Welcome to lecture 6!

Agenda

Session Objectives

- Introduction to CSS
- CSS Selectors (Class vs ID)
- display Property: inline, block
- Intro to Pseudo-classes (link, visited, hover, active)
- Quiz
- Recap & Homework (due Sunday, April 6th by 5PM)

Introduction to CSS

Painting the CSS portrait!

- What is CSS?
  - Cascading Style Sheets.
- A stylesheet language used to describe the presentation of a document written in a markup language like HTML
- Controls how HTML elements are displayed on screen, paper, or in other media
- Defines the visual style of a webpage
  - Colors
  - Fonts
  - Layout
  - Spacing
  - And much more!

Why Use CSS?

- Clean Code: separates design from HTML structure
- Easy Updates: change the look by editing CSS only
  - Eg: changing the theme of a website can be done by changing the CSS files *only*
- Consistent Look: ensures uniformity across your website
  - Styles for headers, paragraphs and other elements for all HTML pages can be centrally defined
- Faster Loading: smaller file sizes and browser caching
  - To load a website, the browser needs to fetch the HTML/CSS files from a server
  - Large HTML/CSS files can impact website load times
- Works Everywhere: adapts to different screen sizes (responsiveness!)
- Good for Everyone: improves website accessibility

Introduction to CSS Selectors

- What are selectors?
  - Patterns used to select the HTML elements you want to style
- They act as a bridge between your CSS rules and the HTML elements
- Different types of selectors target elements with varying degrees of precision. Include:
  - Basic selectors: Select elements by their name, class, ID, or universally (all elements)
  - Combinators: Refine selections based on hierarchical or sibling relationships in the DOM
  - Attribute Selectors: Select elements by the presence or value (exact or partial) of their attributes
  - Pseudo-class Selectors: Select elements by the presence or value (exact or partial) of their attributes
  - Pseudo-element Selectors: Style generated content or specific parts of elements (e.g., ::before, ::first-letter)

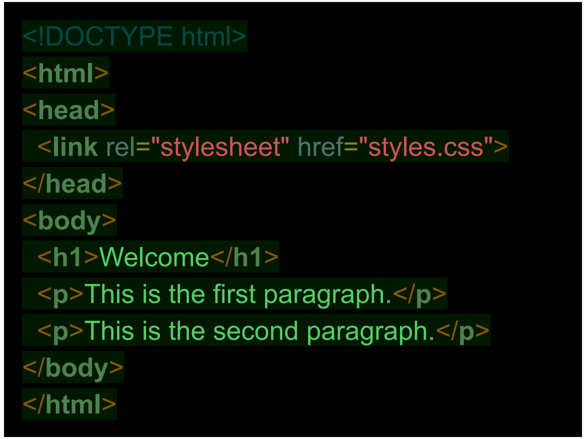
Basic Selectors

- What are basic selectors?
  - The foundation of CSS element targeting
- Types of basic selectors
  - Element (Tag) Name: Targets all HTML elements of a specific type (e.g., p, h1).
  - Class: Targets elements with a specific class attribute.
  - ID: Targets a single, unique element with a specific id attribute
  - Universal Selectors: Targets all elements in an HTML page

Basic Selectors: Tag based selector

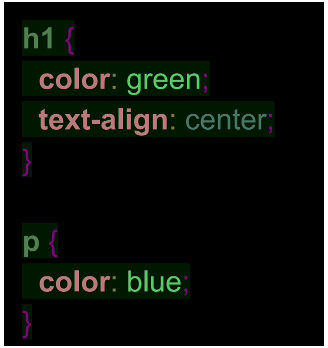
- Targets all HTML elements of a specific type
- A basic selector
  - Syntax: elementname (e.g., p, h1, div) { ... }
  - Targets all HTML elements of a specific type.
  - Example: p { color: blue; }
    - Will turn all paragraph text to blue color

Index.html (Linked CSS)



```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Welcome</h1>
  <p>This is the first paragraph.</p>
  <p>This is the second paragraph.</p>
</body>
</html>
```

styles.css

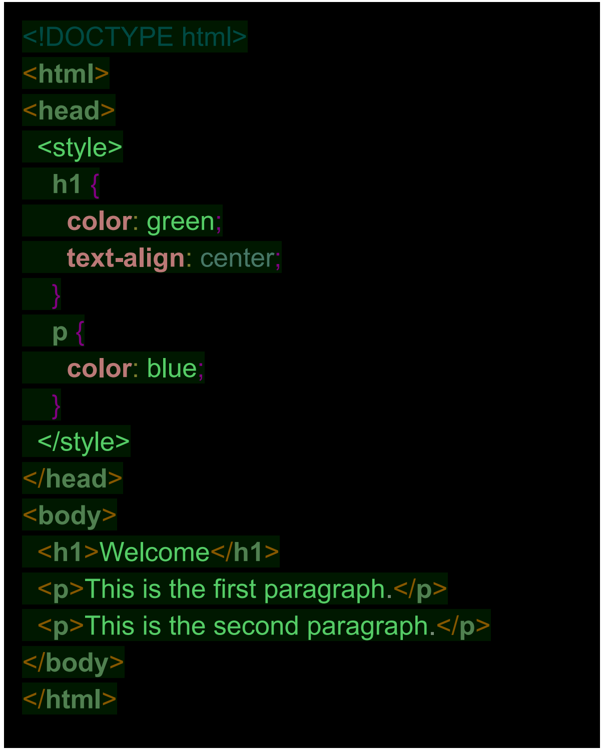


```
h1 {
  color: green;
  text-align: center;
}

p {
  color: blue;
}
```

OR

Index.html (internal CSS)



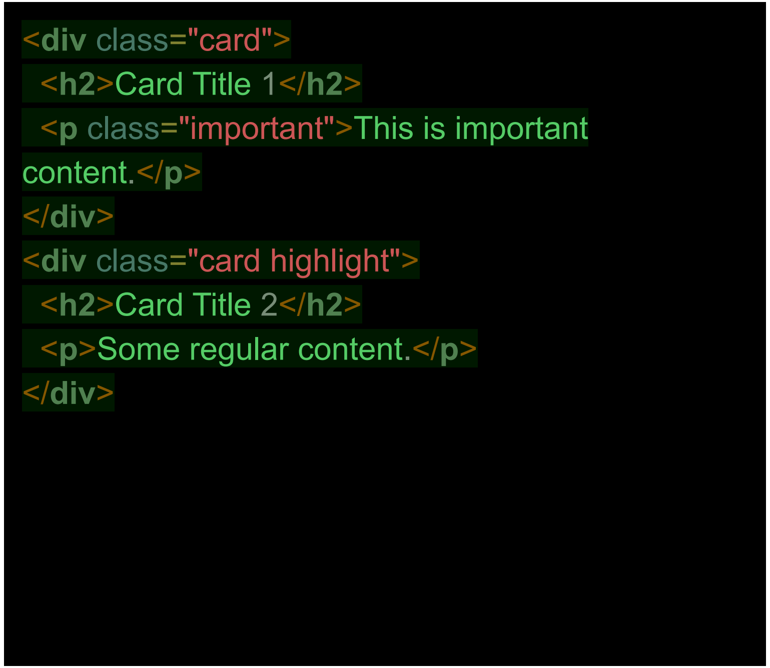
```
<!DOCTYPE html>
<html>
<head>
  <style>
    h1 {
      color: green;
      text-align: center;
    }
    p {
      color: blue;
    }
  </style>
</head>
<body>
  <h1>Welcome</h1>
  <p>This is the first paragraph.</p>
  <p>This is the second paragraph.</p>
</body>
</html>
```

Class Selector

- What are class selectors?
    - Targets HTML elements with a specific class attribute
    - Allows you to apply the same styles to multiple elements
- Syntax:
    - .classname (e.g., .important, .highlight)
        - The classname, like variable names in Javascript, is decided by the developer
- Key features
    - The class attribute can be added to any HTML element
    - Multiple classes can be applied to an element, separated by spaces.
    - Example: .important { font-weight: bold; }
        - Will make text in elements with the class "important" bold
    - Classes promote reusability of styles across different elements

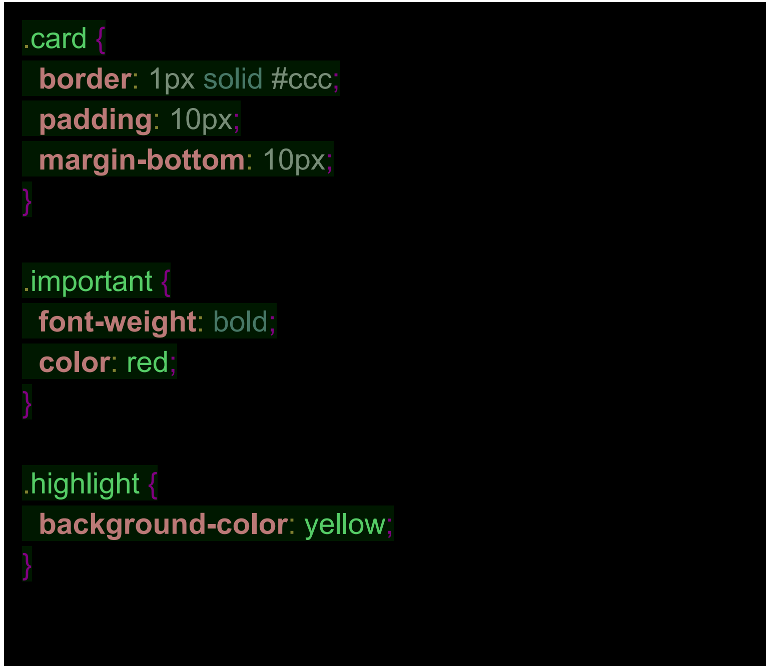
Example: Class Selector

HTML Snippet from Index.html



```html
<div class="card">
  <h2>Card Title 1</h2>
  <p class="important">This is important content.</p>
</div>
<div class="card highlight">
  <h2>Card Title 2</h2>
  <p>Some regular content.</p>
</div>
```

styles.css



```css
.card {
  border: 1px solid #ccc;
  padding: 10px;
  margin-bottom: 10px;
}

.important {
  font-weight: bold;
  color: red;
}

.highlight {
  background-color: yellow;
}
```

ID Selector

- What are ID selectors?
  - Targets a single, unique HTML element with a specific id attribute
- Syntax:
  - #idname (e.g., #main-header, #submit-button)
    - The *idname*, like variable names in Javascript, will be defined by the developer
- Key features
  - The id attribute should be unique within an HTML document
  - Primarily used for JavaScript interaction or very specific styling
  - Example: #main-header { font-size: 24px; }
    - The element with the ID "main-header" will have a font size of 24 pixels
  - ID selectors have a *higher specificity* than class selectors
    - If you have an element with id and class attribute, the styles in the id attribute generally overrides class styles

Examples: ID Selector

Example 1

HTML Snippet from Index.html

```html
<h1 id="main-title">Welcome to Our Website</h1>
<p>Some introductory text.</p>
```

styles.css

```css
#main-title {
  color: purple;
  text-align: center;
  border-bottom: 2px solid purple;
}
```

Example 2: what will be the color of the header?

HTML Snippet from Index.html

```html
<h2 class="section-heading" id="page-title">Page
Title</h2>
```

styles.css

```css
.section-heading {
  color: orange;
  font-size: 1.5em;
}

#page-title {
  color: green;
  text-align: center;
}
```

Class vs ID: Choosing the Right Selector

- When to use class selector?
  - Use when you want to style multiple elements the same way
  - Reusable across different elements
  - Can be used multiple times on a single page
- When to use ID selector?
  - Use when you want to style a single, unique element
  - Should be unique within the HTML document
  - Often used for specific layout elements or JavaScript targeting
  - Has higher specificity than class selectors
- Best Practice: Generally, prefer using classes for styling and IDs for JavaScript functionality or unique layout sections

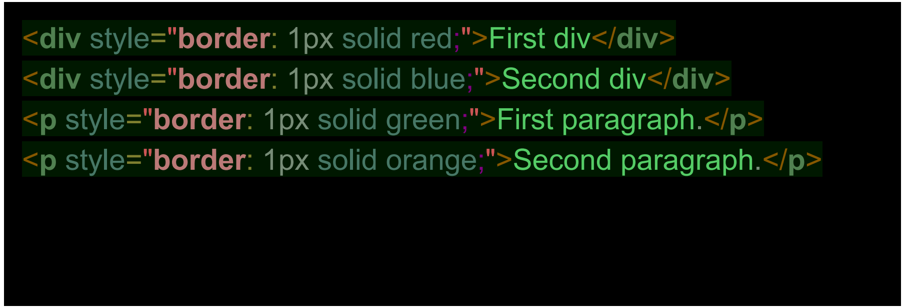
Controlling Element Display

Introduction to the ‘display’ Property

- The display property in CSS controls how an element is rendered on the page.
    - Dictates the flow of the element in relation to other elements
    - Key values for the display property that we will cover today
        - block
        - Inline
- What does display: block; mean?
    - Block-level elements behave like a block of content
    - They always start on a new line
    - They take up the full width available in their parent container (stretching to the left and right)
    - By default, their height is determined by their content
    - The width and height properties on block-level elements can be set by the developer
    - Examples of block-level elements by default
        - <div>, <p>, <h1> to <h6>, <ul>, <ol>, <li>, <form>

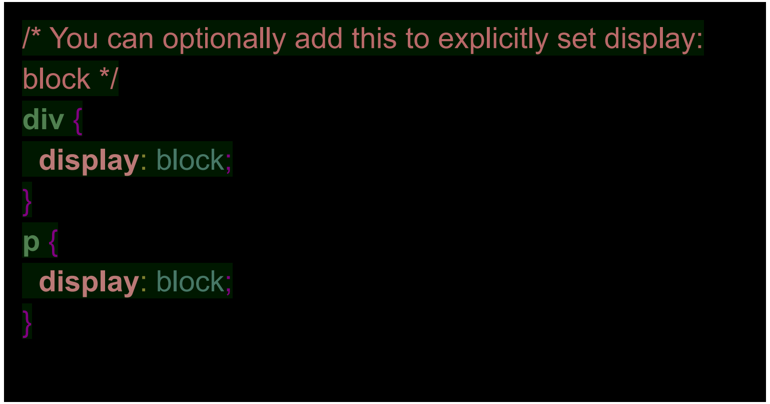
Example: Class Selector

HTML Snippet from Index.html



```html
<div style="border: 1px solid red;">First div</div>
<div style="border: 1px solid blue;">Second div</div>
<p style="border: 1px solid green;">First paragraph.</p>
<p style="border: 1px solid orange;">Second paragraph.</p>
```

styles.css



```css
/* You can optionally add this to explicitly set display:
block */
div {
  display: block;
}
p {
  display: block;
}
```

Display property: Inline elements

- What are inline elements?
  - Inline elements flow along with the surrounding content, like words in a sentence
  - They do not start on a new line
  - They only take up as much width as their content requires
  - You cannot set the width and height properties on inline elements directly; they are determined by the content
  - Examples of inline elements by default: <span>, <a>, <img>, <strong>, <em>
- Example:

HTML Snippet

```html
<span>First span</span>
<span>Second span</span>
<a href="#">First link</a>
<a href="#">Second link</a>
```

styles.css

```css
span {
  display: inline;
  border: 1px solid purple;
}
a {
  display: inline;
  border: 1px solid brown;
}
```

Key Differences: inline vs block

- Setting display to block
  - Use for structural elements and content that needs to be on its own line
  - Good for containers and larger blocks of text
  - Allows setting width and height
- Setting display to inline
  - Use for small elements within a line of text
  - Good for formatting specific words or phrases
  - width and height properties are ignored

Introduction to Pseudo-classes

Styling Element States: Pseudo-classes

- Keywords added to a selector that specify a special state of the selected element
- Allow you to style elements dynamically based on user interaction or their position within the document
- Pseudo-classes are denoted by a colon(:) followed by the name of the pseudo-class (e.g., :hover, :active)
- Commonly used pseudo-classes:
    - :link: Represents links that have not yet been visited by the user
        - Allows you to set a default style for unvisited links
        - Example: a:link { color: blue; }
    - :visited: Represents links that have been visited by the user
        - Useful for providing visual feedback to the user about which links they have already clicked
        - Example: a:visited { color: purple; }

Example: Pseudo-classes

HTML Snippet

```html
<p><a href="https://www.google.com">Visit Google (Unvisited)</a></p>
<p><a href="https://www.example.com">Visit Example (Might be visited)</a></p>
```

styles.css

```css
a:link {
  color: blue;
  text-decoration: none;
}

a:visited {
  color: purple;
  text-decoration: underline;
}
```

Pseudo-classes: hover

- The :hover pseudo-class applies styles when the user moves their mouse pointer over an element.
- Commonly used to provide interactive feedback, such as changing the appearance of buttons or links.
- Example
    - a:hover { color: red; }
        - Turn the color of the link text red when the mouse is over it

HTML Snippet

```
<p><a href="https://www.google.com">Visit Google (Unvisited)</a></p>
<p><a href="https://www.example.com">Visit Example (Might be visited)</a></p>
```

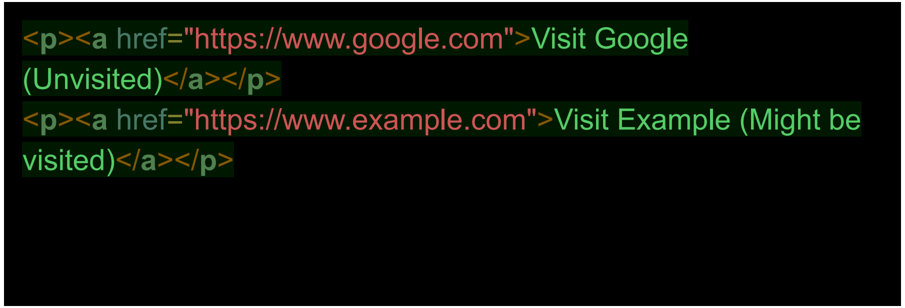
styles.css

```
a:hover {
  color: red;
  font-weight: bold;
}
```

Pseudo-classes: active

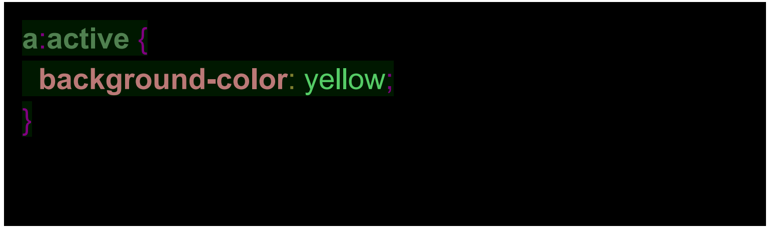
- The :active pseudo-class applies styles when an element is being activated by the user. Eg: the moment a mouse button is pressed down on a link
- Often used to provide a subtle visual cue that an element is being interacted with.
- Example
    - a:active { background-color: yellow; }
        - Briefly changes the background color of the link to yellow when it's clicked

HTML Snippet



```html
<p><a href="https://www.google.com">Visit Google (Unvisited)</a></p>
<p><a href="https://www.example.com">Visit Example (Might be visited)</a></p>
```

styles.css



```css
a:active {
  background-color: yellow;
}
```

Order of Pseudo-classes

- When styling links, it's important to define the pseudo-classes in a specific order for them to work correctly
- The recommended order is:
  - :link
  - :visited
  - :hover
  - :active
- If these are not in the correct order, some styles might not be applied as expected (e.g., :hover might override :visited)

Quiz Time!

1. Selector

- What is the primary use of the class attribute in HTML when working with CSS?
    - a) To uniquely identify an element for JavaScript.
    - b) To apply the same styles to multiple elements.
    - c) To specify the element's position on the page.
    - d) To define the element's content.

Correct Answer: b

2. Display

- Which CSS property controls whether an element starts on a new line and takes up the full width?
  - a) width
  - b) display
  - c) float
  - d) align

Correct Answer: b

3. Hoisting & Scope

- Which display value makes an element behave like inline text?
    - a) block
    - b) inline-block
    - c) inline
    - d) none

Correct Answer: c

4. Pseudo-class

- Which pseudo-class is used to style a link when the mouse cursor hovers over it?
  - a) :link
  - b) :visited
  - c) :hover
  - d) :active

Correct Answer: c

Final stretch!

Recap

- CSS, aka, Cascading Style Sheets is used to style HTML elements
- CSS selectors target specific HTML elements for styling:
  - Element Selectors target by tag name.
  - Class Selectors target by the class attribute (using .)
  - ID Selectors target by the id attribute (using #). IDs have higher specificity
- The display property controls the layout behavior of elements:
  - block: Elements start on a new line and take full width
  - inline: Elements flow with surrounding content and only take necessary width
- Pseudo-classes style elements based on their state (e.g., :link, :visited, :hover, :active)

Homework (due Sunday, April 6th by 5PM)

- Navigation Bar: Create a vertical navigation bar with a few links that change color on hover
  - Reference
- Business Card: Design a card-like layout for a business card for yourself. Your name, your title (work title, or student), and other details can be included on your business card
  - Reference
    - Please don’t share your real phone number or other sensitive information
- Contact Us Form: Create a simple HTML form with labels and input fields, and then use CSS to add basic styling like borders, padding, and alignment
  - Reference

Great job!