CoGrammar

Welcome to this session:
Cascading Style Sheets (CSS)

The session will start shortly...

Questions? Drop them in the chat.
We'll have dedicated moderators answering questions.

Safeguarding & Welfare

We are committed to all our students and staff feeling safe and happy; we want to make sure there is always someone you can turn to if you are worried about anything.

If you are feeling upset or unsafe, are worried about a friend, student or family member, or you feel like something isn't right, speak to our safeguarding team:

Ian Wyles
Designated Safeguarding Lead

Simone Botes

Nurhaan Snyman

Rafiq Manan

Ronald Munodawafa

Tevin Pitts

Scan to report a safeguarding concern

or email the Designated Safeguarding Lead:
Ian Wyles
safeguarding@hyperiondev.com

CoGrammar HyperionDev

Skills Bootcamp Cloud Web Development

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

Skills Bootcamp Cloud Web Development

- For all **non-academic questions**, please submit a query: ***www.hyperiondev.com/support***
- **Report a safeguarding incident: *www.hyperiondev.com/safeguardreporting***
- We would love your feedback on lectures: ***Feedback on Lectures.***
- Find all the lecture **content** in you **Lecture Backpack** on GitHub.
- If you are hearing impaired, please kindly use your computer's function through Google chrome to enable captions.

Learning Outcomes

- Apply CSS to style HTML elements using classes and IDs for customization.
- Manage layout and positioning of elements using CSS box model properties.
- Develop the fundamental concepts of CSS including the box model, selectors, and properties.

Question

- How would you transform the house skeleton?

Lecture Overview

- Introduction to CSS
- Styles
- Selectors
- The Box Model

CSS

Cascading Style Sheets (CSS) is a language used to change the presentation and styling of a document written in a markup language e.g. HTML

- Helps us create visually appealing and user-friendly websites.
- HTML structures the content, CSS controls how the content looks.
- CSS uses a set of rules written in a certain syntax to style HTML.
- We use CSS to create style sheets, which define the appearance and layouts of the elements on a webpage.
- The various properties which we can control with CSS can be found here.

Styles: Inline Style

- HTML elements are described using attributes and properties.
- One of the attributes of an element is style, which we can change by adjusting its properties using CSS rules.
- Attributes are adjusted inside the element's beginning tag.

Styles: Inline Style

- For example: **Text Elements:**

attributes Property values

```
<p style="font-family:Montserrat;color: cornflowerblue;font-size:22px">
    Let's test inline styling on this paragraph. <br>
    This paragraph should be blue, in the Montserrat font, size 22px.</p>
```

Styles: Internal CSS

- CSS rules can be defined in the **head** part of the HTML template, inside the **style element**. This is known as **internal CSS**.
- Rules can be defined for every type of element in the HTML document.

```
<head>
    <style>
        p {
            font-style: italic;
            color: chartreuse;
        }
    </style>
</head>
<body>
    <p style="font-family:Montserrat;
    color:cornflowerblue;
    font-size:22px;">
        Let's test inline styling on this paragraph.
        <br>This paragraph should be blue,
        in the Arial font, size 22px.</p>
</body>
```

- The style sheet consists of **selectors** and **declarations**
  - **Selectors:** indicates which element you want to style
  - **Declaration block:** contains one or more declarations, separated by semicolons and enclosed in curly brackets.
  - **Declaration:** includes a property and a value separated by a colon

Styles: External CSS

- Another way to define the style for an HTML file is by writing all the style rules in a **separate .css** file. This is called **external CSS**.
- The external file can be **linked** to any HTML file to apply the style rules.
- This method is useful when **applying the same style rule to multiple HTML files**.

```
<head>
    <link href="externalStyle.css" rel="stylesheet" type="text/css" />
</head>
```

➔ In the **head** part of the HTML file, in a **link element** define

- **href:** define the name and path of your file (relative to the current working directory)
- **rel:** describes the type of relation the external file is to the HTML (i.e. stylesheet)
- **type:** tells the browser what sort of file it is (only necessary for old browsers)

Let's take a break

CoGrammar

Best Approaches to Styling

- Styling is applied depending on which rules are **closest to the element.**
- Inline styling will be applied to individual elements **overwriting the internal or external CSS** defined for the whole web page.
- Internal styling will overwrite any external styling defined.
- **External CSS** should be chosen over internal CSS where possible
  - **Readability:** separating CSS code and HTML makes code easier to read and follow.
  - **Maintainability:** updating and debugging styling rules is easier since only external CSS files need to change or be replaced.

CSS Selectors

CSS selectors attach to the HTML elements on web pages which allows for customized styling

- There are three common CSS selectors that we will look at:
  - **Element selector**
    - The same style is applied to elements with the same tag.
  - **ID selector**
    - Styles are applied to specific elements using a unique ID.
  - **Class selector**
    - The same style is applied to elements in the same class.

Element Selectors

- The most basic type of CSS selector.
- Style rules are defined for all elements of the same type of tag.
- The selector pinpoints an **element tag** and applies the **same style** to **all elements with that specific tag name**.

For example: Styling the body element

```
body {
    background-color: aliceblue;
    outline-width: 5px;
    outline-color: darkcyan;
    outline-style: groove;
}
```

ID Selectors

- ID selectors apply styles to HTML elements which are identified by its **unique ID name**.
- The ID of an element is an **attribute** defined at the beginning of the HTML tag. The value assigned to this attribute must be **unique**.
- The ID selector is called using a **hash (#)**, followed by the **ID name**.

```
<!-- Here we will be testing ID selectors -->
<h2 id="heading2"> Welcome everyone! </h2>
```

```
#heading2 {
    text-align: center;
    font-family: Montserrat, Helvetica;
    font-size: 26px;
    font-style: italic;
    color: darkgoldenrod;
}
```

CoGrammar

Class Selectors

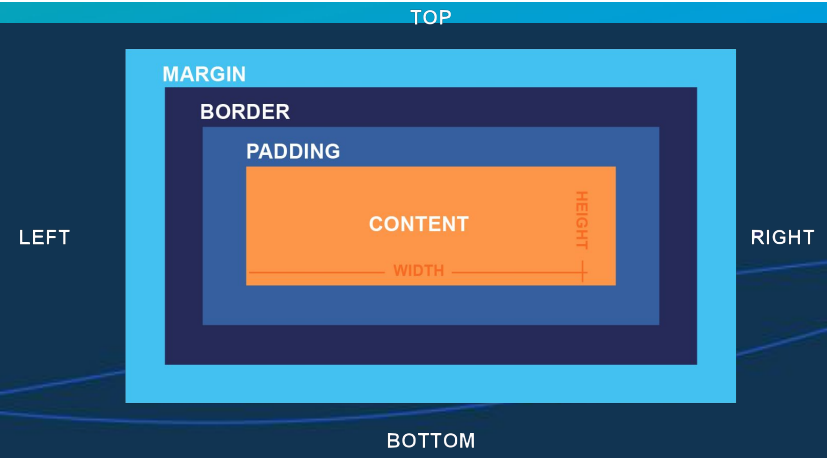
- Class selector aims to change **all HTML elements associated with a specific class**.
- **Class** is also an **attribute**, defined like an ID, but it is not unique.
  - It is called using a **dot (.) followed by the class name**.
  - The **element tag** belonging to that class can be referenced as well.

```css
.endingMessage {
    text-align: center;
    font-family: Bubbly;
    font-size: 20px;
    color: darkslateblue;
    margin-bottom: -18px;
}

p.endingMessage {
    padding-bottom: 20px;
}
```

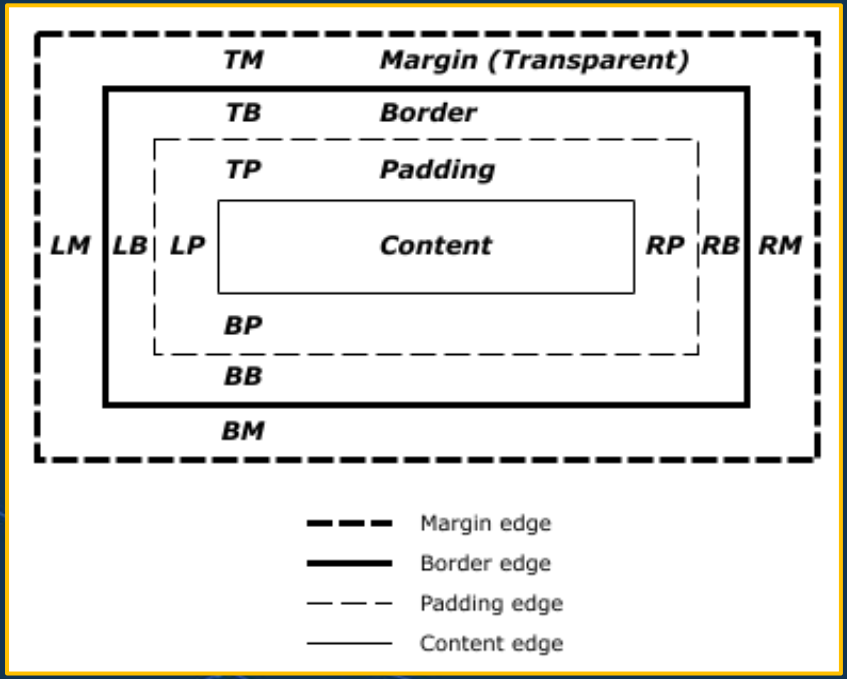
```html
<h3 class = endingMessage>
    Thank you for joining us :)
</h3>

<p class = endingMessage>
    Please let us know if you have any
    questions regarding the code presented.
</p>
```

The Box Model

- A **rectangle** is created for each element in the HTML document.
- The **box model** describes how the **padding, border, and margin** are added to the content to create the rectangle.
- Each area is surrounded by a perimeter called an **edge**.



Source: GCFGlobal



- **Content Edge or Inner Edge**
  - Surrounds the rectangle given by the width and height of the box, depending on the content.
- **Padding Edge**
  - Surrounds the box padding.
  - *padding, padding-top, padding-bottom, padding-left, padding-right*
- **Border Edge**
  - Surrounds the box's border.
  - *border, border-top, border-bottom, border-left, border-right*
- **Margin Edge or Outer Edge**
  - Surrounds the box margin.
  - *margin, margin-top, margin-bottom, margin-left, margin-right*

CSS Validator

- An important step in your development journey is **testing** and **debugging** your code.
- Using tools like VSCode allows us to identify errors in our **syntax** and **formatting**, but some errors may go unnoticed.
- We can use other tools like this CSS Validation Service, to check our CSS code as well.
- When our code doesn't behave as expected, or our web pages don't look the way we intended, understanding how to **identify errors** is an important first step before we can **debug**.

Questions and Answers

CoGrammar

Thank you for attending

CoGrammar

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education