기초 웹

22 Winter CNU 기초 스터디

21 남정연
21 박준서

tag

문단을 나타내는 태그

-> 대부분의 문서는 문단 단위로 구성되어 있음!

<h1~6> tag

<h1~6></ h1~6>

제목을 나타내는 태그

숫자가 클 수록 제목의 크기가 작아짐

Example

```
<h1>The Crushing Bore</h1>

<p>By Chris Mills</p>

<h2>Chapter 1: The dark night</h2>

<p>It was a dark night. Somewhere, an owl hooted. The rain lashed down on the ...</p>

<h2>Chapter 2: The eternal silence</h2>

<p>Our protagonist could not so much as a whisper out of the shadowy figure ...</p>

<h3>The specter speaks</h3>

<p>Several more hours had passed, when all of a sudden the specter sat bolt upright and exclaimed, "Please have mercy on my soul!"</p>
```

 tag

```
<ul>
    <li>milk</li>
    <li>eggs</li>
    <li>bread</li>
    <li>hummus</li>
</ul>
```

- milk
- eggs
- bread
- hummus

 tag

```
<ol>
    <li>Drive to the end of the road</li>
    <li>Turn right</li>
    <li>Go straight across the first two roundabouts</li>
    <li>Turn left at the third roundabout</li>
    <li>The school is on your right, 300 meters up the road</li>
</ol>
```

1. Drive to the end of the road
2. Turn right
3. Go straight across the first two roundabouts
4. Turn left at the third roundabout
5. The school is on your right, 300 meters up the road

Emphasis

텍스트가 강조되어야 함을 나타냄
(Emphasis)

Browser Default Style : *italic*

```
<p>I am <em>glad</em> you weren't <em>late</em>.</p>
```

I am *glad* you weren't *late*.

Strong importance

텍스트가 중요함을 나타냄
(strong importance)

Browser Default Style : **bold**

```
<p>This liquid is <strong>highly toxic</strong>.</p>

<p>I am counting on you. <strong>Do not</strong> be late!</p>
```

This liquid is **highly toxic**.

I am counting on you. **Do not** be late!

Nesting Tag

```
<p>This liquid is <strong>highly toxic</strong> —
if you drink it, <strong>you may <em>die</em></strong>.</p>
```

This liquid is **highly toxic** — if you drink it, **you may *die***.

Semantics vs Presentational Elements

Semantics

어떠한 “의미”를 나타내는 태그

Ex)
<p>
<h1~6>
,

Presentational Elements

특정 서식을 적용하기 위한 태그(HTML5이전)
HTML5에서는 Semantic하게 재정의됨

Ex)
<i>
<b>
<u>

Semantics vs Presentational Elements

- `<i>` is used to convey a meaning traditionally conveyed by italic: foreign words, taxonomic designation, technical terms, a thought...
- `<b>` is used to convey a meaning traditionally conveyed by bold: key words, product names, lead sentence...
- `<u>` is used to convey a meaning traditionally conveyed by underline: proper name, misspelling...

Hyperlink - <a> tag

Syntax

```
<a target="_blank|_self|_parent|_top|framename">
```

Attribute Values

Value	Description
_blank	Opens the linked document in a new window or tab
_self	Opens the linked document in the same frame as it was clicked (this is default)
_parent	Opens the linked document in the parent frame
_top	Opens the linked document in the full body of the window
framename	Opens the linked document in the named iframe

Hyperlink - <a> tag

```
<p>I'm creating a link to
    <a href="https://www.mozilla.org/en-US/"
       title="The best place to find more information about Mozilla's
              mission and how to contribute">the Mozilla homepage</a>.
</p>
```

I'm creating a link to the Mozilla homepage.

The best place to find more information about Mozilla's mission and how to contribute

Hyperlink - block level link

Block level links

As mentioned before, almost any content can be made into a link, even block-level elements. If you have an image you want to make into a link, use the `<a>` element and reference the image file with the `<img>` element.

```
<a href="https://www.mozilla.org/en-US/">
  <img src="mozilla-image.png" alt="mozilla logo that links to the mozilla homepage">
</a>
```

URLs and Paths

- **Same directory**: If you wanted to include a hyperlink inside `index.html` (the top level `index.html`) pointing to `contacts.html`, you would specify the filename that you want to link to, because it's in the same directory as the current file. The URL you would use is `contacts.html`:

```
<p>Want to contact a specific staff member?
Find details on our <a href="contacts.html">contacts page</a>.</p>
```

- **Moving down into subdirectories**: If you wanted to include a hyperlink inside `index.html` (the top level `index.html`) pointing to `projects/index.html`, you would need to go down into the `projects` directory before indicating the file you want to link to. This is done by specifying the directory's name, then a forward slash, then the name of the file. The URL you would use is `projects/index.html`:

```
<p>Visit my <a href="projects/index.html">project homepage</a>.</p>
```

- **Moving back up into parent directories**: If you wanted to include a hyperlink inside `projects/index.html` pointing to `pdfs/project-brief.pdf`, you'd have to go up a directory level, then back down into the `pdf` directory. To go up a directory, use two dots — `..` — so the URL you would use is `../pdfs/project-brief.pdf`:

```
<p>A link to my <a href="../pdfs/project-brief.pdf">project brief</a>.</p>
```

Document Fragment

```
<h2 id="Mailing_address">Mailing address</h2>
```

Then to link to that specific `id`, you'd include it at the end of the URL, preceded by a hash/pound symbol (`#`), for example:

```
<p>Want to write us a letter? Use our <a href="contacts.html#Mailing_address">mailing address</a>.</p>
```

You can even use the document fragment reference on its own to link to *another part of the current document*:

```
<p>The <a href="#Mailing_address">company mailing address</a> can be found at the bottom of this page.</p>
```

Absolute and Relative URLs

Absolute URL

Points to a location defined by its absolute location on the web, including protocol and domain name

프로토콜과 도메인 이름을 포함함
https://www.example.com/projects/index.html

Relative URL

Points to a location that is relative to the file you are linking from.

현재 폴더 : **./**
상위 폴더 : **../**

현재 폴더를 기준으로 브라우저가 탐색함
https://www.example.com/a/b/c.html 에서의
../../d.html 는
https://www.example.com/a/d.html

Linking Resources

```
<!-- Download File -->
<p><a href="https://www.example.com/large-report.pdf">
    Download the sales report (PDF, 10MB)
</a></p>

<!-- Download File with Specific Name(Default Filename) -->
<a href="https://download.mozilla.org/?product=firefox-latest-ssl&os=win64&lang=en-US"
download="firefox-latest-64bit-installer.exe">
    Download Latest Firefox for Windows (64-bit) (English, US)
</a>

<!-- Video Stream -->
<p><a href="https://www.example.com/video-stream/" target="_blank">
    Watch the video (stream opens in separate tab, HD quality)
</a></p>

<!-- Open Flash -->
<p><a href="https://www.example.com/car-game">
    Play the car game (requires Flash)
</a></p>
```

Special Links

Linking to an email address

To create links that open in the user's email program to let them send a new message, use the `mailto:` scheme:

```
<a href="mailto:nowhere@mozilla.org">Send email to nowhere</a>
```

For details about `mailto:` URLs, such as including a subject or body, see Email links or RFC 6068.

Linking to telephone numbers

```
<a href="tel:+49.157.0156">+49 157 0156</a>
<a href="tel:+1(555)5309">(555) 5309</a>
```

Description Lists

```
<dl>
    <dt>aside</dt>
    <dd>In drama, where a character shares a comment only with the audience for humorous or dramatic effect. This is usually a feeling, thought, or piece of additional background information.</dd>
    <dd>In writing, a section of content that is related to the current topic, but doesn't fit directly into the main flow of content so is presented nearby (often in a box off to the side.)</dd>
</dl>
```

aside

In drama, where a character shares a comment only with the audience for humorous or dramatic effect. This is usually a feeling, thought, or piece of additional background information.

In writing, a section of content that is related to the current topic, but doesn't fit directly into the main flow of content so is presented nearby (often in a box off to the side.)

Quotations

```
<p>Here below is a blockquote...</p>
<blockquote cite="https://developer.mozilla.org/en-US/docs/Web/HTML/Element/blockquote">
    <p>The <strong>HTML <code>&lt;blockquote&gt;</code> Element</strong> (or <em>HTML Block
    Quotation Element</em>) indicates that the enclosed text is an extended quotation.</p>
</blockquote>
```

Here below is a blockquote...

> The **HTML `<blockquote>` Element** (or *HTML Block Quotation Element*) indicates that the enclosed text is an extended quotation.

Citations

```
<p>According to the <a href="/en-US/docs/Web/HTML/Element/blockquote">
<cite>MDN blockquote page</cite></a>:
</p>

<blockquote cite="https://developer.mozilla.org/en-US/docs/Web/HTML/Element/blockquote">
    <p>The <strong>HTML <code>&lt;blockquote&gt;</code> Element</strong> (or <em>HTML Block
    Quotation Element</em>) indicates that the enclosed text is an extended quotation.</p>
</blockquote>

<p>The quote element — <code>&lt;q&gt;</code> — is <q cite="https://developer.mozilla.org/en-US/docs/Web/HTML/Element/q">intended
for short quotations that don't require paragraph breaks.</q> -- <a href="/en-US/docs/Web/HTML/Element/q">
<cite>MDN q page</cite></a>.</p>
```

According to the *MDN blockquote page*:

> The **HTML `<blockquote>` Element** (or *HTML Block Quotation Element*) indicates that the enclosed text is an extended quotation.

The quote element — `<q>` — is "intended for short quotations that don't require paragraph breaks." -- *MDN q page*.

Abbreviations

```
<p>We use <abbr title="Hypertext Markup Language">HTML</abbr> to structure our web documents.</p>

<p>I think <abbr title="Reverend">Rev.</abbr> Green did it in the kitchen with the chainsaw.</p>
```

We use HTML to structure our web documents.

I think Rev. Gre… with the chainsaw.

Hypertext Markup Language

Superscript and Subscript

```
<p>My birthday is on the 25<sup>th</sup> of May 2001.</p>
<p>Caffeine's chemical formula is C<sub>8</sub>H<sub>10</sub>N<sub>4</sub>O<sub>2</sub>.</p>
<p>If x<sup>2</sup> is 9, x must equal 3 or -3.</p>
```

My birthday is on the 25^{th} of May 2001.

Caffeine's chemical formula is $C_8H_{10}N_4O_2$.

If x^2 is 9, x must equal 3 or -3.

<time> tag

```
<!-- Standard simple date -->
<time datetime="2016-01-20">20 January 2016</time>
<!-- Just year and month -->
<time datetime="2016-01">January 2016</time>
<!-- Just month and day -->
<time datetime="01-20">20 January</time>
<!-- Just time, hours and minutes -->
<time datetime="19:30">19:30</time>
<!-- You can do seconds and milliseconds too! -->
<time datetime="19:30:01.856">19:30:01.856</time>
<!-- Date and time -->
<time datetime="2016-01-20T19:30">7.30pm, 20 January 2016</time>
<!-- Date and time with timezone offset -->
<time datetime="2016-01-20T19:30+01:00">7.30pm, 20 January 2016 is 8.30pm in France</time>
<!-- Calling out a specific week number -->
<time datetime="2016-W04">The fourth week of 2016</time>
```

Line Breaks

```
<p>There once was a man named O'Dell<br>
Who loved to write HTML<br>
But his structure was bad, his semantics were sad<br>
and his markup didn't read very well.</p>
```

There once was a man named O'Dell
Who loved to write HTML
But his structure was bad, his semantics were sad
and his markup didn't read very well.

Horizontal Rules

```
<p>Ron was backed into a corner by the marauding
   netherbeasts. Scared, but determined to protect his friends, he raised his
   wand and prepared to do battle, hoping that his distress call had made it through.</p>
<hr>
<p>Meanwhile, Harry was sitting at home, staring at his royalty statement
  and pondering when the next spin off series would come out, when an enchanted
  distress letter flew through his window and landed in his lap. He read it
  hazily and sighed; "better get back to work then", he mused.</p>
```

Ron was backed into a corner by the marauding netherbeasts. Scared, but determined to protect his friends, he raised his wand and prepared to do battle, hoping that his distress call had made it through.

Meanwhile, Harry was sitting at home, staring at his royalty statement and pondering when the next spin off series would come out, when an enchanted distress letter flew through his window and landed in his lap. He read it hazily and sighed; "better get back to work then", he mused.

Basic Sections of a Document

header:

Usually a big strip across the top with a big heading, logo, and perhaps a tagline. This usually stays the same from one webpage to another.

navigation bar:

Links to the site's main sections; usually represented by menu buttons, links, or tabs. Like the header, this content usually remains consistent from one webpage to another — having inconsistent navigation on your website will just lead to confused, frustrated users. Many web designers consider the navigation bar to be part of the header rather than an individual component, but that's not a requirement; in fact, some also argue that having the two separate is better for accessibility, as screen readers can read the two features better if they are separate.

Basic Sections of a Document

main content:

A big area in the center that contains most of the unique content of a given webpage, for example, the video you want to watch, or the main story you're reading, or the map you want to view, or the news headlines, etc. This is the one part of the website that definitely will vary from page to page!

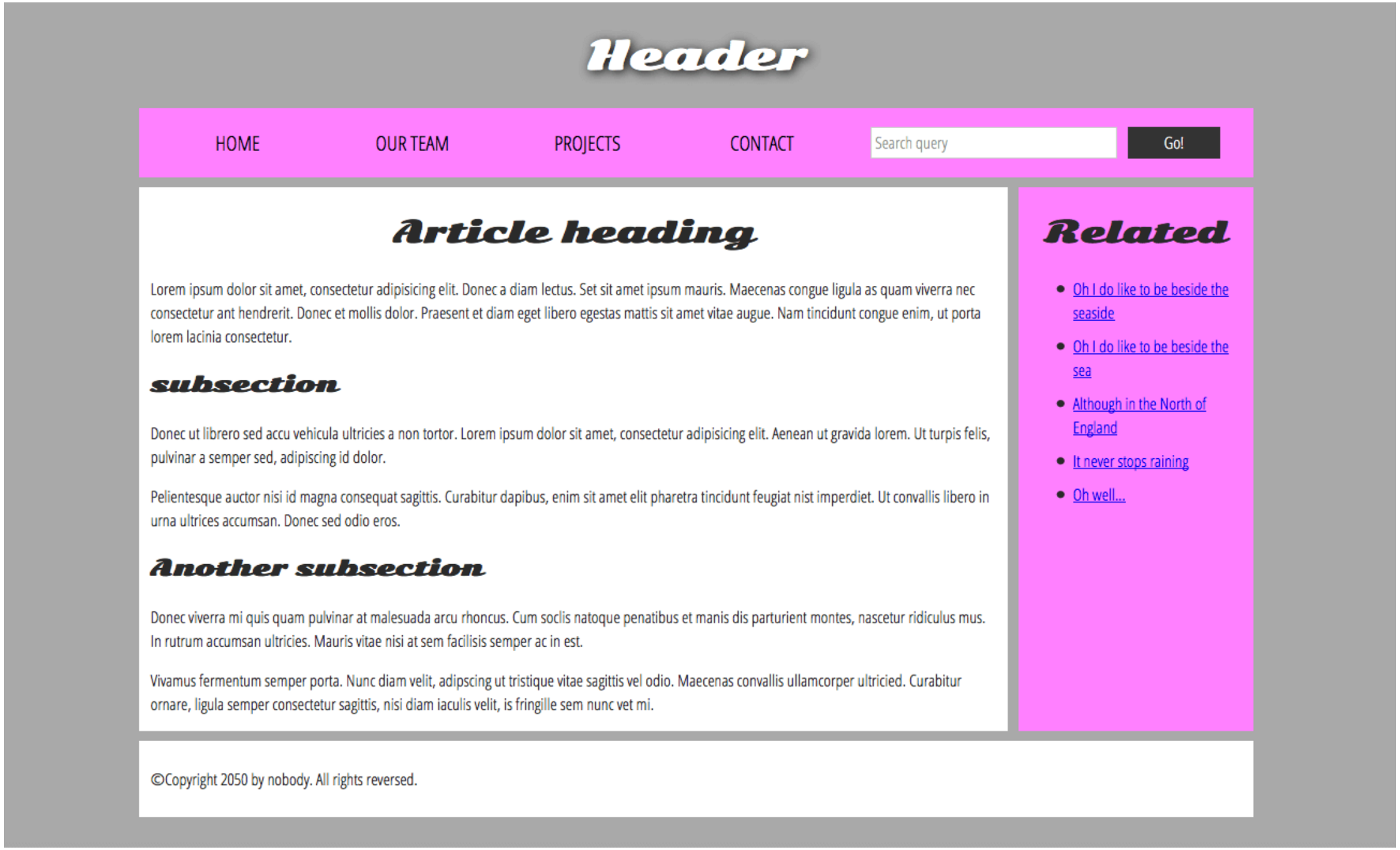
sidebar:

Some peripheral info, links, quotes, ads, etc. Usually, this is contextual to what is contained in the main content (for example on a news article page, the sidebar might contain the author's bio, or links to related articles) but there are also cases where you'll find some recurring elements like a secondary navigation system.

footer:

A strip across the bottom of the page that generally contains fine print, copyright notices, or contact info. It's a place to put common information (like the header) but usually, that information is not critical or secondary to the website itself. The footer is also sometimes used for SEO purposes, by providing links for quick access to popular content.

Basic Sections of a Document



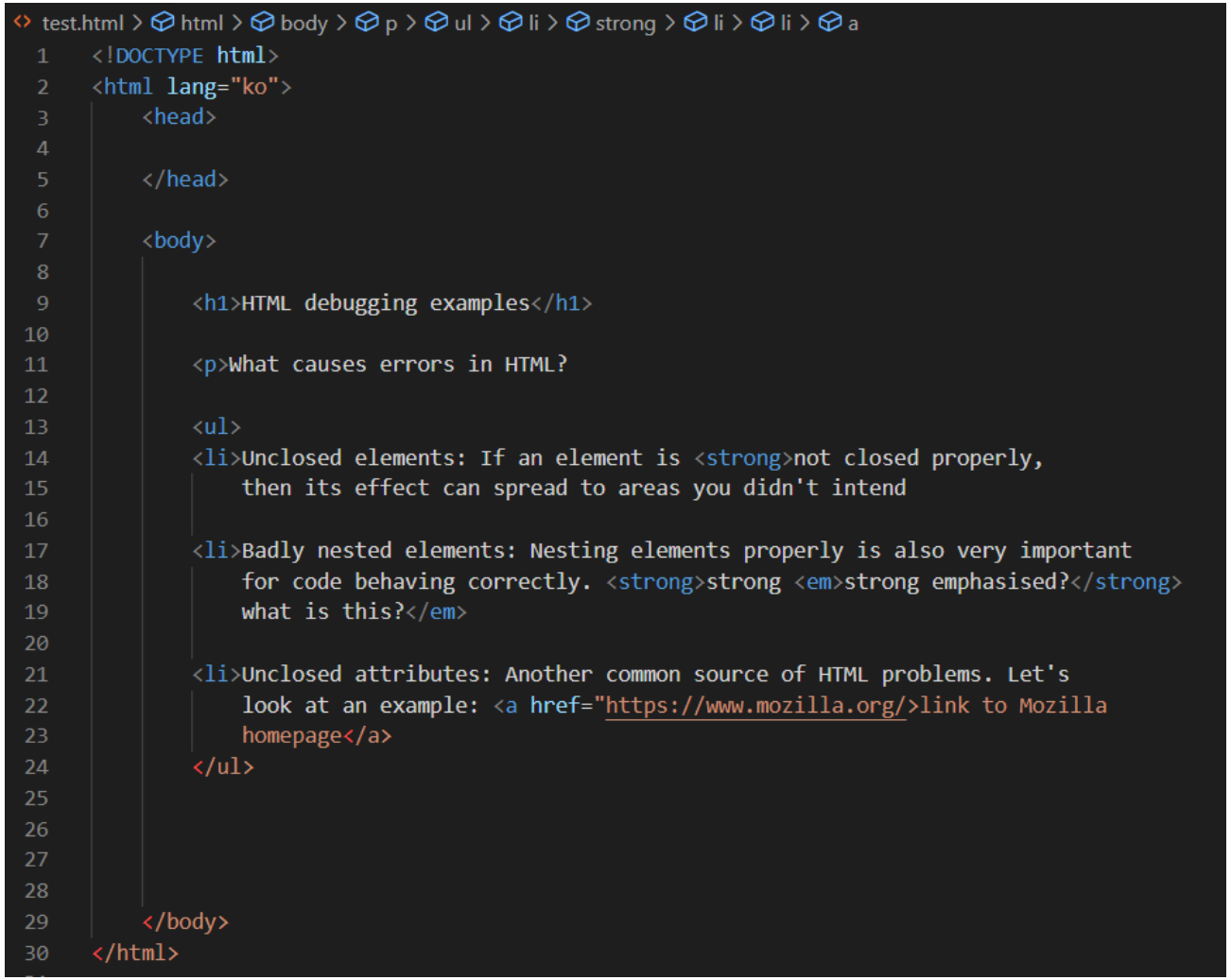
Basic Sections of a Document

- **header:** `<header>`.
- **navigation bar:** `<nav>`.
- **main content:** `<main>`, with various content subsections represented by `<article>`, `<section>`, and `<div>` elements.
- **sidebar:** `<aside>`; often placed inside `<main>`.
- **footer:** `<footer>`.

Basic Sections of a Document

- `<main>` is for content *unique to this page*. Use `<main>` only *once* per page, and put it directly inside `<body>`. Ideally this shouldn't be nested within other elements.
- `<article>` encloses a block of related content that makes sense on its own without the rest of the page (e.g., a single blog post).
- `<section>` is similar to `<article>`, but it is more for grouping together a single part of the page that constitutes one single piece of functionality (e.g., a mini map, or a set of article headlines and summaries), or a theme. It's considered best practice to begin each section with a heading; also note that you can break `<article>`s up into different `<section>`s, or `<section>`s up into different `<article>`s, depending on the context.
- `<aside>` contains content that is not directly related to the main content but can provide additional information indirectly related to it (glossary entries, author biography, related links, etc.).
- `<header>` represents a group of introductory content. If it is a child of `<body>` it defines the global header of a webpage, but if it's a child of an `<article>` or `<section>` it defines a specific header for that section (try not to confuse this with titles and headings).
- `<nav>` contains the main navigation functionality for the page. Secondary links, etc., would not go in the navigation.
- `<footer>` represents a group of end content for a page.

Debugging HTML



Debugging HTML

Syntax errors

These are spelling or punctuation errors in your code that actually cause the program not to run, like the Rust error shown above. These are usually easy to fix as long as you are familiar with the language's syntax and know what the error messages mean.

Logic errors

These are errors where the syntax is actually correct, but the code is not what you intended it to be, meaning that the program runs incorrectly. These are often harder to fix than syntax errors, as there isn't an error message to direct you to the source of the error.

Debugging Tools

Developer Tool : Ctrl+Shift+I or F12
Markup Validation Service : https://validator.w3.org/