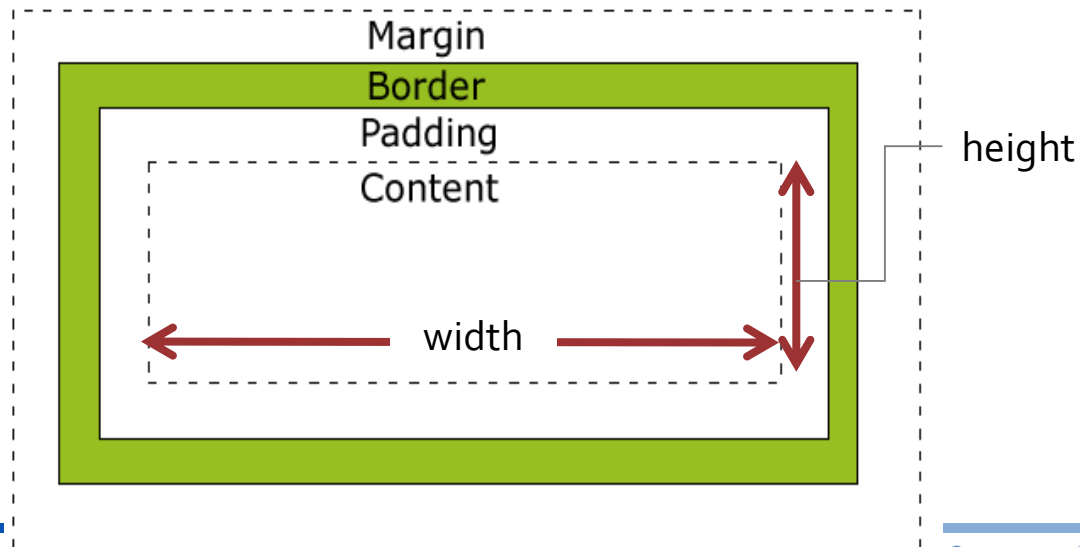# 6. CSS Tutorial II

충남대학교 컴퓨터공학과
데이타베이스시스템 연구실

CSS Box Model

The CSS Box Model

- A box that wraps around HTML elements. It consists of
  - **Content** : The content of the box, where text and images appear
  - **Padding** : Clears an area around the content. The padding is transparent.
  - **Border** : A border that goes around the padding and content
  - **Margin** : Clears an area outside the border. The margin is transparent
- Allow us to add a border around elements, and to define space between

```
Margin
Border
Padding
Content

width
height
```

for example   *Try it!*

```
width : 300px;
padding : 25px;
border:25px solid gray;
marin:25px;
```

total width of the element (450px)
= 300px (width) + 50px (left+right
padding + 50px (left+right border) +
50px (left+right margin)

CSS Border

CSS Border Properties

The CSS border properties allow you to specify the style, size, and color of an element's border.

Border Style

- The border-style property specifies what kind of border to display.
- Note : None of the border properties will have ANY effect unless the **border-style** property is set!
- border-style values : *Try it!*

CSS Border (cont'd)

- **Border Width**   *Try it!*
  - The border-width property is used to set the width of the border
  - The width is et in pixels, or by using one of the three pre-defined values : thin, medium, or thick
  - **Note** : The "border-width" property does not work if it used alone. Use the "border-style" property to set the borders first.

- **Border Color**   *Try it!*
  - The border-color property is used to set the color of the border.
  - If the border color is set it is inherited from the color property of the element.
  - **Note** : The "border-color" property does not work if it used alone. Use the "border-style" property to set the borders first.

CSS Border (cont'd)

- ## Border – Individual sides
  - To specify different borders for different sides *Try it!*
  - The border-style property can have from one to four values
    - Border-style : dotted solid double dashed
      - top, right, bottom, left
    - Border-style : dotted solid double
      - top, right and left, bottom
    - Border-style : dotted solid *Try it!*
      - top and bottom, right and left
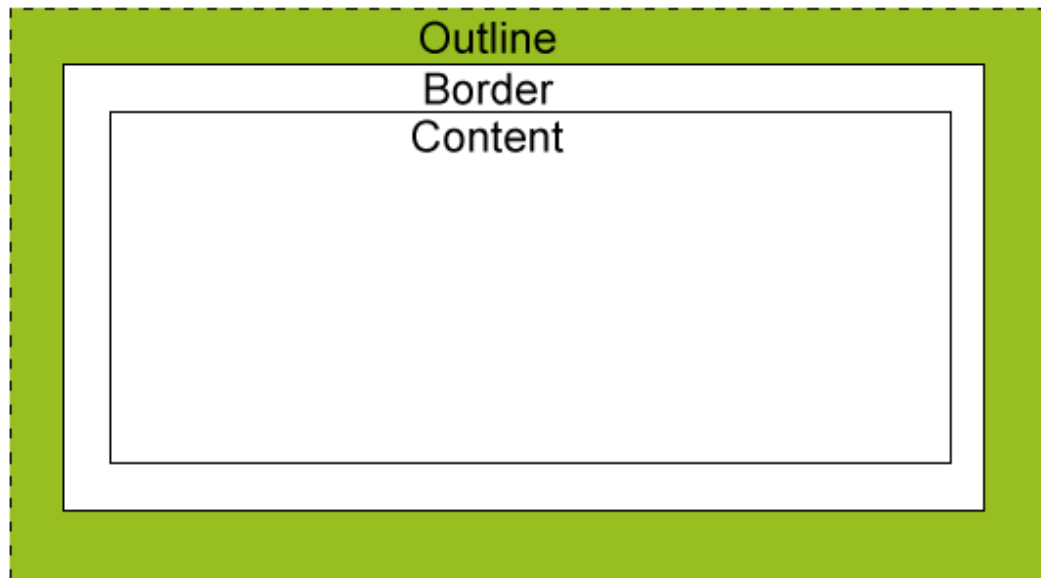   - Border-style : dotted solid double dashed
      - All four borders

- ## Border – Shorthand property *Try it!*
  - A shorthand for the following individual border properties
    - border-width border-style(required) border-color

CSS Outlines

🔷 **CSS Outline**   *Try it!*

- 🔸 An outline is a line that is drawn around elements to make the element "stand out".
- 🔸 The element's total width and height is not affected by the width of the outline

Outline
Border
Content

CSS Margin

- **Margin**
  - Clears an area around an element (outside the border).
  - Not have a background color

- **Possible Values**

Value	Description
auto	The browser calculates a margin
length	Specifies a margin in px, pt, cm, etc. Default value is 0px
%	Specifies a margin in percent of the width of the containing element
inherit	Specifies that the margin should be inherited from the parent element

CSS Margin (cont'd)

🔹 **Margin – Individual sides**   *Try it!*

- To specify different margins for different sides of an element.

🔹 **Margin – Shorthand property**   *Try it!*

- To specify all the margin properties in one property.
  - margin : 25px 50x 75px 100px
    - top, right, bottom, left
  - margin : 25px 50x 75px
    - top, right and left, bottom
  - margin : 25px 50x
    - top and bottom, right and left
  - margin : 25px
    - All four margins

CSS Padding

- **Padding**
  - CSS padding properties define the space between the element border and the element content.
  - Is affected by the background color of the element.

- **Possible Values**

Value	Description
length	Defines a fixed padding (in pixels, pt, em, etc.)
%	Defines a padding in % of the containing element

CSS Padding (cont'd)

- **Padding – Individual sides**   *Try it!*

- **Padding – Shorthand property**   *Try it!*

- *<< Here is the end of "Introduction to CSS – Design a Button for Your Website" in Codecademy.com. >>*

CSS Dimension

- CSS dimension properties allow you to control the height and width of an element.

- Examples
  - *Set the height of elements*
  - *Set the height of an image using percent*
  - *Set the width of an element using a pixel value*
  - *Set the maximum height of an element*
  - *Set the maximum width of an element using percent*
  - *Set the minimum height of an element*
  - *Set the minimum width of an element using a pixel value*

CSS Display and Visibility   *Try it! – using JavaScript*

- **Hiding an Element – display:none or visibility:hidden**
  - display property
    - Specifies if/how an element is displayed
    - display:none hides an element, and it will not take up any space   *Try it!*
  - Visibility property
    - Specifies if an element should be visible or hidden
    - visibility:hidden hides an element, but it will still take up the same space as before.   *Try it!*

- **Changing How an Element is Displayed**
  - Changing an inline element to a block element, or vice versa, can be useful for making the page look a specific way
    - Display <li> elements as inline elements   *Try it!*
    - Display <span> elements as block elements   *Try it!*

CSS Positioning

Decide which element to display in front!

...sitioning can be tricky sometimes!

Elements can overlap!

🔷 **Positioning**

- Allow you to position an element.
- Elements can be positioned using the top, bottom, left, and right properties
- Will not work unless the position property is set first
- Work differently depending on the positioning method

CSS Positioning (cont'd)

- **Static Positioning**
  - By default
  - Positioned according to the normal flow of the page.

- **Fixed Positioning**   *Try it!*
  - Is positioned relative to the browser window
  - Will not move even if the window is scrolled
  - The document and other elements behave like the fixed positioned element does not exist
  - Can overlap other elements

CSS Positioning (cont'd)

- **Relative Positioning**   *Try it!*
  - Is positioned relative to its normal position
  - The content of relatively positioned elements can be moved and overlap other elements, but the reserved space for the element is still preserved in the normal flow.

- **Absolute Positioning**   *Try it!*
  - Is positioned relative to the first parent element
  - The document and other elements behave like the absolutely positioned element does not exist
  - Can overlap other elements

CSS Positioning (cont'd)

- **Overlapping Elements** *Try it!*
  - The z-index property specifies the stack order of an element
  - An element can have a positive or negative stack order
    - An element with greater stack order is always in front of an element with a lower stack order.

CSS Float

- ## What is CSS Float?   *Try it!*
  - With CSS float, an element can be pushed to the left or right, allowing other elements to wrap around it
  - Float is very often used for images, but it is also useful when working with layouts

- ## How Elements Float
  - Elements are floated horizontally
    - This means that an element can only be floated left or right.
  - If an image is floated to the right, a following text flows around it, to the left *Try it!*

CSS Float (cont'd)

- **Floating Elements Next to Each Other**
  - *Example : Image gallery using the float property*

- **Turning off Float – Using Clear**   *Try it!*
  - The clear property specifies which sides of an element where other floating elements are not allowed

CSS Combinators

- **CSS Combinators**
  - Something that explains the relationship between the selectors.
  - Four different combinators
    - Descendant selector, child selector, adjacent sibling selector, general sibling selector

- **Descendant Selector**   *Try it!*
  - Matches all element that are descendants of a specified element

- **Child Selector**   *Try it!*
  - Selects all elements that are the immediate children of a specified element

CSS Combinators (cont'd)

- **Adjacent Sibling Selector**   *Try it!*
  - Selects all elements that are the adjacent siblings of a specified element.
    - Sibling elements must have the same parent element
    - "adjacent" means "immediately following.

- **General Sibling Selector**   *Try it!*
  - Selects all elements that are siblings of a specified element.

CSS Pseudo-classes

What are Pseudo-classes?
- Used to define a special state of an element.
- For example
  - Style an element when a user mouses over it
  - Style visited and unvisited links differently

Syntax

```
selector:pseudo-class {
      property:value;
}
```

Anchor Pseudo-classes   *Try it!*

CSS Pseudo-classes (cont'd)

- ## CSS – The :first-child Pseudo-class
  - Matches a specified element that is the first child of another element.
    - Match the first <p> element   *Try it!*
    - Match the first <i> element in all <p> elements   *Try it!*
    - Match all <i> elements in all first child <p> elements   *Try it!*

- ## CSS – The :lang Pseudo-class   *Try it!*
  - Allows you to define special rules for different languages

- ## All CSS Pseudo Classes   *Try it!*

CSS Pseudo - elements

- **What are Pseudo-Elements?**
  - Used to style specified parts of an element.
  - For example
    - Style the first letter, or line, or an element
    - Insert content before, or after, the content of an element

- **Syntax**

  ```
  selector::pseudo-element {
        property:value;
  }
  ```

CSS Pseudo – elements (cont'd)

- **The ::first-line Pseudo-element** _Try it!_
  - Used to add special style to the first line of a text.
  - Can only be applied to block elements

- **The ::first-letter Pseudo-element** _Try it!_
  - Used to add a special style to the first letter of a text.
  - Can only be applied to block elements

- **The ::before Pseudo-element** _Try it!_
  - Used to insert some content before the content of an element.

CSS Pseudo – elements (cont'd)

- **The ::after Pseudo-element** *Try it!*
  - Used to insert some content after the content of an element.

- **The ::selection Pseudo-element** *Try it!*
  - Matches the portion an element that is selected by a user.

CSS Attribute Selectors

- **Style HTML Elements With Specific Attributes**
  - It is possible to style HTML elements that have specific attributes or attribute values

- **CSS [attribute] Selector**
  - Used to select elements with a specified attribute.   *Try it!*

- **CSS [attribute=value] Selector**
  - Used to select element with a specified attribute and value.   *Try it!*

- **CSS [attribute~=value] Selector**
  - Used to select elements with an attribute value containing a specified word. *Try it!*

CSS Attribute Selectors (cont'd)

- **CSS [attribute|=value] Selector**
  - Used to select elements with the specified attribute starting with the specified value. *Try it!*

- **CSS [attribute^=value] Selector**
  - Used to select elements whose attribute value begins with a specified value. *Try it!*
    - The value does not have to be a whole world!

- **CSS [attribute$=value] Selector**
  - Used to select elements whose attribute value ends with a specified value. *Try it!*
    - The value does not have to be a whole world!

CSS Attribute Selectors (cont'd)

- **CSS [attribute*=value] Selector**
  - Used to select elements whose attribute value contains with a specified value. *Try it!*
    - The value does not have to be a whole world!

- *<< Here is the end of "CSS Classes and ID" in Codecademy.com. >>*
- *<< Here is the end of "CSS Element Positioning" in Codecademy.com. >>*