



Web programming I

Section 7

Eng/ Mona AbdelAziz

The Element Box

In CSS, every HTML element is considered a **box**, and that box is made up of **four layers**:

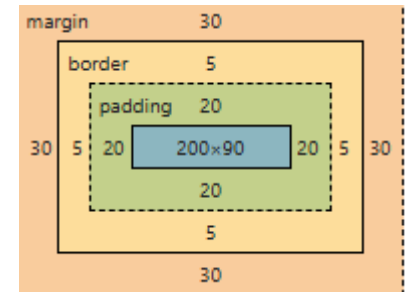
- **Content** → the actual text or image inside.
- **Padding** → space between content and border.
- **Border** → the frame around the padding.
- **Margin** → space outside the border, separating elements.

example

```
12.html > html > head > style > div
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>CSS Box Model</title>
6 <style>
7 div {
8   width: 200px;           /* content width */
9   background-color: #ffe4b5; /* content background */
10
11   padding: 20px;          /* inside spacing */
12   border: 5px solid orange; /* border thickness */
13   margin: 30px;           /* space outside */
14 }
15 </style>
16 </head>
17
18 <body>
19 <h2>CSS Box Model Example</h2>
20 <div>
21   This is a simple box.
22   The text inside is the **content**,
23   around it is the **padding**,
24   then the **border**,
25   and outside is the **margin**.
26 </div>
27 </body>
28 </html>
29
```

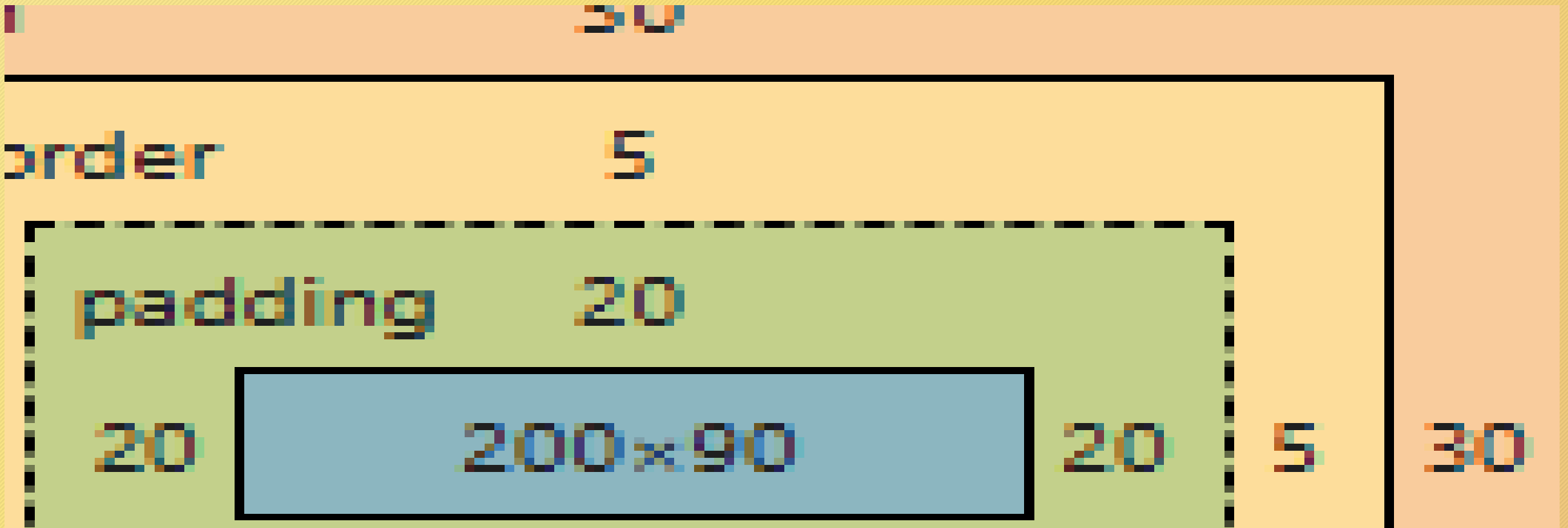
CSS Box Model Example

This is a simple box. The text inside is the **content**, around it is the **padding**, then the **border**, and outside is the **margin**.



CSS Padding

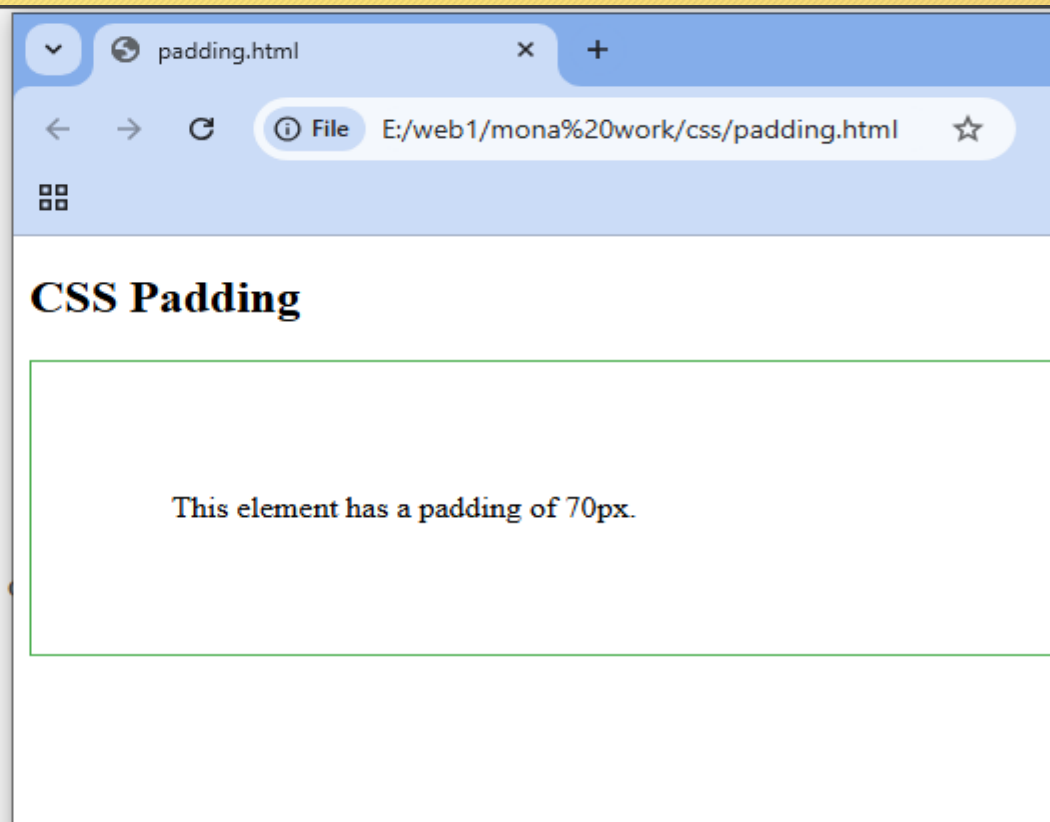
- Padding is used to create space around an element's content, inside of any defined borders.



CSS Margins

- Margins are used to create space around elements, outside of any defined borders.

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  div {
6    padding: 70px;
7    border: 1px solid #4CAF50;
8  }
9  </style>
10 </head>
11 <body>
12
13 <h2>CSS Padding</h2>
14 <div>This element has a padding
15
16 </body>
17 </html>
```



Total Element Size

- Total width of an element =
width + padding-left + padding-right + border-left + border-right +
margin-left + margin-right

example

Help

E:/web1/mona work/css/box model.html (Getting Started) - Brack

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <style>
5 div{
6     background-color: lightgrey;
7     width: 300px;
8     border: 15px solid green;
9     padding: 50px;
10    margin: 20px;
11 }
12 </style>
13 </head>
14 <body>
15
16 <h2>Demonstrating the Box Model</h2>
17
18 <p>The CSS box model is essentially a box that wraps around every HTML
19 of: borders, padding, margins, and the actual content.</p>
20 <div>This text is the content of the box. We have added a 50px padding, 20px margin and a
21 15px green border</div>
22 </body>
23 </html>
```

box model.html

File E:/web1/m...

← → ↺ ⚙

☰

All Bookmarks

Demonstrating the Box Model

The CSS box model is essentially a box that wraps around every HTML element. It consists of: borders, padding, margins, and the actual content.



CSS Display Property

- The CSS display property is used to adjust the layout of an element(whether it appears as a block, inline, flex container, grid, etc.)
- the syntax of display Display:value
- **Inline:** Displays an element as an inline element
- **Block:** Displays an element as block element
- **Inline-block :** formats the element as inline-level but also allows to set height/width like block-level element
- **flex :** sets the element as a flex container to have a flexible layout of its child elements (one column or one row)
- **Grid :** sets the element as a grid container to create complex layouts (both column and row)

In-line example

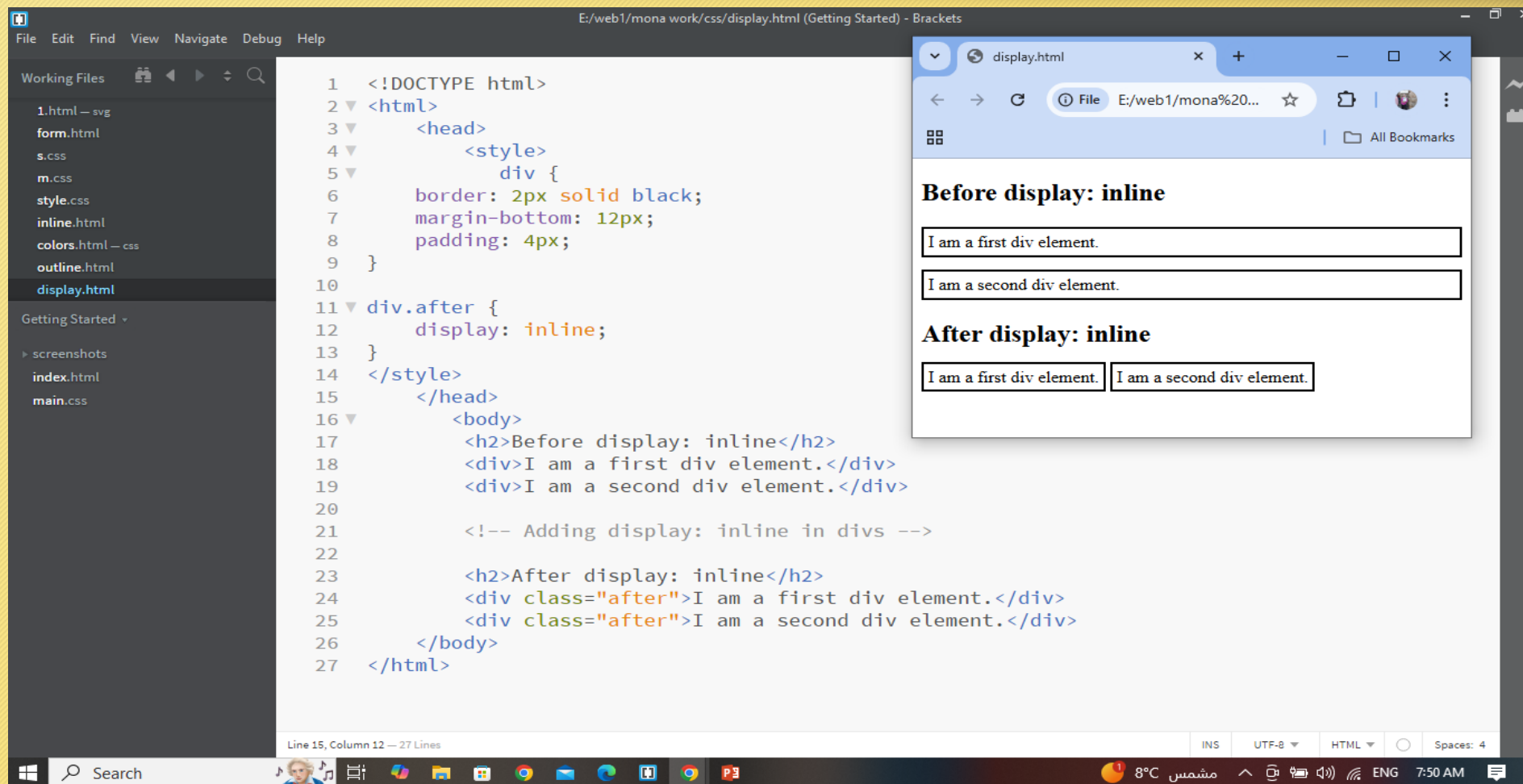
The image shows a Brackets code editor window with the file `display.html` open. The code defines a CSS class `div.after` with `display: inline;` and uses it in the HTML body to demonstrate inline display.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <style>
5       div {
6         border: 2px solid black;
7         margin-bottom: 12px;
8         padding: 4px;
9       }
10
11   <div.after {
12     display: inline;
13   }
14   </style>
15   </head>
16   <body>
17     <h2>Before display: inline</h2>
18     <div>I am a first div element.</div>
19     <div>I am a second div element.</div>
20
21     <!-- Adding display: inline in divs -->
22
23     <h2>After display: inline</h2>
24     <div class="after">I am a first div element.</div>
25     <div class="after">I am a second div element.</div>
26   </body>
27 </html>
```

The browser window on the right shows the rendered output. Under the heading "Before display: inline", two separate div elements are shown, each with a black border and padding. Under the heading "After display: inline", the same two div elements are shown, but they are now inline with each other, sharing the same line of text.

Line 15, Column 12 — 27 Lines

CSS Display Block



The screenshot shows the Brackets code editor with a file named `display.html` open. The code defines a CSS style for a `div` element with a black border, 12px bottom margin, and 4px padding. It then demonstrates the effect of `display: inline` on two div elements.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <style>
5       div {
6         border: 2px solid black;
7         margin-bottom: 12px;
8         padding: 4px;
9       }
10
11   <div class="after">
12     <div>I am a first div element.</div>
13     <div>I am a second div element.</div>
14   </div>
15   <div class="after">
16     <div>I am a first div element.</div>
17     <div>I am a second div element.</div>
18   </div>
19   <div class="after">
20     <div>I am a first div element.</div>
21     <div>I am a second div element.</div>
22   </div>
23   <div class="after">
24     <div>I am a first div element.</div>
25     <div>I am a second div element.</div>
26   </div>
27 </html>
```

The browser window shows the rendered output. The first two div elements are stacked vertically, each with a black border and padding. The second two div elements are displayed inline, side-by-side, each with a black border and padding.

Before display: inline

I am a first div element.

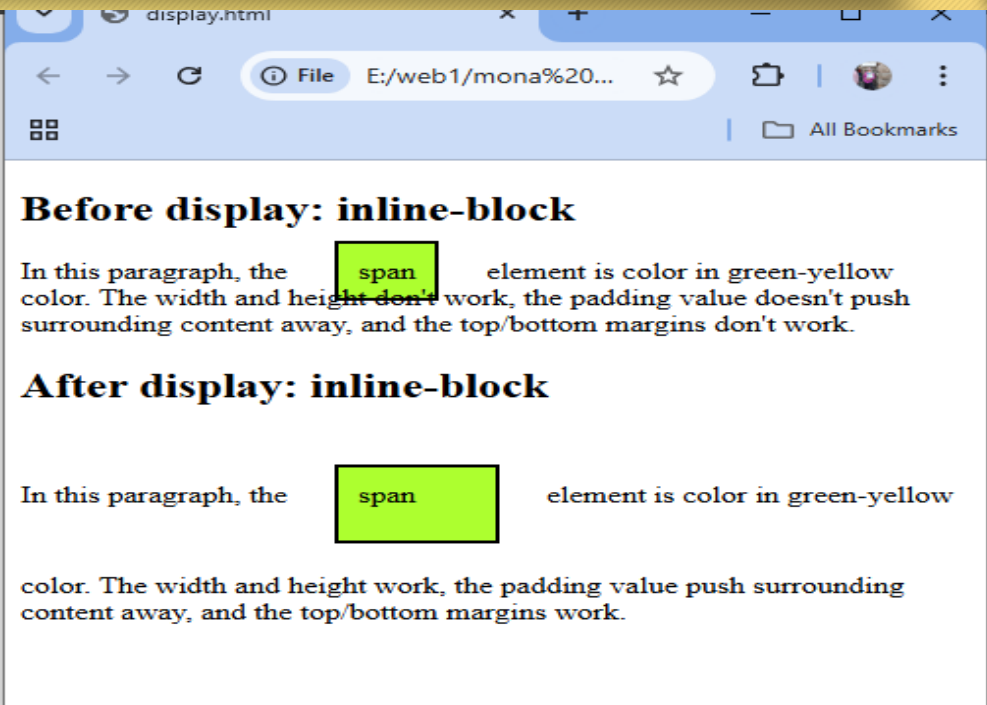
I am a second div element.

After display: inline

I am a first div element. I am a second div element.

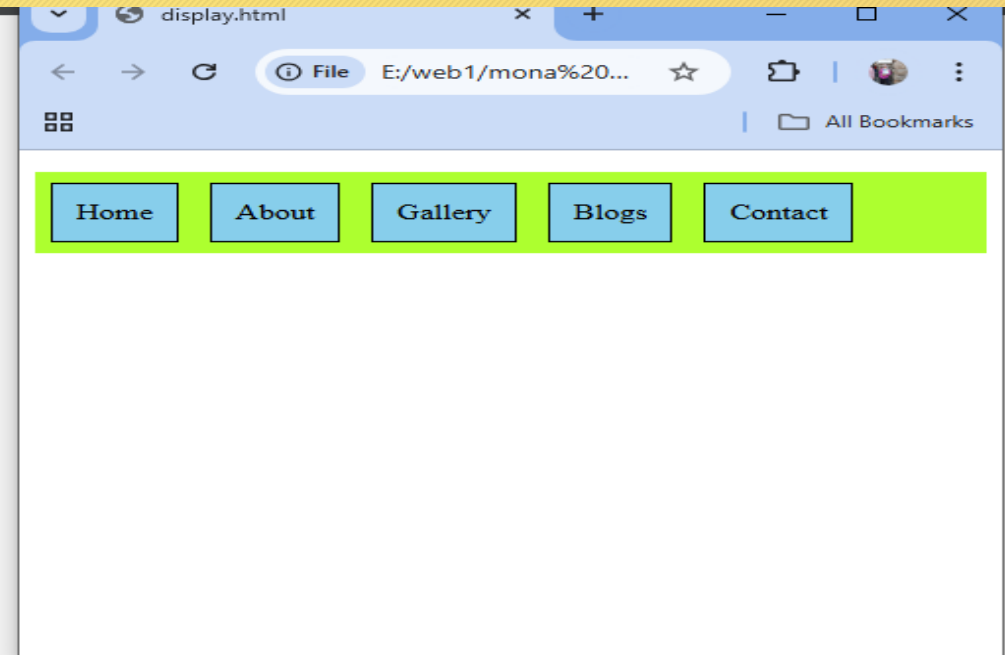
CSS inline-block element

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  span {
6    width: 60px;
7    height: 30px;
8    padding: 10px;
9    margin: 20px;
10
11    border: 2px solid black;
12    background-color: greenyellow;
13  }
14  span.after {
15    display: inline-block;
16  }
17  </style>
18  </head>
19  <body>
20  <h2>Before display: inline-block</h2>
21  <p>
22    In this paragraph, the <span>span</span> element is color in
23    green-yellow color. The width and height don't work, the padding
24    value doesn't push surrounding content away, and the top/bottom
25    margins don't work.
26  </p>
27  <!-- Adding display: inline-block the divs -->
28  <h2>After display: inline-block</h2>
29  <p>
```



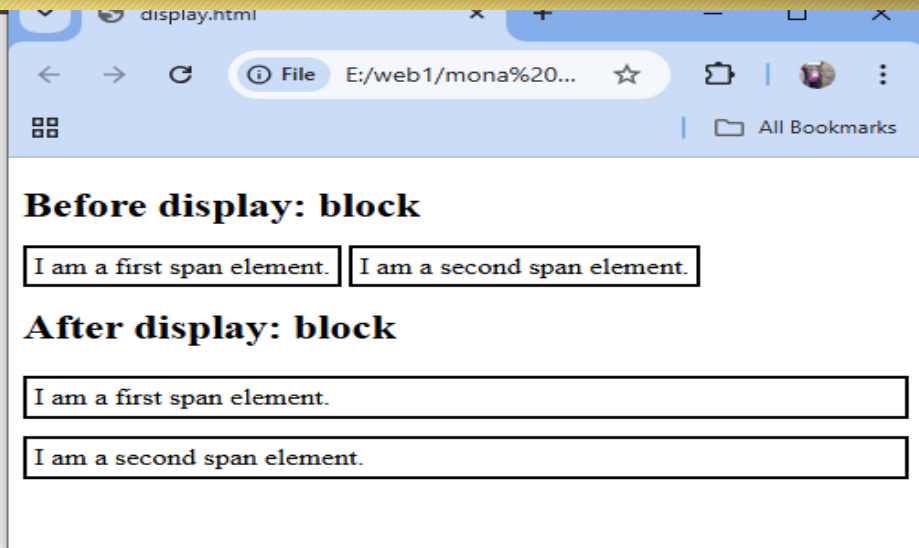
CSS flex

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  ul.parent {
6    display: flex;
7    background-color: greenyellow;
8    padding: 0px;
9  }
10
11 li {
12   background: skyblue;
13   border: 1px solid black;
14   padding: 12px;
15   margin: 8px;
16   list-style: none;
17 }
18 </style>
19 </head>
20 <body>
21 <ul class="parent">
22   <li>Home</li>
23   <li>About</li>
24   <li>Gallery</li>
25   <li>Blogs</li>
26   <li>Contact</li>
27 </ul>
28 </body>
29 </html>
```



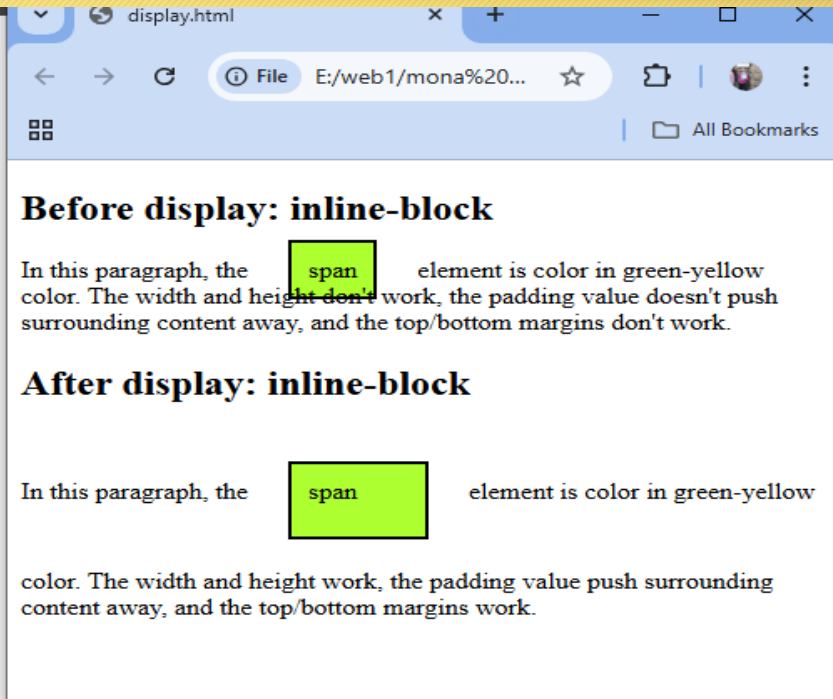
CSS inline-block element

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <style>
5       span {
6         border: 2px solid black;
7         margin-bottom: 12px;
8         padding: 4px;
9       }
10
11   <span.after {
12     display: block;
13   }
14   </style>
15   </head>
16   <body>
17     <h2>Before display: block</h2>
18     <span>I am a first span element.</span>
19     <span>I am a second span element.</span>
20
21     <!-- Adding display: block in spans-->
22
23     <h2>After display: block</h2>
24     <span class="after">I am a first span element.</span>
25     <span class="after">I am a second span element.</span>
26   </body>
27 </html>
```



CSS flex

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <style>
5       span {
6         width: 60px;
7         height: 30px;
8         padding: 10px;
9         margin: 20px;
10
11         border: 2px solid black;
12         background-color: greenyellow;
13       }
14   span.after {
15     display: inline-block;
16   }
17 </style>
18 </head>
19 <body>
20   <h2>Before display: inline-block</h2>
21   <p>
22     In this paragraph, the <span>span</span> element is color in
23     green-yellow color. The width and height don't work, the padding
24     value doesn't push surrounding content away, and the top/bottom
25     margins don't work.
26   </p>
27   <!-- Adding display: inline-block the divs -->
28   <h2>After display: inline-block</h2>
29   <p>
```



After display: inline-block

In this paragraph, the span element is color in green-yellow color. The width and height work, the padding value push surrounding content away, and the top/bottom margins work.

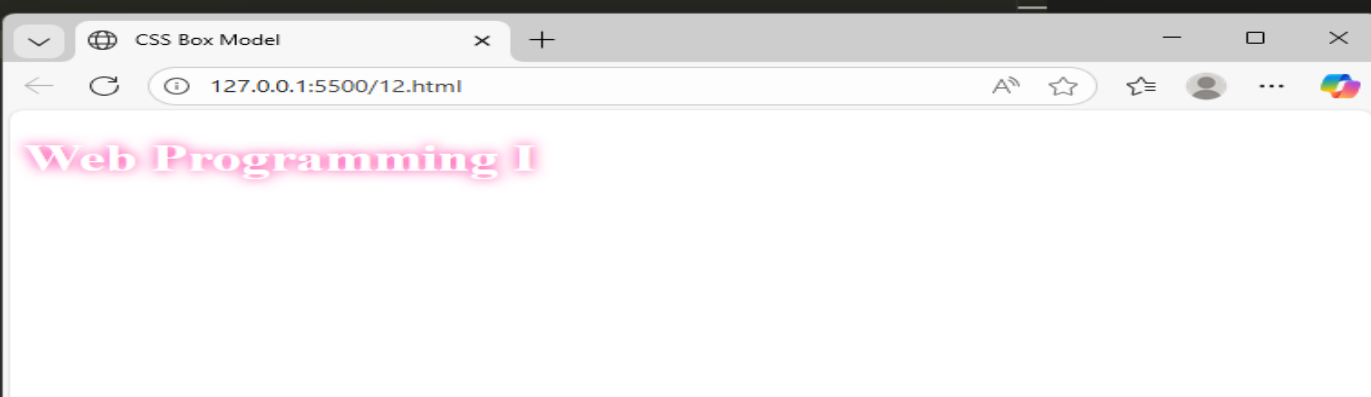
CSS Shadow

- With CSS you can add shadow to text and to elements.
- In these chapters you will learn about the following properties:
 - ✓ text-shadow
 - ✓ box-shadow

Text shadow

- Text-shadow adds a shadow behind the text. It is used to create effects like glow, depth, neon, 3D, or blurry soft shadows.
- Syntax: *text-shadow: offset-x offset-y blur-radius color;*

```
<> 12.html > html > head > style > h1
2   <html lang="en">
3   <head>
4   <title>CSS Box Model</title>
5   <style>
6   h1 {
7     color: rgb(255, 255, 255);
8     text-shadow: 0 0 10px deeppink;
9   }
10
11
12
13 </style>
14 </head>
15
16 <body>
17   <h1>Web Programming I</h1>
18
19 </body>
20 </html>
21
```



The screenshot shows a web browser window with the title 'CSS Box Model' and the URL '127.0.0.1:5500/12.html'. The main content of the page is the text 'Web Programming I', which is displayed in a pink color with a soft, pink shadow effect, demonstrating the CSS text-shadow property.

10 effects example

```
<? 12.html > <? html > <? head > <? style > <? .fire
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="UTF-8">
5  <title>Text Shadow Effects</title>
6  <style>
7  body {
8      background-color: #000;
9      font-family: Arial, sans-serif;
10     text-align: center;
11     padding: 40px;
12     color: white;
13 }
14
15 /* ===== 1. Soft Shadow ===== */
16 .soft {
17     font-size: 50px;
18     text-shadow: 2px 2px 5px rgba(255,255,255,0.3);
19 }
20
21 /* ===== 2. Neon Pink ===== */
22 .neon {
23     font-size: 50px;
24     color: #ff66cc;
25     text-shadow: 0 0 10px #ff66cc, 0 0 20px #ff66cc, 0 0 40px #ff0088;
26 }
27
28 /* ===== 3. 3D Shadow ===== */
29 .threed {
30     font-size: 50px;
31     text-shadow: 4px 4px 0 #555, 8px 8px 0 #777;
32 }
33
34 /* ===== 4. Glow Blue ===== */
35 .glow {
36     font-size: 50px;
37     color: #66ccff;
38     text-shadow: 0 0 15px #66ccff;
39 }
40
41 /* ===== 5. Retro Shadow ===== */
42 .retro {
43     font-size: 50px;
44     color: #ffdd00;
45     text-shadow: 4px 4px 0px #ff5500;
46 }
47
48 /* ===== 6. Embossed (Raised Text) ===== */
49 .emboss {
50     font-size: 50px;
51     text-shadow: -2px -2px 3px #fff, 2px 2px 3px #444;
52 }
53
54 /* ===== 7. Outer Glow White ===== */
55 .whiteglow {
56     font-size: 50px;
57     color: #fff;
58     text-shadow: 0 0 15px rgba(255,255,255,0.8);
59 }
60
61 /* ===== 8. Double Shadow ===== */
62 .double {
63     font-size: 50px;
64     text-shadow: 3px 3px 5px #ff00aa, -3px -3px 5px #00ccff;
65 }
66
67 /* ===== 9. Deep Shadow ===== */
68 .deep {
69     font-size: 50px;
70     text-shadow: 6px 6px 10px rgba(0,0,0,0.8);
71 }
72
73 /* ===== 10. Fire Text ===== */
74 .fire {
75     font-size: 50px;
76     color: #ff3300;
77     text-shadow:
78         0 0 10px #ff6600,
79         0 0 20px #ff3300,
80         0 0 30px #ff0000,
81         0 0 40px #ff2200;
82 }
83
84 </style>
85 </head>
86
87 <body>
88
89     <h1 class="soft">Soft Shadow</h1>
90     <h1 class="neon">Neon Pink</h1>
91     <h1 class="threed">3D Shadow</h1>
92     <h1 class="glow">Glow Blue</h1>
93     <h1 class="retro">Retro Shadow</h1>
94     <h1 class="emboss">Embossed</h1>
95     <h1 class="whiteglow">White Glow</h1>
96     <h1 class="double">Double Shadow</h1>
97     <h1 class="deep">Deep Shadow</h1>
98     <h1 class="fire">Fire Glow</h1>
99
100 </body>
101 </html>
```

output

Soft Shadow

Neon Pink

3D Shadow

Glow Blue

Retro Shadow

Embossed

White Glow

Double Shadow

Deep Shadow

Fire Glow

Box shadow

- The box-shadow property adds a **shadow effect** around an element's box. You can control its **position, blur, spread, and color**.
- syntax

box-shadow: offset-x offset-y blur-radius spread-radius color;

example

Simple

Glow

Deep

Inset

Double

Neon

Orange

Lift

Long

Border

```
.box {
  width: 220px;
  height: 120px;
  background: white;
  border-radius: 10px;
  display: flex;
  align-items: center;
  justify-content: center;
  font-weight: bold;
  font-size: 20px;
}

/* 1 - Simple */
.s1 { box-shadow: 5px 5px 10px gray; }

/* 2 - Soft Glow */
.s2 { box-shadow: 0 0 15px rgba(0,0,0,0.3); }

/* 3 - Deep Shadow */
.s3 { box-shadow: 10px 10px 25px rgba(0,0,0,0.4); }

/* 4 - Inner Shadow */
.s4 { box-shadow: inset 0 0 10px #555; }

/* 5 - Double Shadow */
.s5 { box-shadow: 5px 5px 15px #999, -5px -5px 15px #eee; }

/* 6 - Neon Glow */
.s6 { box-shadow: 0 0 20px #ff66cc; }

/* 7 - Soft Orange */
.s7 { box-shadow: 0 0 20px rgba(255,150,50,0.8); }

/* 8 - Lift Effect */
.s8 { box-shadow: 0 15px 25px rgba(0,0,0,0.3); }

/* 9 - Flat Long Shadow */
.s9 { box-shadow: 20px 20px 0 rgba(0,0,0,0.2); }

/* 10 - Border-like Shadow */
.s10 { box-shadow: 0 0 0 5px #ffcc66; }
```