

CSS

CSS stands for Cascading Style Sheet. It is designed primarily to enable separation of document content (written in HTML) from document presentation. CSS is used for styling the UI components. CSS is not case sensitive.

Example of CSS syntax:

```
H1 { color: #f00; font-size: 30px; }
```

Text highlighted in yellow is the selector. Text highlighted in green is the property. Text highlighted in blue is the value. Property + Value together is known as declaration.

Declarations are enclosed within { } and each declaration ends with a ;. CSS does not care about white space.

Style can be applied to any HTML page in three ways:

1. **Inline styling** using the style attribute of any element.

```
<div style="color: #fff;">Test</div>
```

2. Writing style inside the **style** section of **HTML head**.

```
<style>
    body { color: blue; }
</style>
```

3. External style sheet by using the **link** element in **HTML head**, and linking the css file using the **href** attribute of link.

```
<link href="style.css" rel="stylesheet" type="text/css" />
```

id and **class** attribute can be applied to almost every HTML element. **id** is applicable for just a single HTML element whereas **class** is applicable multiple HTML elements.

Example of id usage in style:

```
<style>
    #Test { color: blue; }
</style>
```

```
<div id='Test'>My Test ID</div>
```

id value should be case sensitive.

Example of class usage in style:

```
<style>

    .Test { color: blue; }

</style>

<p class="Test">Paragraph</p>

<div class="Test">Hello</div>
```

Comments in CSS are used as **/* */**

Background image can be used as

```
div { background-image: url('background.gif'); height: 60px; }
```

background-repeat: no-repeat; prevents repeating of background image.

background-attachment: fixed; prevents the background image to move with the page.

background-position: center center; or **background-position: 100px 500px;** is used for placing background image.

Important font properties are **font-size, font-family, font-style, font-weight.**

Important text properties are **text-align, text-decoration, text-indent, text-overflow, text-transform.**

LoVe/HaTe rule should be followed when styling links. It stands for **:link :visited :hover :active** pseudo's in that order.

```
div:hover {border:3px solid red;}
```

Lists can be styled by using **list-style-type** in the **style** attribute.

```
<ul style="list-style-type:circle;"> <li> Default </li> </ul>
```

Images can be used in lists as shown below:

```
li { background-image: url('star.png'); }
```

Margin is spacing around the element where as padding is spacing inside the element.

HTML elements can be displayed inline or as a block using **display** in **style** attribute.

Display has following values: none, block, inline, inline-block

Using display: none; hides the HTML element.

Using display: block; displays that HTML element in a single row.

Using display: inline; displays the HTML element inline without any width applied to it.

Using display: inline-block; displays the HTML element inline and with the specified width applied to it.

Overflow property can be used to customize the behaviour of any over flown HTML content.

It accepts following values: visible, hidden, auto, scroll, inherit. Visible is the default option.

Grouping and nesting can be used to reduce the number of lines of CSS code.

Grouping is useful when some HTML elements share similar style.

`#MyID, .blue, h1 { color: blue; }` is an example of grouping. Selectors are separated by comma.

Nesting is useful to change style of a specific HTML element, having a class, and some id. It leverages the understanding of Document Object Model.

`div button span { border: 2px solid red; }` gets applied to the below HTML element.

```
<div><button><span>another child</span></button></div>
```

For nesting, selectors are separated by a space.

For guaranteed width of an HTML element, use max-width and min-width.

For positioning an HTML element, use position:absolute OR position:relative OR position:fixed.

Z-index can be used to overlay an element on top of another element. Higher the z-index value, higher the status for that element.

Elements can be made to float either right or left.

```
<style>
    .star { border: 1px solid red; float:right; height: 50px; width:50px; }
</style>
```

Floating element is pushed to one side (left/right) and allows other elements to float around it.

To prevent elements from floating around an HTML element, **clear** should be used.

Clear accepts one of the following values: **none, left, right, both, inherit**

:first-child pseudo will match just the first child of an HTML element.

Parent:first-child childElement { /* Styling */ }

:first-letter pseudo is similar to :first-child element and gets applied to the first letter of an element.

```
div:first-child {border: 1px solid red;}
```

```
div > u:first-child {border: 1px solid red;} --- style gets applied to the first underline element in div.
```

```
div:first-child u {border: 1px solid red;} --- style gets applied to all the underline sections of the first child element.
```

```
p:first-letter { font-size: 32px; }
```

```
body:first-line { word-spacing: 15px; } --- this style gets applied to the first line of the body only
```

```
textarea:focus { border: 1px solid red;}
```

```
strong:before { content: "NOTICE: "; } --- will add that content before the strong element
```

```
span:after { content: "bye"; } --- will add that content after the span element
```

Text can be aligned using text-align and vertical-align.

Sprites are used for faster loading of website. Sprite is a bundle of images in one larger image.

One particular image in a sprite can be identified with a style as follows:

```
<style>

.cards  {

    Background-image: url ('sprite.png');

    Background-position: -17px -373px;

    Background-repeat: no-repeat;

    Width: 16px;

    Height: 15px;

</style>
```

Background position takes only negative x and y values. Negative x is the pixel distance from left and negative y is the pixel distance from right.

Sprite is a .png file and looks like below:



Opacity of an element can be changed using an opacity property. It accepts value from 0.0 to 1.0. Any Internet Explorer browser before IE9 uses filter: alpha(opacity=50).

```

```

```

```



Media elements can be styled differently for different kinds of devices as:

```
<style>

@media screen { /* Computer screen */ }

@media handheld { /* Mobile Devices */ }

@media print { /* Printable page that are usually bland page for saving printer's ink */ }

@media tv { /* Television type Devices */ }

</style>
```

Styling for a particular attribute and value combination can be applied as:

```
<style>

    [lang=abc] { border: 1px solid red; } /* Styles only elements having lang attribute and its value as abc*/

    [lang] { border: 1px solid red; } /* Styles any element having lang attribute */

</style>

<div lang="abc">test</div>
```

Prefixes for different browsers are:

-ms- for Microsoft IE

-moz- for Mozilla Firefox

-o- for Opera

-webkit- for Chrome and Safari

Styling that supports different browsers can be applied by appending prefixes to the property at the beginning:

```
-webkit-transition: -webkit-box-shadow 4s;
```

Default option should always be specified at the end.

To ensure that a particular style gets applied and also prevent that style from being updated later, **!important** is used.

```
<style>  
Body{ font-weight: bold !important;}  
Body{ font-weight: normal;}  
</style>
```

! Important adds more weight to the style declaration, telling the browser that it is more important than any feature that tries to overwrite this. This should only be used as a backup plan when no other option is available.

CSS3

CSS3 is comprised of multiple files called modules. Modules allow each browser to interpret different declarations, such as border-radius.

Border-radius property can be used to get a curved border.

```
Border-radius: 10px;
```

Border-top-right-radius, Border-top-left-radius, Border-bottom-right-radius, Border-bottom-left-radius, are also valid selections. Same thing can also be achieved using just the border radius property as:

```
Border-radius: 10px 0px 20px 10px; /* top-left top-right bottom-right bottom-left  
(clockwise direction) */
```

Images can be applied to border as:

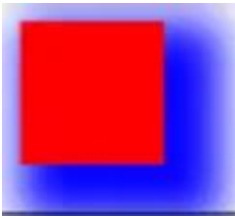
```
Border-image: url(BorderImage.gif) 20% 20% 20% 20% repeat; /* top right bottom left */
```

```
Border-image: url(BorderImage.gif) 20 20 20 20; /* All values are considered in pixels,  
Never specify px here */
```

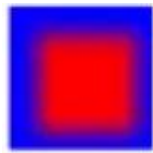
Box-shadow is used for styling shadow of a border.

box-shadow property must have horizontal and vertical values (first two values). Third value (optional) is for blurriness. Fourth value (optional) is for spread.

```
Box-shadow: 2px 2px 10px 8px blue;
```



```
Box-shadow: 2px 2px 10px 8px blue inset;
```



To make a background image spread over an entire page, we use background-size.

```
background-size: 40px 40px; /* Values in % are also accepted */
```

"background-size: contain;" will enclose the child element unlike "background-size: cover;"


```
<style>
  .small {
    background-image:url('star.png');
    background-size:40px 40px;
    background-repeat:no-repeat;
    border:1px solid red;
    width:50px;
    height: 50px;
  }
</style>
```

Displays:



```
<style>
  .big {
    background-image:url('star.png');
    background-size:200% 200%;
    background-repeat:no-repeat;
    border:1px solid red;
    width:50px;
    height: 50px;
  }
</style>
```

Displays:

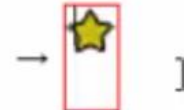


Contain will always show the background image irrespective of the size of the parent element.

```
.contain1 {
  background-image:url('star.png');
  background-size:contain;
  background-repeat:no-repeat;
  border:1px solid red;
  width:50px;
  height: 50px;
}
```



```
.contain2 {
  background-image:url('star.png');
  background-size:contain;
  background-repeat:no-repeat;
  border:1px solid red;
  width:25px;
  height: 50px;
}
```



Cover will expand or contract the background image so that it covers the entire element but will not adjust to fit in the element like it did with contain.

```
.cover1 {
  background-image:url('star.png');
  background-size:cover;
  background-repeat:no-repeat;
  border:1px solid red;
  width:50px;
  height: 50px;
}
```



```
.cover2 {
  background-image:url('star.png');
  background-size:cover;
  background-repeat:no-repeat;
  border:1px solid red;
  width:25px;
  height: 50px;
}
```



Text shadow is similar as the box shadow. IE does not support it.

Horizontal and vertical values are required. Blur and color are optional.

```
.shadow3 {
  text-shadow: 2px 2px 10px blue;
}
```

→ **Text-shadow**

@font-face is used for creating a new font family

```
<style>
  @font-face {
    font-family: myName;
    src: url('octin_spraypaint.ttf'), url('octin_spraypaint.otf');
  }
  .font {
    font-family: myName;
  }
</style>
```

Elements can be rotated as: **-prefix-transform: rotate(15deg);**

-ms-transform: rotate(15deg);

transform: rotate(15deg);

Elements can be resized using **resize** property. Valid values are horizontal, vertical, both, and none.

```
.horizontal {
  resize: horizontal;
  overflow: auto;
}
```



Reference: <https://www.udemy.com/thecompletewebdeveloper/learn/#/>