Planning a Web Page Structure

Our goal for this module is to identify and correctly use HTML page sectioning elements. Learn to organize your page content into logical pieces!

Please Review:
https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Document_and_website_structure

1. **Basic Page Structure** (We've used these in class)
2. **Headings** (A review)
3. **Page Sub Sections** (These are new!)

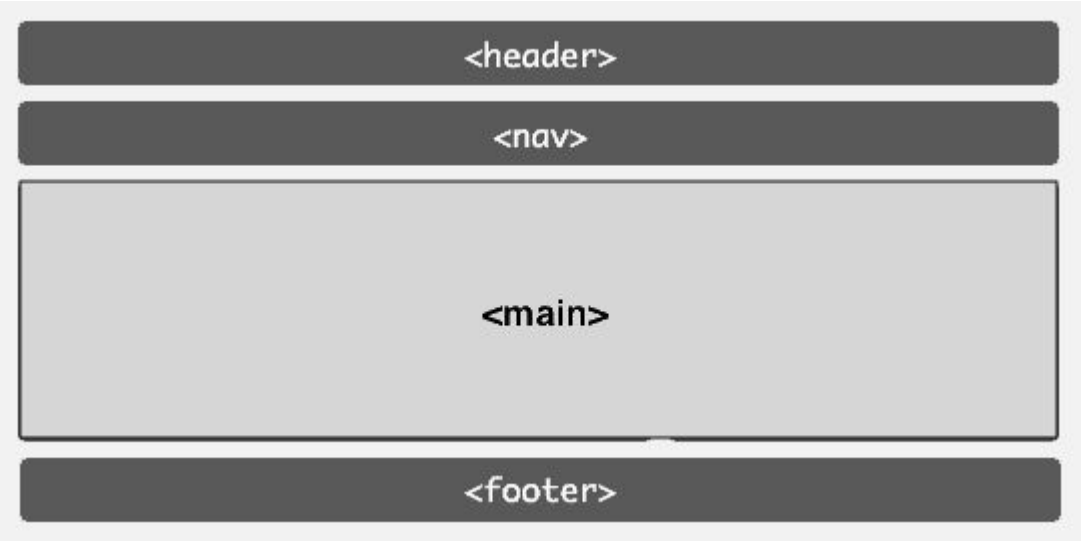
Basic Sections of a Web Page

Text Excerpts from [MDN Web Docs]

"*Webpages can and will look pretty different from one another, but they all tend to share similar standard components, unless the page is displaying a fullscreen video or game, is part of some kind of art project, or is just badly structured*"

Header, Navigation, Main Content Area, Footer

Web pages follow the same pattern in most cases. Think of websites you visit, looks familiar, right?



```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>A Basic Web Page</title>
  </head>
  <!-- A Common Web Page Structure -->
  <body>
    <!-- Introductory Content -->
    <header>

    </header>
    <!-- Links to other sections of the website -->
    <nav>

    </nav>
    <!-- The main topic, content or functionality of this page -->
    <main>

    </main>
    <!-- Additional Site Information like legal info, contact or others -->
    <footer>

    </footer>
  </body>
</html>
```

<header>

You usually see pieces of information like a logo, page name, business name, navigation (sometimes), search functionality, and other important links at the top of the page. It should be defined by the header element.

“The **HTML `<header>` element** represents introductory content, typically a group of introductory or navigational aids. It may contain some heading elements but also a logo, a search form, an author name, and other elements.”
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/header

<nav>

Links to other places in the website - these links are usually available everywhere and are represented in lists of hyperlinks, buttons or tabs. You can have seperate lists of links to current page sections only or web pages that are in context to a particular area of the website.

“The **HTML <nav> element** represents a section of a page whose purpose is to provide navigation links, either within the current document or to other documents. Common examples of navigation sections are menus, tables of contents, and indexes.”
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/nav

<main>

The main element defines the main topic or functionality of a given web page. This is unique to the current page - it shouldn't be used to contain parts of the page that repeat across an entire site like the header, main navigation links, footer or in some cases a sidebar.

"The **HTML <main> element** represents the dominant content of the <body> of a document. The main content area consists of content that is directly related to or expands upon the central topic of a document, or the central functionality of an application."
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/main

<footer>

Located at the bottom of the document, the footer page area usually contains legal information, contact information and any other important site information that might not be suitable for the header.

The **HTML `<footer>` element** represents a footer for its nearest sectioning content or sectioning root element. A footer typically contains information about the author of the section, copyright data or links to related documents.
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/footer

HTML is a mark-up *language*, its usefulness relies on proper use of syntax and structure. The more meaningful (semantic) HTML has become - through development and use of purpose-driven *elements*, the addition of a variety of *attributes,* and by maintaining the role of HTML as purely structural - the stronger the web becomes. Proper use of HTML is only one part of Web Standards, but it's an important piece of which we'll focus on heavily in this course.

HTML Semantics - referring to the use of HTML elements based on the role or meaning of the content within - can be applied by, choosing the most meaningful element to contain page content, while also ensuring the elements used are accurate according to what they mean. *For example, you wouldn't use a heading element (h1-h6) to contain text that is not a heading.* This would be confusing to technologies (like bots, screen readers) interpreting the content of the page.

Headings

Headings are important to overall page structure as they provide a way to scan the page for relevant page sections that are of interest to the reader. They help provide context to the page sections defined above to assistive devices like screen readers as well as to other technologies interpreting the page like bots. Proper heading use greatly increases accessibility of web applications.
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/Heading_Elements

<h1> - <h6>

- Use within page content areas to give context and meaningful structure to that particular area
- Headings should be used in order, beginning with the **<h1>** as it is the most important heading on the page. (Often page level title or company/site name).
  - Only use one <h1>
- Each number increment implies less importance.
- Avoid skipping a level (if you have an h2 on the page, your next used heading should be h3 when the importance of the section is not as great as the previous)

Sub-Sections of a Web Page

Review the following HTML elements which are used to define page **sub-sections**. You will not always use every one of these elements.

Article, Aside, Section, Div

`<article>`

Pieces of content that can be removed from the document and retain all of the context needed to stand-alone (headings paragraphs and subsections which make up a whole)

The **HTML `<article>` element** represents a self-contained composition in a document, page, application, or site, which is intended to be independently distributable or reusable (e.g., in syndication). Examples include: a forum post, a magazine or newspaper article, or a blog entry.
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/article

<article>

```
<!-- The main topic, content or functionality of this page -->
<main>
  <!-- If you do have an article on the page, it will likely go in the main element
  as its usually the focus of the page its on -->
  <article>
    <h2>An Introduction to Lacrosse</h2>
    <p>The sport has four versions that have different sticks, fields, rules and equipment: field lacrosse, women's lacrosse,
    box lacrosse and intercrosse. The men's games, field lacrosse (outdoor) and box lacrosse (indoor), are contact sports and
    all players wear protective gear: helmet, gloves, shoulder pads, and elbow pads. The women's game does not allow body
    contact but does allow stick to stick contact. The only protective gear required for women players is eyegear, while
    goalies wear helmets and protective pads. Intercrosse is a mixed-gender non-contact sport that uses an all-plastic stick
    and a softer ball.</p>
    <h3>Lacrosse Facts</h3>
    <p>Lacrosse is based on games played by various Native American communities as early as 1100 AD. By the seventeenth
    century, a version of lacrosse was well-established and was documented by Jesuit missionary priests in the territory of
    present-day Canada.</p>
    <h3>Lacrosse Equipment</h3>
    <h4>Sticks</h4>
    <p>Stick length is governed by NCAA regulations, which require that men's sticks be from 40 to 42 inches long for offensive
    players, 52 to 72 inches long for defensemen, and 40 to 72 inches long for goalies.</p>
  </article>
</main>
```

<aside>

This element defines content on the page which is related to the main topic or functionality but indirectly - it could be an ad, navigational, resource lists, a figure, and may be present on other pages as well.

The **HTML `<aside>` element** represents a portion of a document whose content is only indirectly related to the document's main content. Asides are frequently presented as sidebars or call-out boxes.
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/aside

<aside>

```
30      <!-- This collection of page content is indirectly related to the main topic of the page  -->
31      <aside>
32        <h3>Books About Lacrosse</h3>
33        <p>American Indian Lacrosse: Little Brother of War by Thomas Vennum</p>
34        <h4>Read Reviews</h4>
35        <p>I can't say enough good things about this book! I picked it up while at a book sale and
          immediately had to have it. It's full of photographs of American Indian teams and tells you
          the full story of Lacrosse and you get so much more--it's an in-depth look at how it affected
          the peoples and tidbits about their daily lives. Interspersed between each chapter are short
          'stories' about lacrosse events. This is a great book, and a wealth of information. It's about
          more than just lacrosse. Don't hesitate to get this!</p>
36      </aside>
37    </main>
```

<section>

A section element defines content within as being related however it does not imply a relationship outside of it. A heading is recommended to provide enough context for this page area.

The **HTML `<section>` element** represents a standalone section — which doesn't have a more specific semantic element to represent it — contained within an HTML document. Typically, but not always, sections have a heading.
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/section

<section>

You can use the section element as many times as you need to in a single web document. You can even nest a section within a section if it is semantically appropriate.

```
34          <!-- Keep content on the same topic together as a sub-section  -->
35          <section>
36            <h4>Read Reviews</h4> <!-- Sections should contain a heading for context -->
37            <p>I can't say enough good things about this book! I picked it up while at a book sale and
              immediately had to have it. It's full of photographs of American Indian teams and tells you
              the full story of Lacrosse and you get so much more--it's an in-depth look at how it
              affected the peoples and tidbits about their daily lives. Interspersed between each chapter
              are short 'stories' about lacrosse events. This is a great book, and a wealth of
              information. It's about more than just lacrosse. Don't hesitate to get this!</p>
38          </section>
```

`<div>`

The division element is a *semantically meaningless* element useful for containing page content within when no other sectioning element makes sense. It’s better to use a div element then to misuse a more meaningful element!

As a "pure" container, the `<div>` element does not inherently represent anything. Instead, it's used to group content so it can be easily styled using the `class` or `id` attributes, marking a section of a document as being written in a different language (using the `lang` attribute), and so on.
https://developer.mozilla.org/en-US/docs/Web/HTML/Element/div

The content within doesn't need to be related. You can use div's for scripting and styling purposes as needed but always use a more semantic alternative if there is one available!

```
<header>
  <!-- Containing parts of the header seperately from the form for later styling purposes -->
  <div>
    <h1>The Sport of Lacrosse - Sports Encyclopedia</h1>
    <img src="images/logo.png" alt="Sports Encyclopedia Logo">
    <p>Community Curated</p>
  </div>
  <form method="get" action="scripts/search.php">
    <!-- Containing a label with it's input for layout purposes -->
    <div>
      <label for="search">Search The Encyclopedia</label>
      <input type="search" name="search" id="search">
    </div>
    <button type="submit">Search</button>
  </form>
</header>
```

Assignment One

Please read the rubric uploaded on BlackBoard for full instructions.

Now that you know more about the basic page structure, use most of these sectioning elements in your recipe page (header, main, footer, nav, article, aside, section, div) along with all of the required pieces of HTML (doctype, character encoding, title, head, body and html *in the correct order*). Focus on using the **correct element for the job.**

Remember to Validate! https://validator.w3.org/