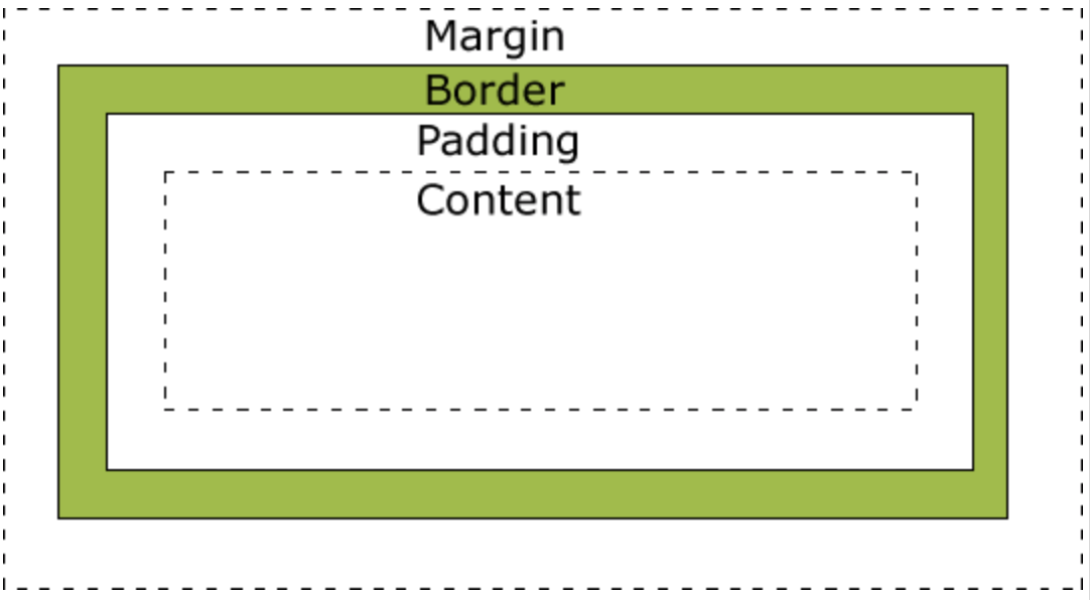
**Learning Objective**

* Understand specific concepts of CSS box model
* Master the occurrence and avoidance of margin folding
* Master attributes related to the CSS box model

**Content**

* In a document, each element is represented as a rectangular box. Each box has four sides: a margin edge, a border edge, a padding edge, and a content edge.



* The box model allows elements to be placed in the space between other elements and the borders of surrounding elements. The descriptions for the four different sides are as follows:
  + Margin – it clears the area outside the border, and the margin is transparent
  + Border – it is a border around the padding and outside the content
  + Padding –it clears the area around the content and the padding is transparent
  + Content – it is the content of the box, which can be used to display text and various types of multimedia files
* Width in the box model:
  + specify the width of the content box
  + the percentage of the content box is relative to the parent container (including the block)
* Height in the box model:
  + specify the height of content box
  + the percentage of the content box is relative to the parent container (including the block)
  + the percentage of height only takes effect when the height containing block does not depend on the element
* Padding in the box model:
  + refers to the padding
  + padding-top, padding-right, padding-bottom, padding-left
  + Abbreviations: padding: top value, right value, bottom value, left value; (up-right-down-left) -Margin in the box model:
  + refers to the margins
  + margin-top, margin-right, margin-bottom, margin-left
  + Abbreviations: margin: top value, right value, bottom value, left value; (up-right-down-left)
* Margin folding
  + In CSS, when boxes (possibly a parent-child element or a sibling element) in two or more adjacent normal document flows is folded in margin in vertical direction, the resulting margin in this case is called margin folding.
  + Margin folding occurs in the following cases:
    - Block-level boxes that all belong to normal document flows and participate in the same block-level formatting context
    - Not separated by padding, border, clear and line box
    - Both belong to the edge of vertically adjacent box:
      * The margin-top of the box and the margin-top of its first normal document flow element
      * The margin-bottom of the box and the margin-top of its next normal document flow brother
      * The margin-bottom of the box and the margin-bottom of its parent element
      * Margin-top and margin-bottom of the box, and no new block-level formatting context is created, and there is a min-height calculated as 0, a height calculated as 0 or auto, and there is no ordinary flow sub-element
  + So how to avoid margin folding? (As mentioned earlier, margin folding meets 4 conditions: two or more, adjacent, normal document flow and vertical direction, so you can avoid margin folding as long as you interrupt any of them.)
    - Floating elements do not have overlay with any element, also including its child element
    - An element that creates BFC does not have a margin superimposition with its child elements
    - Absolute positioning element does not have margin superimposition with any other element, including its child elements
    - Inline-block element does not have margin superimposition with any other element, including its child elements
    - The margin-bottom of the block-level element in the normal flow is always superimposed with the margin-top of the next block-level element adjacent to it, unless the adjacent sibling element is clear
    - The margin-level elements in a normal flow (no border-top, no padding-top) has margin-top superimposition with its child element in its first normal flow (no clear)
    - The block-level elements in the normal flow (height is auto, min-height is 0, no border-bottom, no padding-bottom) has margin-bottom superimposition with its child element in its last normal flow (no its own margin superimposition or clear)
    - If an element has a min-height of 0, no border, no padding, a height of 0 or auto, and no sub-elements, then its own margins will be superimposed.
* Box-sizing:
  + Define the calculation method to change box model
  + Value: border-box | content-box
    - Border-box: the width or height of the content in this case contains border, padding, width or height of the content (the width or height of the content = the width or height of the box - the border - the padding)
    - Content-box: the width and height of the element (width/height) equals to the border width of the element plus the padding plus the element width/height, that is, the element width/ Height = border + padding + content width / height
    - Some experts even recommend that all web developers set the box-sizing of all elements to border-box.
  + Initial value: content-box
* border:
  + Three elements of border:
    - border-width:<length> | thin | medium | thick
    - border-style: none | solid | dashed | dotted | double
    - border-color: <color>
  + Four directions:
    - border-left
    - border-top
    - border-right
    - border-bottom
* Overflow:
  + refers to overflow control
  + Value: visible | hidden | scroll | auto
  + Initial value: visible
* Visibility:
* The purpose of this attribute is to control the display of the element
* Value: visible | hidden | collapse
* Initial value: visible

**Recommendation**

* [Margin Series - Margin Folding](https://blog.doyoe.com/2013/12/04/css/margin%E7%B3%BB%E5%88%97%E4%B9%8B%E5%A4%96%E8%BE%B9%E8%B7%9D%E6%8A%98%E5%8F%A0)
* [Box Model - CSS: Cascading Style Sheets](https://developer.mozilla.org/zh-CN/docs/Web/CSS/CSS_Box_Model/Introduction_to_the_CSS_box_model)
* [CSS Box Model](https://www.w3schools.com/css/css_boxmodel.asp)
* [Box Model](https://www.w3.org/TR/CSS2/box.html)