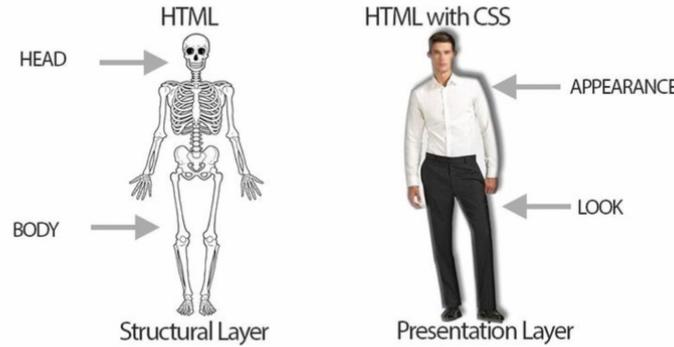
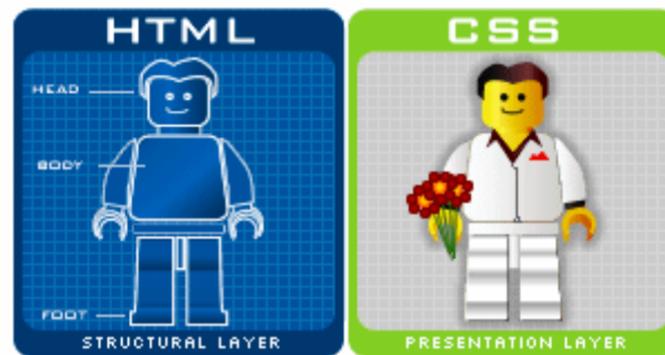
A close-up photograph of liquid ink swirling in water. The colors are vibrant, with shades of blue, red, and white. The blue ink is more concentrated on the left, while the red and white ink are more prominent on the right, creating a dynamic and organic pattern.
2

Day-2: CSS Styles & Box Model

Introduction to styles (CSS):

History of CSS

- Before CSS we used to have black and white layouts, just like emails



- There are various properties in CSS to set and change the color of the elements.

Types of CSS color values

- There are various color values in CSS through which you can specify the color to your HTML elements. These values are in different formats, which are explained below.

CSS Color Keywords

- Using a keyword (such as `blue` or `transparent`).
- Color keywords are case-insensitive identifiers that represent a specific color, such as `red`, `blue`, `black`, or `lightseagreen`. Although the names more or less describe their respective colors, they are essentially artificial, without a strict rationale behind the names used.
- The complete list of such keywords is available [here](#).

Example:

```
<p style="color: "blue">I am paragraph</p>
```

CSS RGB Colors

- RGB value is the combination of Red, Green, and Blue colors.
- An RGB color value is specified with : `rgb(red, green, blue)`. The range of these three colors is defined from 0 to 255 and it defines the intensity of the color. The colors can be changed by changing these values.

Example:

```
<p style="color: "rgb(0, 0, 255)">I am paragraph</p>
```

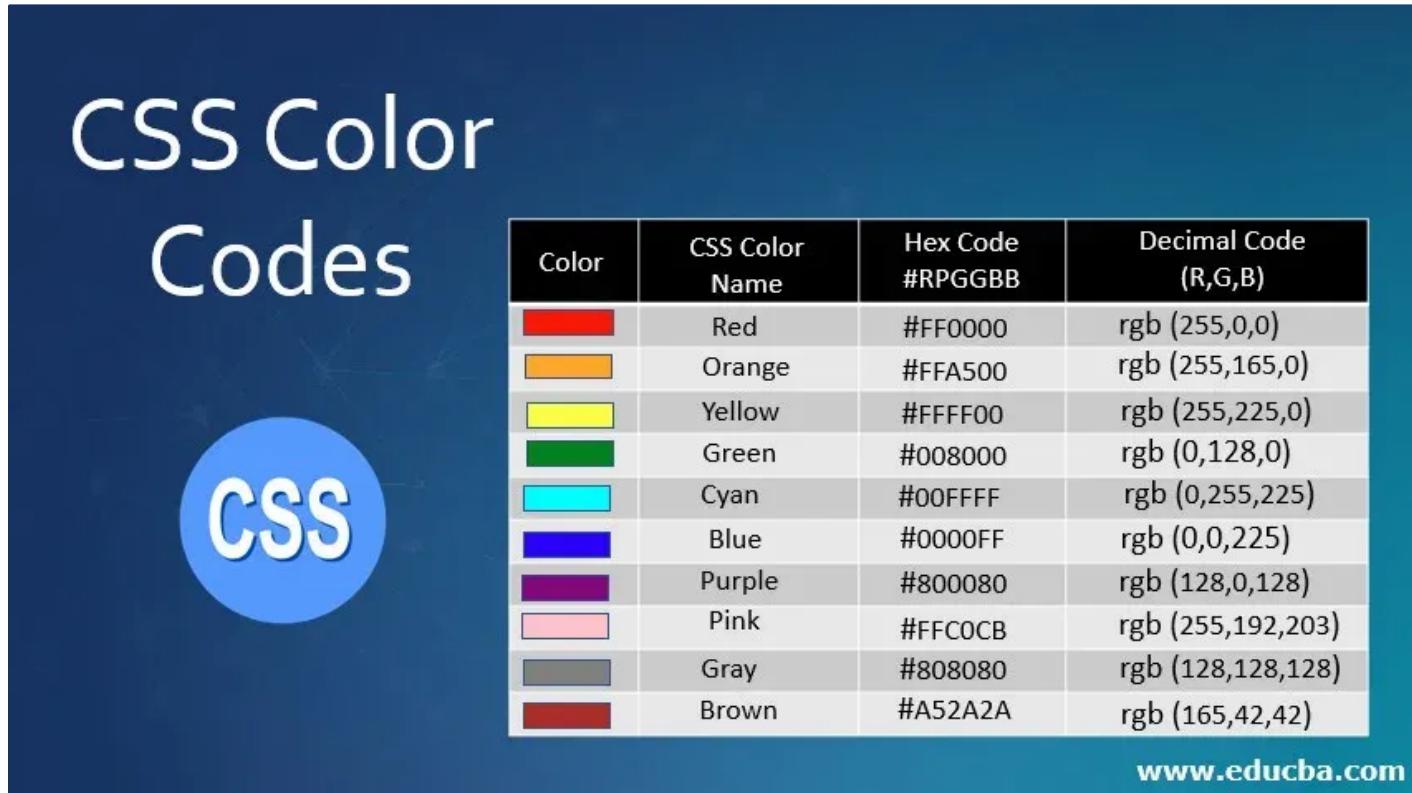
CSS HEX Colors

- The colors in CSS can be specified in Hexadecimal values also.
- A hexadecimal color is a 6 digit representation of the color.
- The notation of the HEX value starts with the "#" symbol followed by the six characters within the range of 0 to F.

Example:

```
<p style="color:#bfff00">I am paragraph</p>
```

- Overview of all types of colors



The slide features a large title 'CSS Color Codes' on the left, with a blue circle containing the letters 'CSS' below it. On the right is a table mapping colors to their names, hex codes, and decimal RGB values.

Color	CSS Color Name	Hex Code #RPGGBB	Decimal Code (R,G,B)
Red	Red	#FF0000	rgb (255,0,0)
Orange	Orange	#FFA500	rgb (255,165,0)
Yellow	Yellow	#FFFF00	rgb (255,225,0)
Green	Green	#008000	rgb (0,128,0)
Cyan	Cyan	#00FFFF	rgb (0,255,225)
Blue	Blue	#0000FF	rgb (0,0,225)
Purple	Purple	#800080	rgb (128,0,128)
Pink	Pink	#FFC0CB	rgb (255,192,203)
Gray	Gray	#808080	rgb (128,128,128)
Brown	Brown	#A52A2A	rgb (165,42,42)

www.educba.com

CSS Font-size

- In CSS the font-size property is used to set or tweak the size of the font.
- It can have several values that can be absolute (eg.- xx-small, medium, xx-large.) or relative (larger, smaller, %) or length (numbered- 12px, 1em, etc.)

The screenshot shows a code editor interface with a dark theme. On the left is the code editor pane displaying the file `test.html`. The code contains HTML and CSS:<!DOCTYPE html>
<html lang="en">
<head>...
</head>
<body>
 <p style="font-size: 40px">Masai School</p>
 <p style="font-size: 20px">
 Masai School - The Coding School that cares ab
 </p>
</body>
</html>On the right is the browser preview pane showing the rendered HTML. The title "Masai School" is displayed in a large, bold, black font. Below it, the text "Masai School - The Coding School that cares about you - Masai School" is displayed in a smaller black font.

Text-align:

- We use the CSS `text-align` property to align content inside a block-level element.
- This alignment affects the horizontal axis only.

Basic Syntax

- The `text-align` property accepts `left`, `center`, `right`, `justify` as values.
- The `left` Value: The `left` value of the `text-align` property is the default. So, every content inside a block-level element is aligned to the left by default.
- The `center` Value: With the center value, spaces are created on the left and right, so, everything gets pushed to the center.
- The `right` Value: Assigning a value of `right` to the `text-align` property pushes the content inside a block-level element to the right.

The screenshot shows a code editor interface with a dark theme. On the left is the code editor pane displaying the file `test.html` with the following content:

```
<!DOCTYPE html>
<html lang="en">
  <head>...
  </head>
  <body>
    <p style="text-align:left">Masai School</p>
    <p style="text-align:center">Masai School</p>
    <p style="text-align:right">Masai School</p>
  </body>
</html>
```

On the right is the browser preview pane showing the rendered HTML. It displays three `p` elements with different text alignments: "Masai School" aligned left, "Masai School" aligned center, and "Masai School" aligned right.

- The `justify` Value: The `justify` value of the `text-align` property lines up the content on the left and right edges of the block-level element (the box). If the last line isn't a full line, then it leaves it alone. It's easier to see how this works in the image below:

The screenshot shows a code editor on the left and a browser preview on the right. The code editor displays the following HTML and CSS:

```

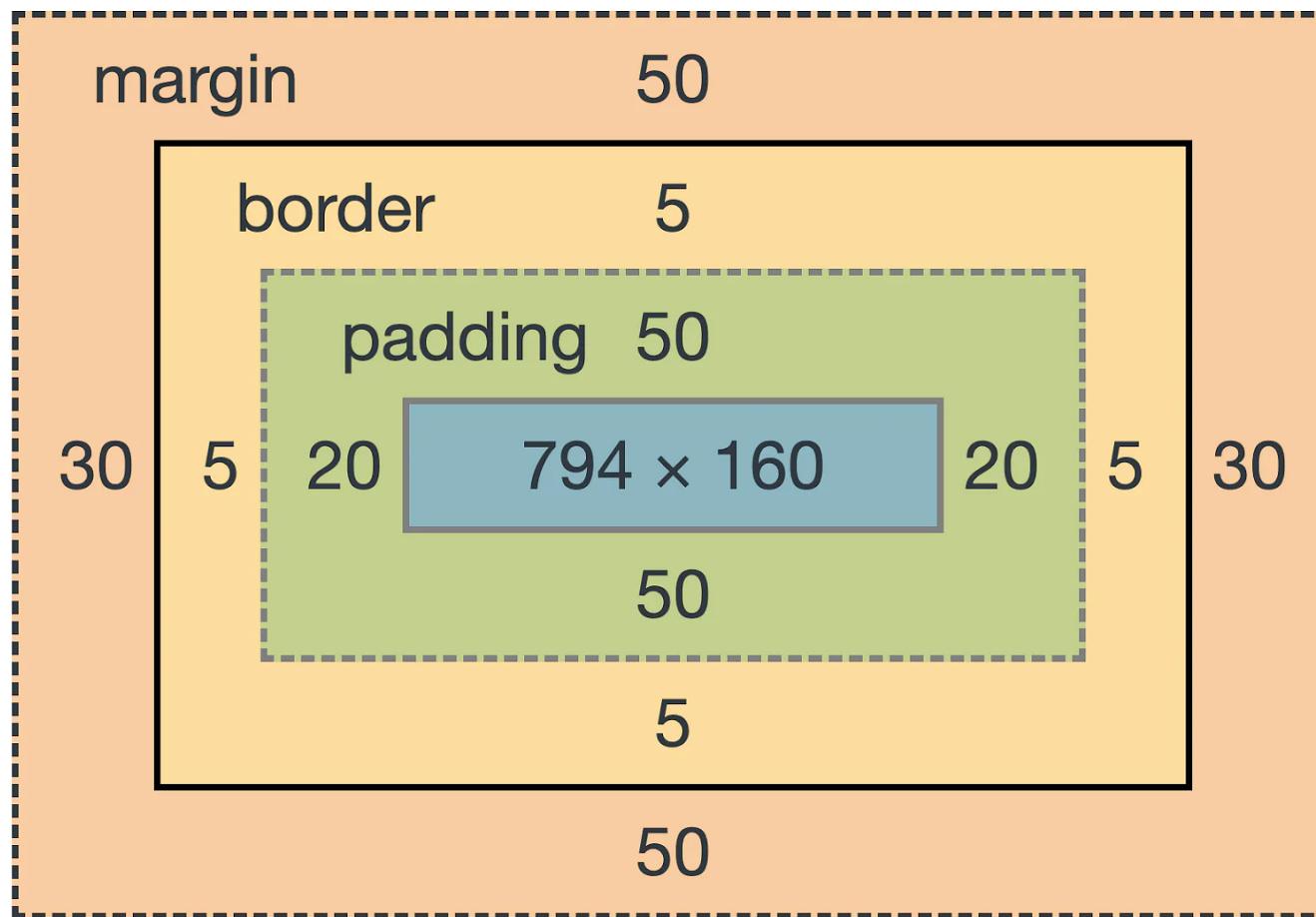
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>...
4  ...
5  <body>
6      <h1>Normal Alignment</h1>
7      <p style="text-align: left">
8          Lorem ipsum, dolor sit amet consectetur adipisicing elit. Volupta
9          earum aspernatur minima optio, quidem ex, laudantium ipsam vitae
10         praesentium deleniti laboriosam esse asperiores dignissimos modi
11         magnam. Eaque, repellendus! Repellat error quae amet sint pariatur
12         repellendus ipsam asperiores adipisci ullam illum obcaecati earum
13         molestiae ipsum libero quaerat facere neque, eligendi voluptate.
14         dolorem impedit culpa ea cum. Dolore eos, odit et itaque dolor re
15     </p>
16     <hr />
17     <h1>Text-align:justify</h1>
18     <p style="text-align: justify">
19         Lorem ipsum, dolor sit amet consectetur adipisicing elit. Volupta
20         earum aspernatur minima optio, quidem ex, laudantium ipsam vitae
21         praesentium deleniti laboriosam esse asperiores dignissimos modi
22         magnam. Eaque, repellendus! Repellat error quae amet sint pariatur
23         repellendus ipsam asperiores adipisci ullam illum obcaecati earum
24         molestiae ipsum libero quaerat facere neque, eligendi voluptate.
25         dolorem impedit culpa ea cum. Dolore eos, odit et itaque dolor re
26     </p>
27     </body>
28 </html>
29
30
31
32
33

```

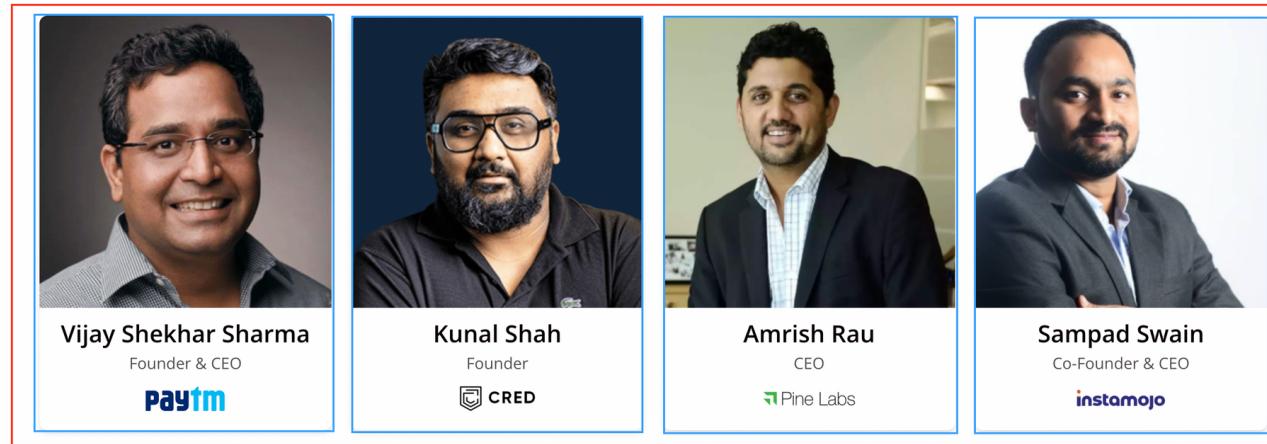
The browser preview shows two sections of text. The first section, under "Normal Alignment", has "text-align: left". The second section, under "Text-align:justify", has "text-align: justify". Both sections contain the same Lorem ipsum text.

CSS Box-Model:

- The CSS box model is a container that contains multiple properties including borders, margin, padding, and the content itself. It is used to create the design and layout of web pages. According to the CSS box model, the web browser supplies each element as a square prism.



Visualizing Box-model



- In the above figure, we have
 - one parent box(red-box)
 - four child boxes (blue-box)

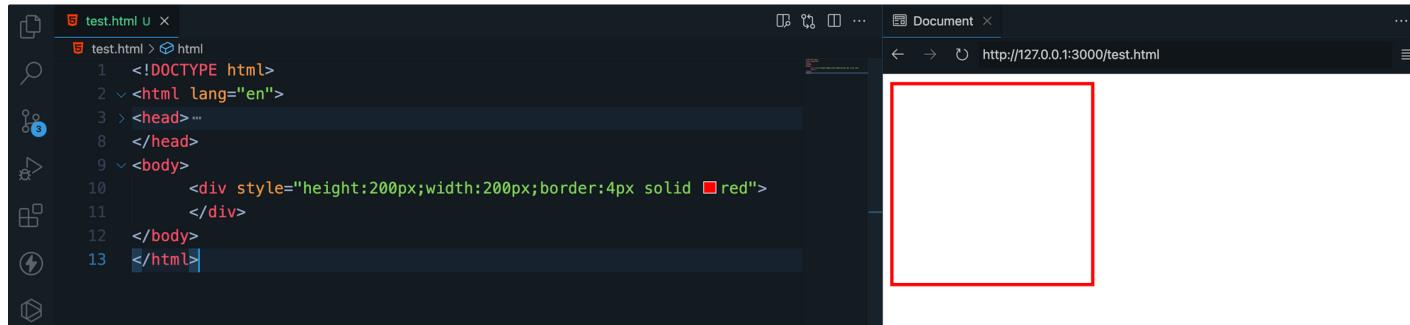
Pre-requisites of building box

- Height
- Width
- Border-color

Height & Width

- The CSS `height` and `width` properties are used to set the height and width of an element.

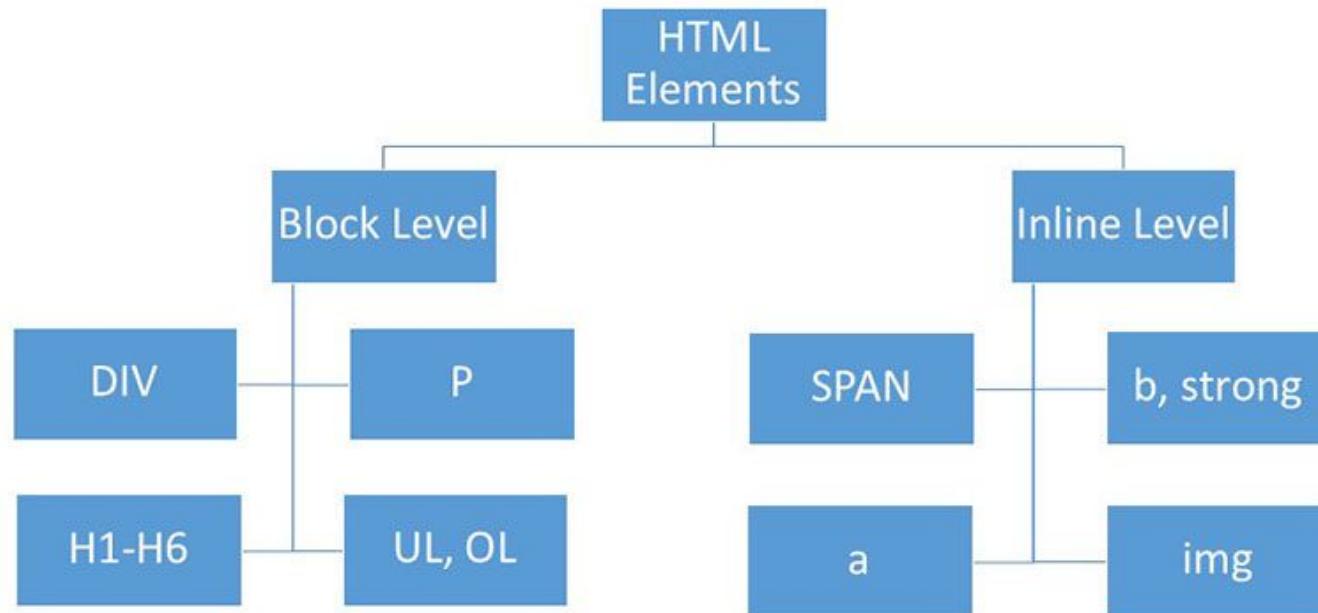
- The `height` and `width` properties may have the following values:
 - `auto` - This is default. The browser calculates the height and width
 - `length` - Defines the height/width in px, cm etc.
 - `%` - Defines the height/width in percent of the containing block



Block vs inline elements

Block	Inline
The element will start on a new line and occupy the full width available. And you can set width and height values.	The element does not occupy the full width it requires.
Block elements accept all the margins	Inline elements accept top and

Various block and inline elements



test.html

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>...
4  </head>
5  <body>
6      <h1>Block elements</h1>
7      <p>I am paragraph</p>
8      <div>I am box</div>
9      <h4>I am heading-4</h4>
10     <h2>I am heading-2</h2>
11
12     <h1>Inline elements</h1>
13     <span>I am inline element</span>
14     <b>I am bold text</b>
15     <a href="#">I am link</a>
16
17 </body>
18 </html>
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
  
```

Document

Block elements

- I am paragraph
- I am box
- I am heading-4
- I am heading-2

Inline elements

- I am inline element
- I am bold text
- I am link

Properties of the Box Model

Content

- The content area consists of content like image, text, or other forms of media content. The height and width properties help to modify the box dimensions.

Padding

- The padding area is the space around the content area and within the border-box. It can be applied to all sides of the box or to the specific, selected side(s) - top, right, bottom, and/or left.

Border

- The border area surrounds the padding and the content, and can be applied to all the sides of the box or to selected side(s) - top, right, bottom, and/or left.

Margin

- The margin area consists of space between the border and the margin. The margin does not possess its own background color and is completely transparent. It shows the background color of the element, like the body element.

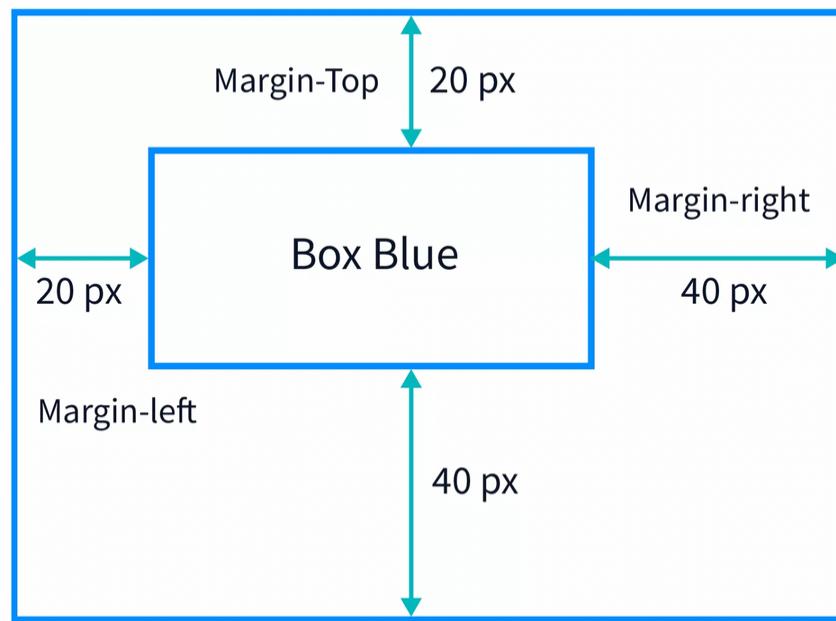
What is Margin in CSS?

- Margin is a CSS property that defines the empty space around an HTML element. Margins ensure that the specified region around an element remains unoccupied by any neighboring element.

Syntax

There are four margin properties, one for each side of the HTML element box.

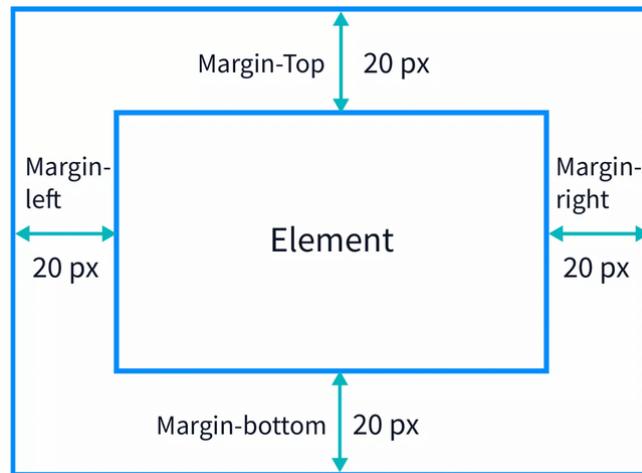
- **margin-top:** adds margin space on top of the element.
- **margin-right:** adds margin space on the right of the element.
- **margin-bottom:** adds margin space on the bottom of the element.
- **margin-left:** adds margin space on the left of the element.



- In above example, we are setting a margin of 20px on the top and left sides and also a margin of 40px on the right and bottom side.

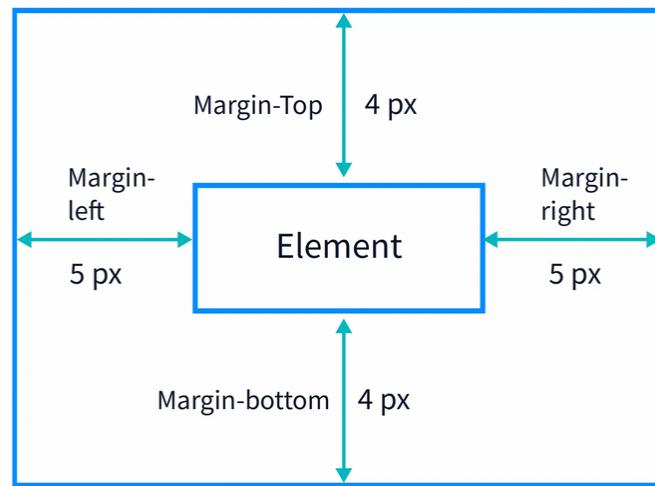
CSS Margin Shorthand Property

- Single Value (margin:20px) :
 - The single value is set as margin on all four sides.

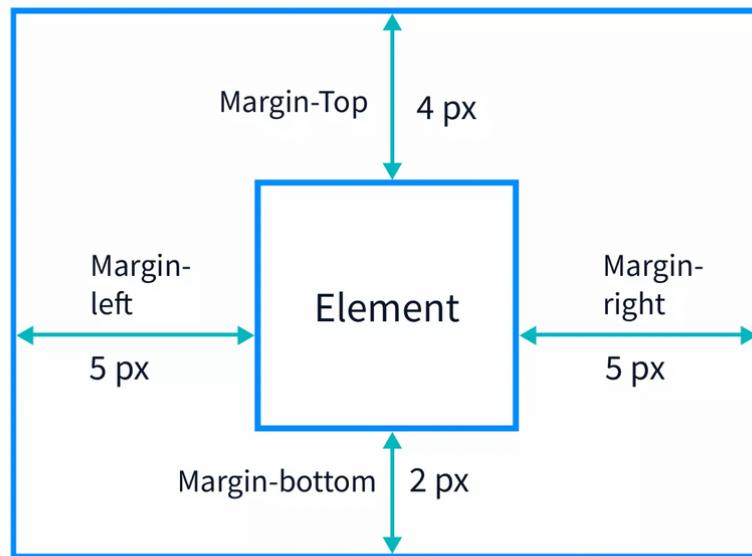


- Two Values (margin:4px 5px):

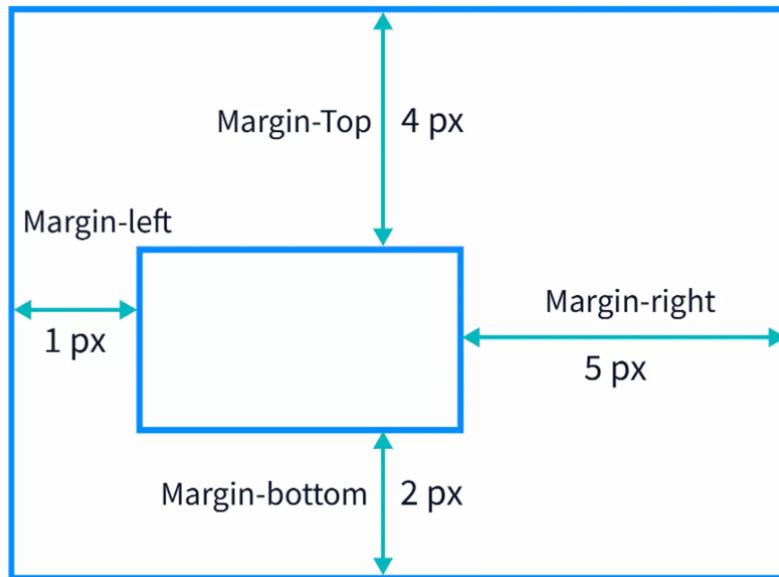
- The first value is set as vertical margin (top & bottom) while the second is set as horizontal margin (right & left).



- Three Values (margin:4px 5px 2px):
 - The first value is set as top margin, second as horizontal margin (right & left) while the third is set as bottom margin.



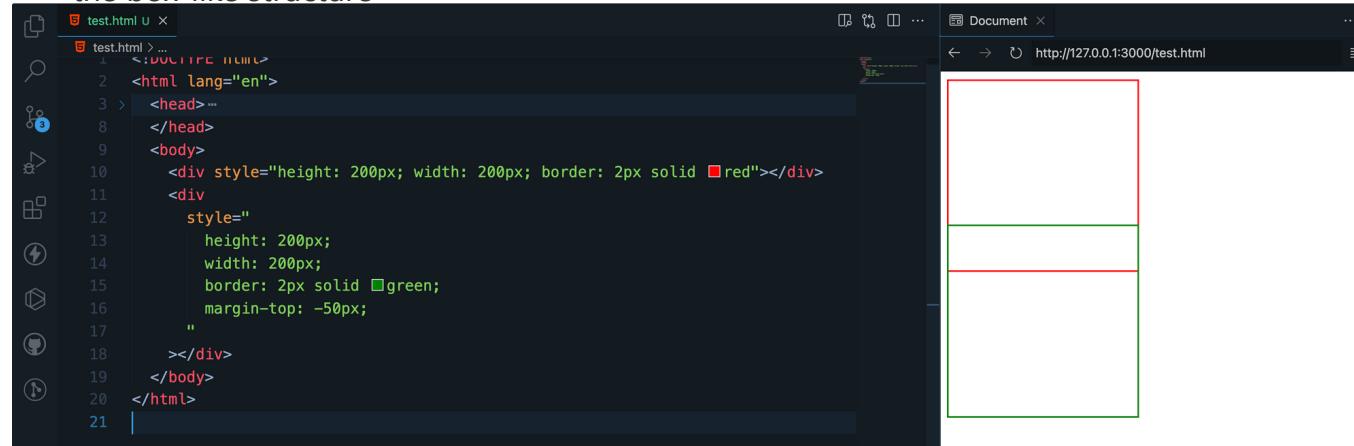
- Four values (margin:4px 5px 2px 1px):
 - The values are assigned to margins on sides starting from top and moving in a clockwise direction, i.e. the four values are set as top, right, bottom and left margin respectively.



Negative margin:

- Positive margin values move the content towards the inner side of its position or page. But

if we take negative margin values, it moves the content towards outside of its position or page. The margin property in html gives space around the outermost element's content of the box-like structure



The screenshot shows a code editor on the left and a browser preview on the right. The code editor displays the following HTML and CSS:

```
test.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head> ...
4 </head>
5 <body>
6   <div style="height: 200px; width: 200px; border: 2px solid red"></div>
7   <div
8     style=""
9       height: 200px;
10      width: 200px;
11      border: 2px solid green;
12      margin-top: -50px;
13    "
14   ></div>
15 </body>
16 </html>
```

The browser preview shows a red-bordered box at the top and a green-bordered box below it. The green box is positioned 50px above the bottom of the red box, demonstrating the effect of a negative margin.

- A funny example of negative margin



Extensions used :

Name: Live Preview

Id: ms-vscode.live-server

Description: Hosts a local server in your workspace for you to preview your webpages on.

Version: 0.2.12

Publisher: Microsoft

VS Marketplace Link: <https://marketplace.visualstudio.com/items?itemName=ms-vscode.live-server>