

CSS

Hoang Thi Kieu Hoa

CSS Overview

CSS Selector

Box model

Styling Text and List

Section 1

CSS OVERVIEW

- **CSS** is a language that describes the style of an HTML document
- CSS stands for **C**ascading **S**tyle **S**heets
 - CSS describes how HTML elements should be rendered on screen, on paper, in speech, or on other media.
 - CSS is one of the core languages of the open Web and is standardized across Web browsers according to the W3C specification

- CSS can be added to HTML elements in 3 ways:
- **Inline** - by using the style attribute in HTML elements
 - **Internal** - by using a <style> element in the <head> section
 - **External** - by using an external CSS file

- **Inline style:** An inline CSS is used to apply a unique style to a single HTML element

❖ *Syntax*

```
<element style="style_definitions">
```

Value	Description
<i>style_definitions</i>	One or more CSS properties and values separated by semicolons (e.g. style="color:blue;text-align:center")

- ❖ **Note:** The style attribute will override any style set globally, e.g. styles specified in the <style> tag or in an external stylesheet.

➤ Internal style:

- An internal CSS is used to define a style for a single HTML page.
- It is defined in the **<head>** section of an HTML page, within a **<style>** element

```
<head>
  <meta charset="utf-8">
  <title>My CSS experiment</title>
  <style>
    h1 {
      color: blue;
      background-color: yellow;
      border: 1px solid black;
    }

    p {
      color: red;
    }
  </style>
</head>
```

➤ External style:

- An external style sheet is used to define the style for many HTML pages.
- With an external style sheet, you can change the look of an entire web site, by changing one file.
- Each HTML page must include a reference to the external style sheet file inside the <link> element, inside the head section.

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

mystyle.css

```
body {
  background-color: lightblue;
}

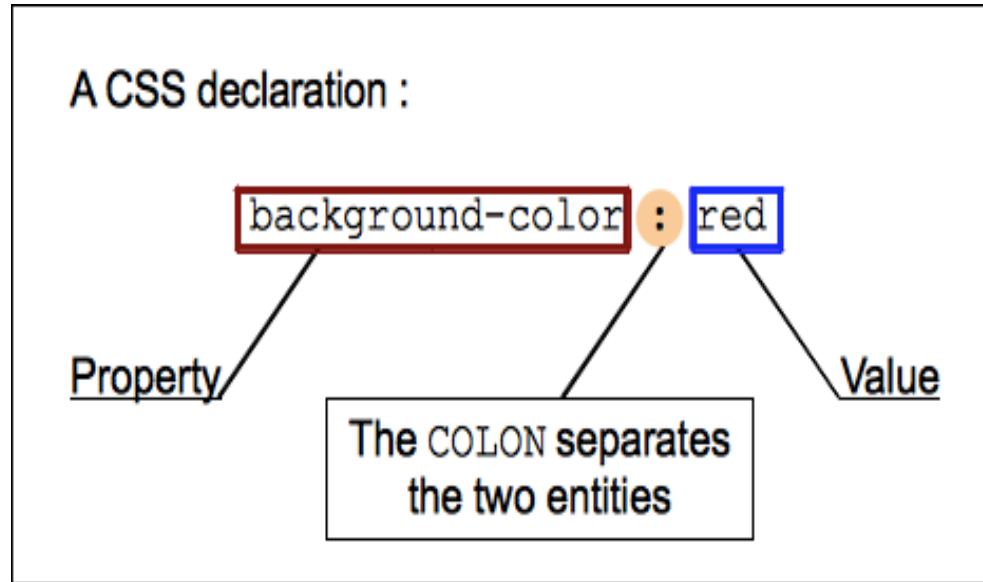
h1 {
  color: navy;
  margin-left: 20px;
}
```


➤ **Cascading order:** All the styles in a page will "cascade" into a new "virtual" style sheet by the following rules, where number one has the highest priority:

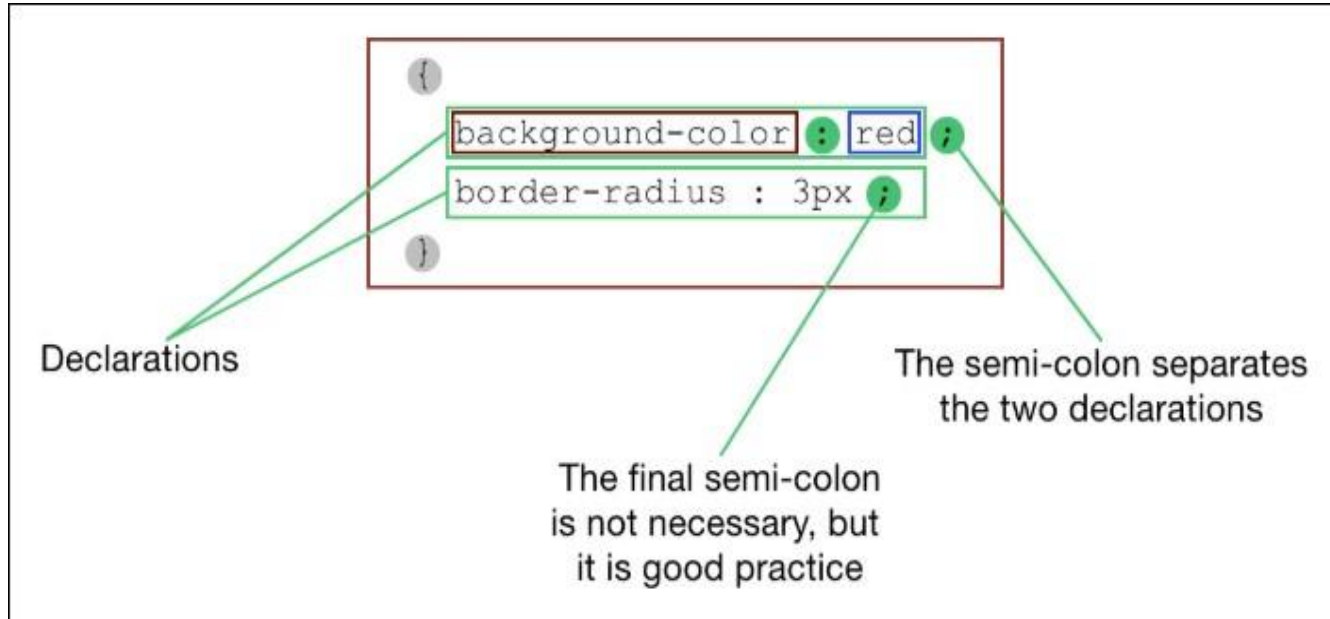
1. Inline style.
2. External and internal style sheets.
3. Browser default

- The basic goal of the **CSS** language is to allow a browser engine to paint elements of the page with specific features, like colors, positioning, or decorations.
- **CSS** Syntax includes:
 - The **property** which is an identifier, that is a human-readable name, that defines which feature is considered.
 - The **value** which describe how the feature must be handled by
 - the engine. Each property has a set of valid values, defined by a formal grammar, as well as a semantic meaning, implemented by the browser engine.

- **CSS Declarations:** A property and value pair is called a **declaration**



➤ **CSS Declarations block:** Declarations are grouped in **blocks**



Session 2

CSS SELECTORS

- **CSS selectors** are used to "find" (or select) the HTML elements you want to style.
- We can divide CSS selectors into four categories:
 1. Simple selector (type, id, class)
 2. Combinator selector
 3. Pseudo classes & pseudo elements
 4. Attribute selectors

- ***The element selector:*** Selects HTML elements based on the element name
- ***Id selectors:*** Uses the id attribute of an HTML element to select a specific element.
- ***Class selectors:*** Selects HTML elements with a specific class attribute

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

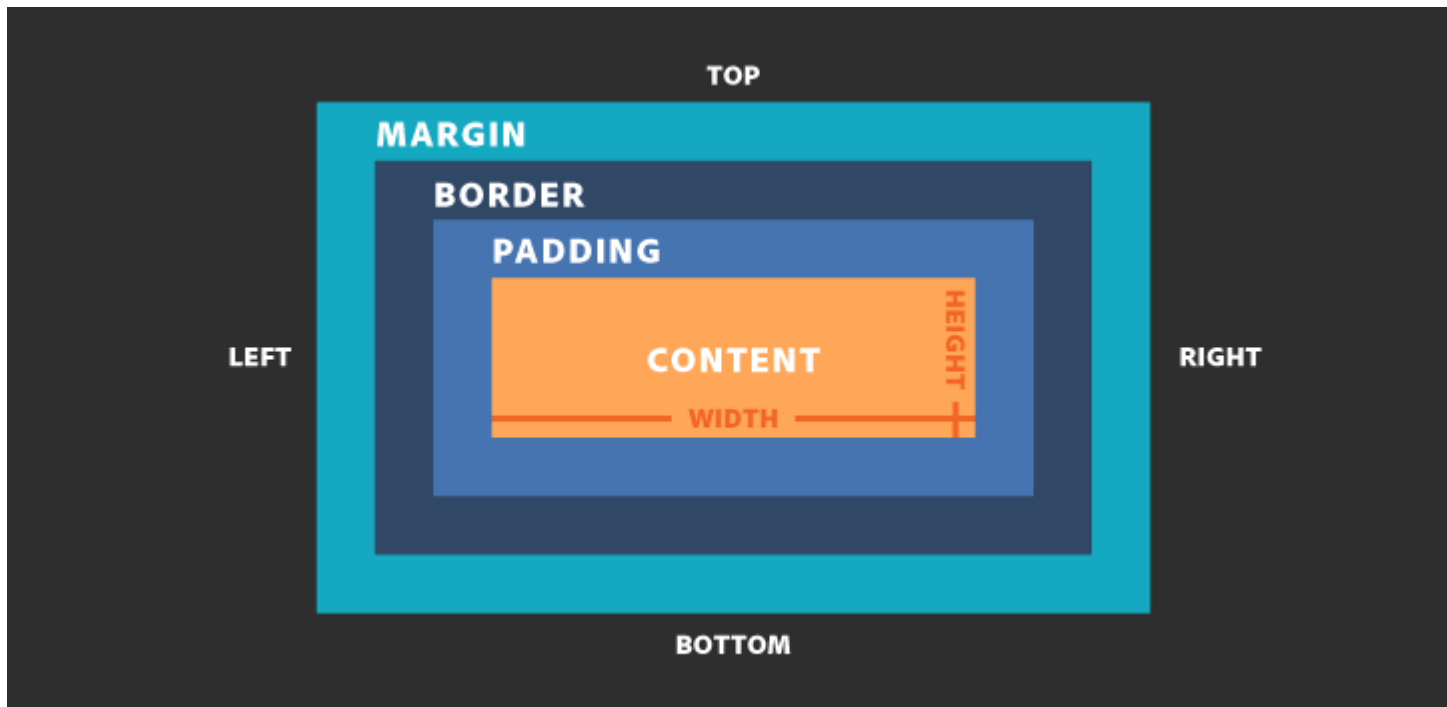
```
#para1 {  
  text-align: center;  
  color: red;  
}
```

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

Session 3

BOX MODEL

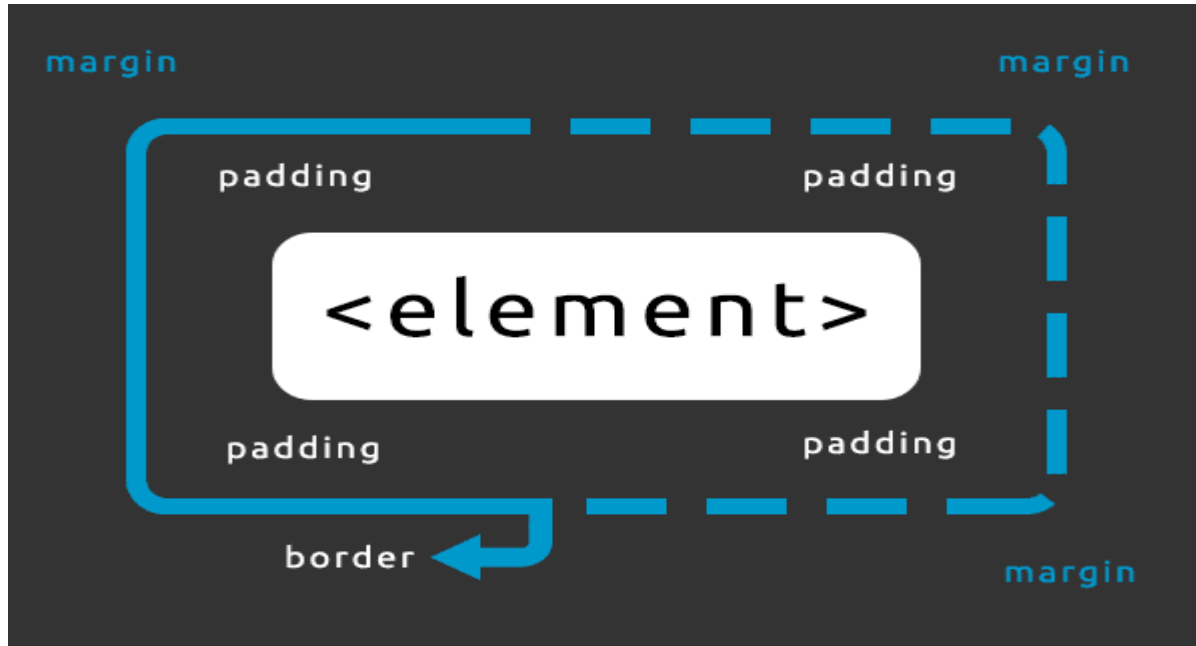
The alternative CSS Box Model



➤ **Making up a block box in CSS we have:**

- ***Content box:*** The area where your content is displayed, which can be sized using properties like width and height.
- ***Padding box:*** The padding sits around the content as white space; its size can be controlled using padding and related properties.
- ***Border box:*** The border box wraps the content and any padding. Its size and style can be controlled using border and related properties.
- ***Margin box:*** The margin is the outermost layer, wrapping the content, padding and border as whitespace between this box and other elements. Its size can be controlled using margin and related properties.

➤ Take a look



- **Margin:** Is an invisible space around your box. It pushes other elements away from the box. Margins can have positive or negative values. Setting a negative margin on one side of your box can cause it to overlap other things on the page
- We can control all margins of an element at once using the margin property, or each side individually using the equivalent longhand properties:
 - ***margin-top***
 - ***margin-right***
 - ***margin-bottom***
 - ***margin-left***

- **Padding:** Sits between the border and the content area. Unlike margins you cannot have negative amounts of padding, so the value must be 0 or a positive value, it is typically used to push the content away from the border.
- We can control all margins of an element at once using the margin property, or each side individually using the equivalent longhand properties:
 - ***padding-top***
 - ***padding-right***
 - ***padding-bottom***
 - ***padding-left***

- **Border:** is drawn between the margin and the padding of a box.
- To set the properties of each side individually, you can use:
 - ***border-top***
 - ***border-right***
 - ***border-bottom***
 - ***border-left***
- To set the width, style, or color of all sides, use the following:
 - ***border-width***
 - ***border-style*** (*dotted, dashed, solid, double, groove, inset, outset, ..*)
 - ***border-color***

Session 4

STYLING TEXT AND LIST

- The CSS properties used to **style text** generally fall into two categories:
- ***Font styles:*** Properties that affect the font that is applied to the text, affecting what font is applied, how big it is, whether it is bold, italic, etc
 - ***Text layout styles:*** Properties that affect the spacing and other layout features of the text, allowing manipulation of, for example, the space between lines and letters, and how the text is aligned within the content box.

➤ Fonts

- **Color:** The color property sets the color of the foreground content of the selected elements.

```
1 | p {  
2 |   color: red;  
3 | }
```

Tommy the cat

Well I remember it as though it were a meal ago...

Said Tommy the Cat as he reeled back to clear whatever foreign matter may have nestled its way into his mighty throat. Many a fat alley rat had met its demise while staring point blank down the cavernous barrel of this awesome prowling machine. Truly a wonder of nature this urban predator — Tommy the cat had many a story to tell. But it was a rare occasion such as this that he did.

➤ Fonts

- **Font families:** The font-family property specifies the font for an element.
 - The **font-family** property can hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font.
 - There are two types of font family names:
 - ***family-name*** - The name of a font-family, like "times", "courier", "arial", etc.
 - ***generic-family*** - The name of a generic-family, like "serif", "sans-serif", "cursive", "fantasy", "monospace".

➤ Fonts

- **font-size:** The font-size property sets the size of a font.
- **font-style:** The font-style property specifies the font style for a text
- **font-weight:** The font-weight property sets how thick or thin characters in text should be displayed.
- **text-transform:** The text-transform property controls the capitalization of text.
- **text-decoration:** The text-decoration property specifies the decoration added to text.

■ Fonts

- ✓ **Font size:** The font-size property sets the size of a font.
- ✓ Font size can take values measured in most of these units (and others, such as percentages), however the most common units you'll use to size text are **px, rem, em**

➤ Text layout

- **Text alignment:** The text-align property is used to control how text is aligned within its containing content box (*left, right, center, justify*).
- **Line height:** The line-height property specifies the height of a line.
- **Letter and word spacing:** The letter-spacing and word-spacing properties allow you to set the spacing between letters and words in your text.

- **Bullet styles:** The list-style-type property specifies the type of list item marker.

```
1 | ol {  
2 |   list-style-type: upper-roman;  
3 | }
```



- I. Toast pitta, leave to cool, then slice down the edge.
- II. Fry the halloumi in a shallow, non-stick pan, until browned on both sides.
- III. Wash and chop the salad
- IV. Fill Pitta with salad, humous, and fried halloumi.

➤ **Bullet styles:** some value of list-style-type property

- disc
- circle
- square
- decimal
- decimal-leading-zero
- lower-roman
- lower-greek
- ...

- **Bullet position:** The list-style-position property specifies the position of the list-item markers.

▪ list-style-position: out

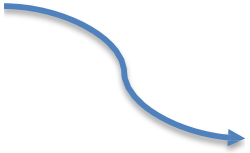
- | |
|---|
| • Coffee - A brewed drink prepared from roasted coffee beans... |
| • Tea |
| • Coca-cola |

▪ list-style-position: inside

- | |
|---|
| • Coffee - A brewed drink prepared from roasted coffee beans... |
| • Tea |
| • Coca-cola |

- **Using a custom bullet image:** The `list-style-image` property specifies an image as the list item marker

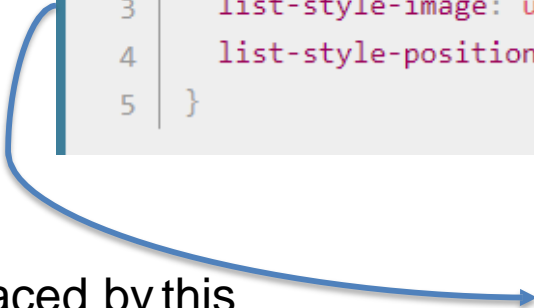
```
ul {  
  list-style-image: url('sqpurple.gif');  
}
```

- 
- Coffee
 - Tea
 - Coca Cola

- **list-style shorthand:** The three properties mentioned above can all be set using a single shorthand property, `list-style`

```
1 | ul {  
2 |     list-style-type: square;  
3 |     list-style-image: url(example.png);  
4 |     list-style-position: inside;  
5 | }
```

Could be replaced by this



```
1 | ul {  
2 |     list-style: square url(example.png) inside;  
3 | }
```

Thank you