

Module 3 Day 1

CSS

What makes an application?

- Program Data

- ✓ Variables & .NET Data Types
- ✓ Arrays
- ✓ More Collections (list, dictionary, stack, queue)
- ✓ Classes and objects (OOP)

- Program Logic

- ✓ Statements and expressions
- ✓ Conditional logic (if)
- ✓ Repeating logic (for, foreach, do, while)
- ✓ Methods (functions / procedures)
- ✓ Classes and objects (OOP)
- ❑ Frameworks (MVC)

- Input / Output

- User

- ✓ Console read / write

- ❖ HTML / CSS

- ❑ Front-end frameworks (HTML / CSS / JavaScript)

- Storage

- ✓ File I/O
- ✓ Relational database
- ❑ APIs

HTML Basics

- Document hierarchy (Tree)
- Tags / Elements
- html / head / body elements
- Attributes

```
<!DOCTYPE html>
<html>
<head>
|   <title>Basic HTML Sample Document</title>
</head>
<body>
|   <h1>A Major Heading</h1>
|   <p>
|       A paragraph with <a href="/home.htm">a hyperlink</a>.
|   </p>
</body>
</html>
```



Demo

Semantic HTML

- Semantics == meaning
- Semantic elements – imply some meaning to parts of a page
 - main, article, section, aside, header, footer, nav
- Non-semantic elements
 - div, span, b, l
- Semantic HTML is preferred
- HTML applies *semantics*
- CSS applies *style*
- [HTML5 Semantic Elements](#)

Styles / Cascading Stylesheets (CSS)

- Settings that *change the look* of an element
- May be applied directly to an element using the *style* attribute
- May be added as an element in the HTML document `<style></style>`
- May be included in a stylesheet
 - A separate file linked to from the HTML doc
 - Usually .css file extension



Let's
Code

CSS Rules - Properties

```
Selector(s) {  
    property : value;  
    property : value; ...  
}
```

- Property : value pairs
 - color : Red;
 - background-color : Yellow;
- https://developer.mozilla.org/en-US/docs/Web/CSS/Reference#Keyword_index

CSS Rules - Selectors

```
Selector(s) {  
    property : value;  
    property : value; ...  
}
```

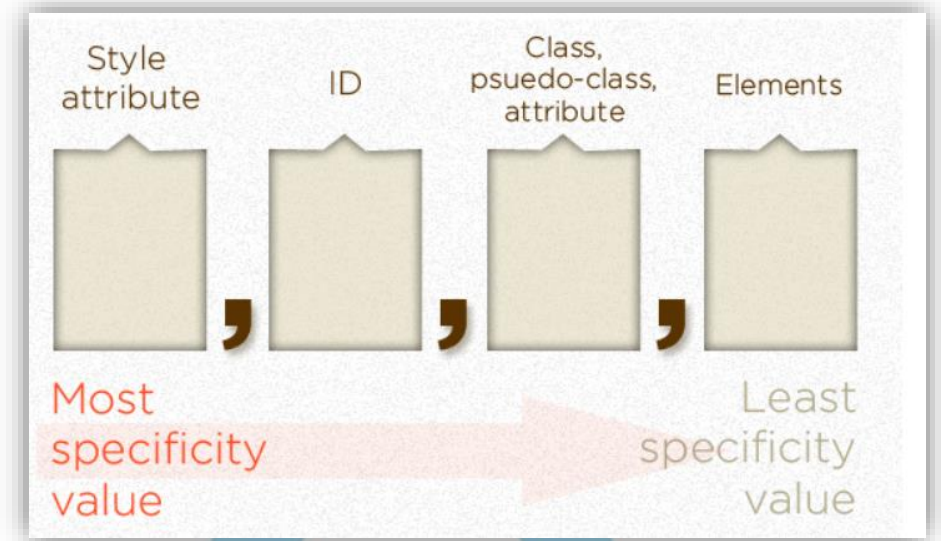
- Type (element) selectors: p, body
- Class selectors: .className
- ID selectors: #mainDiv
- Multiple selectors: A, B
- Descendant: A B
- Child: A > B
- https://developer.mozilla.org/en-US/docs/Glossary/CSS_Selector

Let's
Code

Specificity

- The most “specific” definitions apply
 - Inline styles trump stylesheets
 - Ids trump classes
 - Etc.

- https://www.w3schools.com/css/css_specificity.asp
- <https://css-tricks.com/specifics-on-css-specificity/>
- <https://specificity.keegan.st/>



Box Model

- Block Elements
 - div, h1-6, p, header, footer, section
 - Margin, padding, height effective
- Inline elements
 - span, a, img
 - Vertical settings have no effect
- [Inline-block elements](#)
 - select, button
- Change using the *Display* property



Demo

Position

- Default
 - Left to right, top to bottom, in order they appear in the HTML document
- Relative
 - Relative to where it would otherwise be positioned in the normal flow
 - *top, right, bottom, and left*
- Absolute
 - Place the element relative to the parent ancestor
 - Elements are removed from the flow of the page.
 - Setting both *top* and *bottom*, or both *left* and *right*, you can "stretch" an element's dimensions.
- Fixed
 - Relative to the browser window
 - Does not scroll with the page.



Demo

Visual Studio Code (VS-Code)

- Open the *folder* “With Code”
- Install Live Server extension
- Running Live Server
- Updating code
- Stopping Live Server



Let's
Code

CSS Variables (bonus 😊)

```
:root {  
  --gutter: 20px;  
  --blue: ■ #00adee;  
  --green: ■ #8cc36f;  
  --bg-color: □ rgba(192, 192, 192, 0.1);  
}  
  
body {  
  background-color: var(--bg-color);  
}
```