Article 07 – CS Box Model

Everything that we see on a web page is contained within a box (including the <body> element that is visible to us as the boundaries of the viewport). Thus, understanding the CSS Box Model is crucial for creating responsive and intuitive web pages.

**The CSS Box Model**

The representation of all elements wrapped up in rectangular boxes is known as the CSS Box Model. Each box consists of four parts:  
  
Content: Where the text or image appears. The core area of the element

Padding: The space between the content and the border. Gives breathing room inside the box.

Border: A visible line surrounding the padding. It delineates the area of the box.

Margin: The space outside the border that separates it from the other elements in the DOM.

**Box Model Properties**

1) Content

width: Sets the width of the content area.

height: Sets the height of the content area.

2) Padding

Shorthand property 1: `padding: 50px` (all four sides)

Shorthand property 2:`padding 10px 20px 30px 40px` (top | right | bottom | left)

Individual properties: padding-top, padding-right, padding-bottom, padding-left

3) Border

Shorthand property: `border: width, style, color`

Individual properties: border-width, border-style, border-color, border-left, border-right, border-top, border-bottom, border-left-width, etc.

border-radius: controls the curvature of the corners of an element's border, giving it a softer, more modern appearance.

4) Margin

Shorthand property 1: `margin: 50px` (all four sides)

Shorthand property 2: `margin 10px 20px 30px 40px` (top | right | bottom | left)

Individual properties: margin-top, margin-right, margin-bottom, margin-left

**CSS Specificity Algorithm**

The CSS Specificity Algorithm is a set of rules that the browser uses to determine which CSS rule wins when multiple rules target the same element. It is how the browser resolves CSS conflicts.

Specificity: A numeric score assigned to each selector. The more specific each selector is, the higher its weight and priority are. Calculated as a 4-part score (a, b, c, d):

a: in-line styles (not selectors, but style attributes in HTML)  
b: number of ID selectors (#id)  
c: number of class/ pseudo-class selectors, attributes (.class, ::hover, type=”text”)  
d: number of element and pseudo-element selectors (div, p, ::before)

#main-header 🡪 (0, 1, 0, 0)

.menu li.active a:hover 🡪 (0, 0, 3, 2)

body.home #main .nav li a 🡪 (0, 1, 2, 3)

**Conflict Resolution Order**

1) In case of multiple rules applying to the same element, browser checks specificity.

2) In case specificity is equal, browser uses the one last declared in the stylesheet.

**The Override Trump Card**

Using !important in the stylesheet will override all other rules, even if they have a high specificity. Must be used sparingly as it reduces code readability and makes debugging difficult.