

Session 3 – CSS



Agenda – CSS

1. Understanding CSS
2. CSS Syntax
3. CSS Selectors
4. CSS Selectors Type
5. Combining Selectors
6. CSS Attribute Selectors
7. Attaching CSS Style
8. Display Property
9. CSS Box Model
10. margin: auto, max-width



Agenda – CSS

- 11. CSS background-color Property**
- 12. CSS Links**
- 13. CSS Units**
- 14. Position Property**
- 15. CSS Float Property**
- 16. Problem with Float**
- 17. Inheritance**
- 18. Specificity**
- 19. CSS Media Queries**



Understanding CSS

- **What is style?**
 - Style is a list of formatting instructions.
- A Cascading Style Sheet is a file with a list of formatting instructions.
- **CSS style sheets** are the modern way to control the appearance and layout of your web pages.



CSS Syntax

- A CSS rule set consists of a selector and a declaration block:

Selector

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



CSS Selectors

- **CSS selectors** allow you to select and manipulate HTML elements.
- **CSS selectors** are used to "find" (or select) HTML elements based on
 - Element/ Tag Selector
 - Id Selector
 - Class Selector



CSS Selectors Type

The element / tag selector selects elements based on the element name.

Syntax: `htmlElementName { property1 : value1 ; ... }`

The id selector uses the id attribute of an HTML element to select a specific element.

Syntax : `CSS : #idName {property1 : value1 ; ... }`

`HTML : <tag_name id="idName">... </tag_name>`

The class selector selects elements with a specific class attribute.

Syntax : `CSS : #className {property1 : value1 ; ... }`

`HTML : <tag_name id="className">... </tag_name>`



Combining Selectors

.className tagName: Note the space between .className and tagName. This selects all the specific tagName element inside an element with the class className.

tagName.className: Note that there is no space between tagName and .className. This selects the tagName with a className.

```
<div class="className">
  <p>...</p>
  <p>...</p>
  <p class="cheese">...</p>
</div>
```

```
.className p {
  background: brown;
}
p.cheese {
  background: yellow;
}
```




CSS Attribute Selectors

- The **[attribute] selector** is used to select elements with a specified attribute.
- The **[attribute=value] selector** is used to select elements with a specified attribute and value.
- The **[attribute~=value] selector** is used to select elements with an attribute value containing a specified word.
- The **[attribute|=value] selector** is used to select elements with the specified attribute starting with the specified value.
- The **[attribute*=value] selector** is used to select elements whose attribute value contains a specified value.
- The **[attribute\$=value] selector** is used to select elements whose attribute value ends with a specified value.



Attaching CSS Style

Attach a style sheet to a page by adding the code to the <head> section of the HTML page. There are 3 ways to attach CSS to a page:

- **1. External Style Sheet:** Best used to control styling on multiple pages.

```
<link rel="stylesheet" type="text/css" media="all" href="css/styles.css" />
```

- **2. Internal/ Embedded Style Sheet:** Best used to control styling on one page.

```
<style type="text/css">  
    CSS Selector Statements  
</style>
```

- **3. Inline Style Sheet *:** CSS is not attached in the <header> but is used directly within HTML tags.

```
<tagName style="property:value;...">... </tagName>
```



Display Property

- **A block element** is an element that takes up the full width available, and has a line break before and after it.

Examples of block elements: `<h1>`, `<p>`, ``, `<div>`

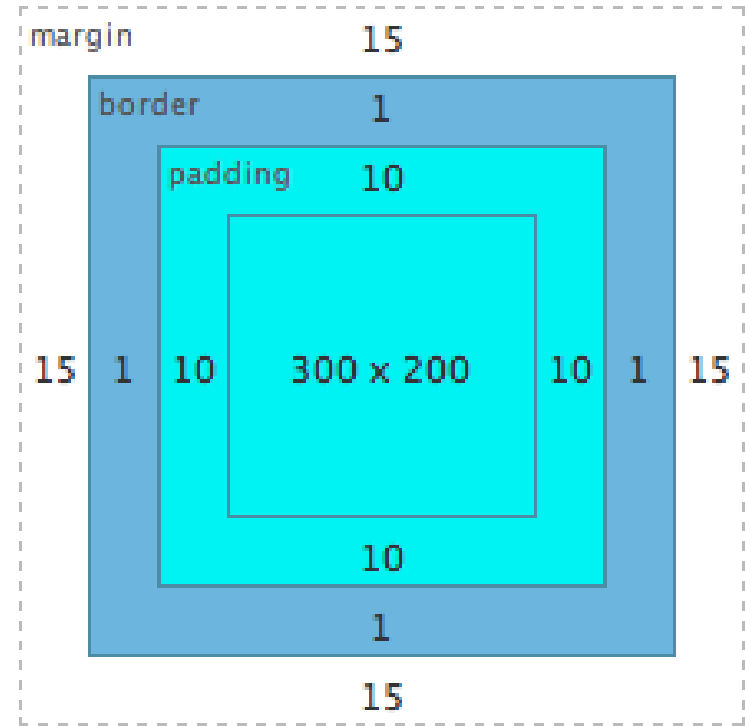
- **An inline element** only takes up as much width as necessary, and does not force line breaks.

Examples of inline elements: ``, `<a>`

- **Note:** Setting **display to none** will render the page as though the element does not exist. `visibility: hidden` hides the element, but the element takes up the space as it would if it was fully visible.

CSS Box Model

- `.box {`
`width: 300px;`
`height: 200px;`
`padding: 10px;`
`border: 1px solid #000;`
`margin: 15px; }`



- The total size of the element above will be calculated as follows:
Total width = $15 + 1 + 10 + 300 + 10 + 1 + 15 = 352\text{px}$
Total height = $15 + 1 + 10 + 200 + 10 + 1 + 15 = 252\text{px}$



margin: auto, max-width

Setting the **margin property to auto** automatically aligns it to the center.

Note: We have given two values. 0 auto which means the top and bottom margins are 0 and left and right are auto. This is shorthand notation

Setting **max-width property** limits the stretch of the element when the screen resolution is higher . It's used for making a site usable on mobile.

```
#outer {  
  max-width: 600px;  
  margin: 0 auto;  
}
```



CSS background-color Property

Keyword color values are names (such as red, green) that map to a given color

```
background-color: yellow;
```

Hexadecimal color value is of a pound, or hash, #, followed by three or six-characters

```
background-color: #00ff00;
```

RGB color value is stated using the rgb() function, which stands for red, green, and blue.

```
background-color: rgb(255,0,255);
```



CSS Links

- Links can be styled in different ways.
- Links can be styled with any CSS property (e.g. color, font-family, background, etc.).

```
/* unvisited link */
```

```
a:link {  
  color: #FF0000;  
}
```

```
/* visited link */
```

```
a:visited {  
  color: #00FF00;  
}
```

```
/* mouse over link */
```

```
a:hover {  
  color: #FF00FF;  
}
```

```
/* selected link */
```

```
a:active {  
  color: #0000FF;  
}
```



CSS Units

Absolute length values are the simplest length values, as they are fixed to a physical measurement, such as inches, centimetres, or millimetres.

The most popular absolute unit of measurement is known as the pixel and is represented by the **px** unit notation.

Relative Lengths is not fixed units of measurement; they rely on the length of another measurement.

We have **percentages and ems** for relative length



Position Property

- The position property specifies the type of positioning method used for an element (static, relative, fixed or absolute).
- **The position Property** - It specifies the type of positioning method used for an element.
- There are four different position values:
 1. static
 2. relative
 3. fixed
 4. absolute



CSS Float Property

- **The float property** removes the element from the document flow and moves them to the edge specified in the float property
- Using float property we can let them float and to the right or left inline within the container.
- **float:** left / right / none



Problem with Float

- Float has one major problem. If we float any elements inside a container, then the container doesn't expand as per the size of elements

- **Solution**

```
.clearfix2{  
  overflow:hidden;  
}  
.clearfix3{  
  clear:both;  
}  
.clearfix4{  
  clear:both; /*or left*/  
}
```

```
.clearfix1 : after {  
  content: ".";  
  clear: both;  
  display: block;  
  height: 0;  
  visibility: hidden;  
}
```



Inheritance

- Inheritance is the process by which some CSS properties applied to one tag are passed on to nested tags.
- Many CSS properties don't pass down to descendant tags at all. For example, the border property (which lets you draw a box around an element) isn't inherited
- A general rule is that properties that affect the placement of elements on the page or the margins, background colors, and borders of elements aren't inherited.



Specificity

- **Specificity** is the means by which a browser decides which property values are the most relevant to an element and gets to be applied.
- So here is the rule. The pattern goes like that
`<ID selector count>:<class selectors count>:< tags selector count>`

Example:

Selector with score of 1:0:0 is higher than score of 0:4:1.

Selector with score 0:1:0 is higher than 0:0:4

Note : A selector with a greater specificity score overwrites the other selector with lesser specificity irrespective of order where the rules are written.



CSS Media Queries

- **A media query** consists of a media type and at least one expression that limits the style sheets' scope by using media features, such as width, height, and color.

- **<!-- CSS media query on a link element -->**

```
<link rel="stylesheet" media="(max-width:800px)" href="example.css" />
```

- **<!-- CSS media query within a stylesheet -->**

```
<style>
```

```
  @media (max-width: 600px)
```

```
    { .facet_sidebar
```

```
      { display: none; }
```

```
    }
```

```
</style>
```



Lets Discuss Assignments