

1. HTML & CSS – Page Layout

Let’s explore the basics of webpage layout. That means we need to understand the structure of HTML and CSS. Let’s get started.

2. Key Topics

In this lesson,

- We’ll start by identifying the foundational code (or markup) that you will find on any webpage
- Then we’ll learn the syntax of HTML and CSS. By learning the patterns, you will be able to write your own code statements, and you will be able to leverage the code of other developers.
- We’ll also take a closer look at the HTML5 semantic elements, especially those that relate to page layout.

4. Foundation

0. We’ve got a blank page with a web extension, let’s say .HTML

1. Our first line is a declaration that tells the web browser, “Hey, this here code as follows is written in HTML, specifically HTML5.
2. Then we add the opening and closing tags `<html>` and `</html>`. And this element tells the browser where the HTML code (or markup) is contained

Two side notes...

1. HTML stands for Hyper Text Markup Language – it’s actually less “code” and more “markup.” That is to say, we define (or contain) content by using the open and close tags of an HTML element. For example, paragraph begins, here’s the paragraph, paragraph ends. Or in this example, HTML stuff begins, here’s all the HTML stuff, and were done. It’s reminiscent of WordPress Reveal Codes or XML or the Karate Kid exercise of Wax On / Wax Off.
 2. We can actually include within this HTML inline blocks of other languages, such as CSS or JavaScript. And we can reference external pages written entirely in CSS or JavaScript etc.
3. Within the HTML, there are two sections... The head and...
 4. ... the body

5. Foundation

0. The head contains information about the page. And the body contains the content that’s on the page

1. For example, the head might contain a page title or metadata like keywords, description, author
2. The head will also contain CSS and other scripts – either embedded internally or, more commonly, and as we see in this example, referenced externally
3. And then in body we add the layout elements and content... let’s take a close look at this body area

6. Layout

A common layout consists of a DIV container, which allows us as developers to define alignment, width, columns, etc. And if we use responsive CSS, then we can tell this DIV how to adapt depending on the device – desktop, tablet, phone – that’s rendering this page

And then we define the container elements for the header, the navbar (navigation area), the main content, and the footer.

And we could optionally include one or two asides, also call sidebars. And using CSS, we can create columns, left-side, right-side, even

7. Patterns – HTML

Both tags and elements refer to the markup used to write HTML. For example, you might use the `<p>` tag to define a paragraph or the `<a>` `` element to create links. Many people use the terms tag and element interchangeably, and if you find that easier, you can too. But there is a slight difference between the two terms. HTML is a markup language, which means the content on a web page is “marked up” with these codes to tell the browser how to display the text. And these markup tags are the HTML tags themselves. When you write HTML, you are writing HTML tags. An element is the basic building block of HTML and is typically made up of two tags: an opening tag and a closing tag.

HTML elements

An element in HTML represents some kind of structure or semantics and generally consists of a start tag, content, and an end tag.

Example: `<p>This is the content of the paragraph element.</p>`

Example: `
`

HTML tags

Tags are used to mark up the start and end of an HTML element.

Example: `<p>`

Example: `<p class="info">`

Example: `</p>`

HTML attributes

An attribute defines a property for an element, consists of an attribute/value pair, and appears within the element’s start tag.

The most popular misuse of the term “tag” is referring to alt attributes as “alt tags.”

8. Patterns – HTML

MOST COMMON!!!

html examples

```
<body> </body>
```

```
<div class="wrapper"> </div>
```

9. Patterns – HTML

empty (void or singleton) elements – hold no content

self-closing empty elements – for xHTML

```
<tag attribute= "value" attribute= "value" />
```

html examples

```

```

```
<br>
```

```
<br />
```

```
<!-- add header -->
```

10. Patterns – HTML / 12. Patterns – CSS

13. Patterns – CSS

Class – reuse per page

ID – once per page

css examples

```
body {  
  font: 100%/1.4 Verdana, Arial, Helvetica, sans-serif;  
  background-color: #f66;  
  margin: 0;  
  padding: 0;  
  color: #000; /* black font color */  
}
```

```
.wrapper {  
  margin: 0 auto;  
  width: 960px;  
  background: #eee; /* same as aside */  
  text-align: left;  
}
```

OR...

```
#wrapper {  
}
```

14. Patterns – CSS / 15. Patterns – CSS / 16. Patterns – CSS / 17. Patterns – CSS

19. Look It Up – References & Platforms

See also: w3schools.com

- www.w3schools.com/tags/default.asp
- www.w3schools.com/cssref/default.asp

20. HTML5 Elements

HTML5 includes several new semantic elements

Semantic elements not only contain content but also imbue meaning as to what that contain is

In the olden days, we used a DIV element and assigned to it the CSS attribute of “class = header” or “id = header”

But now with HTML5, we can use the HEADER element to contain the header – the banner, logo, title, masthead... -- and we can still style the header with CSS

And search engines love semantics for SEO ranking and indexing

And semantic elements make it easier to repurpose or migrate our content

Now the left column has key elements we would use in basic layout...

Header, nav, footer...

And in the content area... section, article, aside

21. HTML5 Elements – Section / Article

22. Think of a Newspaper

Think of a newspaper.

The paper comes in sections. You have the sports section, real estate section, maybe home & garden section etc.

Each of those sections in turn has articles in it. And some of those articles are divided into sections themselves.

– Estelle Weyl’s analogy

SECTION

- Thematic grouping of content
- Typically includes h1-h6 element as a child

ARTICLE

- Represent independently distributable, reusable, or syndicated content
- Examples: blogposts and comments, news articles, RSS feeds

GUIDELINES

- No real hard and fast rules, just guidelines
- Can nest articles inside a section, or vice versa;
- but don’t overdo it

23. DIV Tags and Classes

SECTION or ARTICLE = Semantic elements

DIV = Non-semantic element

- The meaning is unchanged in HTML5 from HTML 4.01
- Use to contain information for styling
- Use to structure a page; e.g., site wrapper (container), intro paragraph with different style, or columns (Use with CSS class selector or ID selector)
- Use when enclosed contents are not related to each other semantically and have no generic heading

24. Key Takeaways

- Foundational markup of all webpages
- Two patterns worth learning
- HTML5 semantic elements

I hope this lesson was informative for you, and I’d like to thank you for viewing

25. Bonus

We’ll conclude with an introduction to CSS responsive layout – a technique for supporting multiple screens and user experiences.

CSS fluid layouts for multiple devices

```
/* ~ Fluid Grid Support ~ */  
img, object, embed, video, figure { max-width: 100%; }  
/* ~ Mobile ~ */  
/* ~ Tablet ~ */  
@media only screen and (min-width: 580px) {
```