


readme file

2. Now click the pencil to edit the file
3. This should bring you to an editor. Here write a bit about yourself using some of the Markdown elements. The Preview tab will show you the output. Here's an example:

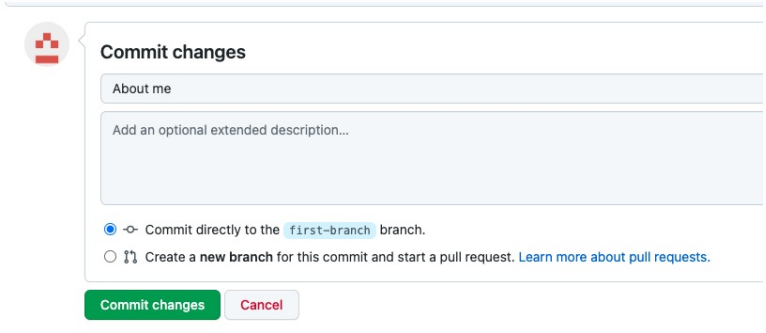


pencil



edit the file

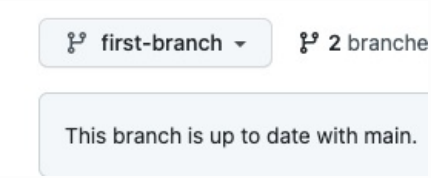
4. In the **Commit changes box** write a message describing your changes



hello-world

Committing Changes

> After you created your branch, Github brought you to the code page for your branch, it looks the same as the previous page which was your `main` branch but it's not, this is the page for your `first-branch`. As you can see from the dropdown, it's an exact copy.



committing changes

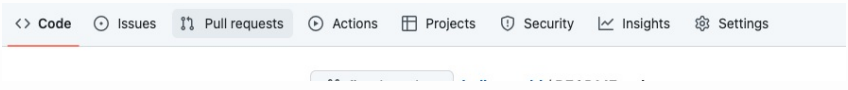
Since this is your new branch and it's separate from your `main` branch, these changes have only been made to this branch. Your `first-branch`.

Pull Requests

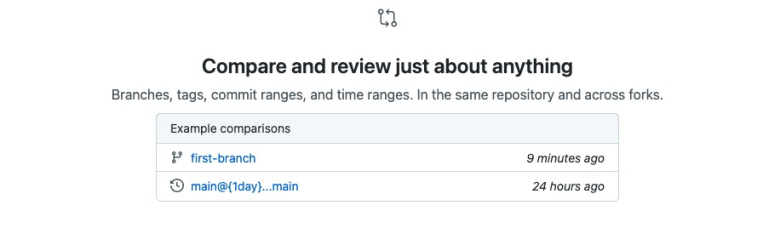
> When you open a pull request you're essentially suggesting to someone to review your changes and merge them into their branch. Pull requests are useful since they show the differences from both branches in different colors. A neat feature of GitHub you can use the `@mention` feature to ask for feedback from anyone. Shorthand for pull request is 'PR'. You can expect, as a developer, to hear this term a lot.

Now let's execute a pull request

1. Remember the tabs at the top of your repository? Click the **Pull requests** tab

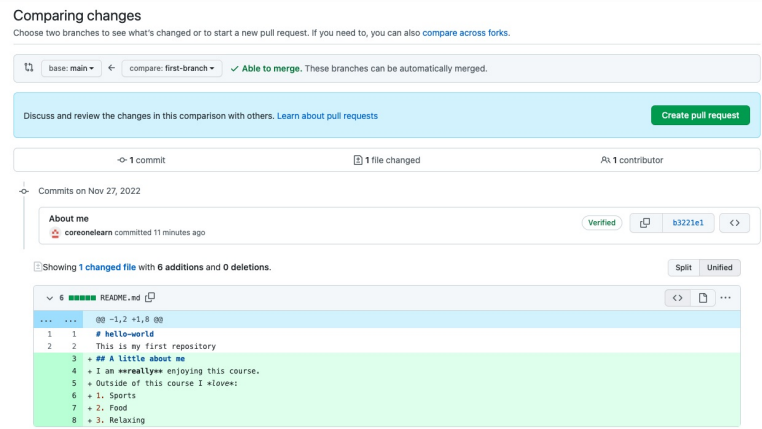


2. Click **New pull request**
3. In the **Example comparisons** click the branch you just made changes to - `your first-branch`



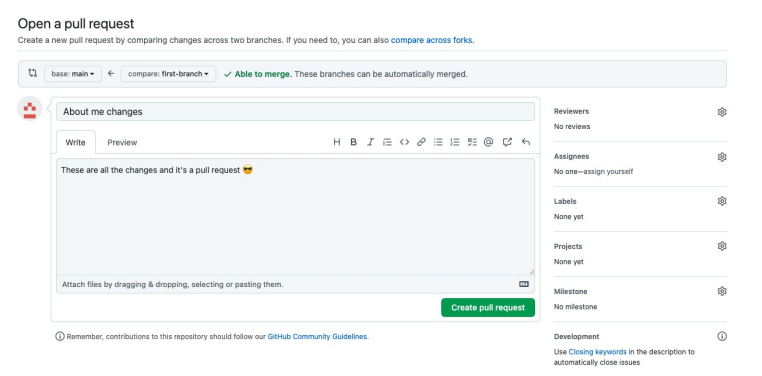
review info

4. You can now see the changes, ensure they are what you want and click **Create pull request**



create PR

5. Write a title for your pull request and a description. You can put images, emojis or gifs in here if you like. On the right you can also click the settings icons and choose any of those options for your pull requests (reviewers, assignees, labels, projects, and milestones).



add to your PR

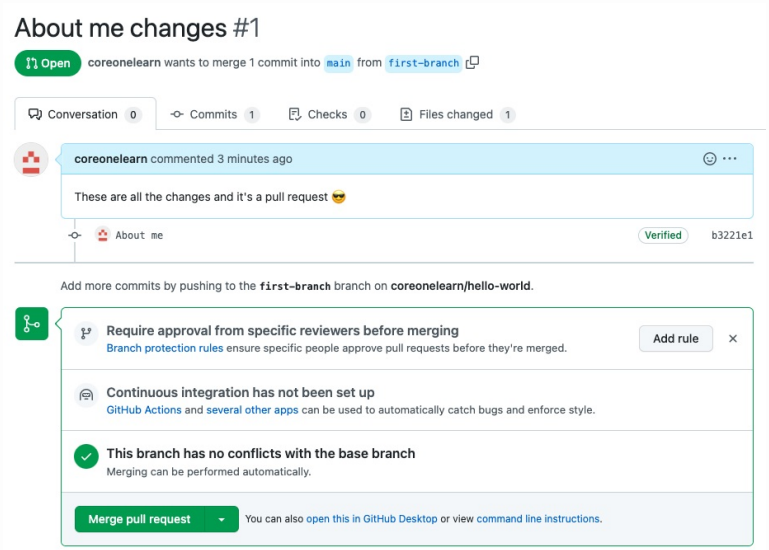
6. Now click **Create pull request**

Merging

This is the final step where you will merge your `first branch` changes back into the `main` branch. If there are conflicts with the code GitHub will alert you and you can review/resolve these conflicts before proceeding. In this example you should not have any conflicts.

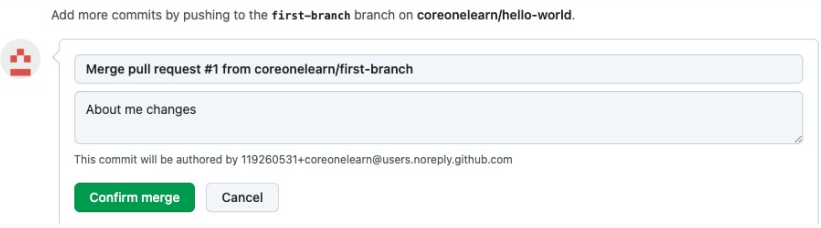
Let's merge your branch into the main branch

1. After seeing that there are no conflicts with the base branch and the green checkmark click Merge pull request

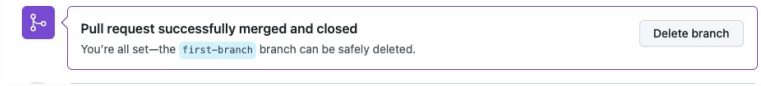


merging

2. You will get a short summary of what is being merged where. Click **Confirm merge**

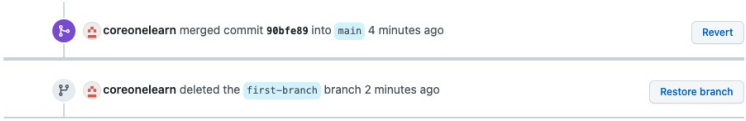


3. Now that your pull request has been merged you'll get the option to delete the branch. Since the changes have been merged to the main branch, you don't need your first branch anymore. You can always create a new branch if you want to make more changes. Click **Delete branch**



deleting branch

If you deleted or merged your branch by accident, GitHub also gives you the ability to revert the merge and restore your branch.



confirm delete

Once you start using GitHub and Git for a while you can start to collaborate with others by either contributing to someone's project using their projects as a place to start for yourself. You can go here for further reading on this.

Conclusion,Takeaways & Attribution

> - What you learned in the module are the basic functionalities needed when you start or work on any GitHub project.
> - The more you immerse yourself in GitHub projects and collaborate with other, the more you will learn and the better you will get.
> - When in doubt, always remember there is an extensive list of documentation to help walk you through any questions or concerns you may have.

Further Reading

Quickstart. (n.d.). GitHub Docs. https://docs.github.com/en/get-started/quickstart

Attribution

(remember to use APA style)

Project: Add Website and Files to Github

> **Goals**
>
> By the end of this lesson you will:
> - Upload and host your site on GitHub
> - Add pages and CSS/SCSS
> - Collaborate on other projects

Introduction

In the previous modules and exercises we created file paths, file structures, linked HTML pages together, filled them with SEO details and then we learnt how to use Git and GitHub with their different commands while also learning how to create repositories. Now we will learn to upload our website from our local machine to GitHub, add pages, CSS/SCSS and collaborate on projects.

Note: You can use the previous files you created for building your website.

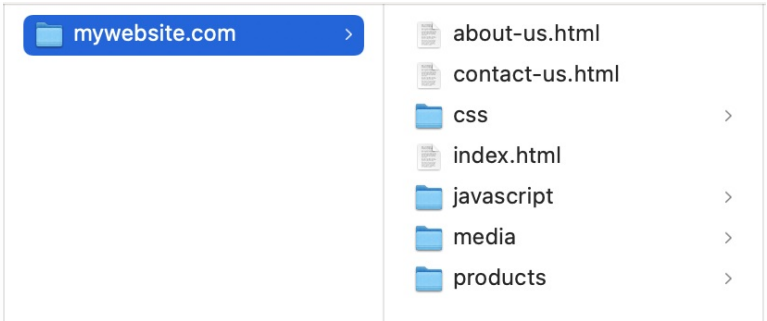
1. Create your personal website with the correct file structure on your local machine - this file structure should include:

a. One index.html page - this page should contain meta tags and all the HTML document linked appropriately
b. At least three other pages you can link to (you can use the ones from the previous exercise: about us, contact us, and products page)
c. One CSS file and one optional Javascript file
d. At least two images on any page of your choice

2. Upload and host your website to GitHub - name the repository after your website's name
3. Add one more additional page
4. Change the color and bold a sentence on any page using pull and merge commands
5. Invite collaborators through their GitHub username

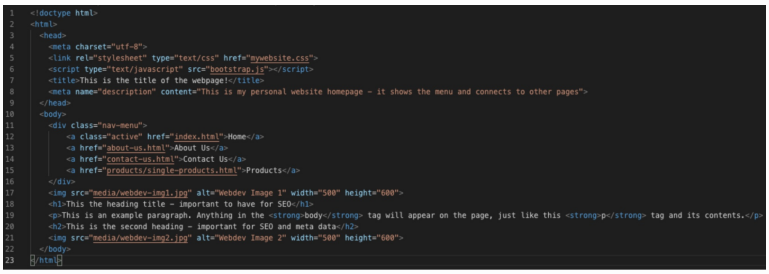
Create!

1. Personal website with correct file structure



file structure

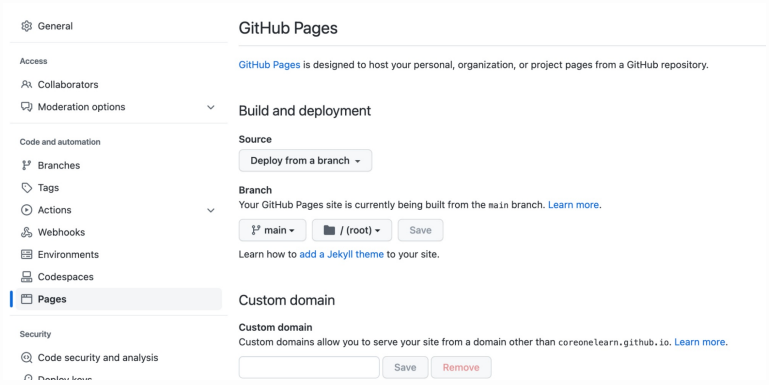
Index.html page with all the meta tags and files linked appropriately. Three pages you can link to. One CSS file, Javascript file, and at least two images.



```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <link rel="stylesheet" type="text/css" href="mywebsite.css">
    <script type="text/javascript" src="bootstrap.js"></script>
    <title>This is the title of the webpage!</title>
    <meta name="description" content="This is my personal website homepage - it shows the menu and connects to other pages">
  </head>
  <body>
    <div class="nav-menu">
        <a class="active" href="index.html">Home</a>
        <a href="about-us.html">About Us</a>
        <a href="contact-us.html">Contact Us</a>
        <a href="products/single-products.html">Products</a>
    </div>
    <img src="media/webdev-img1.jpg" alt="Webdev Image 1" width="500" height="600">
    <h1>This the heading title - important to have for SEO</h1>
    <p>This is an example paragraph. Anything in the <strong>body</strong> tag will appear on the page, just like this <strong>p</strong> tag and its contents.</p>
    <h2>This is the second heading - important for SEO and meta data</h2>
    <img src="media/webdev-img2.jpg" alt="Webdev Image 2" width="500" height="600">
  </body>
</html>
```

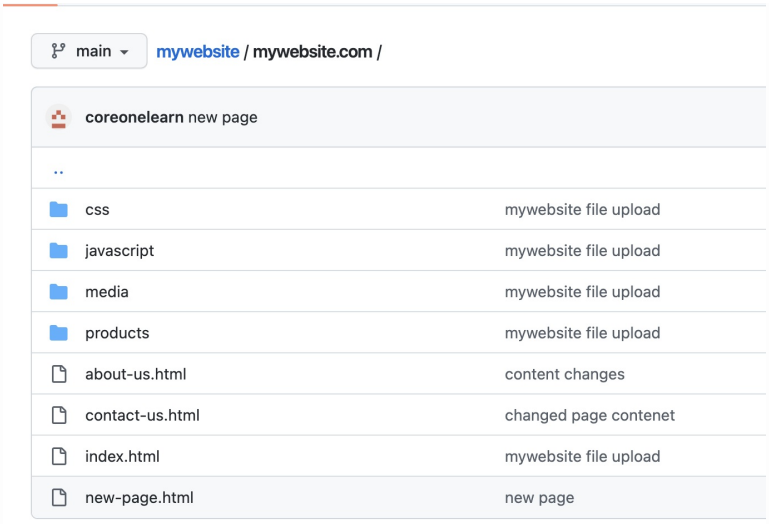
links in code

2. Now we need to push the files into a new GitHub repository. After uploading the files to GitHub we go to the GitHub pages and select the branch and save to publish.



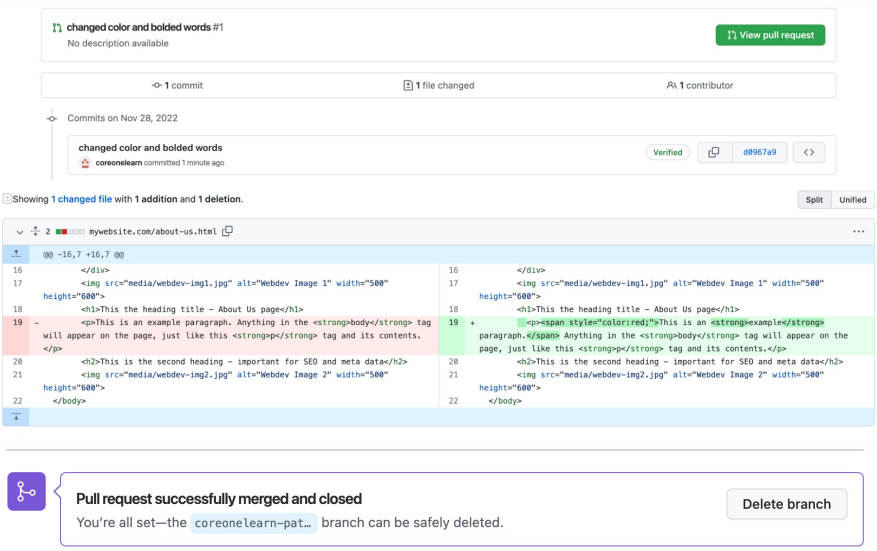
push changes to github

3. Add an additional page. A resume is a good addition here.

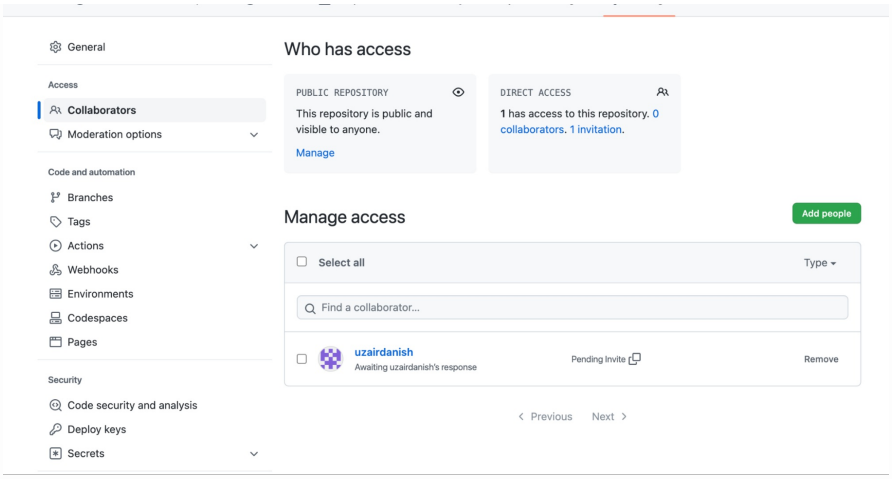


add additional page

4. Change the color and bold a sentence on an existing page using a pull and merge commands



5. Invite collaborators through their GitHub username


General
Access
Collaborators
Moderation options
Code and automation
Branches
Tags
Actions
Webhooks
Environments
Codespaces
Pages
Security
Code security and analysis
Deploy keys
Secrets
Who has access
PUBLIC REPOSITORY
This repository is public and visible to anyone.
Manage
DIRECT ACCESS
1 has access to this repository. 0 collaborators. 1 invitation.
Manage access
Add people
Select all
Type
Find a collaborator...
uzairdanish
Awaiting uzairdanish's response
Pending Invite
Remove
Previous
Next

>

Conclusion, Takeaways & Attribution

> - GitHub is a great way to host a portfolio for free when starting off
> - Being able to create, edit, and manage files in GitHub is essential to keeping your website running smoothly
> - Understanding the different Git commands helps you achieve your goals more efficiently like making CSS changes, file edits