More on HTML

Note: Go through these notes before heading onto the assignment!
After covering the basics of HTML, now you'll learn about some advanced HTML topics. In these notes, you'll come across some new tags and attributes that are required to broaden the prospect of frontend development.

<DIV> Tag

- The **<div> tag** creates a **block-level division or segment** in an HTML document.
- <div> is a **block element.**
- It also **serves as a container** for **other HTML elements**.
- The <div> section **doesn't require any necessary attributes.**
- When working with CSS, <div> is **frequently used to layout a web page**.

SEMANTIC ELEMENTS

- Semantic HTML is an HTML that, rather than a presentation to the website, provides context. For instance, a <p> tag says the text contains a paragraph.
- Semantic HTML tags provide the content of tags that go beyond the way they look on a page. For example, text inserted under the **<code> tag** is **recognized as a coding language by the browser immediately.**
- Non-semantic examples: <div> and <span> - say nothing about their content.
- Semantic element examples: <h1> through <h6>, <p> and <ul> clearly describe their content.

Now, here is an example of how a page is made with some common semantic tags:

```
<header>
<nav>
<section>
<aside>
<article>
<footer>
```
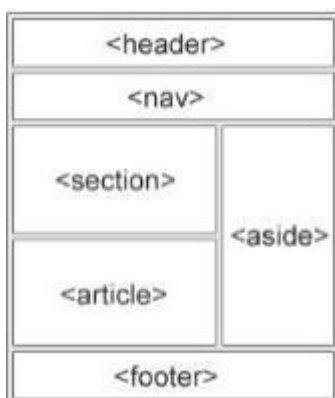
EXTRA:

To know more about semantic elements and the benefit of using them, visit:

_https://developer.mozilla.org/en-US/docs/Glossary/Semantics#Semantics_in_HTML_

INTERNAL LINKS

- Instead of scrolling through long pages, you satisfy your readers by offering page hops as an alternative transport mode throughout your website.
- In principle, page hops are links only, but links to certain sections of the same document (i.e. inner links) are the same <a> feature as all links.
- The anchor tag can refer to any element of the page provided the id attribute of the HTML element should be the same as the id that the anchor tag refers to.

You'll know about id attributes later in CSS, so don't put too much emphasis on them now. For eg., here's how you set up a link:

```
<h2 id="heading">This is the top</h2>
    …….
    …….
<a href="#heading">Go to top</a>
```

Explanation: the **heading** is the **id** of the top heading of this page, and it's used in the **href attribute**. The **id attribute** is a **unique identifier** for an HTML element that **can be used to locate any element.** The symbol **'#'** is used to **refer to an id.**

By clicking the link **_Go to top_**, you will go to the line where the heading is created.

BLOCK VS INLINE ELEMENTS

Block	Inline
They use the **entire webpage width**, effectively blocking any other elements from being placed on the left or the right side.	They **only take up as much width as necessary** to show the element's contents and after that, other elements can be made to match the inline element.

Block elements always start on a new line. But, that's not the case with inline elements.	That's not the case with inline elements.
Examples: **<div>, <p>, <h1>, <h6>, <nav>, etc.**	Examples:**<b>, <span>, <img> etc.**

NOTE: You can also check which elements are block and which are inline by inspecting them using chrome dev tools.

Chrome Dev Tools:

You can learn a lot about your website using **Chrome Developer Tools**, which are **built into your Chrome browser**. You can **modify** your page and *see how various items appear on the platform while troubleshooting and debugging problems.* **HTML**, **CSS**, and **JavaScript can be manipulated** using Chrome Dev Tools.

To **open** Chrome Dev Tools, **press CTRL+SHIFT+C** or **right-click an item on the page, then select Inspect.** We'll go over this topic in greater detail **later** in the course.

TEXT FORMATTING TAGS

- HTML allows us to format text just like we do in any other text-editing application like MS Word. The text appearance of your web page will be formatted using various HTML tags.
- This is likely to spice up the look of your website but too much variation can be unpleasant in-text formatting. HTML also distinguishes specific text elements with a particular significance.
- HTML uses elements like **<b>** and **<i>** for f**ormatted output**, such as **bold** or **italic text** respectively.

For eg:

`<b>Hello!</b>` can be used to bolden the text and
`<i>Hi there!</i>` can be used to italicize the text.

- The span element can also be used for text formatting as follows:
  ```
  <p>
      It is very <span style="font-weight: bold;">
  ```

```
            important</span> to learn HTML, to begin with front
            end development.
    </p>
```

- You can format or style specific words in a sentence. Here only the 'important' word will be made bold. You'll learn more about the style attribute in the next lecture.

EXTRA:

To know more about them, visit:

https://developer.mozilla.org/en-US/docs/Web/HTML/Element#Inline_text_semantics

and

https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Advanced_t

Ext_formatting

SPECIAL CHARACTERS

- In HTML, we have **certain characters reserved for the definition of a tag** that is less than **(<)** and is greater than **(>) signs**, called **angle brackets**. By using them as page symbols, the browser may misinterpret them as markup.
- However, several characters are missing from the keyboard.
- **Special characters**, also known as **HTML entities**, are characters that **have reserved meaning in HTML, or they are usually not present on the keyboard, hence can't be typed.**
  The character entities must be replaced in order to display these unique characters.
- **HTML entity** is a text, which starts with an ampersand **(&)** and ends with a semicolon **(;)**.

Eg., these are some HTML entities with how they will look on the browser:

character entities	diacritical marks	mathematical symbols
(single space)	à(à)	∀(∀) or ∀ (∀)

< (<)	Ô(Ô)	&#8721(∑); or ∑ (∑)
' (')		
© (©)		

- some other entities - ←(←), ♥(♥), ™(™)

EXTRA:

To know more about HTML entities, visit:
https://developer.mozilla.org/en-US/docs/Glossary/Entity

You can get all the available entities list here:
https://dev.w3.org/html5/html-author/charref

https://html.spec.whatwg.org/multipage/named-characters.html#named-character-references (NOTE: USE & BEFORE EVERY CODE TO PRINT THE SPECIAL CHARACTER)

TABLES

- **Tables** are **used to display the tabular data.** A lot of tags are used to do this. All table data is included inside the <table> tags.
- A table (with the <tr>-tag) divides the table into rows and divides the data cells in each row (with the <td>-tag).
- **<tr>** represents the table row
- **<td>** is used for **representing a cell's content.**

The cell can contain text, pictures, lists, types, horizontally defined rules, tables, and so on. Eg:

```
<table border=1>
    <tr>
        <td>Row 1, cell 1</td>
        <td>Row 1, cell 2</td>
        <td>Row 1, cell 3</td>
    </tr>
    <tr>
        <td>Row 2, cell 1</td>
        <td>Row 2, cell 2</td>
        <td>Row 2, cell 3</td>
```

```
    </tr>
    <tr>
        <td>Row 3, cell 1</td>
        <td>Row 3, cell 2</td>
        <td>Row 3, cell 3</td>
    </tr>
    <tr>
        <td>Row 4, cell 1</td>
        <td>Row 4, cell 2</td>
        <td>Row 4, cell 3</td>
    </tr>
</table>
```

The table will be seen something like this:

Row 1, cell 1	Row 1, cell 2	Row 1, cell 3
Row 2, cell 1	Row 2, cell 2	Row 2, cell 3
Row 3, cell 1	Row 3, cell 2	Row 3, cell 3
Row 4, cell 1	Row 4, cell 2	Row 4, cell 3

border Attribute

The border attribute is used to specify the thickness of the borders of the table. The table will be shown without any borders if you do not assign a border attribute. It can be useful at times, but the majority of the time, you want to show your boundaries.

Headings in a Table

HTML have a separate tag to add column names. The <th> tag is used to describe headings in a table.
Eg:

```
<table border=1>
    <tr>
        <th>Column 1</th>
        <th>Column 2</th>
        <th>Column 3</th>
    </tr>
    <tr>
        <td>Row 1, cell 1</td>
        <td>Row 1, cell 2</td>
```

```
        <td>Row 1, cell 3</td>
    </tr>
    <tr>
        <td>Row 2, cell 1</td>
        <td>Row 2, cell 2</td>
        <td>Row 2, cell 3</td>
    </tr>
</table>
```

The table will be seen like this:

Column 1	Column 2	Column 3
Row 1, cell 1	Row 1, cell 2	Row 1, cell 3
Row 2, cell 1	Row 2, cell 2	Row 2, cell 3

<thead>, <tbody>, <tfoot>
- The **<thead>** tag is used in an HTML table to group the content of the header.
- For grouping of body content in an HTML table, the **<tbody>** tag is used.
- The **<tfoot>** tag is used for grouping the content of footers.

These are **semantic tags** that give **meaning to the elements** as well as other useful features.
Additionally, when printing a large table that spans multiple pages, these elements will allow the table header and footer to be printed at the top and bottom of each page.

Eg:

```
<table border="1">
    <thead>
        <tr>
            <th>Column 1</th>
            <th>Column 2</th>
            <th>Column 3 </th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td>Row 1, cell 1</td>
            <td>Row 1, cell 2</td>
            <td>Row 1, cell 3</td>
        </tr>
        <tr>
```

```
          <td>Row 2, cell 1</td>
          <td>Row 2, cell 2</td>
          <td>Row 2, cell 3</td>
      </tr>
   </tbody>
   <tfoot>
      <tr>
          <th>Column 1</th>
          <th>Column 2</th>
          <th>Column 3 </th>
      </tr>
   </tfoot>
</table>
```

The table now looks like this:

Column 1	Column 2	Column 3
Row 1, cell 1	Row 1, cell 2	Row 1, cell 3
Row 2, cell 1	Row 2, cell 2	Row 2, cell 3
Column 1	Column 2	Column 3

caption Tag
- The **<caption>** tag defines a table caption.
- Following the **<table>** tag, the **<caption>** tag must be inserted.

Eg: If you add ***<caption>Table Example</caption>*** just after the <table> tag, the table will now look like this:

Table Example

Column 1	Column 2	Column 3
Row 1, cell 1	Row 1, cell 2	Row 1, cell 3
Row 2, cell 1	Row 2, cell 2	Row 2, cell 3
Column 1	Column 2	Column 3

NOTE: *You can specify only one caption per table.*

colspan and rowspan Attribute
These two attributes, **rowspan** and **colspan**, are used to **handle the layout of the tables.** The **rowspan attribute** is **used to indicate the number of rows occupied by a specific cell.** The **colspan attribute** is **used to specify the number of columns occupied by a specific cell.**

Both are used with the <td> tag and also with the <tr> tag.

Eg: adding attributes colspan and rowspan to the table

```html
<table border="1">
    <thead>
        <tr>
            <th>Column 1</th>
            <th colspan="2">Column 2 and 3 heading</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td>Row 1, cell 1</td>
            <td colspan="2">(Row 1, cell 2) and (Row 1, cell 3)</td>
        </tr>
        <tr>
            <td rowspan="2">(Row 2, cell 1) and (Row 3, cell 2)</td>
            <td>Row 2, cell 2</td>
            <td>Row 2, cell 3</td>
        </tr>
        <tr>
            <td>Row 3, cell 2</td>
            <td>Row 3, cell 3</td>
        </tr>
    </tbody>
</table>
```

The table now looks like:

Column 1	Column 2 and 3	
Row 1, cell 1	(Row 1, cell 2) and (Row 1, cell 3)	
(Row 2, cell 1) and (Row 3, cell 2)	Row 2, cell 2	Row 2, cell 3
	Row 3, cell 2	Row 3, cell 3

You can take a look at this cheat sheet for HTML, which will keep you from memorizing everything:

https://html.com/wp-content/uploads/html-cheat-sheet.pdf

Semantic vs Non-Semantic Elements

Semantic elements clearly define the content they carry. Tags used to enclose them are called **semantic tags**. For example a paragraph element, <p>CONTENT</p> clearly tells that its content is a paragraph.

Some of the semantic elements are:
article, aside, details, figcaption, figure, footer, header, main, mark, nav, section, etc.

Non-semantic elements don't tell any information about the content that they carry. Tags used to enclose these are called **non-semantic tags**. For example a div element, <div>CONTENT</div> can carry any content (headings, paragraphs, links, etc) but the content as a whole has no defined semantics. Hence div is a non-semantic element.

Note: **_Semantically correct HTML helps search engines, screen readers, and other user devices determine the significance and context of web content._**

Section tag and Div tag

- <section>CONTENT</section> is similar to a div element. Its content can be a group of elements put together as shown.
  ```
  <section>
      <h2>This is a footer</h2>
      <a href="www.contactABC.com">Contact Us here</a>
  </section>
  ```
- It provides the same functionality as a div i.e. it groups elements together.

So how will you know when to use which element? **What's the difference between a section and a div?**

Section	Div
It is used to group together elements that are thematically related i.e. the elements share a single theme or these elements serve a collective purpose, or these elements are	The div element gives no information about the interrelation between the elements that it

related.	contains. It does not tell if they are connected to a single theme or not.
It is a generic element to divide our webpage into sections. Usually, it is used with a heading.	It is a generic element to group elements for styling purposes. It can also be used to divide our webpage into divisions, like the section element. But its content won't represent a thematic relation.
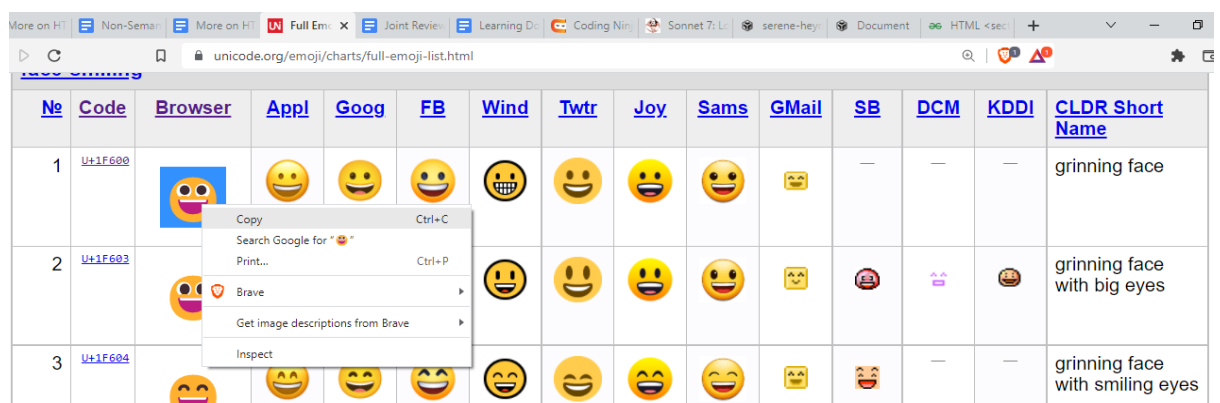
Adding Emojis

Emojis are also special characters. Follow the following steps to add emojis to a website

- Go to this link https://unicode.org/emoji/charts/full-emoji-list.html
- Use ctrl+F to open a search pop-up, then right-click as follows. Select emoji.
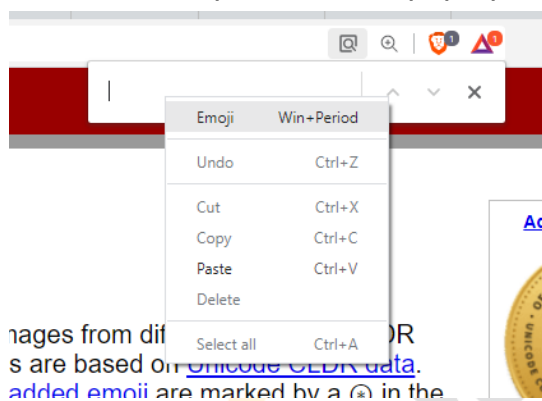
- Then search the emoji you want to use and press enter. It'll redirect you to the emoji on the website you want to use.
- Select the emoji under the 'Browser' column and click on 'Copy' as shown.

- Paste it in your code wherever you wish to add it!

EXTRA - SPECIAL CHARACTERS CODES:

You can get all the available entities list here:
https://dev.w3.org/html5/html-author/charre