CSS Selectors

CCAPDEV T1 AY 2022-2023

Outline

- CSS Structure
- CSS Selector Basics
- CSS Combinators
- CSS Attribute Selector
- Pseudo-classes and Pseudo-elements

CSS Structure

```
selector
{
        property : value ;
}
```

CSS Selectors

```
element
{
        property : value ;
}
```

HTML elements – such as `<p>`, `<h1>`, `<a>`, `<div>`

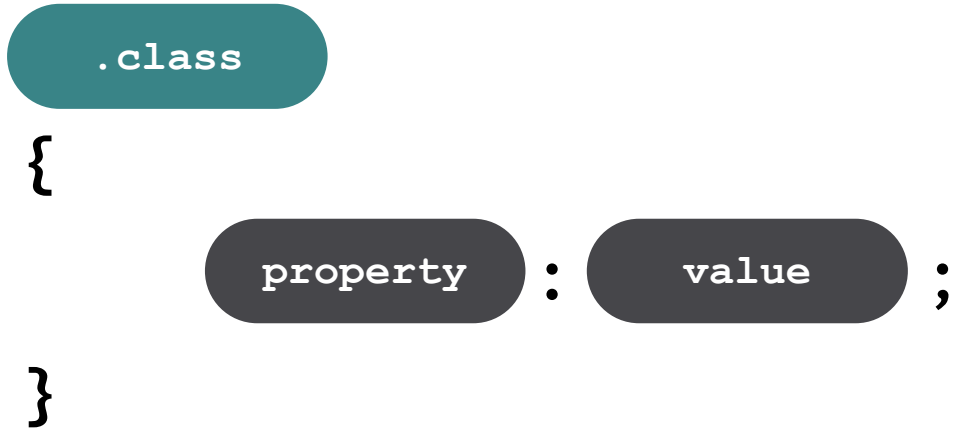
CSS Selectors

HTML Element Example

```
body {
      background-color: silver;
}

h1 {
      color: red;
}
```

CSS Selectors



Class – The value of the **`class`** attribute assigned to an element. This is denoted by a **`.`** before the value

CSS Selectors – Class

HTML

```
<p class="comment">
        Sample comment
</p>
```

CSS

```
.comment {
        color: blue;
}
```

CSS Selectors – Element + Class

HTML

```
<h3 class="comment">
        My name
</h3>
<p class="comment">
        Sample comment
</p>
```

CSS

```
p.comment {
        color: blue;
}
```

CSS Selectors – Multiple Classes

HTML

```
<p class="center bold">
        Text in center and bold
</p>
```

CSS

```
.center {
        text-align: center;
}

.bold {
        font-weight: bold;
}
```

CSS Selectors

```
#id
{
        property : value ;
}
```

ID – The value of the **`id`** attribute assigned to an element. Unlike the class, id must be ***unique***. Denoted by **#**.

CSS Selectors – ID

HTML

```
<div id="menu">
        <ul>
        <li>Home</li>
        <li>Profile</li>
        </ul>
</div>
```

CSS

```
#menu {
        background-color:black;
}
```

ID vs Class

- **IDs (#)** are unique and can only be used once in an HTML file.
- **Classes (.)** can be used as many times as needed. Good for reusability across elements.

NOTE

You can declare values for both id and class attributes for all HTML tags.

CSS Selectors

```
selector1 , selector2
{
    property : value ;
}
```

Selects all occurrence of **selector1** and **selector2**

CSS Selectors – Multiple Elements

HTML

```
<h3>
        My name
</h3>
<p>
        Sample comment
</p>
```

CSS

```
h3, p {
        font-weight: bold;
}
```

CSS Selectors – Multiple Classes

HTML

```
<h3 class="name">
        My name
</h3>
<p class="comment">
        Sample comment
</p>
```

CSS

```
.name, .comment {
        font-weight: bold;
}
```

CSS Selectors – Multiple IDs

HTML

```
<div id="title">
        ...
</div>
<p id="about">
        ...
</p>
```

CSS

```
#title, #about {
        background-color:black;
}
```

CSS Selectors – Multiple Selectors

HTML

```
<div id="title">
        ...
</div>
<div class="thread">
        ...
</div>
```

CSS

```
#title, .thread {
        background-color:black;
}
```

Combinators

CSS Selectors with relationships

CSS Combinators

```
selector1 selector2
{
    property : value ;
}
```

Selects all occurrence of **selector2** inside **selector1** elements.

This is also known as the **descendant selector**.

CSS Combinators – Descendant Selector

HTML

```
<div>
        <p> Text1 </p>
        <b>
                <p> Text2 </p>
        </b>
</div>
```

CSS

```
div p {
        color: blue;
}
```

CSS Combinators

```
selector1 > selector2
{
    property : value ;
}
```

Selects all occurrence of **`selector2`** where the parent is **`selector1`** elements.

This is also known as the **child selector**.

CSS Combinators – Child Selector

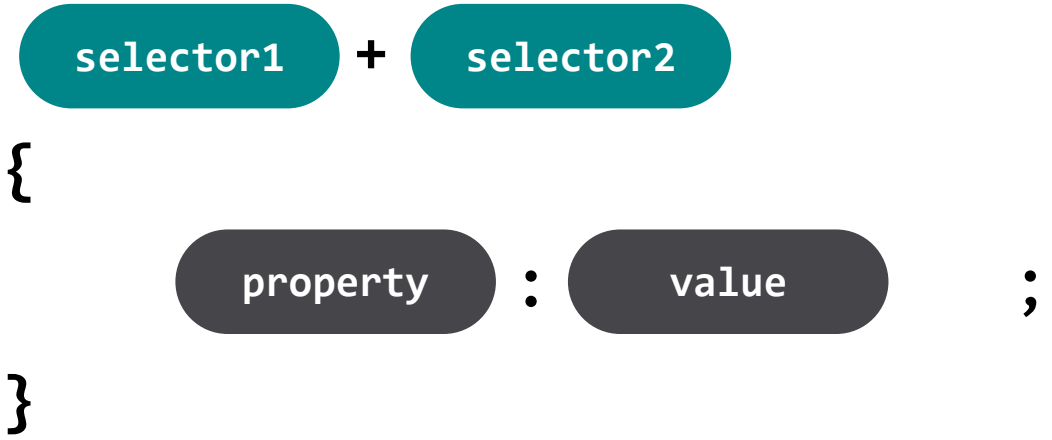
HTML

```
<div>
        <p> Text1 </p>
        <b>
                <p> Text2 </p>
        </b>
</div>
```

CSS

```
div>p {
        color: blue;
}
```

CSS Combinators



Selects all occurrence of **`selector2`** placed right after **`selector1`** elements.

This is also known as the **adjacent sibling selector**.

CSS Combinator – Adjacent Sibling Selector

HTML

```
<h1> Title </h1>
<p> Some Text </p>
<h2> Subtitle </h2>
<p> Some new text </p>
```

CSS

```
h1 + p {
        color: silver;
}
```

CSS Attribute Selectors

```
selector1 ~ selector2
{
    property : value ;
}
```

Selects all occurrences of **`selector2`** that follow a **`selector1`** element.

This is also known as the **general sibling selector**.

CSS Selectors – General Sibling Selector

HTML

```
<h1> Title </h1>
<p> Some Text </p>
<h2> Code and Text </h2>
<pre> some code </pre>
<p> Some new text </p>
```

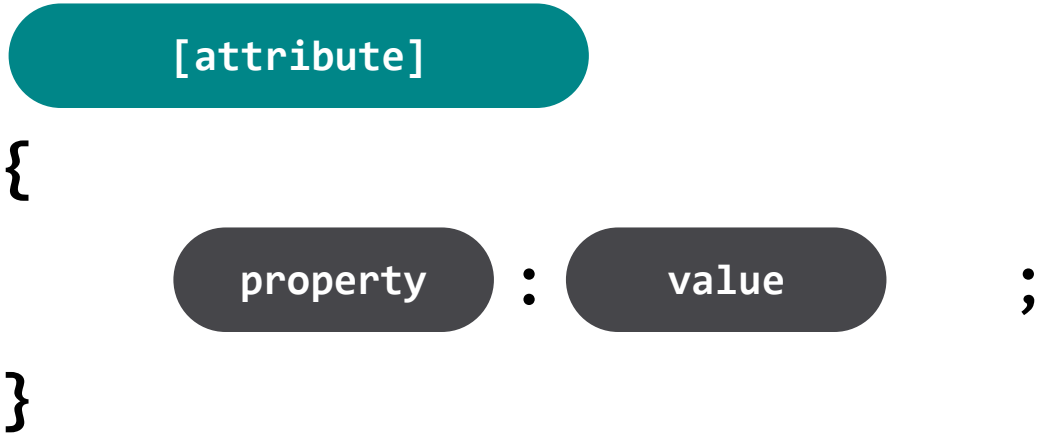
CSS

```
h1 ~ p {
        color: green;
}
```

Attribute Selectors

Making use of an element’s attributes

CSS Attribute Selectors



Selects all elements with the target **`attribute`**.

CSS Selectors – Attribute Selector 1

HTML

```
<p id="title">
        Some title
</p>
<p>
        Some text
</p>
```

CSS

```
p[id] {
        font-weight: bold;
}
```

CSS Attribute Selectors

```
[attribute=val]
{
    property : value ;
}
```

Selects all elements with the **`attribute`** equal to **`val`**.

CSS Selectors – Attribute Selector 2

HTML

```
<p id="title">
        Some title
</p>
<p id="content">
        Some text
</p>
```

CSS

```
p[id="content"] {
        font-weight: bold;
}
```

CSS Attribute Selectors

```
[attribute^=val]
{
    property : value ;
}
```

Selects all elements with the **attribute** starting with the substring **val**.

CSS Selectors – Attribute Selector 3

HTML

```
<div id="menu">
        ...
</div>
<div id="list-groups">
        ...
</div>
<div id="list-groups">
        ...
</div>
```

CSS

```
div[id^="list"] {
        background-color:black;
}
```

CSS Attribute Selectors

```
[attribute$=val]
{
    property : value ;
}
```

Selects all elements with the **`attribute`** ending with the substring **`val`**.

CSS Selectors – Attribute Selector 4

HTML

```
<div id="names">
        ...
</div>
<div id="addresses">
        ...
</div>
<div id="footer">
        ...
</div>
```

CSS

```
div[id$="es"] {
        background-color:black;
}
```

CSS Attribute Selectors

```
[attribute*=val]
{
    property : value ;
}
```

Selects all elements with the **`attribute`** containing with the substring **`val`**.

CSS Selectors – Attribute Selector 5

HTML

```
<div id="lastname">
        ...
</div>
<div id="nickname">
        ...
</div>
<div id="footer">
        ...
</div>
```

CSS

```
div[id*="na"] {
        background-color:black;
}
```

Attribute Selector Summary

Symbol	Purpose
=	equality (words that is equal to the substring)
^=	prefix (words that start with the substring)
*=	infix (words that have the substring)
$=	suffix (words that end with the substring)

Pseudo-class and Pseudo-elements

Styling the states and parts of an element

CSS Pseudo-class

```
selector : pseudo-class
{
    property : value ;
}
```

Pseudo-class denotes a special state of an element. Refer to MDN Docs for the full list of pseudo-classes.

https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes

CSS Pseudo-class

HTML

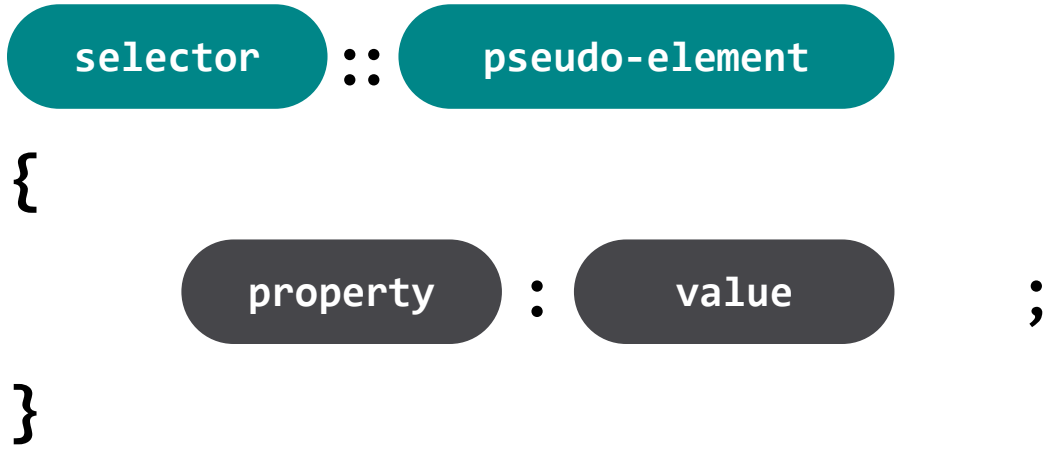
```
<a href="#top">
        Some link
</a>
```

CSS

```
a:hover {
        color: orange;
}

a:visited {
        color: red;
}
```

CSS Pseudo-element



Pseudo-elements represent a special part of an element.
Refer to MDN Docs for the full list of pseudo-elements.

https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-elements

CSS Pseudo-element

HTML

```
<p>
  This is one very long
  paragraph which can span
  multiple lines in the screen.
  The rest will not be styled.
</p>
```

CSS

```
p::first-line{
        color: green;
}
```

CSS Selectors

CCAPDEV T1 AY 2022-2023