

Welcome to the DES Team!

Interactive Designer Onboarding

Updated: 01/23/25

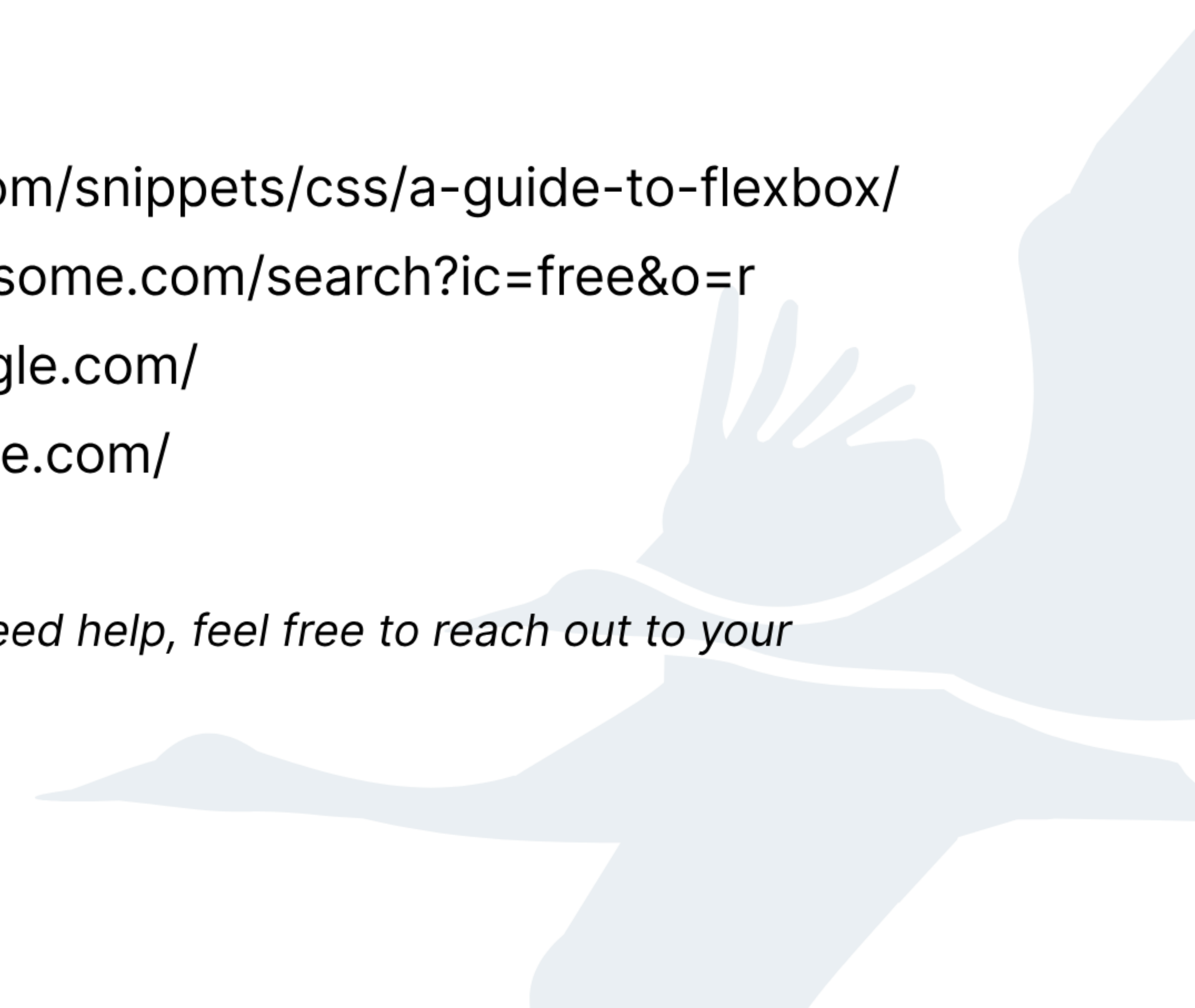


Before We Get Started

Please open your preferred browser and bookmark each of the provided links. You'll be using these resources frequently, so bookmarking them will help you access everything quickly and efficiently.

- **Department Reference Pages:** <file:///sandhills.int/dev/HostDealerPages/Training/SAW/default.htm>
- **CRM:** <http://intranet/SandhillsCRM/contact>
- **Sandhills Cloud:** <https://sandhillscloud.com/>
- **Inventory Filter:** <https://www.sandhillsinventory.com/inventory-filter/>
- **Inventory Category Search:** <https://intranet/mmmcore/CategorySearch>
- **Inventory Error Codes:** <https://sandhills.snowcloud.com/docs/Article/Read/e7181601-bd06-46e5-b346-838cbd9777f0>
- **Showpad:** <https://sandhills.showpad.biz/webapp2/home>
- **Schema Generator:** <https://technicalseo.com/tools/schema-markup-generator/>
- **Schema Validator:** <https://validator.schema.org>
- **Image Reducer:** <https://tinypng.com/>
- **Video Reducer:** <https://www.videosmaller.com/>
- **Favicon Generator:** <https://realfavicongenerator.net/>
- **W3 Schools:** <https://www.w3schools.com/html/default.asp>
- **CSS Tricks:** <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- **Font Awesome:** <https://fontawesome.com/search?ic=free&o=r>
- **Google Fonts:** <https://fonts.google.com/>
- **Adobe Fonts:** <https://fonts.adobe.com/>

Notes: You're also encouraged to create folders within your bookmarks to keep things organized. If you're unsure how to do this or need help, feel free to reach out to your Trainer for guidance.



Before We Get Started

Now open File Explorer. These are folders you will access on a daily basis. They are how you will access all our sites as well as your personal MAC Folder. You can pin these to make accessing them easier.

- **MAC Folder:** \\sandhills.int\file\Typeset\Art\Mac\{Your_Name}
- **HTML Dev:** \\Sandhills.int\dev\HostDealerPages
- **HTML Staging:** \\sandhills.dmz\stg\HostDealerPages
- **WordPress Staging:** \\dsstag.ext\file\WPSTG
- **Shop Dev:** \\Sandhills.int\dev\Websites\HostedStoreTemplates
- **Shop Staging:** \\sandhills.dmz\stg\Websites\HostedStoreTemplates
- **Human Resources:** \\sandhills.int\file\Data\Sandhills\Knowledgebase\Human Resource Information

Bonus: If you want a fun way to brush up on flexbox, try out Flexbox Froggy! <https://flexboxfroggy.com/>



The Project

You'll be building a website over the next 8 steps.

Each step will guide you through essential web development concepts, starting with a blank file and ending with a fully structured and styled site.

Notes: Your trainer is your primary point of contact for any questions throughout your training. Don't hesitate to ask - questions are a key part of the learning process!



01: The Beginning

In this step, you'll create a default website structure and learn about the <head> and <body> elements.

You'll also learn how to enhance your site's metadata with meta tags, canonical links, and schema markup, and how to use CSS versioning to manage stylesheet updates.

- Open Project_1 in Dreamweaver
 - Click on Windows start menu in lower left corner and select Adobe Dreamweaver > right-click and select "Pin to start"
 - At the top of the files panel, click the dropdown and select "Manage Sites...."
 - Click "New Site" in the bottom right of the popup
 - In Site Name, enter "Training"
 - For Local Site Folder, enter
"\\Sandhills.int\dev\HostDealerPages\www.desonboard1.com"
 - Click "Save" - you now have your project set up
 - At the top you'll see 3 buttons for Code, Split and Live. Click on code; we will never use Live or Split.

Notes: You've just opened your first training document in Dreamweaver and we can now begin learning about the head.



01: The Beginning

- Basic HTML 5 website file components (this is the standard code that should be on every html website page)
 - `<!DOCTYPE html>` this is a basic doctype declaration and should be at the top of the page.
 - `<html lang="en"></html>` This contains everything in the website and the lang="en" sets the language as English. All code should go inside of these html tags
 - `<head></head>` This is the opening and closing head tags. The next steps will be added in the head tags. (This is information for search engines users won't see the items in the head.)
 - `<meta name="viewport" content="width=device-width, initial-scale=1.0">` This viewport tag allows mobile devices to respond according to your responsive styles.
 - `<link rel="stylesheet" href="/css/style.css">` This references your websites external stylesheet. A website will have at least one external stylesheet stored in a folder named "css".
 - `<meta name="author" content="Sandhills Global">`
 - `<meta name="robots" content="index, follow">`

Notes: Please copy and paste the underlined items in your document in Dreamweaver inside the Head tags.



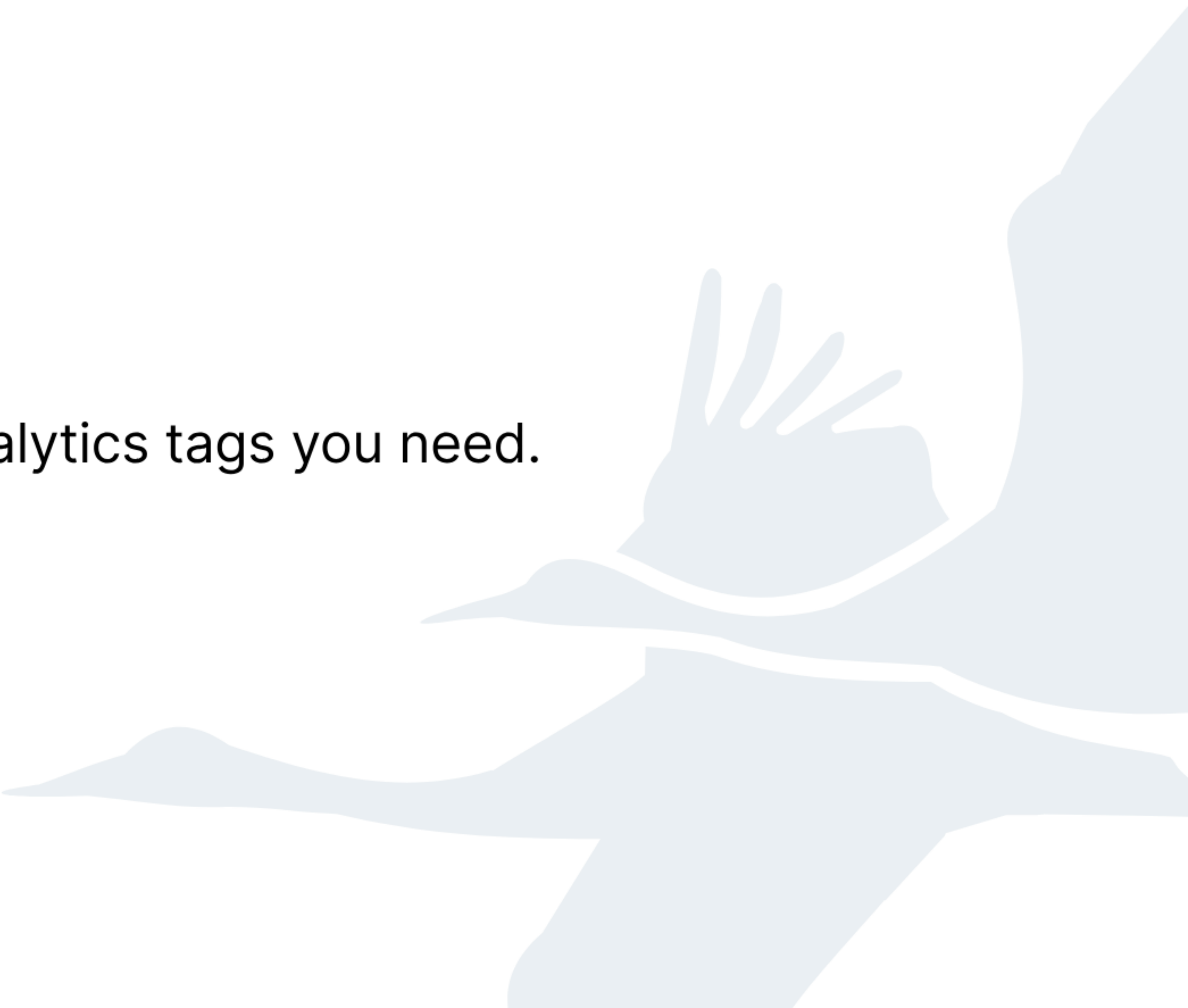
01: The Beginning

In this step, you'll create a default website structure and learn about the <head> and <body> elements.

You'll also learn how to enhance your site's metadata with meta tags, canonical links, and schema markup, and how to use CSS versioning to manage stylesheet updates.

- <Title>Company Name | City, State | A few keywords describing the page</Title> The title tag is required in all HTML documents and it defines the title of the page. Please use this formatting for all sites. Best practice is around 60 characters long but Google indexes up to 180 characters.
 - Please replace the Name with Hilliard Lawn & Garden.
 - Please replace the city, state with Hilliard, OH
 - Please replace the keywords with Residential & Commercial Lawn Equipment
- <meta name="description" content="A couple sentences about the company."> Please add a few sentences about the company in the content=""
- Social Media title and descriptions - The content for these can match the first title and meta description
 - <meta property="og:title" content=" TITLE OF WEBPAGE " />
 - <meta property="og:description" content="A couple sentences about the company." />
- Google Analytics
 - This is used to track users on each website. The manager or dealer you're working with will provide any analytics tags you need.
 - The tag collects analytical data to measure traffic and engagement.

Notes: Please copy and paste the underlined items in your document in Dreamweaver inside the Head tags.



01: The Beginning

In this step, you'll create a default website structure and learn about the <head> and <body> elements.

You'll also learn how to enhance your site's metadata with meta tags, canonical links, and schema markup, and how to use CSS versioning to manage stylesheet updates.

- Dealers can also add a Google Tag Manager (GTM). Customers can combine tracking scripts and track specific actions or events. We do not set these up or provide them for our customers. If they wish to have one they will need to set it up themselves and provide us the script.
- Copy this whole script and paste it in the head.

```
1  <!-- Google Analytics Tag (gtag.js) -->
2  <script async src="https://www.googletagmanager.com/gtag/js?id=G-VC193EJYCG"></script>
3  <script>
4      window.dataLayer = window.dataLayer || [];
5      function gtag(){dataLayer.push(arguments);}
6      gtag('js', new Date());
7
8      gtag('config', 'G-VD123EJZJH');
9  </script>
10 <!-- End Analytics Script -->
```

- <link rel="shortcut icon" href="/siteart/favicon.ico" /> Favicons are the small image you see beside the title in the tab of the browser.
- <meta name="robots" content="index, follow"> The robots tag tells search engines to follow the page and pages linked to it.

Notes: Please copy and paste the underlined items in your document in Dreamweaver inside the Head tags.

01: The Beginning

In this step, you'll create a default website structure and learn about the <head> and <body> elements.

You'll also learn how to enhance your site's metadata with meta tags, canonical links, and schema markup, and how to use CSS versioning to manage stylesheet updates.

- <link rel="canonical" href="https://www.{sitename}/{pagename}"/> This is a self referencing canonical. It tells search engines that this is the correct version of the page to index so duplicates don't get indexed.
- Schema go to <https://technicalseo.com/tools/schema-markup-generator/> to generate your schema then paste the script in the head just on the home page.
 - The schema gives search engines a bit more structured information about the business.
 - To generate schema in the schema generator select Local Business
 - Use Hilliard Lawn & Garden as the name
 - Use <https://www.hilliardlawnequipment.net> for the url
 - Use <https://www.hilliardlawnequipment.net/siteart/logo.png> as the image
 - Use (614) 876-4055 as the number
 - Used 5300 Cemetery, Rd Hilliard, OH 43026 as the address.
- Under the closing </head> add a line with <body> </body>. This will wrap all of the page content the users will see.

Notes: Please copy and paste the underlined items in your document in Dreamweaver inside the Head tags.



01: The Beginning

In this step, you'll create a default website structure and learn about the <head> and <body> elements.

You'll also learn how to enhance your site's metadata with meta tags, canonical links, and schema markup, and how to use CSS versioning to manage stylesheet updates.

- Resources
 - <https://www.w3schools.com/html/default.asp> - Reference for help with HTML
 - <https://www.w3schools.com/css/default.asp> - Reference for help with CSS
 - <https://www.codecademy.com/resources/cheatsheets/language/html-css> - HTML and CSS cheat sheets
 - <https://flexbox.help/> - Flex box (We will learn about this in another step)
 - <https://flexboxfroggy.com/> - Flex Box game

You've finished adding meta content to the head!

STOP please submit your training task to be viewed by your trainer before continuing to Step 2.

Notes: Please copy and paste the underlined items in your document in Dreamweaver inside the Head tags.



02: The Essential Elements

You'll add headings, paragraphs, lists, and more to your homepage. These foundational elements improve structure, readability, and accessibility - especially for users with visual impairments who rely on screen readers

General Notes

- Everything in the `<body>` is visible to users.
- Common containers: `<div>`, `<p>`, `<img>`, `<h1>`--`<h6>`.
- Semantic containers we use often: `<header>`, `<nav>`, `<section>`, `<footer>`.
- Always give `<nav>` a class name to avoid global conflicts.
- Do not style tags directly (`img`, `h1`, etc.). Use classes to avoid affecting the inventory framework.

Build the Page Structure

1. Hero Section

- Add a `<section>` for the hero area.
- Inside it, add:
 - One `<h1>` (each page should have only one).
 - Example: Residential & Commercial Lawn Equipment
 - A link for the shop button:
 - `<a href="">Shop Equipment</a>`
 - Add `target="blank"` if it should open in a new tab.
- Copy the hero text from the "Page Copy" document in your Ex2 folder.

Notes: If you need more guidance on HTML you can learn more here https://www.w3schools.com/html/html_basic.asp



02: The Essential Elements

You'll add headings, paragraphs, lists, and more to your homepage. These foundational elements improve structure, readability, and accessibility - especially for users with visual impairments who rely on screen readers

Quick Links Slider Section

- Add a new <section>.
- Add the H2 and text shown in the mockup.

About Section

- Add another <section>.
- Add the H2 using a and
 instead of multiple H tags:
 - <h2>Welcome To
Hilliard Lawn & Garden</h2>
- Add the paragraph text from the "Page Copy" document.
- Add the "Learn More" button (navigational links should always be a tags buttons are for form submissions or other actions).
- Add the image:
 -
- Images are located in the siteart folder in Ex1.

Services Section

- Add another <section>.
- Add the H2, paragraph, and button similar to the About section.
- Skip the accordion for now (added later).

Notes: If you need more guidance on HTML you can learn more here https://www.w3schools.com/html/html_basic.asp



02: The Essential Elements

You'll add headings, paragraphs, lists, and more to your homepage. These foundational elements improve structure, readability, and accessibility - especially for users with visual impairments who rely on screen readers

Footer

- Add a <footer> container.
- Do not use heading tags in the footer (H tags must stay in descending order on the page).
- Add each contact item in its own <div> so you can flex/float them.
- Add a separate <div> for footer navigation links with a top border.

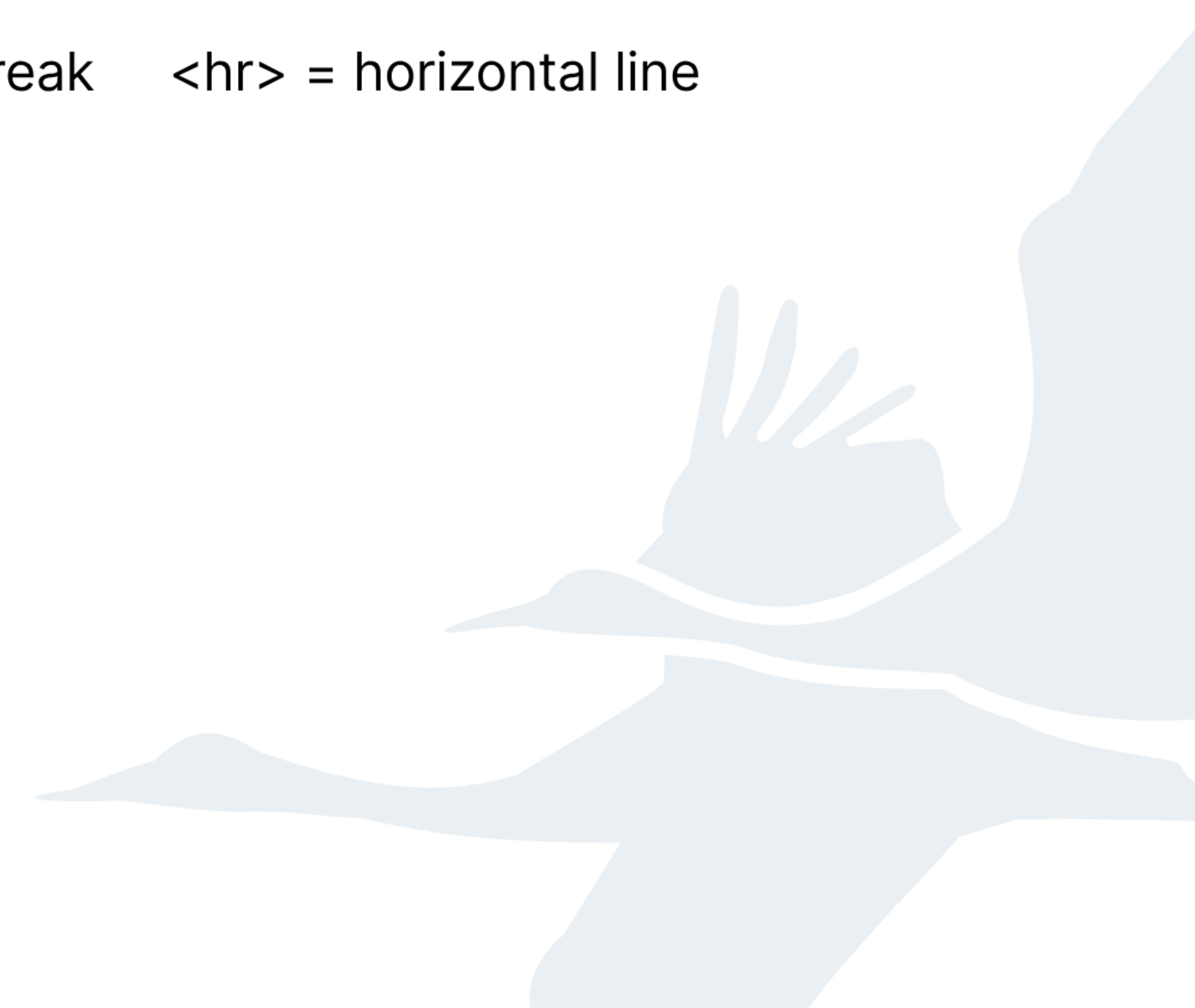
HTML Symbols & Inline Styling

- Common character codes:
 - & = & © = © # = # … = ... • = • – = – = space
 = line break <hr> = horizontal line
- **Inline formatting:**
 - <i> italics bold bold + semantic importance <u> underline

Practice:

- Add bold text inside the About paragraph.
- Add italics to the red text in the H2 (based on the mockup).
- Your HTML body should be built and include sections with divs, paragraphs, images, links, and a footer
- **STOP please submit your task to be viewed by your trainer before moving to the next step.**

Notes: If you need more guidance on HTML you can learn more here https://www.w3schools.com/html/html_basic.asp



03: The Layout

You'll explore layout techniques using Flexbox, CSS Grid, and absolute positioning. This will help you organize the content you've added into a clean, structured, and visually balanced layout.

- https://www.w3schools.com/css/css_grid.asp - Grid Layout
- https://www.w3schools.com/css/css3_flexbox.asp - flexbox layout
- <https://flexboxfroggy.com/> - flex game
- Use Flexbox when you need either a row or a column. Example: nav bars, button groups, simple card layouts.
- Use Grid when you need two-dimensional layouts (both rows and columns) Example: page layouts with header, content, and a sidebar
- Please read through the w3schools resources to become familiar with grids and flexbox



03: The Layout

You'll explore layout techniques using Flexbox, CSS Grid, and absolute positioning. This will help you organize the content you've added into a clean, structured, and visually balanced layout.

Hero Section - Absolute Positioning

- We'll style everything in styles.css (inside the css folder).
- Add the class "hero" to your hero section:
- Add the hero background image inside this section (use the image from your training folder).
- Wrap the hero text elements (H1 + links) in a div so we can position them together with the class "hero-text"

CSS for the Hero

- The parent (.hero) must be position: relative; so the child can be positioned inside it.
- Make the hero image fill the width by targeting it in the css and adding width:100%;
- Position the text over the image width position absolute
 - .hero-text { position: absolute; /* then set top/left using %, VW, or VH */ }
- Fix stacking order using z-index: .hero-text { z-index: 100; } .hero { z-index: 10; }
- Remember: z-index only works on positioned elements.

About Section - Flexbox Layout

- Wrap the entire About section in a div with class "flex".
- Wrap each half (text + image) in its own div with class "flex2".

Notes: If you need more guidance on Flexbox you can learn more here https://www.w3schools.com/css/css3_flexbox.asp



03: The Layout

You'll explore layout techniques using Flexbox, CSS Grid, and absolute positioning. This will help you organize the content you've added into a clean, structured, and visually balanced layout.

CSS Setup

- Base width + padding for the flex container:
 - `.flex { max-width:1600px; width:100%; padding:80px 10px; box-sizing:border-box; }`
- Flexbox layout:
 - `.flex { display:flex; flex-direction:row; }`
 - `.flex2 { width:50%; }`
- This places the two inner sections side-by-side at 50% width each.

Services Section

- Use the same structure as the About section:
 - Outer wrapper: `<div class="flex">`
 - Inner halves: `<div class="flex2">`
- Add `<p>Hello World</p>` as placeholder text until the accordion is added later.

Footer - Flexbox Layout

- Add a wrapping div inside the footer with the class "flex-foot".
- Wrap each footer column in a div with class "flex3":

```
1 <div class="flex3">
2   <p>Give us a call</p>
3   <a href="tel:615-876-4054">(615) 876-4054</a>
4 </div>
```

Notes: If you need more guidance on Flexbox you can learn more here https://www.w3schools.com/css/css3_flexbox.asp

03: The Layout

You'll explore layout techniques using Flexbox, CSS Grid, and absolute positioning. This will help you organize the content you've added into a clean, structured, and visually balanced layout.

CSS for Footer Flexbox

- Flex container:
 - `.flex-foot { max-width:1600px; margin:0 auto; width:100%; display:flex; flex-direction:row; }`
- Footer column widths:
 - `.flex3 { width:33.33%; }`

Footer Link Alignment

- Use inline or inline-block for the bottom navigation links.

Display Property Overview

- `display: block;` - Full width, starts on a new line.
- `display: inline;` - No new line, only takes needed width (good for `<span>` or `<a>`).
- `display: inline-block;` - Behaves like inline but allows width/height/padding.
- `display: flex;` - Turns the element into a flex container.

Your layout should now include:

- A positioned hero section
- Flexbox About and Services section
- Flexbox Footer

STOP and submit your work to your trainer before moving to step 4.

Notes: If you need more guidance on Flexbox you can learn more here https://www.w3schools.com/css/css3_flexbox.asp



04: The Styles

You'll start adding personality to your site through basic CSS styling. This includes setting background colors, fonts, text colors, line heights, and hover effects to bring your content to life.

Core CSS Concepts to Keep in Mind

Box Model

- Margin = space outside an element
- Padding = space inside an element
- Border = optional outline
- Width/height = element size
- Box-sizing = how width/height are calculated (use border-box for predictable sizing)

Common Styling Properties

- Text: color, font-size, font-family, font-weight, line-height, text-align
- Backgrounds: background-color, background-image
- Layout tools: Flexbox (already used), Grid
- Visual extras: opacity, box-shadow, border-radius, transition
- Interaction: hover, focus, active (pseudo-classes)

Hero Section Styling

We'll style this one together - the rest you'll match to the mockup on your own.

Notes: If you need more guidance on CSS you can learn more here https://www.w3schools.com/css/css_intro.asp



04: The Styles

You'll start adding personality to your site through basic CSS styling. This includes setting background colors, fonts, text colors, line heights, and hover effects to bring your content to life.

1. Set up the hero container

- Give the hero section a fixed height and full width.
- Make sure the background image fills the entire area.
- Use object-fit on the image to control how it scales:
 - fill = stretches (may distort)
 - contain = fits entirely (may leave empty space)
 - cover = fills container while keeping aspect ratio (may crop)

2. Improve readability

- Darken the hero image slightly using a filter so the text stands out.

3. Center and limit the text

- Center the text block and give it a max width so it doesn't stretch too wide.

Fonts

The site uses Oswald and Roboto.

How to add them

- Go to Google Fonts and add both fonts to your "bag."
- Open the bag and choose the <link> option.
- Copy the embed code into your <head>; add the CSS classes to your styles.css.

Notes: If you need more guidance on CSS you can learn more here https://www.w3schools.com/css/css_intro.asp



04: The Styles

You'll start adding personality to your site through basic CSS styling. This includes setting background colors, fonts, text colors, line heights, and hover effects to bring your content to life.

Hero Text Styling

- Style the H1 to match the mockup (size, line height, uppercase, color, weight). font-size in mockup is 85px,
- Add a subtle text shadow to improve contrast on the image.
- Add spacing below the heading before the button.

Buttons

To keep all buttons consistent:

- Add a shared class to every button that uses the red style.
- In CSS, style that class with:
 - Padding
 - Centered text
 - Red background
 - White text
 - Matching font
 - A transition for hover effects
- Center the button within the hero text block.

Notes: If you need more guidance on CSS you can learn more here https://www.w3schools.com/css/css_intro.asp



04: The Styles

You'll start adding personality to your site through basic CSS styling. This includes setting background colors, fonts, text colors, line heights, and hover effects to bring your content to life.

Continue Styling the Page

- Use the mockup as your guide for spacing, colors, and alignment.
- Skip the quick links slider and accordion for now.
- Colors used in the design:
 - Black: #202020
 - White: #fff
 - Red: #bf3119
 - Light Red: #f9492a

After you add the CSS it should be looking pretty close to the mockup minus the slider, accordion, and header.

STOP and submit your work to your trainer before moving to step 5.

Notes: If you need more guidance on CSS you can learn more here https://www.w3schools.com/css/css_intro.asp



05: The Header

You'll build a header and a navigation while also learning how our inventory links are structured. You'll also use Font Awesome for icons and create showroom links to showcase new models offered by Manufacturers.

Navigation Files & Setup

- Navigate to the Ex5 folder and you'll find the resources for the navigation
- Copy the menu.css into your css folder the mobilemenu.js and jquery into your js folder and the header.htm in your site root folder
- Add a link in the head to your menu.css
 - <link rel="stylesheet" type="text/css" href="/css/menu.css?v=1.0">
- Add a link before the closing body and html tags to your JS files
 - <script type="text/javascript" src="/js/jquery-3.5.1.min.js"></script>
 - <script type="text/javascript" src="/js/mobilemenu.js"></script>
 - Make sure the jquery is above the mobile menu. The jquery needs to load first.
- Add the includes in your default.htm file in the body where you want the navigation to show.
 - <!--#include file="header.htm" -->
 - The purpose of includes files is to call in repeated HTML files. Typically, we use these for header and footers. Once a change is made in an includes file it will update across all pages.
- Once your includes, css, and js are added go to your header.htm tab to review the navigation code.
- Add an image to your logo and an alt tag.



05: The Header

You'll build a header and a navigation while also learning how our inventory links are structured. You'll also use Font Awesome for icons and create showroom links to showcase new models offered by Manufacturers.

Understanding the Navigation Structure

You'll see two main <nav> elements inside header.htm:

- .primary → desktop navigation
- .mobile → mobile navigation

Whenever you update links, you must update both.

Parent & Child Links

- Top-level items = Parent
- Dropdown items = Child

Mobile Menu Behavior

- Links that open submenus use the class click
- Their child list uses drop-list
- The mobilemenu.js file controls how these open/close
 - Spend a moment reviewing the JS so you understand how the toggle works

menu.css

- This file contains all styling for both desktop and mobile navigation
- Adjusting values here will help you understand spacing, hover states, dropdown behavior, etc.

Notes: You shouldn't need to adjust the structure of the navigation in CSS unless you want to. If you mess up click Ctrl + Z to undo the changes in DW



05: The Header

You'll build a header and a navigation while also learning how our inventory links are structured. You'll also use Font Awesome for icons and create showroom links to showcase new models offered by Manufacturers.

Adding Font Awesome Icons

- link the following stylesheet in the <head>
 - <https://media.sandhills.com/CDN/CSS/Font-Awesome/7.0.0/css/all.min.css>
- Choose an icon from fontawesome.com and paste its code into header.htm where {ICON} is shown

Important Rule for All Resources

Whenever you reference Stylesheets, Scripts, Images, Inventory links, always start the path with a forward slash /
This ensures the resource loads correctly on inventory pages.

Notes: You shouldn't need to adjust the structure of the navigation in CSS unless you want to. If you mess up click Ctrl + Z to undo the changes in DW



05: The Header

You'll build a header and a navigation while also learning how our inventory links are structured. You'll also use Font Awesome for icons and create showroom links to showcase new models offered by Manufacturers.

InventoryLayout.htm (Important)

- This file controls all inventory pages
- In the <body>, you'll see a container that loads the inventory <!--##REPLACE##-->
- Do not edit this container
- Do not add jQuery to this file (inventory pages already load their own)

Learning Inventory Links

Inventory links are the most important new skill in this step.

They connect your navigation to the dealership's live equipment listings.

How to Explore Inventory

1. Go to file:///sandhills.int/dev/HostDealerPages/Training/SAW/facetedinventory.htm
2. Type Hilliard Lawn & Garden into "Enter Filter"
3. Click Generate DS Companies
4. Make sure the dropdown is set to All Equipment
5. Click View All Equipment
6. A new tab opens — this is the customer's full inventory

Notes:



05: The Header

You'll build a header and a navigation while also learning how our inventory links are structured. You'll also use Font Awesome for icons and create showroom links to showcase new models offered by Manufacturers.

/inventory/?/listings/for-sale/equipment/all?DSCompanyID=137820&settingscrmid=29568337

Calls in
InventoryLayout
page

for sale, for rent,
or for retail

Category
number

Dealer specific
inventory

Controls
settings in IM

Building the Base Inventory Link

The URL you see will look something like the link above

To turn this into a working site link, you must add:

- /inventory/?/ at the beginning
- &settingsCRMID= at the end - the settingscrmid for this one is 22449379

Category Links

To create links for specific categories:

1. While viewing All Equipment, click a category in the left sidebar
2. The URL will update
3. Look for &category=



05: The Header

You'll build a header and a navigation while also learning how our inventory links are structured. You'll also use Font Awesome for icons and create showroom links to showcase new models offered by Manufacturers.

1. Copy the number that follows
2. Add &category={number} to your All Equipment link

Repeat this for:

- Outdoor Power

For base categories you can choose the category in the filter drop down. Do this for Farm Equipment, Construction Equipment, and Attachments

Why This Matters

- These links filter inventory dynamically
- They must be accurate or customers will see incorrect results
- Every dealership site uses this structure

Showroom Links

Showroom pages display manufacturer catalogs, not in-stock items.

Key Differences

Showroom links use:

- &LocationID= instead of DSCompanyID



05: The Header

You'll build a header and a navigation while also learning how our inventory links are structured. You'll also use Font Awesome for icons and create showroom links to showcase new models offered by Manufacturers.

- &lsShowroom=1
- &Manufacturer={brand name}

You'll be given the LocationIDs for training:

- Ferris: 350005465736
- Toro: 350004905506
- Stihl: 350004905491

Add these showroom links under New Equipment in the navigation.

Note: Showroom pages are a paid feature - only add them when requested by the sales rep.

Final Steps

- Update both desktop and mobile navs
- Test every link
- Confirm icons display
- Confirm dropdowns work
- Confirm inventory links load the correct filtered pages

STOP Submit your task and notify your trainer before moving to Step 6.

Notes: search for all category numbers here <http://intranet/mmmcore/CategorySearch> (save this page in your browser)



06: The Interactive Components

You'll add interactive components like an image slider and an accordion to enhance user engagement.

These elements help make your site feel more dynamic and user-friendly.

Slider Options You'll See in Real Projects

- <https://kenwheeler.github.io/slick/> - Slick slider (this is commonly used on sites for brands or inventory link sliders)
- <https://jquery.malsup.com/cycle2/> - Cycle slider (this is commonly used for hero sliders)
- <https://swiperjs.com/> - Swiper Slider (this is another slider you might see for sliders)
- Today we'll use the swiper slider

Explore Swiper First

Go to the Swiper demos:

<https://swiperjs.com/demos>

Spend a few minutes looking at:

- Basic sliders
- Navigation arrows
- Pagination dots
- How the HTML structure is set up
- How the JS initializes the slider

You'll be recreating the equipment links slider based on the mockup, so understanding the structure is important.



06: The Interactive Components

You'll add interactive components like an image slider and an accordion to enhance user engagement.

These elements help make your site feel more dynamic and user-friendly.

Implement Swiper Using the Official Docs

Instead of relying on prewritten code, go to swiperjs.com and copy the example structure directly from their documentation.

You'll need to:

- Match the HTML structure Swiper's required markup
- Add the correct classes for Swiper containers, wrappers, and slides
- Initialize Swiper in your JS (based on the examples)
- Adjust settings (slides per view, spacing, navigation, etc.) to match the mockup
- Adjust styles to match the mockup

This step is intentionally open-ended so you can learn how to read documentation and adapt it to a real project.

Notes About Other Sliders (For Future Projects)

You don't need this today, but you'll see it later:

- Slick Slider sometimes requires editing its internal CSS files
(slick-1.6.0 → slick → slick.css)
- Cycle2 uses data attributes in the HTML to control transitions

Swiper is more modern and flexible, which is why we're using it here.

Notes:



06: The Interactive Components

You'll add interactive components like an image slider and an accordion to enhance user engagement.

These elements help make your site feel more dynamic and user-friendly.

Add the Accordion HTML

You'll be given the base HTML for one accordion item.

Use that single item as your template and duplicate it to create the rest.

Important:

The classes accordion and panel must stay exactly the same.

The script depends on these class names to open and close each section.

Add the Script File

- Place the provided accordion.js file into your Ex1/js folder.
- Add a script link to it in the footer of your page (right before `</body>`). `<script type="text/javascript" src="/js/accordion.js"></script>`
- You don't need to modify the script — just make sure it's linked correctly.

```
1 <button class="accordion">Mower Tune-Ups <i class="fa-solid fa-plus" aria-label="View More"></i></button>
2 <div class="panel">
3   <p>Service details: Change oil & filter, Change fuel filter, Clean or replace air filter, Clean & adjust carburetor, Install new spark plug(s), Sharpen &
  Balance blades, Check belts </p>
4 </div>
```

Notes: The text for the rest of the accordion text is in Ex6 labeled Accordion Copy

06: The Interactive Components

You'll add interactive components like an image slider and an accordion to enhance user engagement.

These elements help make your site feel more dynamic and user-friendly.

Add the Accordion CSS

Copy the accordion CSS into your style.css file.

- the `.panel {max-height: 0; overflow:hidden; transition: max-height 0.2s ease-out;}` is necessary for the script to work. Everything else can be edited to match the site you're working on. These are already styled for you.
- You CAN customize colors, padding, borders, fonts, hover styles, active styles, etc.

```
1 .accordion {color:#000; cursor: pointer;padding: 25px 10px;font-size: 22px;width: 100%;text-align: left;border: none;outline: none;transition: 0.4s;display:
  flex;justify-content: space-between;background-color: #fff;border-bottom: 1px solid #000;font-family:'Oswald', sans-serif; }
2 .active, .accordion:hover {
3
4 }
5 .panel {
6   padding: 0px 10px;
7   background-color: #fff;
8   max-height: 0;
9   overflow: hidden;
10  transition: max-height 0.2s ease-out;
11 }
12 .panel p{padding: 25px 10px;}
```

STOP Submit your task and notify your trainer before moving to Step 6.

07: The Responsive Approach

You'll use media queries to make your layout responsive. With over half of all web traffic coming from mobile devices, this ensures your site looks great and functions well on any screen size.

Making the Site Responsive

What Media Queries Do

Media queries let you change your CSS based on screen size.

They're used to:

- Adjust layouts for mobile, tablet, and desktop
- Rearrange content for smaller screens

You'll write your responsive styles inside media queries.

Basic Media Query Structure >>

```
1 @media only screen and (max-width: 1200px) {  
2     /* Styles for screens 1200px and smaller */  
3 }
```

Common Breakpoints

These are general guidelines - adjust them based on the site you're building:

- Large Desktop: 1025px and up
- Small Desktop: 769px–1024px
- Tablet: 481px–768px
- Mobile: 300px–480px



07: The Responsive Approach

You'll use media queries to make your layout responsive. With over half of all web traffic coming from mobile devices, this ensures your site looks great and functions well on any screen size.

Best Practices for Responsive Design

These are the habits you should build into every site:

Design for thumbs

- Buttons and links must be easy to tap. Text must be readable without zooming

Use flexible units

- Percentages and relative units help layouts adapt. Avoid fixed pixel widths unless necessary

Improve load time

- Add loading="lazy" to images below the fold. This delays loading until the user scrolls to them

Prioritize content

- On mobile, important content should appear first
- You may need to reorder elements for a better mobile experience

Keep the mobile header small

- Navigation should not dominate the top of the screen

Avoid horizontal scrolling

- If you see a sideways scroll, something is too wide

Your Task

- Add media queries for the breakpoints listed above. Adjust spacing, font sizes, layout, and images so the site matches the mockup at each size.

Test your site on multiple screen widths. Make sure nothing breaks, overlaps, or scrolls sideways

Once your site looks good at all sizes, submit your task before moving to Step 8.



08: The Page Build

You'll build a contact page on your own with minimal guidance. The goal is to match the styling of your homepage while incorporating some new elements.

Build the Contact Page

Your home page is complete and responsive - now you'll create a contact page that matches the style and structure of the rest of the site.

Customer Request Summary

The client wants a clean, user-friendly contact page that includes:

- A simple contact form
 - Name
 - Email
 - Phone
 - Message
- Their phone number: (614) 876-4054
- A map showing:
5300 Cemetery Rd, Hilliard, OH 43026
- Business hours:
Mon–Fri: 8:00AM–5:00PM
Sat–Sun: Closed

Your layout and styling should match the look and feel of the home page.

Notes: You can use online resources to look up design inspiration for this page just don't copy anything exactly



08: The Page Build

You'll build a contact page on your own with minimal guidance. The goal is to match the styling of your homepage while incorporating some new elements.

Set Up the Contact Page

- Create a new HTML file for your contact page.
- Copy over the head, header includes, footer includes, and scripts from your home page so everything stays consistent.
- Add a link to this new contact page inside header.htm so it appears in the navigation.

Build the Contact Form

- Your form must include the inputs in the dealers request.
- The first input on all forms must have a "Primary_Name" input in order for the forms to process correctly - please review all requirements for our forms here: <file:///sandhills.int/dev/HostDealerPages/Training/SAW/credithowto.htm>
- Style your form and submit button to match the website design.

Add Required Form Scripts & Includes

Open the Form Resources document in Ex8. It contains everything you need.

Add these items:

- Form Validation Script → goes in the <head>
- formoutput.js → add to your js folder and link it in the footer
- Form include code → must be placed directly under <!DOCTYPE HTML>
- Captcha include → add it inside your form before the submit button

The include and captcha code varies by publication. You can find the correct version here: <file:///sihtmldev1.sandhills.int/Websites/Training/SAW/credithowto.htm#inc>

Notes: If you need more guidance on forms you can learn more here https://www.w3schools.com/html/html_forms.asp



08: The Page Build

You'll build a contact page on your own with minimal guidance. The goal is to match the styling of your homepage while incorporating some new elements.

Required Form Structure

Your form tag must include:

- `onsubmit="return validation(this)"`
- `action="contact.htm"` (must match your page name)
- `method="post"`

Required Hidden Inputs

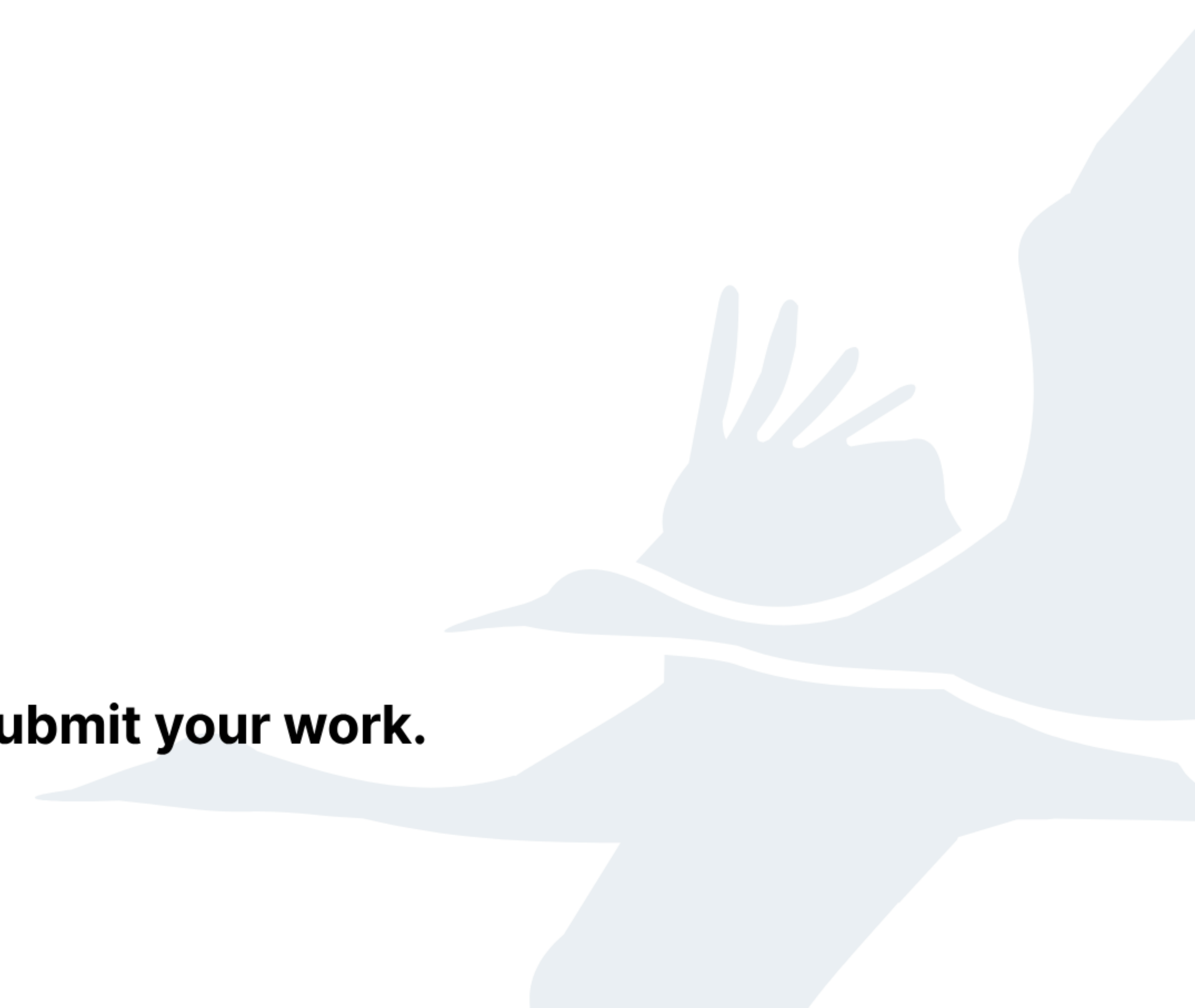
These must be inside your form:

- `form_fieldlist` - Must list the exact names of your form fields
- `form_subject` - The subject line that will appear in the email
- `form_thankyou` - The page users are sent to after submitting
- `form_to` - The email address the form sends to

These hidden fields are essential for the form to submit correctly.

- Next add the Map, Phone Number & Hours
- Make the Contact Page Responsive

Once your contact page matches the site design, includes all required fields, and is fully responsive please submit your work.



09: The Customer Requests

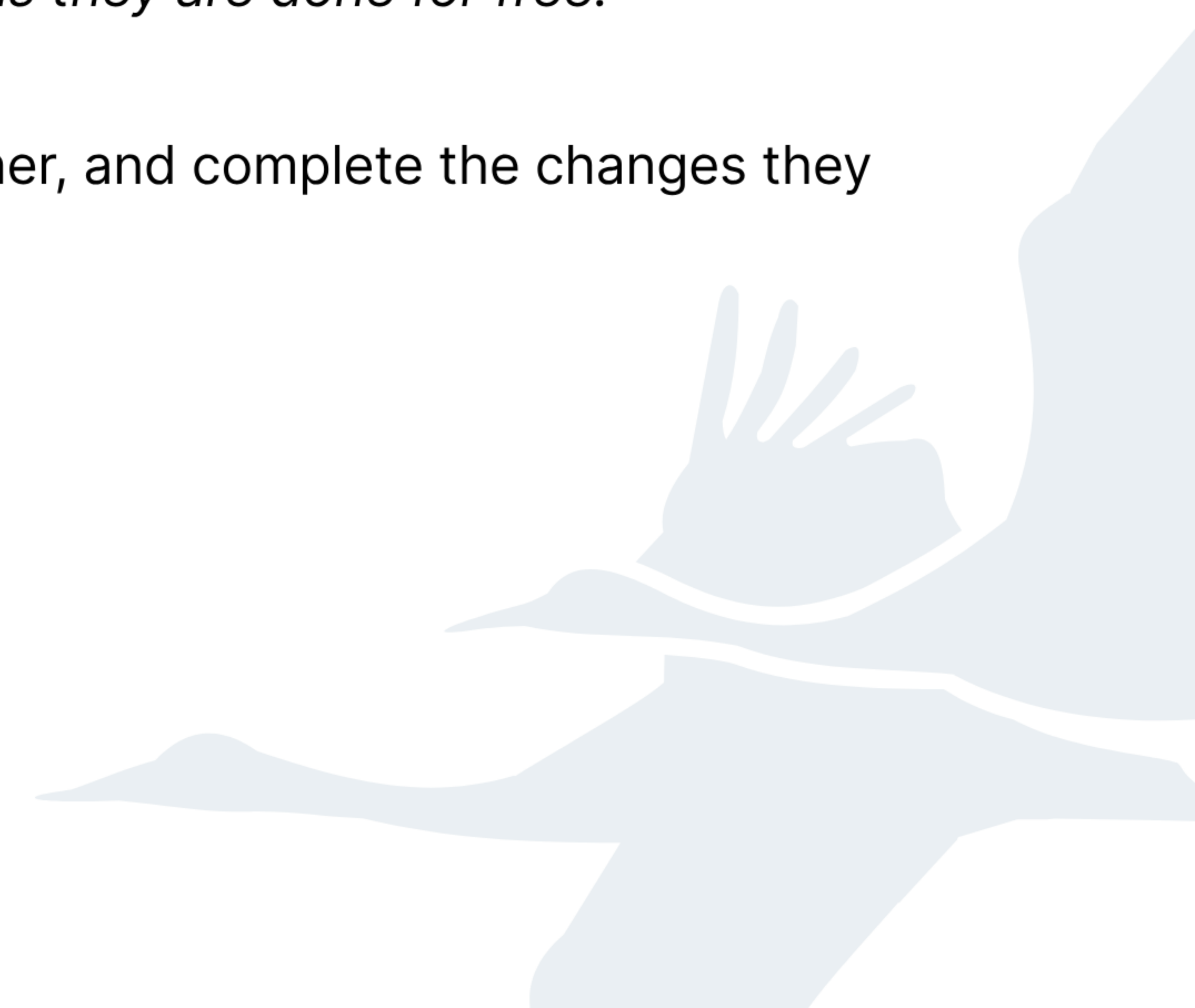
You'll learn how to estimate the time required for common website updates and practice editing existing code. You'll get comfortable with real-world change requests, like updating contact info, replacing images, and applying 301 redirects.

- Now that you have a fully built Home page and Contact page we'll practice quoting a completing a customer request.
- Changes get quoted at 50 dollars an hour and prorated by 15 minute increments. The smallest a change could be charged is 12.50
- You submit your quotes in quarters so 15 minutes = .25, 30 minutes = .5, 45 minutes = .75, etc.
 - We'll show you how to do this when you begin working on live sites and using our TO system. For now please just reach out to your trainer with your quote and they'll approve or work with you to come up with a new quote.

Note: If the change is an inventory link change or the website is broken and needs fixed, those are quoted at 0 as they are done for free.

There is a website request scenario in Ex9 titled Website Changes. Please ready it, send your quote to your trainer, and complete the changes they requested.

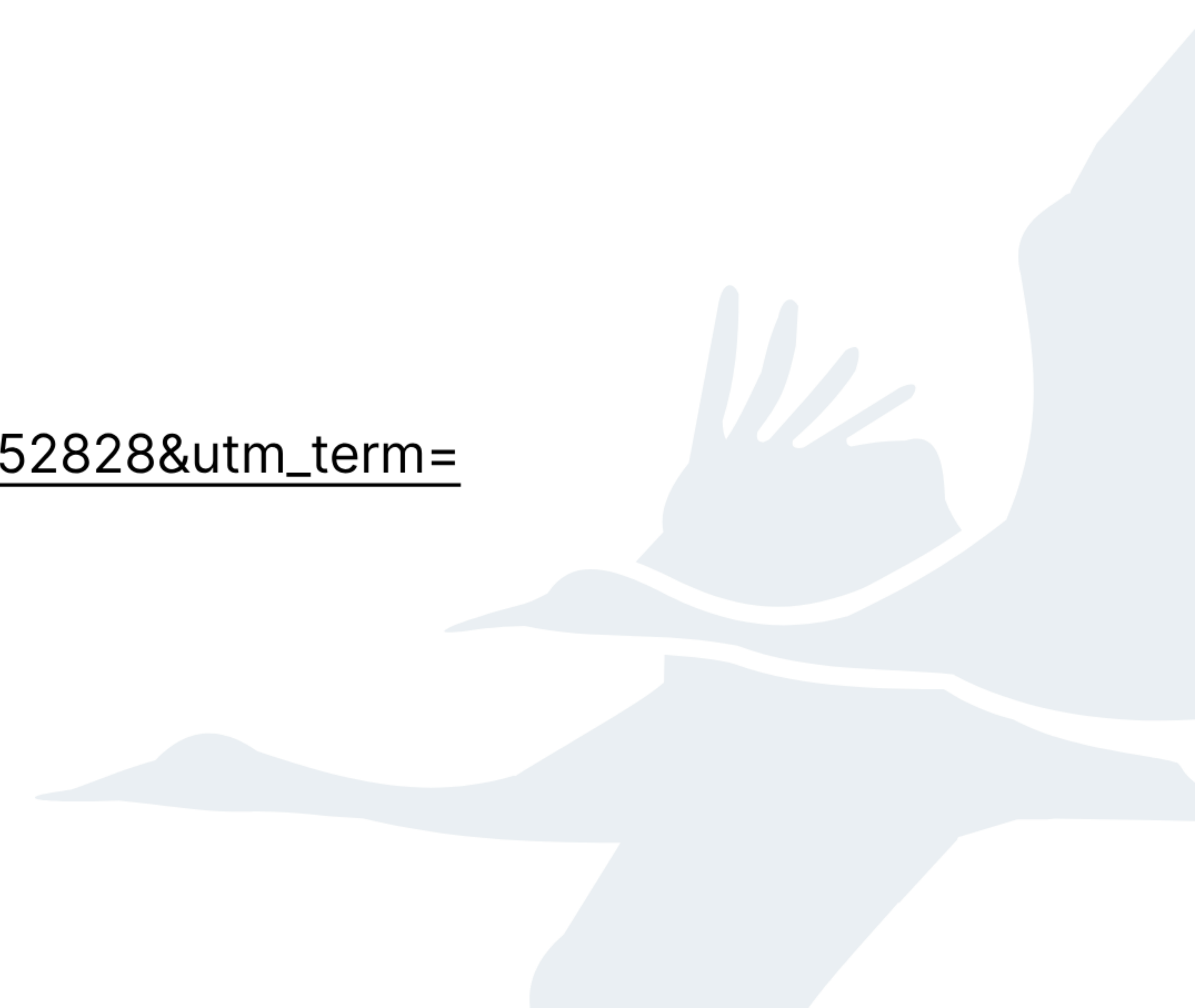
Submit your task and let your trainer know it's been submitted.



10: The Design

This final step is a design challenge where you'll create a homepage entirely on your own. It's an opportunity to showcase your design style and apply everything you've learned across the previous steps - without direct guidance.

- Go to Ex10 there should be a js, css, and siteart folder.
- In this exercise, you are going to mockup a fictitious business. The home page should include a logo, text, images/photographs and navigational links with hover states. These links can all link to Google.com for the time being. Have fun with this exercise! It is just for practice and the only person to see it will be your training partner.
- First, create a home page mockup within Photoshop, Adobe XD, or Figma
 - The desktop should be 2000px wide and however long you need.
- You can use a few places for inspiration including pinterest, <https://www.awwwards.com/>, <https://99designs.com/inspiration/websites> or any other resources you can think of.
- Please browse UX best practices
 - <https://lawsofux.com/>
 - <https://www.siegemedia.com/contrast-ratio>
 - <https://www.siegemedia.com/contrast-ratio>
 - https://elementor.com/blog/grid-design/?utm_source=google&utm_medium=cpc&utm_campaign=10759652828&utm_term=



10: The Design

This final step is a design challenge where you'll create a homepage entirely on your own. It's an opportunity to showcase your design style and apply everything you've learned across the previous steps - without direct guidance.

- We are not permitted to use any stock images we do not own or the customer does not own.
- We have stock images here: \\sandhills.int\\file\\WebDesignData
 - Please pin this to your quick access if you haven't done so already
- Once the mockup is complete email over a jpg copy to your trainer to review.
- Once they approve you may start on the build of the site.
- Build the site and make it responsive. Please ask for help if any questions arise.

Submit your task and let your trainer know it's complete.



11: The Next Steps

Now that you've completed the training, you'll start working on customer change requests. Before you get started, look over the following resources.

- Congrats! You've completed the initial training there are still many concepts to cover, but it is impossible to list them all in training documentation, so now it is time to learn from real customer requests and experiences.

Over the next few months, we will be introducing you to these concepts:

- WordPress
- New website mockups and builds
- eCommerce pages
- SEO Audits
- Google Analytics
- Bobcat WP Themes
- How to work in CRM
- How to work within the Sandhills Cloud

