INFO2180 - LECTURE 2

HTML

Photo by Goran Ivos on Unsplash

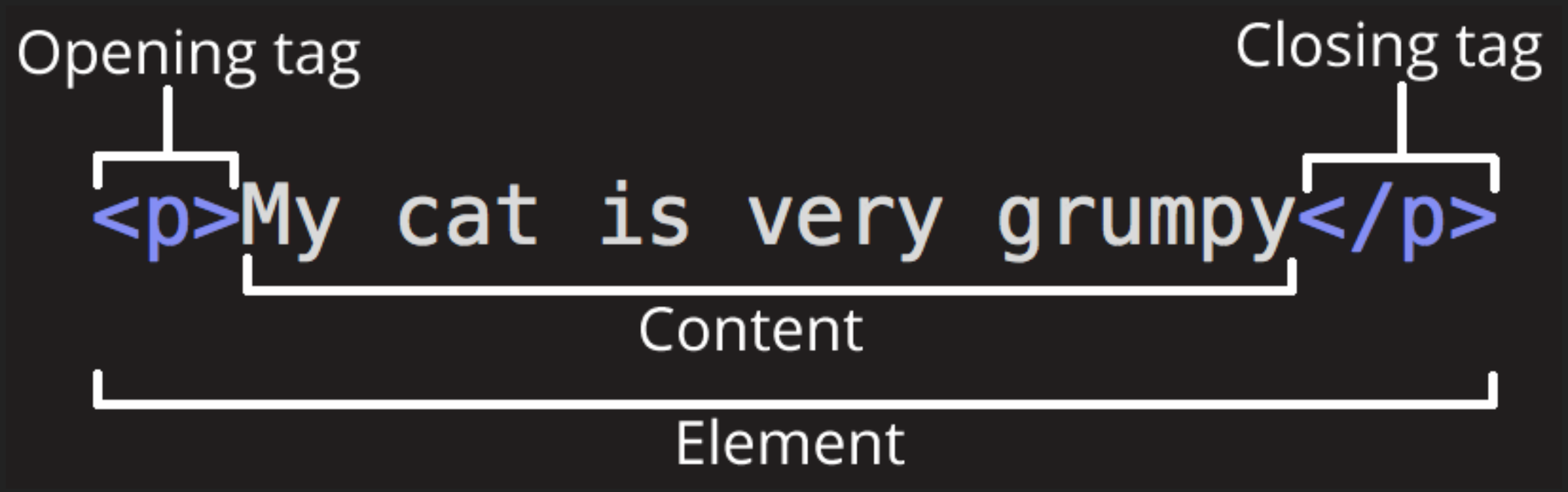
HYPERTEXT MARKUP LANGUAGE (HTML)

HTML is a markup language, used to tell your browser how to structure the webpages you visit.

Mozilla Developer Network

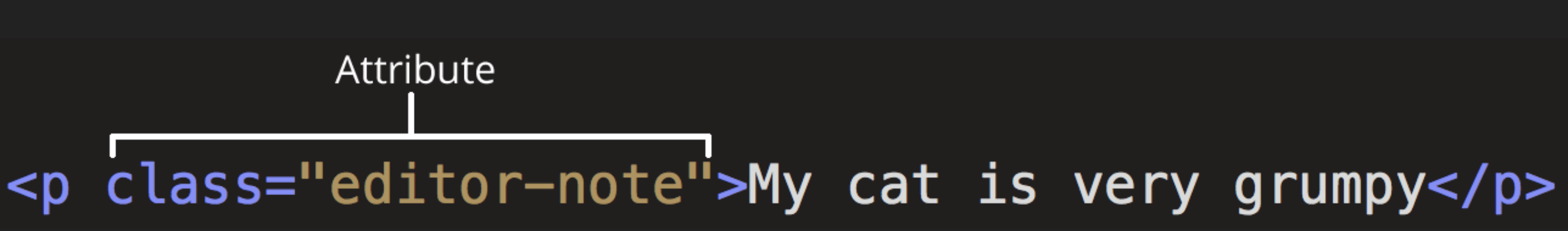
HTML consists of a series of elements, which you use to enclose, or wrap, different parts of the content to make it appear a certain way, or act a certain way.

ANATOMY OF AN HTML ELEMENT



HTML ATTRIBUTES

- Elements can also have attributes
- And some elements have different attributes



NESTING ELEMENTS

- You can place elements inside other elements.

```
<p>My cat is <strong>very</strong> grumpy.</p>
```

- Ensure that they are properly nested (as shown above) and not the following:

```
<p>My cat is <strong>very</p> grumpy.</strong>
```

EMPTY ELEMENTS

- Some elements have no inner content.
- `<img src="images/firefox-icon.png" alt="My test image" />`
- Some other examples are the `<input />`, `<link />`, `<meta />`, `<hr />` and `<br />`.

ANATOMY OF AN HTML DOCUMENT

- HTML documents are typically saved with a `.html` file extension
- All HTML documents have a required structure that includes the following declaration and elements: `<!DOCTYPE html>`, `<html>`, `<head>`, and `<body>`.

ANATOMY OF AN HTML DOCUMENT

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My First HTML Page</title>
  </head>
  <body>
    <p>This is my paragraph</p>
    <img src="images/firefox-icon.png" alt="My test
image" />
  </body>
</html>
```

DOCTYPE

An instruction to the web browser about what version of HTML the page is written in.

```
<!DOCTYPE html>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD
XHTML 1.0 Strict//EN" "http://
www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
```

HTML ELEMENT

- The **<html>** element. This element wraps all the content on the page. It is sometimes known as the root element.

```
<html>
…
</html>
```

HEAD ELEMENT

The **<head>** element acts as a container for machine-readable information (metadata) about the HTML page. This includes things like the *title*, *keywords* and a *page description* that would appear in search results, styles (CSS), scripts (JavaScript), character set declarations, and more.

```
<head>
  <meta charset="utf-8">
  <title>My First HTML Page</title>
</head>
```

TITLE ELEMENT

The **<title>** element. This sets the title of the page, which is the title that appears in the browser tab the page is loaded in. The page title is also used to describe the page when it is bookmarked and used in Search Engines as the title for search engine results.

```
<title>My First HTML page</title>
```

META TAGS

- They provide additional information about your page for web browsers, search engines, etc.

```
<meta charset="utf-8">
<meta name="description"
    content="Learn to create awesome websites." />
<meta name="keywords" content="html, css, javascript" />
<meta name="author" content="John Doe">
```

These **<meta>** tags are placed in the **<head></head>** of the document.

BODY ELEMENT

The **<body>** element. This contains all the content that displays on the page, including text, images, videos, games, playable audio tracks, or whatever else.

```
<body>
  <p>This is my paragraph</p>
  <img src="images/firefox-icon.png" alt="My test image" />
</body>
```

PARAGRAPHS

- Paragraphs are defined using the **<p>** block-level element. Paragraphs can appear one after the other, adding information to a page as desired.

```
<p>This is my paragraph</p>
```

IMAGES

```
<img src="images/firefox-icon.png"
alt="Firefox logo" width="250"
height="350" />
```

HEADINGS

```
<h1>My main title</h1>

<h2>Level 2 heading</h2>

<h3>Level 3 heading</h3>

<h4>Level 4 heading</h4>

<h5>Level 5 heading</h5>

<h6>Level 6 heading</h6>
```

LINKS

- Links are what makes the Web **A WEB**. They link to other web pages or files.

```
<a href="https://www.mona.uwi.edu/">UWI
Mona</a>

<a href="documents/
mydocument.pdf">Download Document</a>
```

LINKS

- Links can also be used to link to parts of the same page.

```
<body id="top">
  ...
  <a href="#top">Back to top</a>
  ...
</body>
```

COMMENTS

```
<!-- This will not show on your page -->
```

LISTS

- Ordered List uses **<ol>** element
- Unordered List uses **<ul>** element
- Definition Lists uses **<dl>** element

EXAMPLE ORDERED LIST

```
<ol>
  <li>technologists</li>
  <li>thinkers</li>
  <li>builders</li>
</ol>
```

EXAMPLE UNORDERED LIST

```
<ul>
  <li>technologists</li>
  <li>thinkers</li>
  <li>builders</li>
</ul>
```

EXAMPLE DEFINITION LIST

```
<dl>
    <dt>Fun</dt>
    <dd>amusing or entertaining</dd>
    <dt>Learn</dt>
    <dd>gain or acquire knowledge</dd>
</dl>
```

HTML CHARACTER ENTITIES

- Reserved characters in HTML must be replaced with character entities. (e.g. `< >` are `< >`)
- Characters that are not present on your keyboard can also be replaced by entities. E.g. `é è ñ` are `&eacute; &egrave; &ntilde;`
- `™ ©` are `&trade; &copy;`
- And there are many others

BLOCK AND INLINE ELEMENTS

- Elements are usually either block-level or inline elements.
- Block-Level elements begin on new lines, stacking one on top of the other (e.g. `<div>`, `<p>`, `<h1>`, `<ol>`, `<ul>`, `<table>`). They typically occupy the entire width of its parent element (container) and can contain inline elements or other block-level elements.
- Inline elements do not begin on a new line. They can start anywhere in a line and only occupies the width of their content. They should not wrap block-level elements and are typically used with text. (e.g. `<a>`, `<span>`, `<strong>`, `<em>`)

TABLES

- HTML Tables are used to to mark up structured tabular data.
- The **<table></table>** element is used to create a table on a page.
- The **<tr></tr>** element is then used to define rows in a table.
- The **<td></td>** element defines a table cell and when more than on is used within a table row, it creates columns.
- You can also create table headings use the **<th></th>** element.
- If you need to add a caption to your table, use the **<caption></caption>** element.

```
<table>
  <caption>Items available in store.</caption>
  <tr>
    <th scope="col">Item</th>
    <th scope="col">Availability</th>
    <th scope="col">Qty</th>
    <th scope="col">Price</th>
  </tr>
  <tr>
    <td>Textbook</td>
    <td>In Stock</td>
    <td>1</td>
    <td>$130.02</td>
  </tr>
  <tr>
    <td>Mechanical Pencils</td>
    <td>In Stock</td>
    <td>2</td>
    <td>$52.94</td>
  </tr>
</table>
```

TABLES

- You can also add **<thead>**, **<tbody>** and **<tfoot>** elements to your table.
- The **<thead>** element defines a set of rows defining the head of the columns of the table.
- The **<tbody>** encapsulates a set of table rows, indicating that they comprise the body of the table.
- The **<tfoot>** element defines a set of rows summarizing the columns of the table.

```
<table>
  <thead>
    <tr>
      <th scope="col">Items</th>
      <th scope="col">Cost</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Textbook</td>
      <td>$130.02</td>
    </tr>
    <tr>
      <td>Mechanical Pencils</td>
      <td>$52.94</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Total</td>
      <td>$182.96</td>
    </tr>
  </tfoot>
</table>
```

FORMS

- An HTML form is used to collect user input. And the input is often sent to a server to be processed.
- To add a form you use the **<form>** element and this will wrap other elements.
- Forms consist of other elements, such as **<label>**, **<input />**, **<textarea>**, **<select>**, **<button>**, among other things.
- These elements all consist of certain ***attributes*** which allow us to define other properties of our form elements.

FORMS

- The **<form>** element typically contains the following attributes:
  - The ***action*** attribute which determines were to send the data when the form is submitted.
  - The ***method*** attribute which specifies which HTTP method (either **GET** or **POST**) to use when submitting the data.
- e.g.

```
<form action="process_data.php" method="post">
```

FORMS

- The **<input>** element is the most frequently used and can be displayed differently depending on it's **type** attribute.
- **<input type="text">** - Displays a single-line text input field
- **<input type="radio">** - Displays a radio button (for selecting one of many choices)
- **<input type="checkbox">** - Displays a checkbox (for selecting zero or more of many choices)
- **<input type="submit">** - Displays a submit button (for submitting the form)

FORMS

- So a simple form could look like the following:

```
<form action="/process_data.php" method="post">
  <div>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="John">
  </div>
  <div>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Doe">
  </div>

   <input type="submit" value="Submit">

</form>
```

First name:

John

Last name:

Doe

Submit

FORMS - RADIO BUTTONS

```
<form action="/process_data.php" method="post">
  <p>Choose your favorite Web language:</p>
  <div>
    <input type="radio" id="html" name="fav_language" value="HTML">
    <label for="html">HTML</label>
  </div>
  <div>
    <input type="radio" id="css" name="fav_language" value="CSS">
    <label for="css">CSS</label>
  </div>
   <div>
     <input type="radio" id="javascript" name="fav_language"
value="JavaScript">
    <label for="javascript">JavaScript</label>
  </div>

</form>
```

Choose your favorite Web language:

- ○ HTML
- ○ CSS
- ○ JavaScript

FORMS - CHECKBOXES

```
<form action="/process_data.php" method="post">
  <div>

<input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">
    <label for="vehicle1">I have a bike</label>
  </div>
  <div>
    <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">
    <label for="vehicle2">I have a car</label>
  </div>
  <div>
   <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">
   <label for="vehicle3">I have a boat</label>
  </div>
</form>
```

- [ ] I have a bike
- [ ] I have a car
- [ ] I have a boat

FORMS - TEXTAREA

```
<form action="/process_data.php" method="post">
  <div>
    <label for="email">Email:</label>
    <input type="email" id="email" name="email">
  </div>
  <div>
   <label for="message">Message:</label>
<textarea rows="10" cols="30" id="message" name=
"message"></textarea>
  </div>
</form>
```

Email:

Message:

SETTING THE LANGUAGE OF YOUR HTML DOCUMENT

- You can (and really should) set the language of your web page. This can be done by adding the lang attribute to the opening html tag.

```
<html lang="en-US">
```

USING SEMANTIC HTML

SEMANTIC HTML

- Semantic HTML introduces meaning to the code we write.
- Always write HTML code that describes the content rather than how that content should look. Presentation (how it should look), is the sole responsibility of CSS.
- Using semantic tags makes it clear to the browser what the meaning of a page and its content is.

SEMANTIC HTML EXAMPLE

Bad

```
<font size="6">
    <strong>This is a heading</strong>
</font>
```

Good

```
<h1>This is a heading</h1>
```

SEMANTIC HTML EXAMPLE

Bad

```
<p>Supermarket list</p>
<p>Sugar<br>
Butter<br>
Eggs<br>
</p>
```

Good

```
<h1>Supermarket List</h1>
<ul>
  <li>Sugar</li>
  <li>Butter</li>
  <li>Eggs</li>
</ul>
```

SOME OTHER SEMANTIC ELEMENTS

- **<blockquote>** is used for quotes.
- **<abbr>** is used for abbreviations.
- Elements such as **<header>**, **<nav>**, **<main>**, **<footer>**, **<aside>**, **<section>** and **<article>** can be used to define parts of a web page.
- We also have **<video>**, **<audio>** elements.
- And there are others…

SOME OTHER SEMANTIC ELEMENTS

- **<header>** - container for introductory content. Typically contains one more heading elements, logo or icon, author information.
- **<nav>** - defines a set of navigation links
- **<aside>** - defines some content indirectly related to the surrounding or the main content (like a sidebar)
- **<main>** - specifies the main content of a document. There should only be one main element.

SOME OTHER SEMANTIC ELEMENTS

- **<footer>** - footer for a document or section. Typically contains author, copyright or contact information
- **<section>** - defines a section in a document. Pages can be split into sections for introduction, content, and contact information
- **<article>** - can be used for forum posts, blog posts or news articles.

WHY IS SEMANTIC CODE IMPORTANT?

- Visually impaired people rely on special devices to read pages back to them. Semantic code aids with accessibility. It can also help them to navigate the page more quickly.
- Search engines need to understand what your content is about to rank you properly on search engines. Semantic code can improve your ranking.
- Semantic code can be more straightforward for another developer to understand.

WHY IS SEMANTIC CODE IMPORTANT?

- Because semantic code does not contain design elements, it is possible to change the look and feel of your site without recoding all of the HTML.

WEB STANDARDS

ARE RULES AND GUIDELINES ESTABLISHED BY THE WORLD WIDE WEB CONSORTIUM (W3C) TO PROMOTE CONSISTENCY IN THE CODE WHICH MAKES UP A WEB PAGE

http://www.soswebdesign.com/gallery/webstandards.cfm

WEB STANDARDS

- Its important to ensure you follow the Standards
  - It ensures your pages will work across different browsers.
    - more likely to display properly in the future.
- Current Standard is HTML 5
- Use the W3C Validator to check your HTML markup. https://validator.w3.org/

RESOURCES

- W3 Schools HTML Tutorial - https://www.w3schools.com/html/default.asp
- Codecademy Learn HTML - https://www.codecademy.com/learn/learn-html

ANY QUESTIONS?