

Lecture 15: CSS Box Model

- **css border style**: This property specifies what kind of border we want to display. For example, dotted, dashed, solid, double, groove, etc.
- **border-width**: This property sets the width of an element.
- **border-color**: This property sets the color of the border.
- **ways to apply border on an element**:

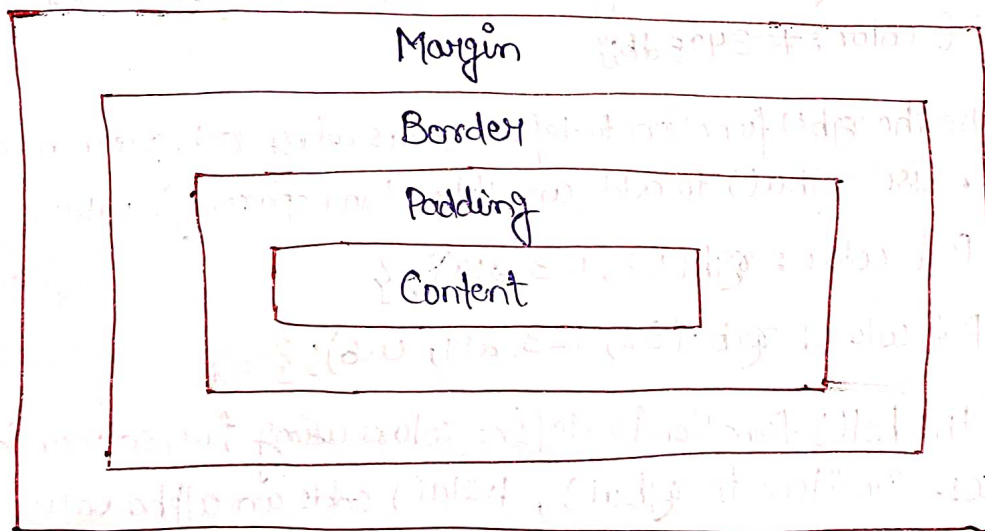
`border-style: solid;`
`border-color: red;`
`border-width: 2px;` } These two ways are same. We can define `border: solid red 2px;` in any of these two ways.

- **border-radius**: This property defines the radius of the element corner. This property allows us to add rounded corners to elements.

→ Why can't we apply height or width property on span tag?

As span tag is an inline element and their height or width are determined by the content it contains. So, it will completely ignore these properties. However, we can apply these properties by setting display as block element.

- **CSS Box Model**: The term box model is used when we talk about design and layout. So, basically this model says whatever element we render on viewport, it will be rendered in rectangle box only.



- **Content**: The content of the box where text or image appears.
- **Padding**: Clears the area around the content. The padding is transparent.
- **Border**: A border that goes around padding and content.
- **Margin**: Clears the area outside the border. The margin is transparent.

→ Ways to add padding and margins:

`padding-left: 20px;`
`padding-right: 40px;`
`padding-top: 60px;`
`padding-bottom: 80px;`

or `padding: 25px 35px 45px 55px;`

↑ right ↑ left
↓ top ↓ bottom

padding : 20px 40px;
 ↓ ↓
 top-bottom right-left

Just like padding we can apply margins also to an element in same way.

→ For best practice set padding and margins to 0 initially -

* { padding : 0px;
 margin : 0px; }

} → it is an universal selector