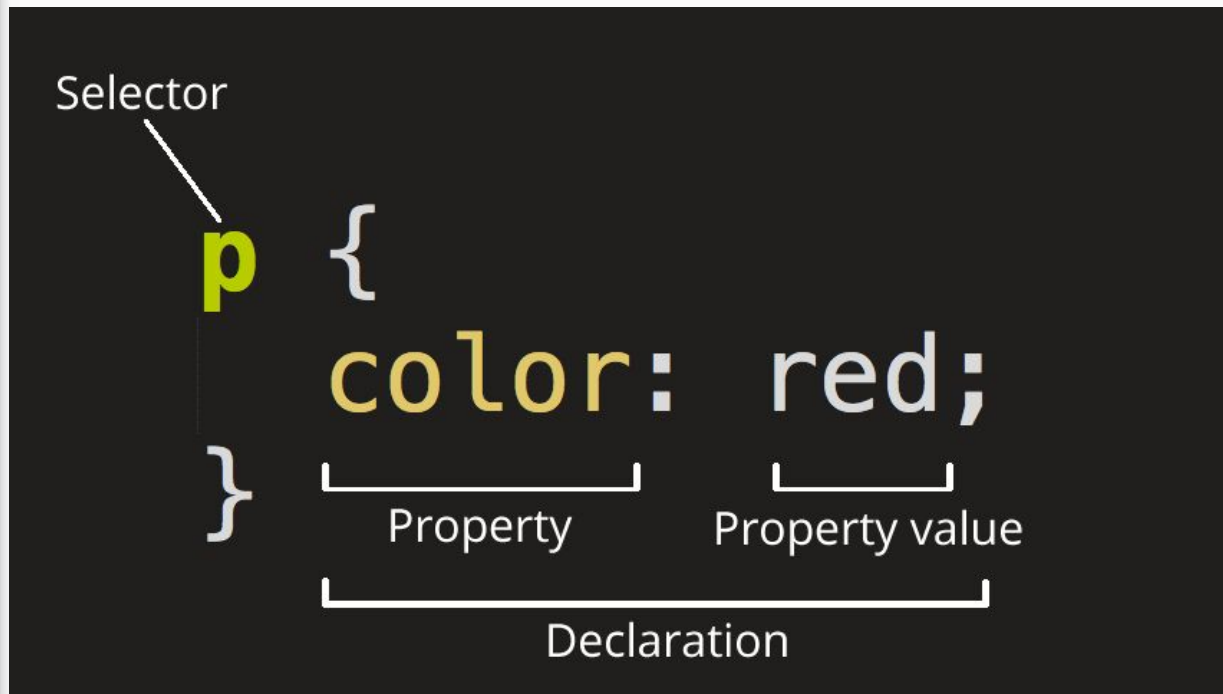# Intro to CSS

What is CSS?

- **C**ascading **S**tyle **S**heets specify how a web page looks. Allow you to apply styles selectively to elements in HTML.
- CSS let's you adjust layout and formatting such as line spacing, justification, borders, colors, and whitespace.
- Used to define rules on the specific styles that should be applied to the page.

CSS Syntax

[CSS rules](#) consist of an HTML element -- the selector -- followed by a style property and value to make up the CSS rule *declaration*

Let's say we wanted to make all `<p>` elements the color red...

CSS Ruleset

Important parts of the CSS ruleset:

- After the HTML selector the ruleset is wrapped in curly braces { }
- In the declaration, the property and the property value must be separated by a colon :
- Every declaration must be terminated with a semicolon before starting the next declaration ;

```css
p {

  color: red;

  background-color: blue;

  width: 500px;

}
```

How to apply CSS to an HTML file

CSS rules can be defined in the `<head>` element of an HTML page, in the `style` attribute of an HTML element, or in a *.css* document pulled in by the HTML file.

The <head>

CSS rules can be placed in the <head> element of an HTML file, wrapped in a <style> element. This method is called an *internal stylesheet*

```html
<!-- Pretend there is more going on up here -->

<head>

    <style>

        p {

            color: red;

            background-color: blue;

            width: 500px;

        }

    </style>

</head>

<!-- Pretend there is more going on down here -->
```

Inline `style`

CSS rules can also be added to an HTML element *inline* via the element's `style` attribute

```
<p style="color: red; background-color: blue; width: 500px;">Hello, World!</p>
```

When inlining styles, you do not need to wrap the ruleset in curly braces, but colons and semicolons are still in effect.

In a separate *.css* file

You can create a separate *.css* stylesheet that gets referenced in your HTML file via the <link> element. This lets the browser know where to find the CSS file.

```
<!-- in index.html -->

<head>

  <meta charset="utf-8">

  <title>Test CSS</title>

  <link rel="stylesheet" href="style.css">

</head>
```

```
/* in a file named style.css */

p {

  color: red;

  background-color: blue;

  width: 500px;

}
```

CSS Selectors

There are many ways to specify a CSS selector:

- Element/type selectors
- `class` or `id` of the element
- Attribute *
- Pseudo classes and pseudo elements *
- Combinators * and multiple selections

* will cover next lecture

Type Selectors

Type selectors match the selector name with the given HTML element name. Type selectors match all HTML elements of the selector name.

```css
/* matches all paragraph elements */

p {

  color: red;

  font-family: Arial;

}


/* matches all h2 elements */

h2 {

  color: blue;

  font-family: Georgia;

}
```

Selecting multiple elements

To select multiple elements for the same ruleset, separate the different elements with a comma ,

```css
h1, h2, h3 {

  color: red;

  background-color: blue;

  width: 500px;

}


p,

li {

  color: red;

  background-color: blue;

  width: 500px;

}
```

Class selectors

Class selectors match HTML elements via their `class` attribute.

- `class` value is decided by the developer.
- `class` selectors begin with a period `(.)`
- An HTML element can have several class attribute values.
- Classes can be used more than once in the HTML file.

```css
/* style.css file */

.red { color: red; }

.short-desc { width: 300px }
```

```html
<!-- index.html file →

<h1 class="red">Greetings</h1>

<p class="red">Hello, World</p>

<p class="red text-desc">This one has two classes!</p>
```

ID selectors

ID selectors match HTML elements via their `id` attribute.

- `id` value is decided by the developer.
- `id` selectors begin with a hash symbol (#)
- An HTML element can have several ID attribute values.
- An `id` can only be used once in an HTML document - they are unique identifier

```css
/* style.css file */

.red { color: red; }

.short-desc { width: 300px; }

#greet-title { text-decoration: underline; }
```

```html
<!-- index.html file -->

<h1 id="greet-title" class="red">Greetings</h1>

<p class="red">Hello, World</p>

<p class="red text-desc">This one has two classes!</p>
```

Common CSS properties

There are hundreds of CSS properties available for you to use -- but we won't be going over all of them. Getting comfortable in writing CSS involves discovery of the different properties and how they behave. Here is a short list of useful properties.

font-family

font-size

width

height

color

background-color

border

text-align

text-decoration

padding

margin

CSS Validation

Just like with HTML validation, we can have a service check to make sure our CSS code is valid and error free using the W3C CSS Validation Service

Helpful Resources

MDN Introduction to CSS

A Guide to HTML5 and CSS3: Part 4 - The Basics of CSS

Learning PHP, MySQL and Javascript: Chapter 18