BRO Academy

Manual

Lesson 2

HTML5 HyperText Markup Language

– is the standard markup language for describing the structure of documents displayed on the web

Types of Elements by Tag

void elements

- Do not contain text or nested elements

```
<br>, <col>, <embed>, <hr>,
<img>, <input>, <link>,
<meta>, <source>, <track>,
<wbr>
```

replaced elements

- Information for display is taken from their attributes

```
<iframe>, <video>, <embed>
<img src="switch.svg"
alt="light switch"/>
<input type="range"/>
```

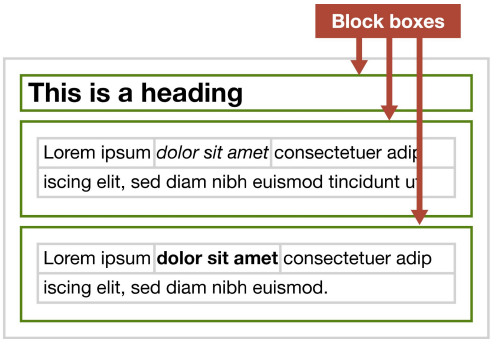
Non-replaced elements

- Have opening and closing tags;
- may contain content and other tags

```
<p>
  This paragraph has some
  <strong> strong </strong>
  content
</p>
```

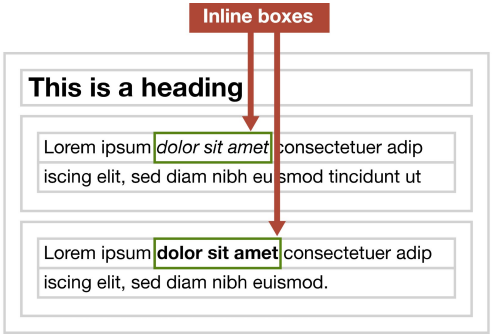
Types of Elements by Type

Block Elements



- Occupy the full width of the page (or the available space of the parent)
- Cannot be nested within inline elements, but can be nested within block elements
- <p> and <h1>-<h6> cannot contain block elements, only inline elements

Inline Elements



- Occupy only as much space as their content
- Can contain only data (text) and other inline elements, but <a> can contain block elements
- Used to highlight a piece of text within a block element

Anatomy of an HTML Document

1. Doctype: Historical artifact
2. Root Element
3. Element for connecting resources not visible on the page
4. Element that sets the document's UTF-8 encoding
5. Element for the page title (displayed in the browser tab or bookmarks)
6. Element for the content of the entire page

Let us see how we can break the code up in different components:

`<html> … </html>`

This tag specifies that the webpage is written in HTML. It appears at the very first and last line of the webpage. It is mainly used to show that the page uses HTML5 – the latest version of the language. Also known as the root element, this tag can be thought of as a parent tag for every other tag used in the page.

`<head> … </head>`

This tag is used to specify meta data about the webpage. It includes the webpage's name, its dependencies (JS and CSS scripts), font usage etc.

`<title> … </title>`

As the name suggests, this tag contains the title/name of the webpage. You can see this in your browser's title bar for every webpage open in the browser. Search engines use this tag to extract the topic of the webpage, which is quite convenient when ranking relevant search results.

`<body> … </body>`

Everything the user sees on a webpage is written inside this tag. It is a container for all the contents of the webpage.

Document summary

Example

```
<html>
	<head>
		<title>My First Website</title>
	</head>
	<body>

	</body>
</html>
```

Document Information

<base/>
Used to specify the base URL of your site, this tag makes linking to internal links on your site cleaner.

<meta/>
This is the meta data tag for the webpage. Can be useful for mentioning the page's author, keywords, original published date etc.

<link/>
This is used to link to scripts external to the webpage. Typically utilized for including stylesheets.

<style> … </style>
The style tag can be used as an alternative to an external style sheet, or complement it. Includes the webpage's appearance information.

<script> … </script>
Used to add code snippets, typically in JavaScript, to make webpage dynamic. It can also be used to just link to an external script.

Example

```
<html>
	<head>
		<meta charset="utf-8">

		<base href="http://myfirstwebsite.com"
			target="_blank" />

		<title>My Beautiful Website</title>

		<link rel="stylesheet"
		href="/css/master.css">

		<script type="text/javascript">
			var dummy = 0;
		</script>
	</head>
	<body>

	</body>
</html>
```

1 Semantics

– Using tags for their intended purpose.

1. Cannot be checked automatically
2. Why use it:
    a. Search engines and SEO
    b. Accessibility through screen readers and reading mode
    c. Code clarity
    d. Developer status

Base

tags

`<h1>...<h6>`	Headings on the page for structuring content
`<p>`	Marks a paragraph of text
`<div>`	**Universal building block without semantic meaning; wrapper for grouping elements. Some websites are 99% made up of this tag, which is not ideal.**
`<span>`	**Helps to style a part of text within a block; no semantics**
`<header>`	Header of a page or block
`<footer>`	Footer of a page or block
`<main>`	Block with the main content of the page
`<section>`	Tag for logical blocks on the page
`<article>`	Tag for creating independent content units, from cards to comments
`<nav>`	Tag that groups navigation elements on a site
`<img>`	Tag for inserting images on the page
`<a>`	Tag for links
`<button>`	Tag for buttons
`<ol>/<ul>/<li>`	Ordered list / Unordered list / List item
`<dl>/<dt>/<dd>`	Three tags for creating a list of terms. Outer wrapper / Term / Term definition
`<table>/<tr>/<td>`	Table / Table row / Cell in a row
` `	Line break in text; should be used sparingly and only if styles cannot achieve the same effect

Document structure

`<h1..h6> … </h1..h6>`
Six different variations of writing a heading. <h1> has the largest font size, while <h6> has the smallest.

`<div> … </div>`
A webpage's content is usually divided into blocks, specified by the div tag.

`<span> … </span>`
This tag injects inline elements, like an image, icon, emoticon without ruining the formatting / styling of the page.

`<p> … </p>`
Plain text is placed inside this tag.

`<br/>`
A line break for webpages. Is used when wanting to write a new line.

`<hr/>`
Similar to the above tag. But in addition to switching to the next line, this tag also draws a horizontal bar

Example

```
<div>
      <h1>Top 5 Greatest Films</h1>
      <p>These are considered the greatest
            <span>reel-icon</span> of all time
      </p>
      <hr>
      <h2>1. The Godfather</h2>
      <p>This 1972 classic stars Marlon Brando
and Al Pacino.</p>
</div>
```

Text formatting

*** – very rare used tags**

`<strong> … </strong>`
Makes text bold. Used to emphasize a point

`<b> … </b>`
Alternative to the above tag, also creates bold text.

`<em> … </em>`
Another emphasis tag, but this displays text in italics.

`<i> … </i>`
Also used to display text in italics, but does not emphasize it like the above tag.

`<cite> … </cite>`
Tag for citing author of a quote.

`<del> … </del>`
Pre-formatted, ‘monospace’ text laid out with whitespace inside the element intact.

`<small> … </small>`
Reduces text size. In HTML5, it often refers to redundant or invalid information.

Example

```
<p>
      <strong>Bold text</strong> Regular text
      <em>some words in italics</em> regular
text once again.
</p>
<pre>
      Some pre-formatted text
</pre>

<p>A code snippet:
      <code>some code</code>
</p>
```

Text formatting

<ins> … </ins>
Denotes text that has been inserted into the webpage.

<blockquote> … </blockquote>
Quotes often go into this tag. Is used in tandem with the <cite> tag.

<q> … </q>
Similar to the above tag, but for shorter quotes.

<abbr> … </abbr>
Denotes abbreviations, along with the full forms.

<address> … </address>
Tag for specifying author's contact details.

<dfn> … </dfn>
Tag dedicated for definitions.

<code> … </code>
This is used to display code snippets within a paragraph.

Example

```
<blockquote>
      Anyone who has never made a mistake has
never tried anything new.<cite>- Albert
Einstein</cite>
</blockquote>

<p>A code snippet: <code>some code</code></p>
```

Superscript & Subscript

<sup>

The <sup> element is used to contain characters that should be superscript such as the suffixes of dates or mathematical concepts like raising a number to a power such as 22.

```
<p>On the 4<sup>th</sup> of September you
will learn about E=MC<sup>2</sup></p>

<p>The amount of CO<sub>2</sub> in the
atmosphere grew by 2ppm in 2009 <sub>1</sub>
</p>
```

On the 4^{th} of September you will learn about $E=MC^2$

The amount of CO_2 in the atmosphere grew by 2ppm in 2009 $_1$

<sub>

The <sub> element is used to contain characters that should be subscript. It is commonly used with foot notes or chemical formulas such as H20.

Links

`<a href="catalog.html"> … </a>`
Anchor tag. Primarily used for including hyperlinks.

`<a href="mailto:"> … </a>`
Tag dedicated to sending emails.

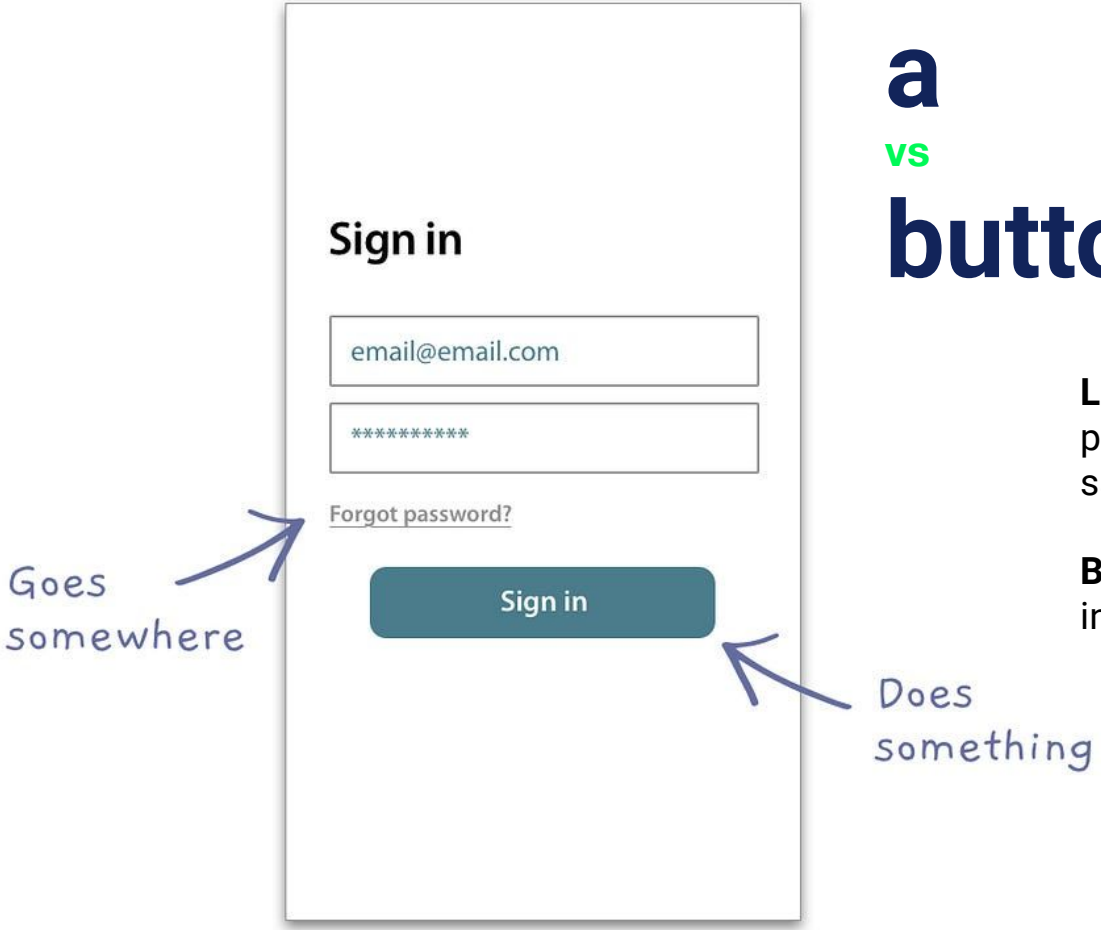
`<a href="tel://###-###"> … </a>`
Anchor tag for mentioning contact numbers. As the numbers are clickable, this can be particularly beneficial for mobile users.

`<a name="name"> … </a>`
This tag can be used to quickly navigate to a different part of the webpage.

`<a href="#name"> … </a>`
A variation of the above tag, this is only meant to navigate to a div section of the webpage.

Example

```
<a href="mailto:test@gmail.com">Text to our email</a>
```



a

vs

button

Link - we need to go to another page or to another place on the same page

Button – something should happen in the same place on the page

img, figure & figcaption

<img>
A tag to display images in the webpage.

<figure> can be used to contain any content that is referenced from the main flow of an article (not just images).

It is important to note that the article should still make sense if the content of the <figure> element were moved (to another part of the page, or even to a different page altogether).

For this reason, it should only be used when the content simply references the element (and not for something that is absolutely integral to the flow of a page).

Examples of usage include:

- Images
- Videos
- Graphs
- Diagrams
- Code samples
- Text that supports the main body of an article

The <figure> element should also contain a **<figcaption>** element which provides a text description for the content of the <figure> element. In this example, you can see a <figure> has been added inside the <article> element.

Example

```
<img
      src="planets.gif"
      width="145"
      height="126"
      alt="Planets"
>

<figure>
      <img src="images/bok-choi.jpg" alt="Bok
Choi">
      <figcaption>Bok Choi</figcaption>
</figure>
```

Lists

`<ol> … </ol>`
Tag for ordered or numbered list of items.

`<ul> … </ul>`
Contrary to the above tag, used for unordered list of items.

`<li> … </li>`
Individual item as part of a list.

`<dl> … </dl>`
Tag for list of items with definitions.

`<dt> … </dt>`
The definition of a single term inline with body content.

`<dd> … </dd>`
The description for the defined term.

Example

```
<ol>
	<li>Monday</li>
	<li>Tuesday</li>
	<li>Wednesday</li>
</ol>

<ul>
	<li>France</li>
	<li>Germany</li>
	<li>Italy</li>
</ul>

<dl>
	<dt>Toyota</dt>
	<dd>Japanese car brand</dd>

	<dt>Armani</dt>
	<dd>Italian fashion brand</dd>
</dl>
```

Tables

<table> … </table>
Marks a table in a webpage.

<caption> … </caption>
Description of the table is placed inside this tag.

<thead> … </thead>
Specifies information pertaining to specific columns of the table.

<tbody> … </tbody>
The body of a table, where the data is held.

<tfoot> … </tfoot>
Determines the footer of the table.

<tr> … </tr>
Denotes a single row in a table.

<th> … </th>
The value of a heading of a table's column.

<td> … </td>
A single cell of a table. Contains the actualcvalue/data.

<colgroup> … </colgroup>
Used for grouping columns together.

<col>
Denotes a column inside a table.

Example

```
<table>
	<colgroup>
		<col span="2">
		<col>
	</colgroup>
	<tr>
		<th>Name</th>
		<th>Major</th>
		<th>GPA</th>
	</tr>
	<tr>
		<td>Bob</td>
		<td>Law</td>
		<td>3.55</td>
	</tr>
	<tr>
		<td>Alice</td>
		<td>Medicine</td>
		<td>3.61</td>
	</tr>
</table>
```

Headers & Footers

The `<header>` and `<footer>` elements can be used for:

●● The main header or footer that appears at the top or bottom of every page on the site.

●● A header or footer for an individual <article> or <section> within the page.

In this example, the <header> element used to contain the site name and the main navigation.
The <footer> element contains copyright information, along with links to the privacy policy and terms and conditions.
Each individual <article> and <section> element can also have its own <header> and <footer> elements to hold the header or footer information for that section within the page.

For example, on a page with several blog posts, each individual post can be thought of as a separate section. The <header> element can therefore be used to contain the title and date of each individual post, and the <footer> might contain links to share the article on social networking sites.

Please note that all of the code shown in this chapter is referenced in one HTML

Example

```
<header>
	<h1>Yoko's Kitchen</h1>
	<nav>
	<ul>
		<li><a href="#" class="current">
	home</a></li>
		<li><a href="#">classes</a></li>
		<li><a href="#">catering</a></li>
		<li><a href="#">about</a></li>
		<li><a href="#">contact</a></li>
	</ul>
	</nav>
</header>

<footer>
	&copy; 2011 Yoko's Kitchen
</footer>
```

main

<main>…</main>

Marks the main content of the webpage.

This content is unique to the individual page, and should not appear elsewhere on the site. Repeating content like headers, footers, navigation, logos, etc., is placed outside the element.

The <main> element should only ever be used at most once on a single page.

The <main> element must not be included as a descendant of an article, aside, footer, header or nav element.

Example

```
<body>
	<header>
		<nav>...</nav>
	</header>
	<main>
		<h1>Individual Blog Post</h1>
		<p>An introduction for the
	post.</p>
		<article>
			<h2>References</h2>
			<p>...</p>
		</article>
		<article>
			<h2>Comments</h2>
			…
		</article>
	</main>
	<footer>...</footer>
</body>
```

article

<article>…</article>

Denotes an article.

The <article> element acts as a container for any section of a page that could stand alone and potentially be syndicated.

This could be an individual article or blog entry, a comment or forum post, or any other independent piece of content.

If a page contains several articles (or even summaries of several articles), then each individual article would live inside its own <article> element.

The <article> elements can even be nested inside each other. For example, a blog post might live inside one <article> element and each comment on the article could live inside its own child <article> element.

Example

```
<article>
	<h2>Japanese Vegetarian</h2>
	<h3>Five week course in London</h3>
	<p>A five week introduction to traditional
Japanese vegetarian meals, teaching you a
selection of rice and noodle dishes.</p>
</article>

<article>
	<figure>
		<img src="images/teriyaki.jpg"
	alt="Teriyaki sauce">
		<figcaption>Teriyaki
	Sauce</figcaption>
	</figure>
	<h2>Sauces Masterclass</h2>
	<h3>One day workshop</h3>
	<p>An intensive one-day course looking at
how to create the most delicious sauces for use
in a range of Japanese cookery.</p>
</article>
```

aside

<aside> … </aside>

Denotes content displayed in a sidebar of the webpage. The <aside> element has two purposes, depending on whether it is inside an <article> element or not.

When the <aside> element is used inside an <article> element, it should contain information that is related to the article but not essential to its overall meaning. For example, a pullquote or glossary might be considered as an aside to the article it relates to.

When the <aside> element is used outside of an <article> element, it acts as a container for content that is related to the entire page. For example, it might contain links to other sections of the site, a list of recent posts, a search box, or recent tweets by the author.

Example

```
<aside>
	<section class="popular-recipes">
		<h2>Popular Recipes</h2>
		<a href="#">Yakitori (grilled
	chicken)</a>
		<a href="#">Tsukune (minced chicken
	patties)</a>
		<a href="#">Okonomiyaki (savory
	pancakes)</a>
		<a href="#">Mizutaki (chicken
	stew)</a>
	</section>
	<section class="contact-details">
		<h2>Contact</h2>
		<p>Yoko's Kitchen <br>
		27 Redchurch Street <br>
		Shoreditch <br>
		London E2 7DP</p>
	</section>
</aside>
```

section

`<section>…</section>`

Specifies a particular section in the webpage.
The <section> element groups related content together, and typically each section would have its own heading.

For example, on a homepage there may be several <section> elements to contain different sections of the page, such as latest news, top products, and newsletter signup.

Because the <section> element groups related items together, it may contain several distinct <article> elements that have a common theme or purpose.

Alternatively, if you have a page with a long article, the <section> element can be used to split the article up into separate sections.

The <section> element should not be used as a wrapper for the entire page (unless the page only contains one distinct piece of content). If you want a containing element for the entire page, that job is still best left to the element.

Example

```
<section class="popular-recipes">
       <h2>Popular Recipes</h2>
       <a href="#">Yakitori (grilled chicken)</a>
       <a href="#">Tsukune (minced chicken
patties)</a>
       <a href="#">Okonomiyaki (savory
pancakes)</a>
       <a href="#">Mizutaki (chicken stew)</a>
</section>

<section class="contact-details">
       <h2>Contact</h2>
       <p>Yoko's Kitchen <br>
       27 Redchurch Street <br>
       Shoreditch <br>
       London E2 7DP</p>
</section>
```

nav

<nav>…</nav>

Navigation links for the user in a webpage.

The <nav> element is used to contain the major navigational blocks on the site such as the primary site navigation.

Going back to our blog example, if you wanted to finish an article with links to related blog posts, these would not be counted as major navigational blocks and therefore should not sit inside a <nav> element.

At the time of writing, some of the developers that were already using HTML5 decided to use the <nav> element for the links that appear at the bottom of every page (links to things like privacy policy, terms and conditions and accessibility information).
Whether this will be widely adopted is still yet to be seen.

Example

```
<nav>
	<ul>
		<li><a href="#" class="current">
	home</a></li>
		<li><a href="#">classes</a></li>
		<li><a href="#">catering</a></li>
		<li><a href="#">about</a></li>
		<li><a href="#">contact</a></li>
	</ul>
</nav>
```

Semantic tags

`<dialog>…</dialog>`
Used to create a dialog box.

`<mark>…</mark>`
Used to highlight a particular portion of the text.

`<rt>…</rt>`
Displays East Asian typography character details.

`<time>…</time>`
Tag for formatting date and time.

`<wbr>`
A line-break within the content.

`<details> … </details>`
Used for additional information. User has the option to view or hide this.

`<summary> … </summary>`
Used as a heading for the above tag. Is always visible to the user.

Example

```
<dialog>
	<button autofocus>Close</button>
	<p>This modal dialog has a groovy
backdrop!</p>
</dialog>

<button>Show the dialog</button>

<time datetime="2016-03-13">13th March
2016</time>

<details>
	<summary>Details</summary>
	Something small enough to escape casual
notice.
</details>
```

HTML

Attributes

Attributes provide additional information about an element that you do not want to display in the content.

An attribute must always have:

- A space between it and the element name (or the previous attribute if the element already has one or more attributes)
- Attribute name, followed by an equals sign
- Attribute value, enclosed in quotes
- The order of attributes does not matter

Types of

Attributes

global
Can be applied to any element

required
Without them, the element is invalid

boolean
Do not require a value

custom
Can be created for later use in JS with the **dataset[name]** property

data-*your-name*

```
<p class="box error">Setting 'box' class to the element.</p>
```

```
<img src="image.jpg" alt="this is an image">
```

```
<button disabled>You can’t click on this button</button>
```

```
<button data-machine-learning="workshop">Click</button>
```

Most Used

Attributes

href	<a>	URL for the link
target="_blank"	<a>	Opens the link in a new window
class	all	Used with CSS to style elements
id	all	The value must be unique on the page; used for linking to blocks on the same page
alt / src	<img>	Alternative text if the image cannot be displayed / Path to the image, required for the image to display
width/height	<img>, <iframe>	Sets default width/height for images, which can be overridden in CSS
style	all	Defines styles that override previously set CSS file styles

Attribute

class

- Class name can be arbitrary
- A good class should answer the question, **"What does this element do?"**
- Do not name classes **by content, style, or tag**, as they may change
- **Avoid using numbers** in class names; the order of blocks may change
- Use only **English words**, **no transliteration**
- Written in **lowercase**
- No spaces in class names; use **hyphens** for compound class names
- Preferably no **more than three words** in a class name
- An element can have many classes

Comments in

HTML

```
<div>
    <!-- HTML Comment
     This comment will not be seen in the browser window but
     is sent to the browser which can add to the delay of a
     page load. -->
```

- Invisible to the user
- Explanations in the code for yourself or colleagues
- Hide blocks of code from the page without deleting them from the file
- Cannot nest comments within each other
- To quickly comment/uncomment a line or block of code, use **Ctrl + /** or **Cmd + /** in most editors
- Often used to mark the beginning of a large piece of code and explain its purpose
- Visible in the browser's developer tools, so write only what anyone can see

2 hierarchy of headings

– Creating a hierarchy of headings, which should look like a book's table of contents

1. Core content
2. Why think about it:
    a. Reasons similar to semantics
3. If the page lacks visual headings, add hidden ones

Contents


```
<h1>InDesign book</h1>
  <h2>Basics of InDesign</h2>
    <h3>History of InDesign</h3>
      <h4>InDesign in Ancient Greek</h4>
      <h4>The Birth of InDesign</h4>
      <h4>The Death of InDesign</h4>
    <h3>The best publishing software</h3>
      <h4>What about Quark</h4>
      <h4>What about Office</h4>
    <h3>Whu use InDesign?</h3>
      <h4>I love InDesign</h4>
      <h4>I also love Photoshop</h4>
    <h3>Who Create InDesign?</h3>
  <h2>Typography</h2>
    <h3>Defenition of Typography</h3>
      <h4>Typography is Text in Visual</h4>
      <h4>Typography is Art</h4>
    <h3>Dealing with Typography</h3>
      <h4>Mr.Typography Himself</h4>
```

3 Validity

– Conformity to standards (specification)

The World Wide Web Consortium(W3C) is an international community that develops open standards for the Internet.

1. Correct order of opening and closing tags

2. Presence of closing tags

3. All required attributes

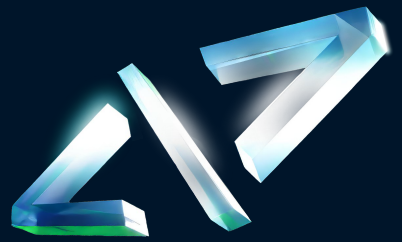
4. Proper use of tags within other tags

1. Correct order of opening and closing tags

 Lorem ipsum <a> dolor set ❌

 Lorem ipsum <a> dolor set ✅

 Lorem ipsum <a> dolor set ✅



2. Presence of closing tags

```
<li> Lorem ipsum dolor set
<li> Lorem ipsum dolor set
<input type="text"></input>
```

❌

```
<li> Lorem ipsum dolor set </li>
<li> Lorem ipsum dolor set </li>
```

✅

```
<img src=”image.png” alt=”icon check”>
<input type="text">
```

✅

3. Availability of all required attributes

```
<img src=”image.png”>
```
❌

```
<img src=”image.png” alt=”icon check”>
```
✅

```
<a> Lorem ipsum </a>
```
✅

```
<div id="value01"></div>
<div id="value01"></div>
```
❌

4. Proper use of tags within other tags

```
<span> Lorem ipsum <div> dolor sit </div> </span>
```

❌

```
<div> Lorem ipsum <span> dolor sit </span> </div>
```

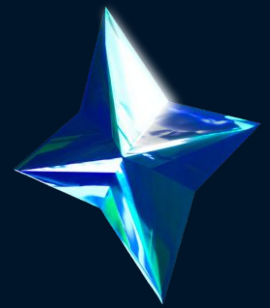
✅

```
<ul> Lorem ipsum
      <p> dolor sit </p>
</ul>
```

❌

```
<ul> Lorem ipsum
      <li> <p> dolor sit </p> </li>
</ul>
```

✅



4 Accessibility

– Ensures the website is usable for people with additional needs

1. Often overlooked

2. If you follow the other three rules, accessibility will be achieved automatically

3. Can be checked using Chrome's audits for accessibility, which also provides fixes

4. Ensures people can use your site with a keyboard and screen reader

structure of

Emmet

`!`	Basic site structure	`<!DOCTYPE html>`
		`<html lang="en">`
		`...`
		`</html>`

`lorem`	First 5 words of dummy text	`Lorem ipsum dolor sit amet.`
`#example`	id	`<div id="page"></div>`
`.example`	class	`<div class="logo"></div>`
`p*2`	Number of elements	`<p></p>` `<p></p>`

`ul>li`	Nesting	`<ul>`
		`  <li></li>`
		`</ul>`

`ul>li.item$*5`	Counter	`<li class="item1"></li>`
		`...`
		`<li class="item5"></li>`

`p>span+em^bq = (^+)`	Move up a level	`<p><span></span><em></em></p>` `<blockquote></blockquote>`

`div>(header>p)+footer`	Grouping	`<div>`
		`  <header><p></p></header>`
		`  <footer></footer>`
		`</div>`

`a[href="index.html"]`	Attribute	`<a href="index.html"></a>`
`a{Click me}`	Text	`<a>Click me</a>`

Rules

Emmet

Class without a tag = ** ** with class	`.wrap>.content = div.wrap>div.content`
If an element has nesting rules, they apply automatically	`ul>.item*3 = ul>li.item*3`
• `li` for `ul` and `ol`	`ol>.item = ol>li.item`
• `tr` for `table`, `tbody`, `thead` and `tfoot`	`table>.row = table>tr.row`
• `td` for `tr`	`table>tr>.cell = table>tr>td.cell`
• `option` for `select` and `optgroup`	`select>.select-item*5 = select>option.select-item*5`
Follow nesting rules by element types	`em>.info = em>span.info` ~~`div`~~