Web Design & Development I

IDD103

Introduction to CSS3

Prepared by Mr. Kinley Tshering

Content

What is CSS3?

- CSS (Cascading Style Sheet) is the language to style an HTML document or the appearance of the page is determined by one or more style sheets written in CSS language.
- The latest version of CSS language is CSS3
- CSS3 is built upon several modules, where each module is focused on a separate design topic.

General Style rules

- The general syntax of a CSS rule is

```
selector{
    property1: value1;
    property2: value2;
    ...
}
```

- Example:

```
h1{
    color:blue;
    font-size: 12px;
}
```

- Selector selects HTML element to be styled
- Declaration sets out how to style the selector using a property. It consist of one or more declaration separated by semicolon

Type of Style Sheets

- Embedded styles or Internal– Styles added to the head of an HTML document
- Inline styles – Styles added as element attributes within an HTML document and applied to only that particular element
- External styles – Styles created by a website author, placed within a CSS file, and linked to the page

Embedded style sheets

- Used when a single HTML document to be styled uniquely.
- The CSS rule set should be within the HTML file in the head section i.e the CSS is embedded within the HTML file.
- Example:

```
<head>
  <style>
  selector{
    property1: value1;
    property2: value2;
    ...
  }
  </style>
</head
```

Inline style

- They are styles applied directly to specific elements using the following style attribute
- Example:

```
<element style="property1:value1; property2:value2; ...">
    content
 </element>
```

- where the property: value pairs define the styles applied directly to that element

External style

- External CSS contains separate CSS file which contains only style property
- CSS property written in a separate file with .css extension and should be linked to the HTML document using link tag
- Example:

```
<head>
  <link rel="stylesheet" href="CSSFile.css"/>
</head>
```

- link tag is used to link the external style sheet with the html webpage.
- rel attribute specifies the relationship between the current document and the linked document/resource
- href attribute is used to specify the location of the external style sheet file

Writing style comments

- Comments are used to provide explanations or notes within your stylesheet and help to edit source code at a later date that are not processed by the browser.
- They are helpful for adding context to your code and for documenting your styles. CSS supports two types of comments:

```
/*
This is a comment.
It can also span across several lines.
*/
.header {
  background-color: #333;
  color: #fff;
}
```

CSS Selectors

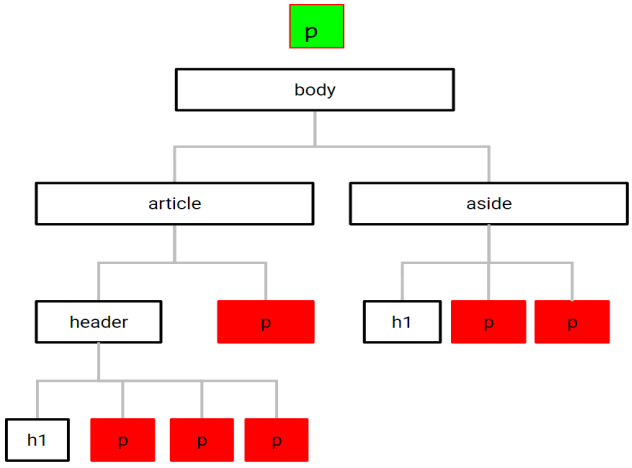
- Selector selects the HTML element(s) to be styled
- The style below rule matches every h1 element in the HTML document, regardless of the location of the h1 heading:

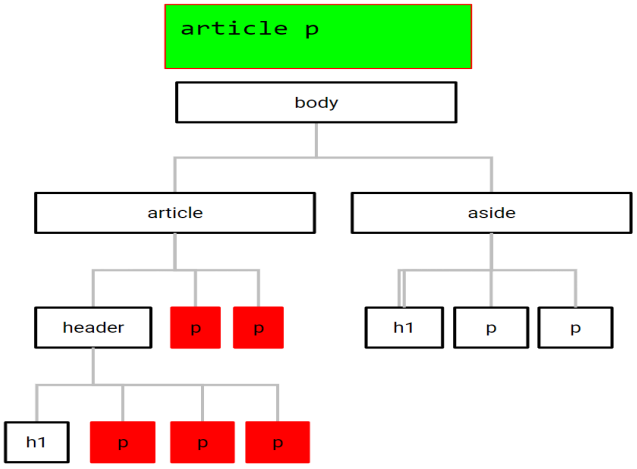
```
h1 {
  color: grey;
}
```

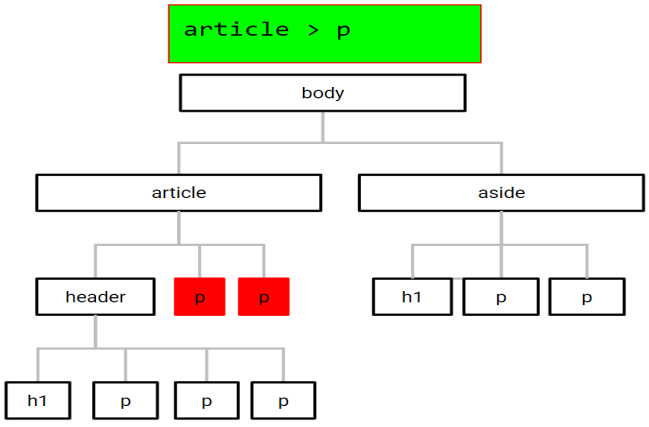
Contextual Selectors

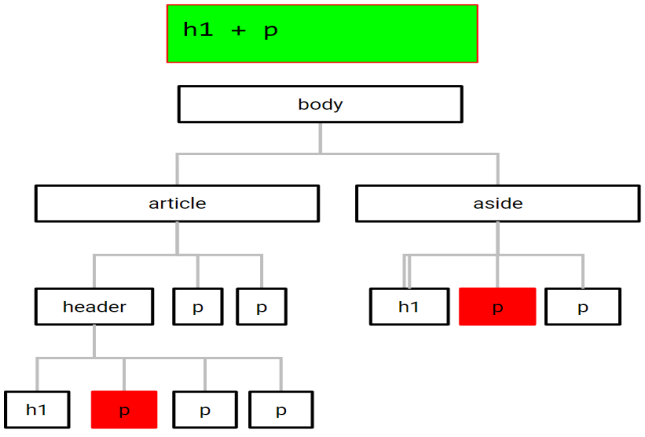
- Contextual selector specifies the context that matches a particular page element.
- Context is based on the hierarchical structure of the document.
- a parent element containing one or more child elements
- Within those child elements there can be several levels of descendant elements.

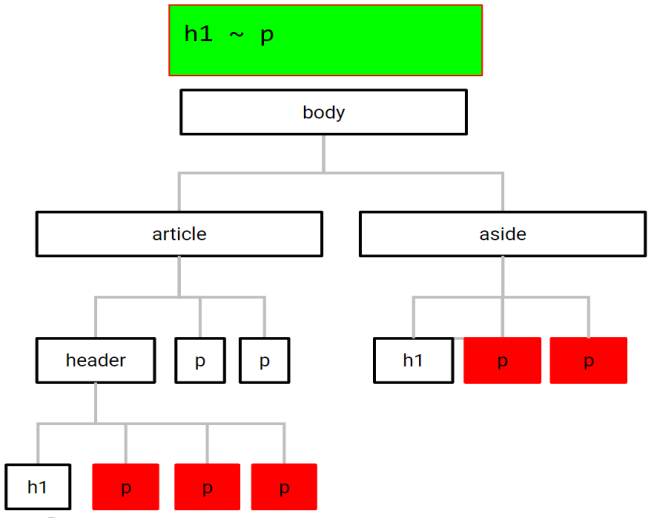
Selector	Name	Description
*	wildcard	Matches any element
elem	p	Matches the element elem located anywhere in the document
elem1, elem2, …	div,p	Matches any of the elements elem1, elem2, etc.
parent descendant	div p	Matches the descendant element that is nested within the parent element at some level
parent > child	div > p	Matches the child element that is a child of the parent element
elem1 + elem2	Sibling selector	Matches elem2 that is immediately preceded by the sibling element elem1
elem1 ~ elem2		Matches elem2 that follows the sibling element elem1

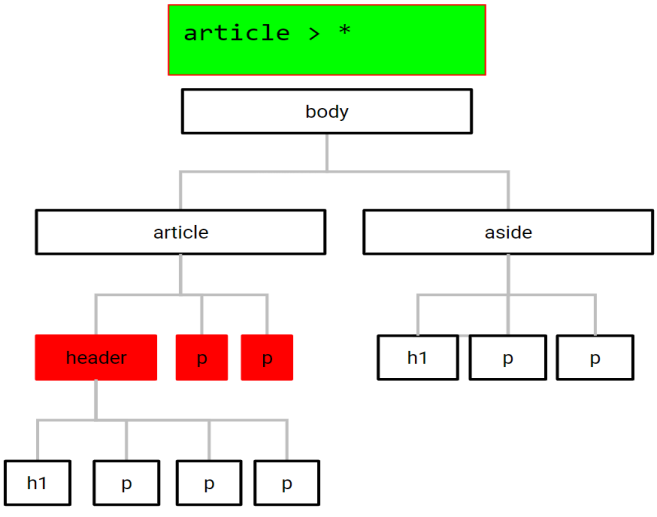

p
body
article
aside
header
p
h1
p
p
h1
p
p
p


article p
body
article
aside
header
p
p
h1
p
p
h1
p
p
p


article > p
body
article
aside
header
p
p
h1
p
p
h1
p
p
p


h1 + p
body
article
aside
header
p
p
h1
p
p
h1
p
p
p


h1 ~ p
body
article
aside
header
p
p
h1
p
p
h1
p
p
p


article > *
body
article
aside
header
p
p
h1
p
p
h1
p
p
p

Attribute Selectors

- Selectors also can be defined based on attributes and attribute values within elements.
- Two attributes, id and class, are often used to apply styles to specific elements.
- Use either the selector #id or the selector elem#id

Selector	Selects	Example	Selects
elem#id	Element elem with the ID value id	h1#main	The h1 heading with the id main
#id	Any element with the ID value id	#main	Any element with the id main
elem.class	All elem elements with the class attribute value class	p.intro	All paragraphs belonging to the intro class
.class	All elements with the class value class	.intro	All elements belonging to the intro class
elem[att]	All elem elements containing the att attribute	a[href]	All hypertext elements containing the href attribute
elem[att="text"]	All elem elements whose att attribute equals text	a[href="a.html"]	All hypertext elements containing the href attribute = a.html
elem[att~="text"]	All elem elements whose att attribute contains the word text	a[href~="html"]	All hypertext elements whose href attribute contains the word html
elem[att|="text"]	All elem elements whose att attribute value is a hyphen-separated list of words beginning with text	a[href|="html"]	All hypertext elements whose href attribute starts with the word html in a hyphen-separated list of words
elem[att^="text"]	All elem elements whose att attribute begins with text	a[href^="html"]	All hypertext elements whose href attribute begin with html
elem[att$="text"]	All elem elements whose att attribute ends with text	a[href$="html"]	All hypertext elements whose href attribute end with html
elem[att*="text"]	All elem elements whose att attribute contains the value text	a[href*="html"]	All hypertext elements whose href attribute contains the text string html

Class Selector

- The class selector is used to target HTML elements that have a specific class attribute assigned to them.
- It is denoted by a period (.) followed by the class name.
- Class selectors are particularly useful when you want to apply the same styling to multiple elements across your webpage.
- Syntax for using a class selector in CSS:

```
.className {
    /* CSS properties here */
}
```

```
<div class="className">This is an element with the class "className".</div>
```

- Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Document</title>
    <style>
        /* Targeting elements with the class "highlight" */
        .highlight {
            background-color: yellow;
            padding: 10px;
            font-weight: bold;
        }
    </style>
</head>
<body>
    <div class="highlight">This is a highlighted text.</div>
    <div>This is a regular text.</div>
</body>
</html>
```

- In the given example, any element with the class highlight will have a yellow background, extra padding, and bold text. The second div without the class won't be affected by these styles.

ID Selector

- The ID selector is used to target a single HTML element with a unique ID attribute.
- It is denoted by a hash (#) followed by the ID name.
- ID selectors are commonly used to apply specific styles or behaviors to individual elements.
- However, it's important to note that while IDs should be unique, using classes is generally more flexible for styling multiple elements with similar characteristics.
- Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Document</title>
    <style>
        #myDiv {
            background-color: lightblue;
            padding: 20px;
            font-size: 18px;
            text-align: center;
        }
    </style>
</head>
<body>
    <div id="myDiv">This is a div with an ID.</div>
</body>
</html>
```

- In the given example, the #myDiv selector in the CSS file targets the element with the ID attribute set to "myDiv".
- The CSS rules specified under this selector will apply only to that specific element.
- In this case, the background color will be light blue, there will be some padding around the content, the font size will be increased, and the text will be centered within the div.

Color in CSS

- CSS supports colors using various formats, such as named colors, hexadecimal notation, RGB values, and HSL values.

- Named Color:
  - CSS provides a set of predefined color names that you can use directly.
  - For example:

```
p {
   color: red;
}
```

- Hexadecimal Notation:
  - You can specify colors using hexadecimal notation, which consists of a hash (#) followed by a six-character code representing the red, green, and blue components of the color.
  - For example:

```
h1 {
  color: #3498db; /* Hexadecimal code for a shade of blue */
}
```

- RGB Values:
  - RGB stands for Red, Green, and Blue, and you can define colors using the amounts of each of these primary colors.
  - Each color component can range from 0 to 255.
  - For example:

```
a {
  color: rgb(255, 0, 127); /* Red with no green and some blue */
}
p{
  background-color: rgba(255, 0, 0, 0.5); /* Semi-transparent red
background */
}
```

- HSL Values:
  - HSL stands for Hue, Saturation, and Lightness.
  - Hue is represented as an angle (0 to 360), Saturation is a percentage (0% to 100%), and Lightness is also a percentage (0% to 100%).
  - For example:

```
blockquote {
  color: hsl(120, 100%, 50%); /* Fully saturated green color */
}
span {
  background-color: hsla(240, 100%, 50%, 0.7); /* Semi-transparent
blue background */
}
```

Fonts

- Typography is the art of designing the appearance of characters and letters on a page.
- Text characters are based on fonts that define the style and appearance of each character in the alphabet.
- To specify a different font for any page element use
  font-family: fonts;
- Specific font is a font that is identified by name, such as Times New Roman
- Based on a font definition file that is stored on the user's computer or accessible on the web
- Generic font describes the general appearance of the characters in the text but does not specify any particular font definition file

Generic font group

- CSS supports the following generic font groups:
  - serif is a typeface in which a small ornamentation appears at the tail end of each character. Ex. Times New Roman
  - sans-serif is a typeface without any serif ornamentation - Ex. Arial
  - monospace is a typeface in which each character has the same width - Ex. Courier New and Consolas

    - cursive is a typeface that mimics handwriting with highly stylized elements and flourishes - Ex. Brush Script
    - fantasy is a highly ornamental typeface used for page decoration; Not to be used as body text - Ex. Impact and Papyrus
- The browser can choose any font definition file for a generic font
- List specific fonts first, in order of preference, and end the font stack with a generic font.
- If the browser cannot find any of the specific fonts listed, it uses a generic font of its own choosing.

  font-family: "Times New Roman", Times, serif;
- Web safe fonts are fonts that will be displayed very similarly in all operating systems and on all devices.

Font Size

- Text size is defined using the font-size property

  font-size: size;
- Size is a CSS unit of measurement.
- Size values can be whole numbers (0, 1, 2 ...) or decimals (0.5, 1.6, 3.9 ...).
- Lengths (and widths) in CSS are expressed in either absolute units or relative units.

Relative units

- Relative units are relative to the size of other objects in a web page.
- Basic unit for devices is pixel (px), representing a dot on the screen.
- A pixel is a relative unit because the actual pixel size depends on the resolution and density of the output device.
    - Desktop monitor pixel density is about 96ppi (pixels per inch)
    - Laptops are about 100 to 135ppi
    - Mobile phones at 200 to 300ppi or more.
- Most browsers will apply a base font size of 16px to body text
- Override the default can be done using example :

  body {font-size: 10px;}

  h1 {font-size: 14px;}

Scaling Font Size

- All font sizes expressed relative to a default font size.
- There are three relative measurements used to provide scalability:

  - percentages
  - ems
  - rems.
- Percentage sets size as a percent of font size used by the containing element. For example, h1 heading is 200% of h1 parent element:
- h1 {font-size: 200%;}
- em is similar to percentage, size relative to font size of the parent.
- Thus, to set the font size of h1 headings to twice the parent elements:,

  h1 {font-size:2em;}

- Relative units cascade through the style sheet can be confusing.
- For example,
- body {font-size: 16px;}

  body > article {font-size: 0.75em;}
  body > article > h1 {font-size: 1em;}

- The font size in the article element is 75% of 16px = 12 pixels. Size of h1 headings in the article is 100% of 12px = 12 pixels.
- rem or root em unit expressed font size relative to the html element.
- Using rems, the following sets the font size of article text to 75% of 16 pixels or 12 pixels while the h1 heading size is set to 16 pixels:

  body {font-size: 16px;}
  body > article {font-size: 0.75rem;}
  body > article > h1 {font-size: 1rem;}

Fonts Style

- Font style can be specified using

  font-style: type;
  - type is normal, italic, or oblique.
- Weight of the text can be specified using

  font-weight: weight;
  - weight is the level of bold formatting applied to the text.
  - keywords bolder or lighter express the relative weight of the text
- Text-decoration property places a line under and over the element text:

  text-decoration: underline overline;
- Text-decoration has no effect on non-textual elements, e.g inline images.
- Text-transform control the case of the text

text-transform: type;

- type is capitalize, uppercase, lowercase, or none

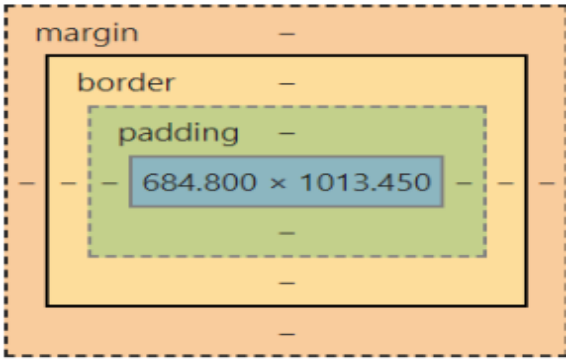
- font-variant property control variations of the text

  font-variant: type;

  - type is normal (for no variation) or small-caps (small capital letters).

Box Model

- The "box model" is a fundamental concept in CSS that defines how elements are displayed and structured on a webpage.
- It conceptualizes HTML elements as rectangular boxes, each with four layers: content, padding, border, and margin.
- These layers collectively determine the element's total size and its spacing relative to other elements.
- The box model is essential for understanding and controlling the layout and spacing of elements in a web page.

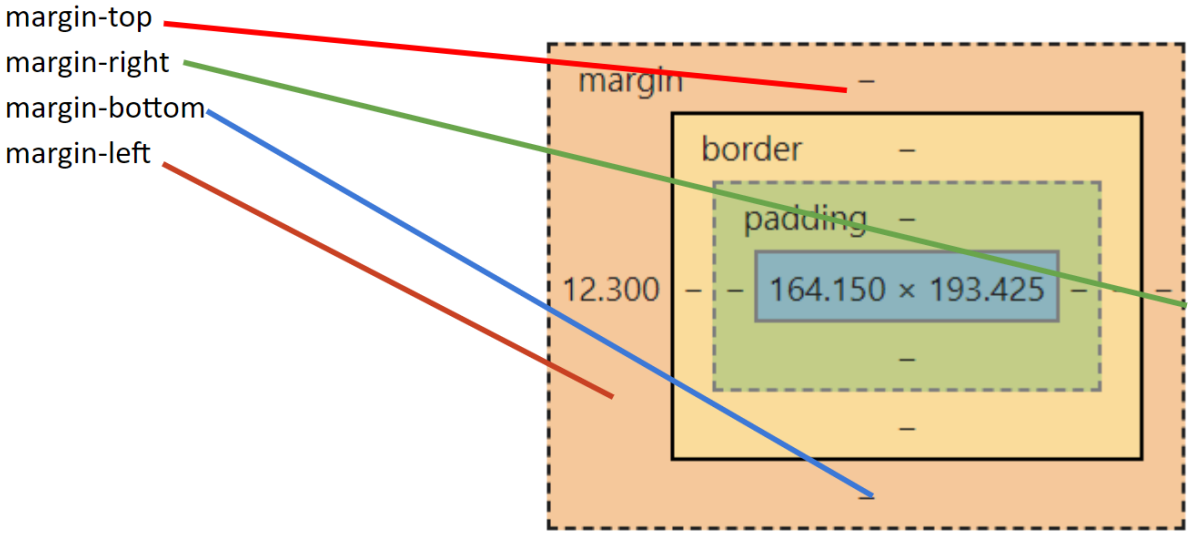


- **Content:**
  - This is the innermost layer of the element and represents the actual content, such as text, images, or other HTML elements.
  - The size of this layer is determined by properties like width and height.
- **Padding**:
  - The padding is the space between the content and the element's border. It provides an area of cushioning around the content.
  - Padding can be controlled using properties like padding-top, padding-right, padding-bottom, and padding-left.
- **Border:**
  - The border is the line that surrounds the padding and content.
  - It separates an element from its surroundings and can be styled using properties like border-width, border-style, and border-color.

- **Margin:**
  - The margin is the space outside the border, creating a gap between this element and adjacent elements.
  - Margins are used to control the spacing between different elements on the page and can be adjusted using properties like margin-top, margin-right, margin-bottom, and margin-left.

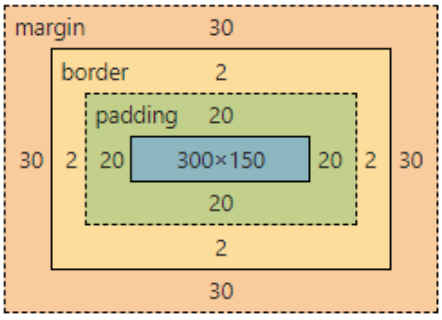
Margins property

- The margin property sets the margins for an element, and is a shorthand property for the following properties:



- Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Document</title>
    <style>
        .styled-box {
            width: 300px;
            height: 150px;
            background-color: blue;
            color: white;
            padding: 20px;
            border: 2px solid black;
            margin: 30px;
        }
    </style>
</head>
<body>
    <div class="styled-box">
        Hello, this is a styled box!
    </div>
</body>
</html>
```



The .styled-box class defines the styling for our box.
We set the width and height to create the size of the content area.
The background color is set to blue, and the text color is set to white for contrast.
A padding of 20px creates space between the content and the border.
A solid black border with a width of 2px surrounds the padding.
A margin of 30px creates space between this box and other elements on the page.

#####Thankyou#####