

# Survey of CSS

## Relevant Links

- Semantic Code<sup>1</sup>
- devdocs<sup>2</sup>
- List of CSS properties<sup>3</sup>
- Learn HTML&CSS<sup>4</sup>
- MDN's links for learning CSS<sup>5</sup>
- "Can I use" site<sup>6</sup>
- Free online books<sup>7</sup>

## Notes

CSS stands for *Cascading Style Sheets*.

CSS is used to control the *appearance* of the web page.

Here are some key notions, before we look at some examples:

**rules** A rule consists of a "selector" and series of property-value specifications. It indicates that those elements that match the selector should have those properties set to those values.

**selectors** Selectors<sup>8</sup> "target" certain elements in the page. For instance a selector could say: "I want to look at the li elements that have a class of todo and are contained inside an element with an id of main."

**properties and values** CSS properties specify the appearance of elements. There is a long list of properties<sup>9</sup>, and each property accepts a specific set of values. Each property-value pair must end with a semicolon.

**specificity** Each selector has a *specificity*<sup>10</sup> value, indicating how "precisely" it is pinpointing the elements. For instance, targetting a specific id is more "specific" than targetting a specific class, which is more "specific" than targetting a specific HTML tag.

---

<sup>1</sup><https://boagworld.com/dev/semantic-code-what-why-how/>

<sup>2</sup><http://devdocs.io/>

<sup>3</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>

<sup>4</sup><http://learn.shayhowe.com/html-css/>

<sup>5</sup><https://developer.mozilla.org/en-US/learn/css>

<sup>6</sup><http://caniuse.com/>

<sup>7</sup><https://github.com/vhf/free-programming-books/blob/master/free-programming-books.md#html-css>

<sup>8</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/Reference#Selectors>

<sup>9</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>

<sup>10</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity>

If two rules try to apply the same property to an element, the rule with higher specificity wins. If they have equal specificity, the rule that appears later in the file wins.

**inheritance** Many of the properties are *inherited*<sup>11</sup> from a parent node to its children. This behavior lends cascading style sheets their name.

**box model** The box model<sup>12</sup> specifies how different properties specifying dimensions of elements should behave. It consists of *content width*, *padding*, *border* and *margins*.

**layout mode** At any given time, depending on the element and its settings, the browser is in one of four main *layout modes*<sup>13</sup>: *block*, *inline*, *table* and *positioned*.

**floats** Floats<sup>14</sup> are a difficult concept to work with at first. A “floating” element is taken out of the normal flow and placed along the left or right side of the page, while other elements wrap around it.

There are too many properties to list here, but we will list the **various selectors**:

**aTag** An HTML tag when used as a selector targets all elements in the page with that tag.

**#anId** Targets only the element with that specific id.

**.aClass** Targets all elements with this class.

**sel1sel2** Putting selectors right next to each other indicates that they should all apply to an element for it to be targetted. For example `div#main` targets only a `div` element with an id of `main`, and nothing else.

**sel1 sel2** A space between two selectors indicates that the second selector should target descendants of elements matched by the first selector. For instance `.todo p` targets paragraph elements only when they appear within an element with class “todo”.

**sel1, sel2** A comma between two selectors indicates that this rule should apply if either selector applies.

**sel1 > sel2** Targets elements that match the second selector, that are children (immediate descendants) of elements that match the first selector.

**sel1 ~ sel2** Targets elements that match the second selector, that are siblings of elements that match the first selector.

**sel1 + sel2** Targets elements that match the second selector, that are adjacent siblings (immediately follow) of elements that match the first selector.

**[attr=value]** Attribute selectors target elements with a specific value for a specific attribute. There are a number of variants<sup>15</sup> depending on the relation between the specified value and the element’s value.

---

<sup>11</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/inheritance>

<sup>12</sup>[https://developer.mozilla.org/en-US/docs/Web/CSS/box\\_model](https://developer.mozilla.org/en-US/docs/Web/CSS/box_model)

<sup>13</sup>[https://developer.mozilla.org/en-US/docs/Web/CSS/Layout\\_mode](https://developer.mozilla.org/en-US/docs/Web/CSS/Layout_mode)

<sup>14</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/float>

<sup>15</sup>[https://developer.mozilla.org/en-US/docs/Web/CSS/Attribute\\_selectors](https://developer.mozilla.org/en-US/docs/Web/CSS/Attribute_selectors)

**::pseudo** There are a few pseudo-elements<sup>16</sup> that can be targetted this way, for instance the first line in a paragraph or the first letter in a paragraph.

**:pseudo** These are pseudo-classes<sup>17</sup> and represent specific behavior of the targetted elements. For instance, you can use `:hover` to indicate something that should happen only when the element is hovered over by the mouse, you can use `:visited` to highlight links that have already been followed, and so on.

For an example of how a CSS stylesheet might look like, have a look at the CSS file<sup>18</sup> used for our class notes.

---

<sup>16</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-elements>

<sup>17</sup><https://developer.mozilla.org/en-US/docs/Web/CSS/pseudo-classes>

<sup>18</sup><https://github.com/skiadas/skiadas.github.io/blob/master/css/course.css>