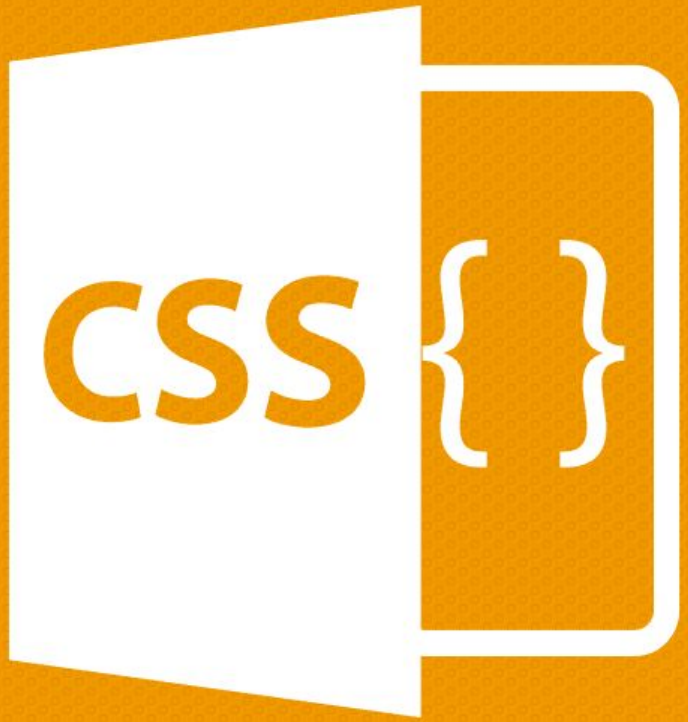

CSS
{ }

Introduction

- CSS is an acronym for Cascading Style Sheets.
- CSS is used to control the style of a web document in a simple and easy way.
- CSS is a style language that defines layout of HTML documents. For example, CSS covers fonts, colours, margins, lines, height, width, background images, advanced positions and many other things.
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- HTML can be (mis-)used to add layout to websites. But CSS offers more options and is more accurate and sophisticated. CSS is supported by all browsers today.

Advantages of CSS

- ★ CSS saves time
- ★ Pages load faster
- ★ Easy maintenance
- ★ Superior styles to HTML
- ★ Multiple Device Compatibility
- ★ Global web standards
- ★ Platform Independence

CSS Versions

Cascading Style Sheets, level 1 (**CSS1**) was came out of W3C as a recommendation in December 1996. This version describes the CSS language as well as a simple visual formatting model for all the HTML tags.

CSS2 was became a W3C recommendation in May 1998 and builds on CSS1. This version adds support for media-specific style sheets e.g. printers and aural devices, downloadable fonts, element positioning and tables.

CSS3 was became a W3C recommendation in June 1999 and builds on older versions CSS. it has divided into documentations is called as Modules and here each module having new extension features defined in CSS2.

CSS Syntax

A style rule is made of three parts −

Selector − A selector is an HTML tag at which a style will be applied. This could be any tag like <h1> or <table> etc.

Property - A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be color, border etc.

Value - Values are assigned to properties. For example, color property can have value either red or #F1F1F1 etc.

selector { property1 : value1; property2 : value2; }

Declaration Declaration

The selector points to the HTML element you want to style.

The declaration block contains one or more declarations separated by semicolons.

Each declaration includes a CSS property name and a value, separated by a colon.

A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

Example

In the following example all <h1> elements will be with a red text color:

```
<!DOCTYPE html>
<html>
	<head>
		<style>
			h1{
				color: red;
			}
		</style>
	</head>
<body>
	<h1>Hello World!</h1>
	<h1>These paragraphs are styled with CSS.</h1>
</body>
</html>
```

What if we have to apply styles to multiple HTML Elements

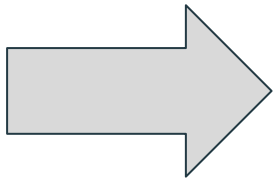
Grouping Selectors

If you have elements with the same style definitions

```
h1 {
      color: red;
}

h2 {
      color: red;
}

h3 {
   text-align: center;
   color: red;
}
```



It will be better to group the selectors, to minimize the code.

To group selectors, separate each selector with a comma.

```
h1, h2, h3 {

    color: red;
}
```

What if we have to apply styles to All HTML Elements

Universal Selectors

Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type −

```
* {
    color: #000000;
  }
```

Class Selectors

All the elements having that class will be formatted according to the defined rule.

To select elements with a specific class, write a period (.) character, followed by the name of the class.

```
.error {
   color: #FF0000;
}
```

ID Selectors

All the elements having that id will be formatted according to the defined rule.

The id of an element should be unique within a page, so the id selector is used to select one unique element!

To select an element with a specific id, write a hash (#) character, followed by the id of the element.

```
#black {
    color: #000000;
}
```

Three Ways to Insert CSS

- Inline style
- Internal style sheet
- External style sheet

Inline style

An inline style may be used to apply a unique style for a single element.

You can use style attribute of any HTML element to define style rules. These rules will be applied to that element only.

The style attribute can contain any CSS property.

```
<h1 style="color:blue;">This is a heading</h1>
```

Internal Style Sheet

- An internal style sheet may be used if one single page has a unique style.
- You can put your CSS rules into an HTML document using the <style> element.
- This tag is placed inside <head>...</head> tags.
- Rules defined using this syntax will be applied to all the elements available in the document.

```
<style type = "text/css" media = "all">
        body {
           background-color: linen;
        }
</style>
```

External Style Sheet

CSS Backgrounds

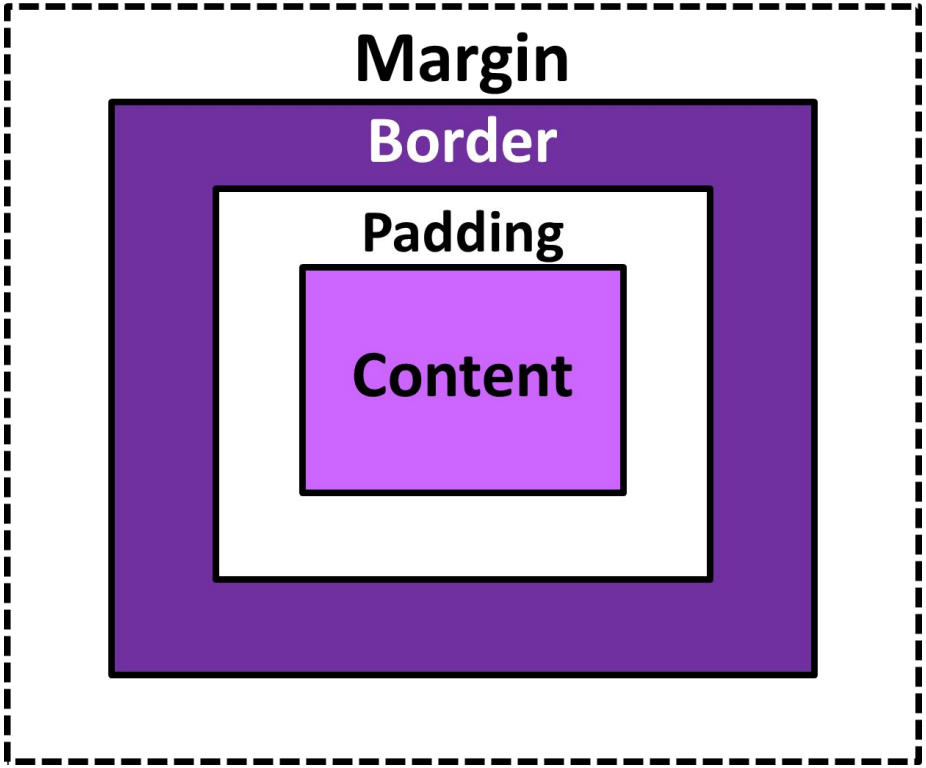
CSS Borders

CSS Outline

CSS has the following outline properties:

- outline-style
- outline-color
- outline-width
- outline-offset
- outline

CSS Box Model



CSS Margin

CSS Padding

CSS Height / Width

The height and width properties are used to set the height and width of an element.

```
div {
        height: 200px;
         width: 50%;
}
```

CSS Text Formatting

- Text Color
  - `color: blue;`
- Text Alignment
  - `text-align: center; [center/left/right/justify]`
- Text Decoration
  - `text-decoration: none; [overline/line-through/underline]`
- Text Transformation
  - `text-transform: uppercase; [lowercase/capitalize]`
- Text Indentation
  - `text-indent: 50px;`

CSS Text Formatting

- Letter Spacing
  - `letter-spacing: 3px;`
- Word Spacing
  - `word-spacing: 10px;`
- Line Height
  - `line-height: 0.8;`
- Text Direction
  - `direction: rtl;`
- Text Shadow
  - `text-shadow: 3px 2px 5px red;`
  - `text-shadow: horizontal verticle blur color;`

Css Links

- `a:link` - a normal, unvisited link
- `a:visited` - a link the user has visited
- `a:hover` - a link when the user mouses over it
- `a:active` - a link the moment it is clicked

CSS lists

```
list-style-type: circle; [square / upper-roman / lower-alpha ]

list-style-image: url('sqpurple.gif');
```

Css Display

```
display: none;
display: inline;
display: block;
```

CSS Combinators

Descendant Selector

The descendant selector matches all elements that are descendants of a specified element.

```
div p {

  background-color: yellow;

}
```

Child Selector (>)

The child selector selects all elements that are the children of a specified element.

```
div > p {
  background-color: yellow;
}
```

Adjacent Sibling Selector (+)

The adjacent sibling selector is used to select an element that is directly after another specific element.

Sibling elements must have the same parent element, and "adjacent" means "immediately following".

```
div + p {
  background-color: yellow;
}
```

General Sibling Selector (~)

The general sibling selector selects all elements that are next siblings of a specified element.

```
div ~ p {
  background-color: yellow;
}
```

Pseudo-classes

:hover
:visited
:active
:first-child
:last-child
:nth-child(n)
:not(selector)
:

Pseudo Elements

::after, ::before

```
p::after {
  content: " -  demo";
}
```

::first-letter

::first-line

::selection

Attribute Selectors

```
a[target] {
  background-color: yellow;
}

a[target="_blank"] {
  background-color: yellow;
}
```

CSS Opacity / Transparency

```
img {
  opacity: 0.5;
}
```