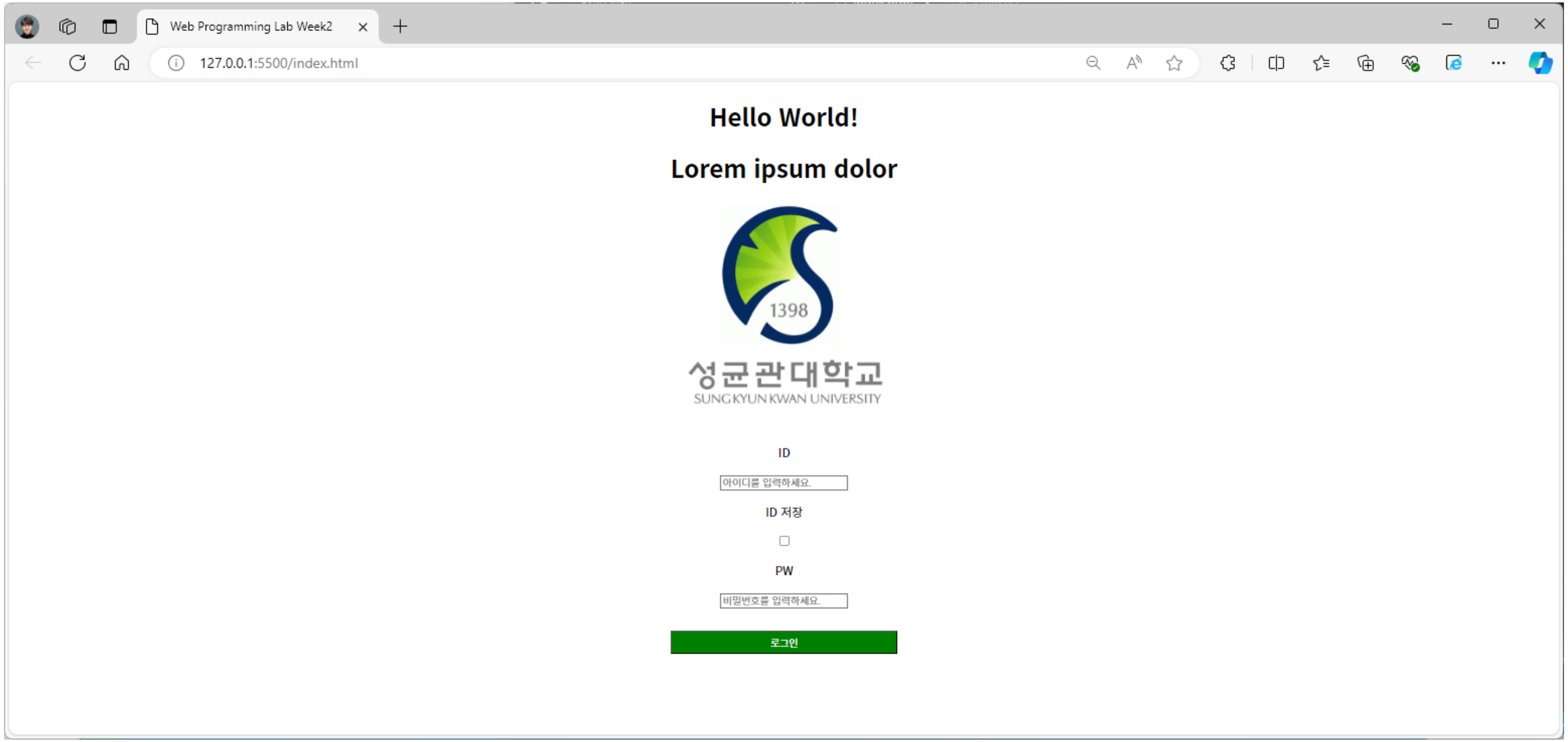
Web Programming Lab

HTML & CSS

Week 3

Week 2 Result


Web Programming Lab Week2
127.0.0.1:5500/index.html

Hello World!

Lorem ipsum dolor

1398
성균관대학교
SUNG KYUN KWAN UNIVERSITY

ID

아이디를 입력하세요.

ID 저장

□

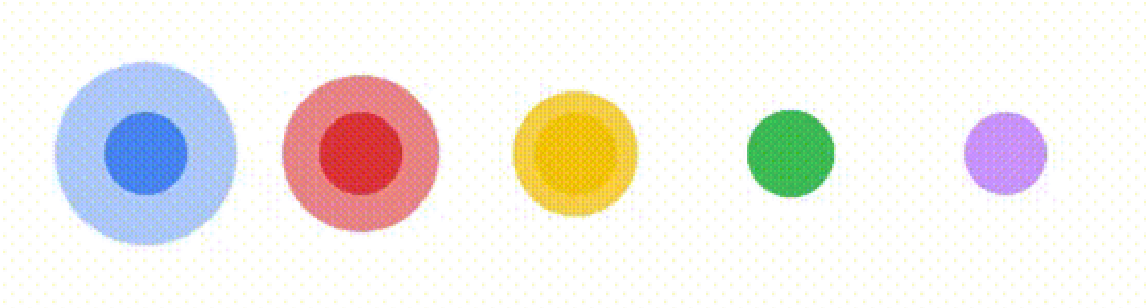
PW

비밀번호를 입력하세요.

로그인

Contents

- Box Model
    - Margin and Padding
- CSS Layout
- Element Positioning
- CSS Flexible Box Layout

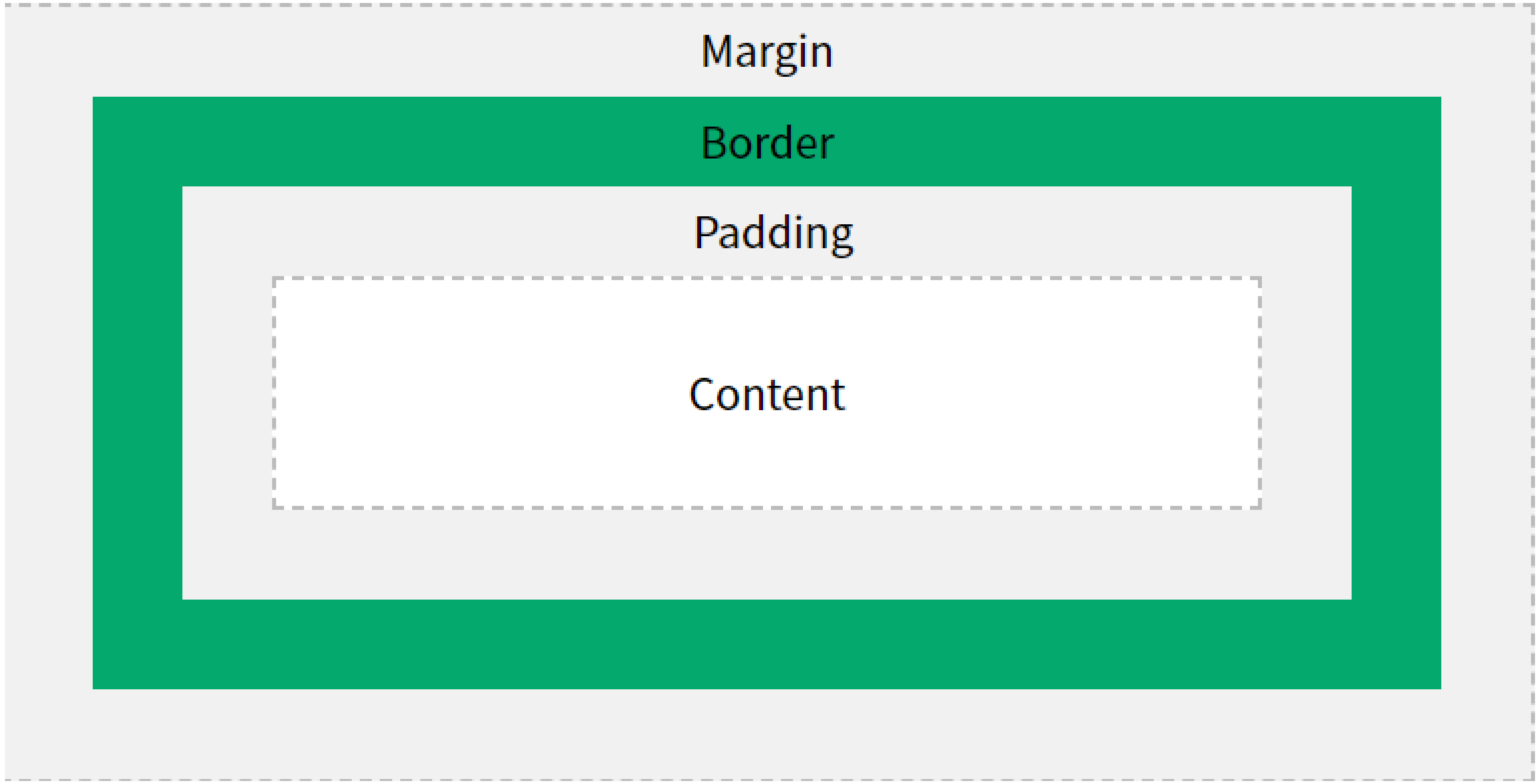


Box Model

Box Model

- Content : Text and images appear
- Padding : The area around the content. Padding is transparent
- Border : The border around the padding and contents. Affected by the border color of the box
- Margin : Area around the border. Margin is transparent

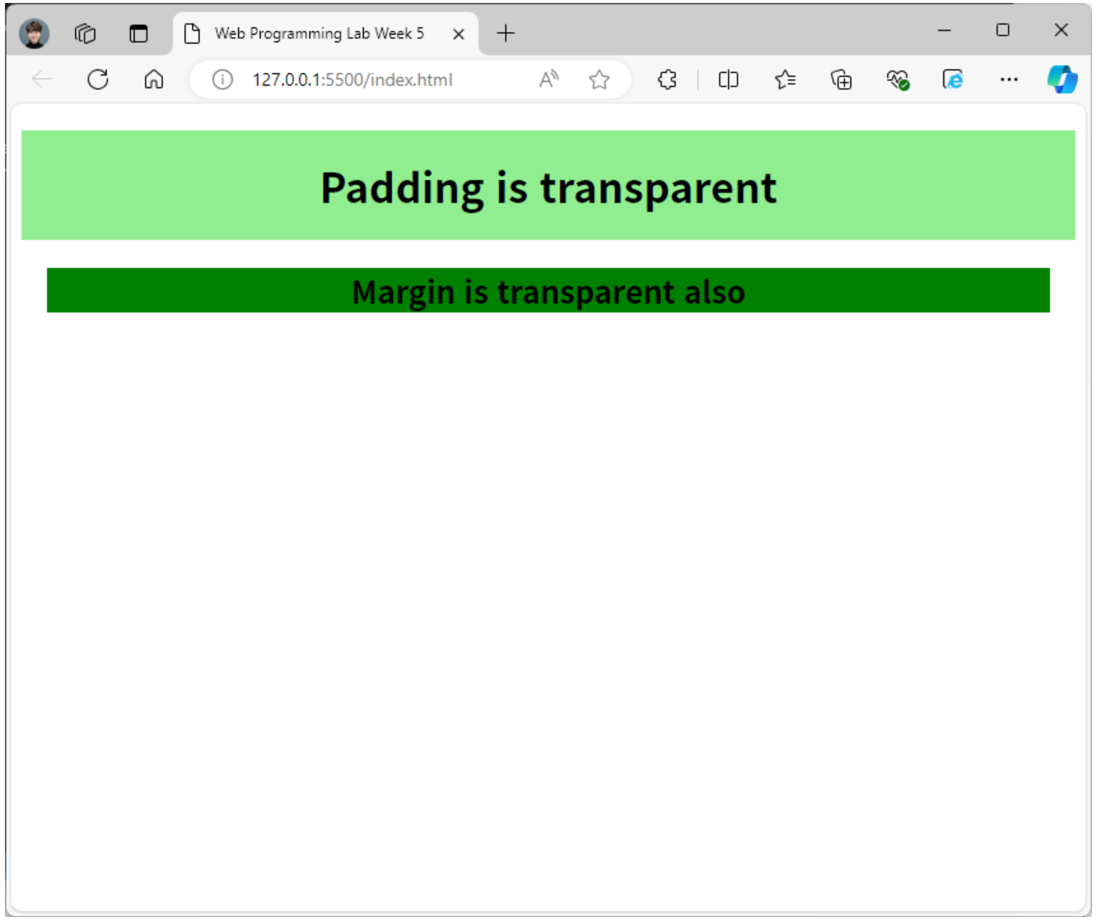
Box Model



Box Model

<index.html>

```
<!DOCTYPE html>
<head>
    <style>
        h1 {
            padding : 20px;
            background-color: lightgreen;
        }
        h2 {
            margin: 20px;
            background-color: green;
        }
    </style>
</head>
<body>
    <h1><center>Padding is transparent</center></h1>
    <h2><center>Margin is transparent also</center></h2>
</body>
```

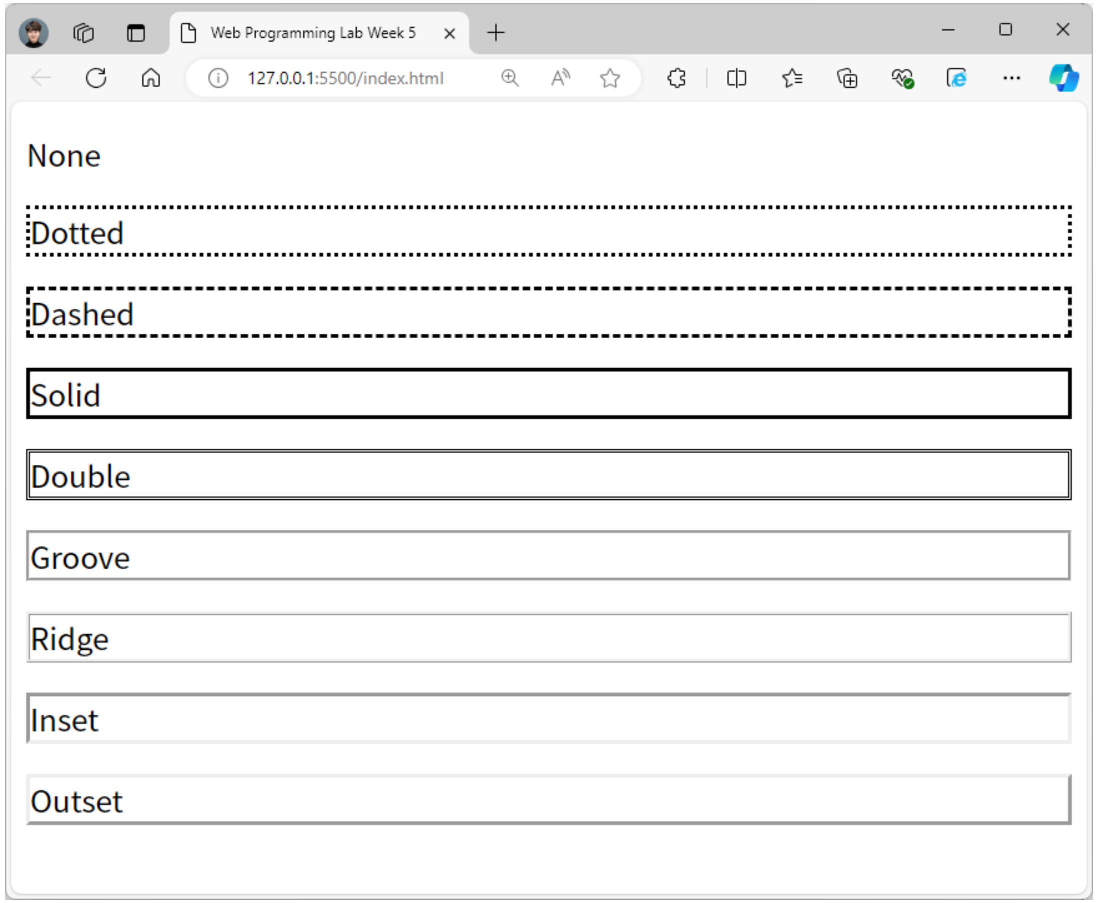


Padding is transparent

Margin is transparent also

Borderline Style

- border-style : none
- border-style : dotted
- border-style : dashed
- border-style : solid
- border-style : double
- border-style : groove
- border-style : ridge
- border-style : inset
- border-style : outset



None

Dotted

Dashed

Solid

Double

Groove

Ridge

Inset

Outset

Borderline Style

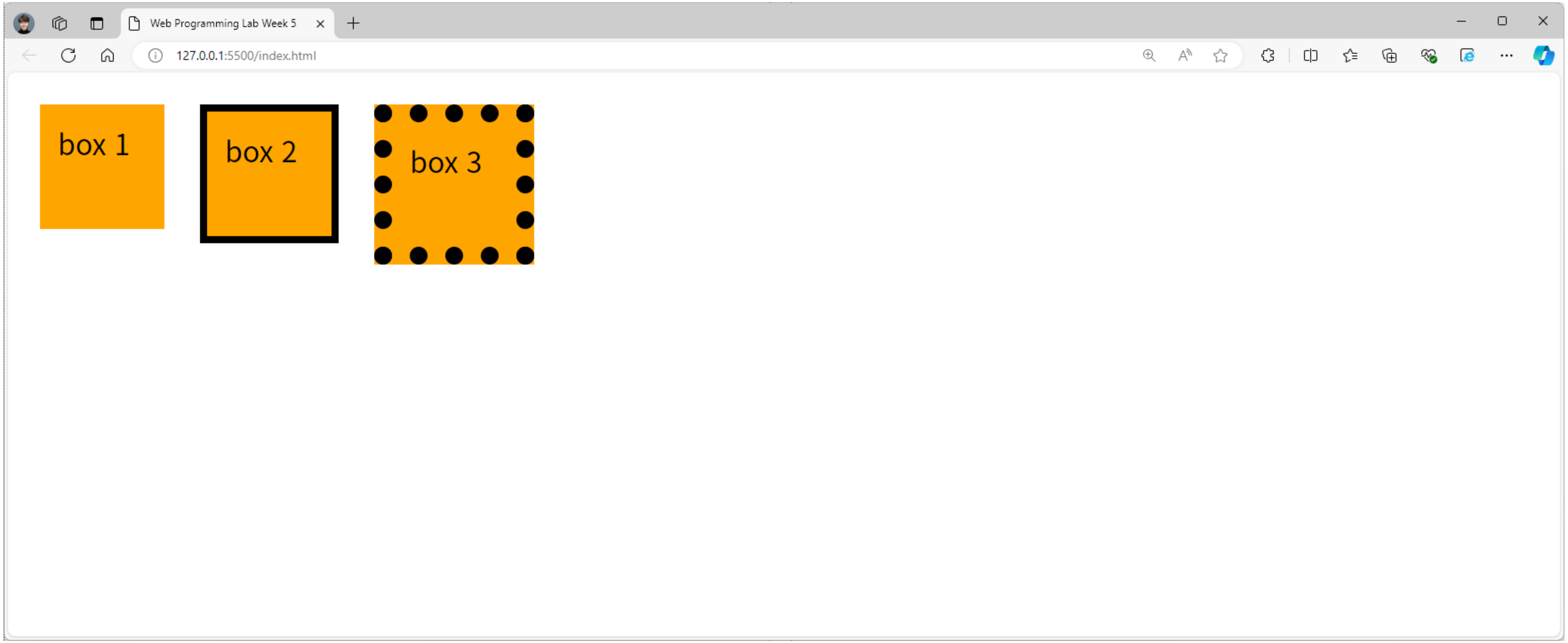
<index.html>

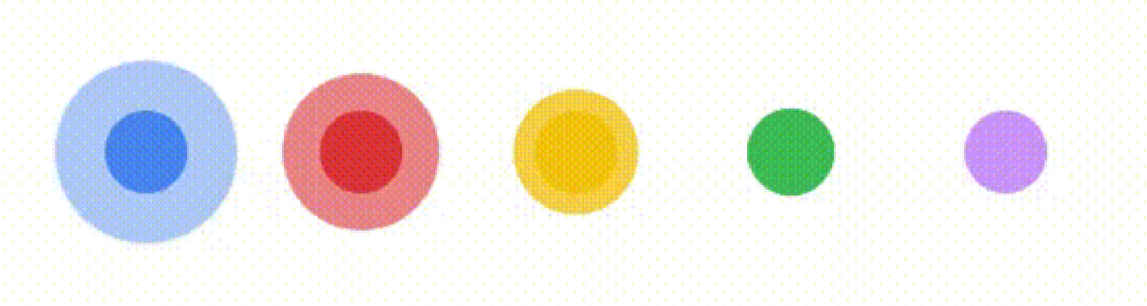
```
<!DOCTYPE html>
<head>
    <title>Web Programming Lab Week 5</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <p id="one">box 1</p>
    <p id="two">box 2</p>
    <p id="three">box 3</p>
</body>
```

<style.css>

```
body {
    display: flex;
}
p {
    width : 50px; height : 50px;
    margin : 10px;
    padding : 10px;
    background-color: orange;
}
#one {
    border : medium none black;
}
#two {
    border : 4px solid black;
}
#three {
    border : 10px dotted black;
}
```

Borderline Style

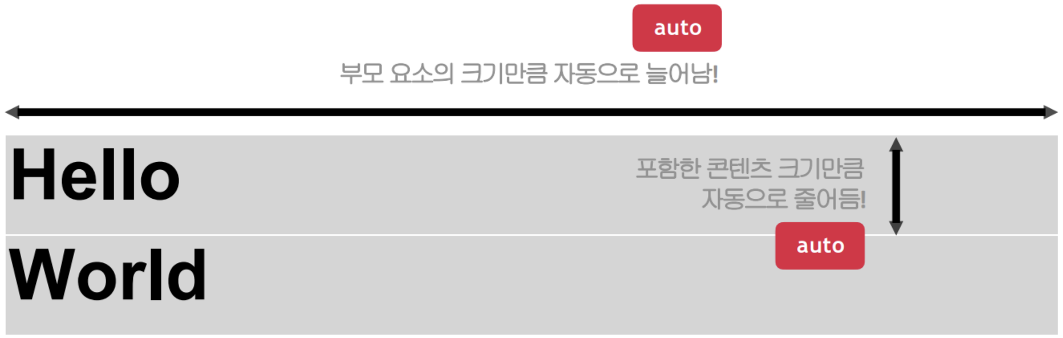




CSS Layout

Block and Inline Element

Block Element



auto

부모 요소의 크기만큼 자동으로 늘어남!

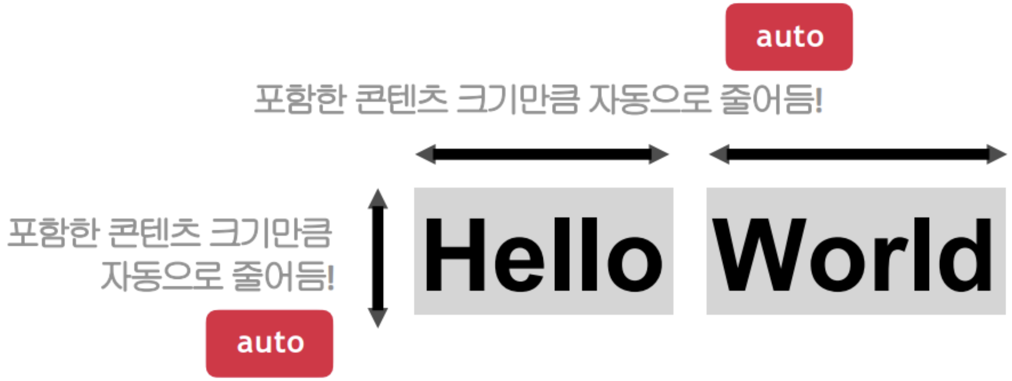
Hello

포함한 콘텐츠 크기만큼
자동으로 줄어듬!

auto

World

Inline Element



auto

포함한 콘텐츠 크기만큼 자동으로 줄어듬!

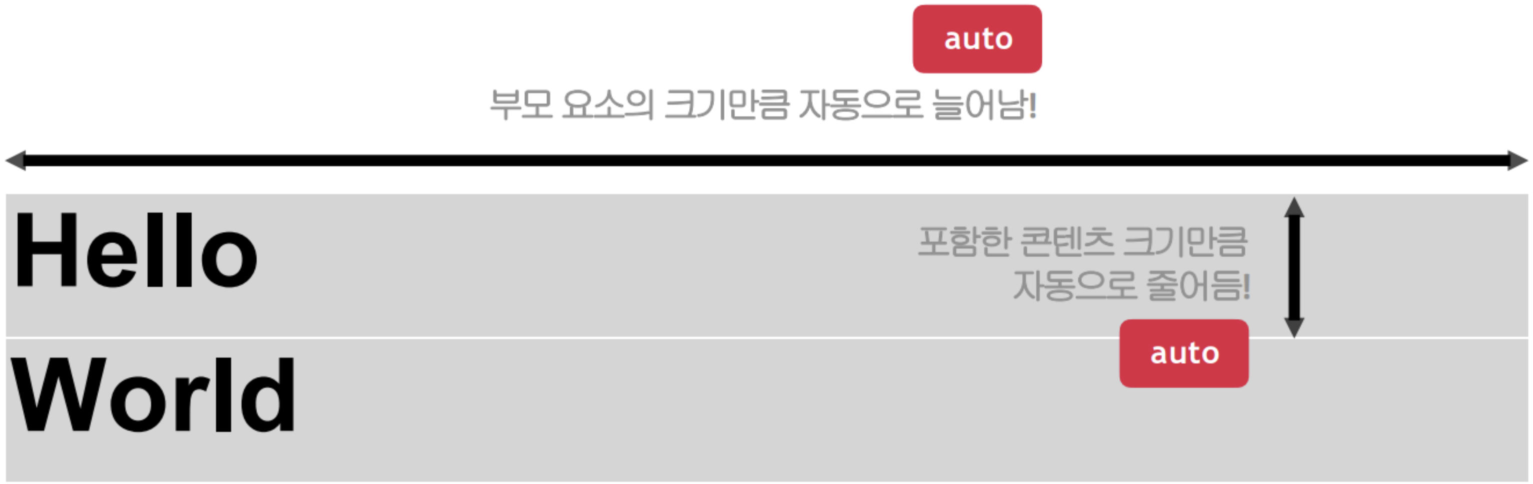
포함한 콘텐츠 크기만큼
자동으로 줄어듬!

auto

Hello World

Block Element

- Form in which a line break occurs by default and the width of the area becomes the entire width of the upper area
- tag : <h1>, <p>, <ul>, <li>, <table>, <blockquote>, <pre>, <div>, <form>, <header>, <nav>



auto

부모 요소의 크기만큼 자동으로 늘어남!

Hello

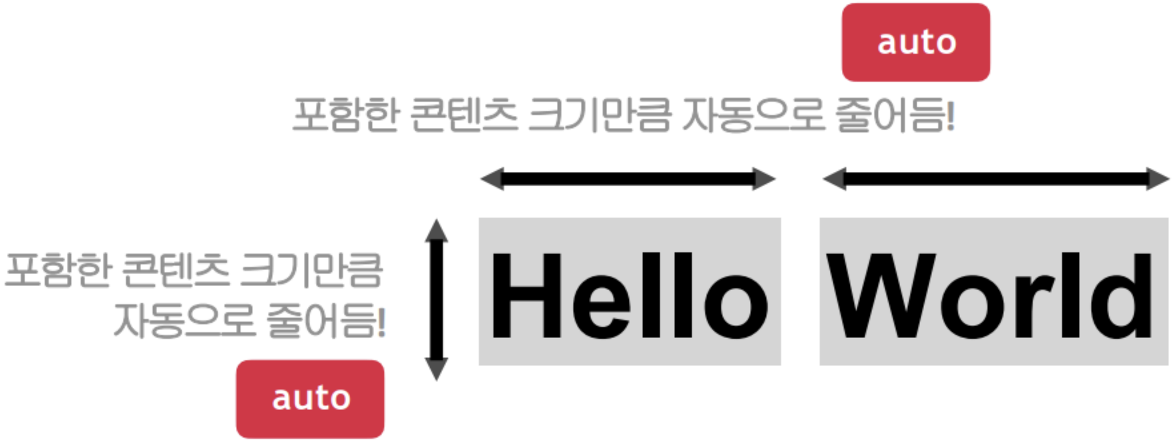
포함한 콘텐츠 크기만큼
자동으로 줄어듬!

auto

World

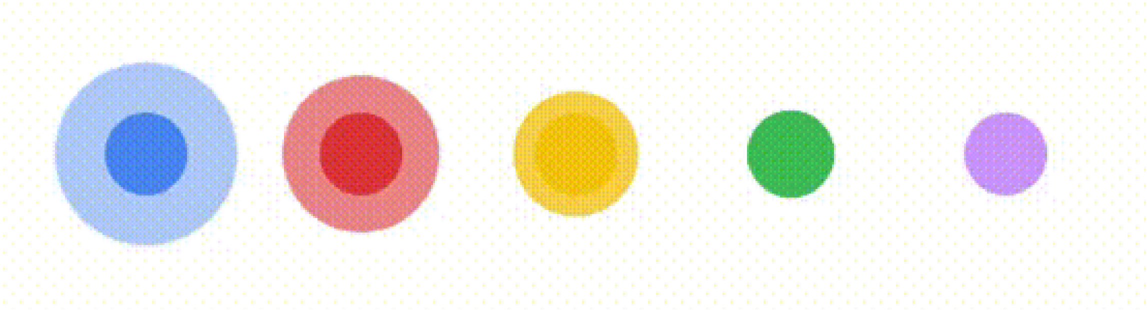
Inline Element

- Opposite of the block element, with no line breaks
- tag : <a>, <img>, <strong>, <span>, <em>, <br>, <input>



Display Property

- If the display property is set to block, it is arranged like a block element
- If you set the display property to inline, it will be laid out like an inline element
- How to use the display property
  - display: block
  - display: inline
  - display: flex
  - display: none
    - Not considered to be missing



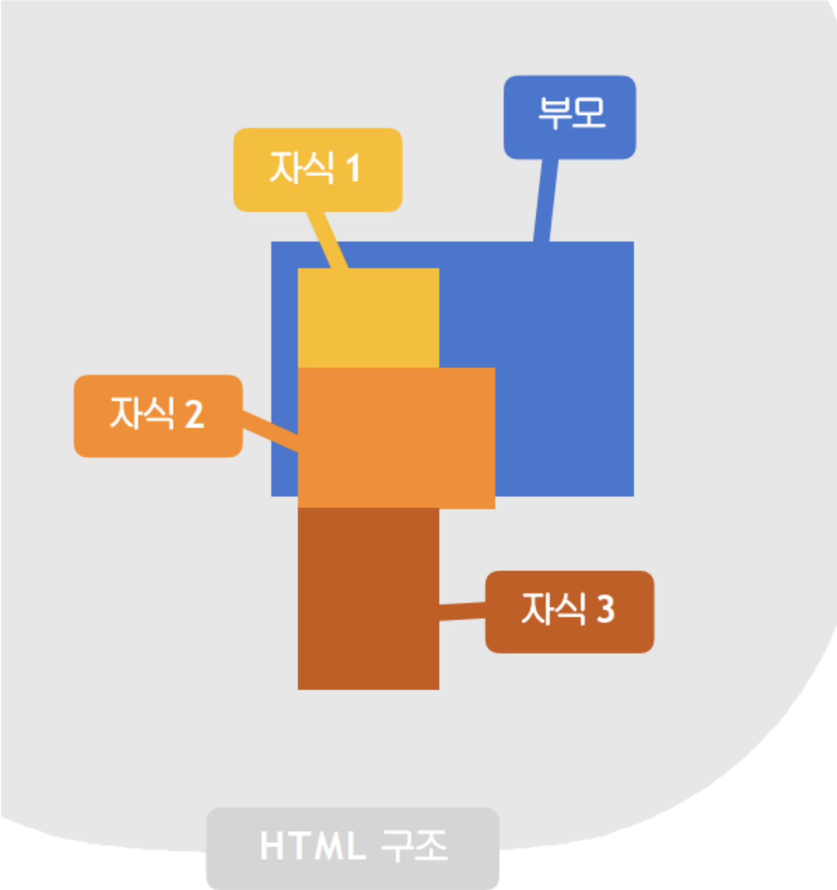
Element Positioning

Position Property

- Static positioning
  - HTML elements are positioned static by default
- Relative positioning
  - An element with position: relative; is positioned relative to its normal position
- Absolute positioning
  - An element with position: absolute; is positioned relative to the nearest positioned ancestor
- Fixed positioning
  - An element with position: fixed; is positioned relative to the viewport

Static Positioning

- Block elements are stacked up and down like a box, and inline elements are arranged in a row
- This property is always placed according to the basic flow of the page
  - This property is not affected by the top, bottom, left and right properties



Static Positioning

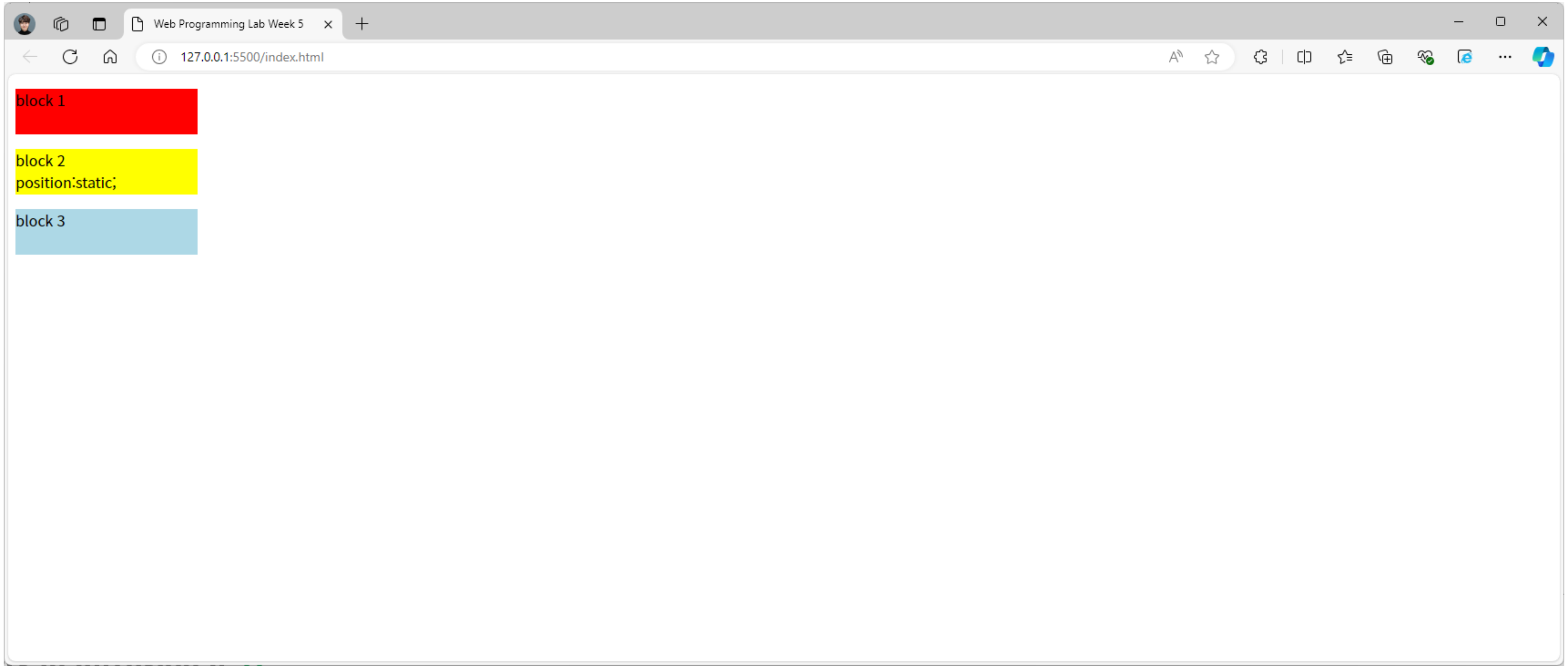
<index.html>

```
<head>
    <link rel="stylesheet"
href="style.css">
</head>
<body>
    <p id="one">block 1</p>
    <div id="two">block 2<br>
position:static;</div>
    <p id="three">block 3</p>
</body>
```

<style.css>

```
#one {
    background-color: red;
    width: 200px;
    height: 50px;
}
#two {
    background-color: yellow;
    position: static;
    width: 200px;
    height : 50px;
}
#three {
    background-color: lightblue;
    width: 200px;
    height: 50px;
}
```

Static Positioning



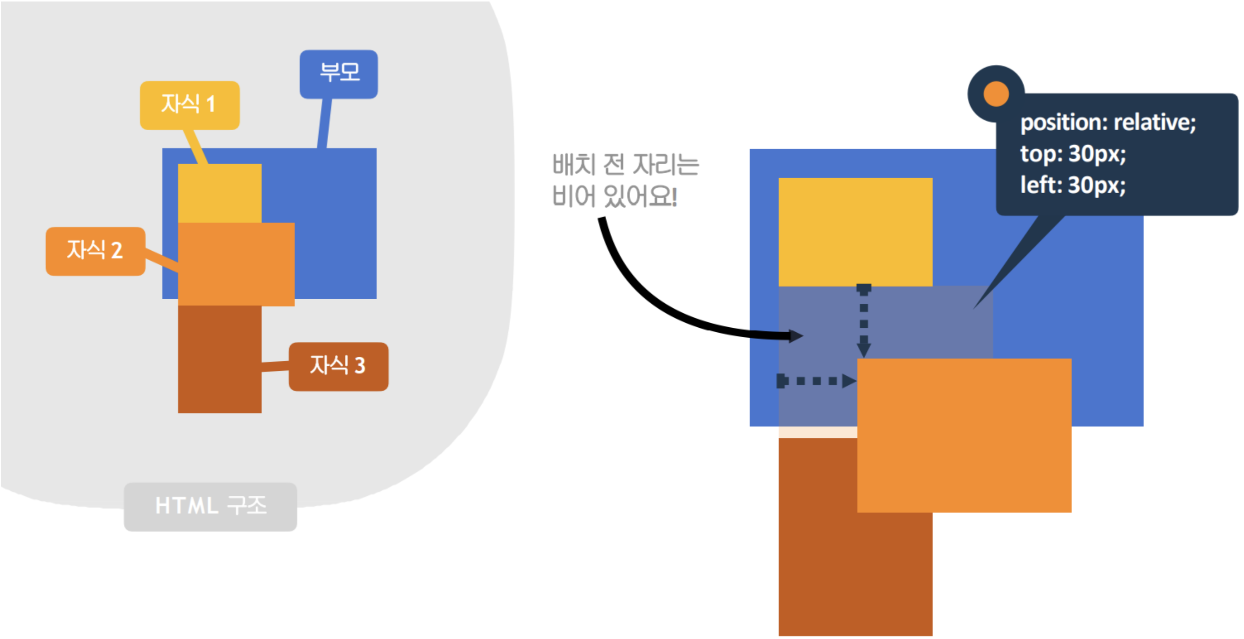
block 1

block 2
position:static;

block 3

Relative Positioning

- Setting a relatively positioned element will cause it to be adjusted away from its normal position



```
position: relative;
top: 30px;
left: 30px;
```

Relative Positioning

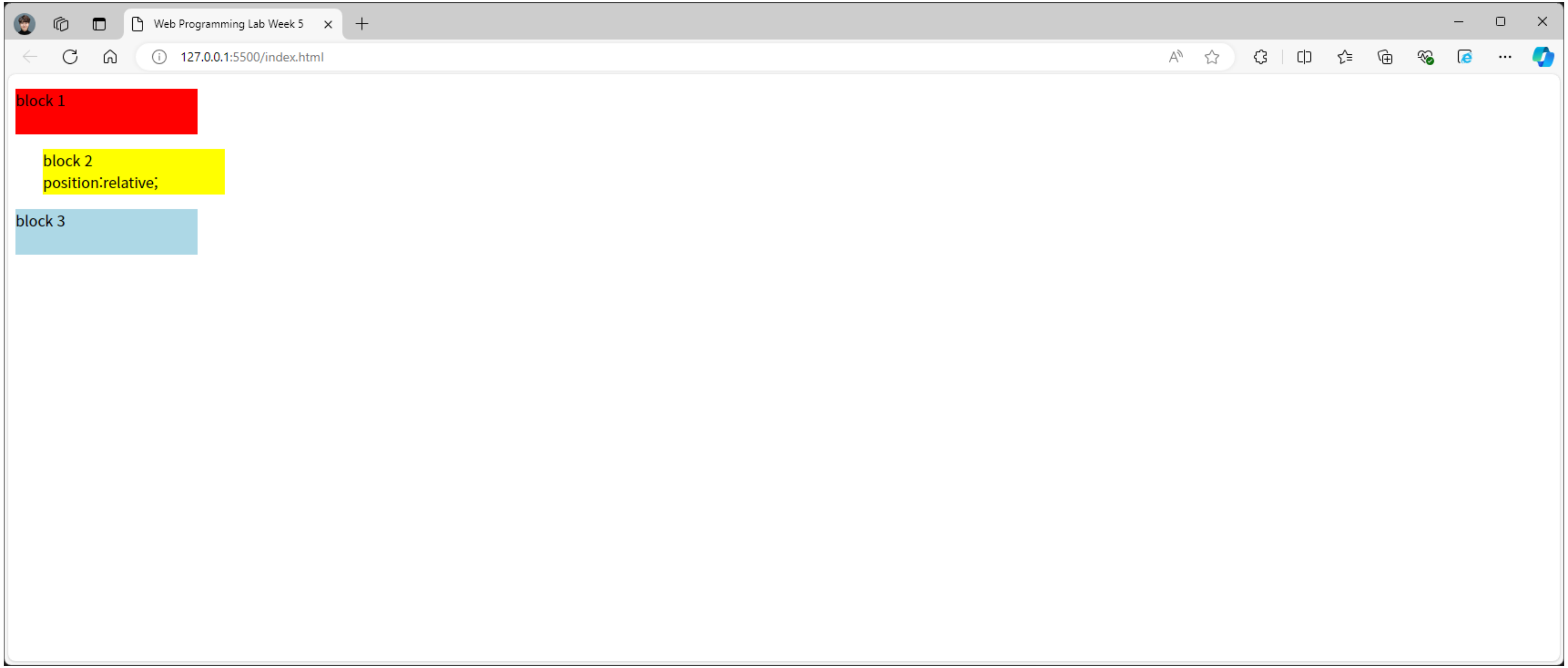
<index.html>

```
<head>
    <link rel="stylesheet"
href="style.css">
</head>
<body>
    <p id="one">block 1</p>
    <div id="two">block 2<br>
position:relative;</div>
    <p id="three">block 3</p>
</body>
```

<style.css>

```
#one {
    background-color: red;
    width: 200px;
    height: 50px;
}
#two {
    background-color: yellow;
    position: relative;
    left : 30px;
    width: 200px;
    height : 50px;
}
#three {
    background-color: lightblue;
    width: 200px;
    height: 50px;
}
```

Relative Positioning



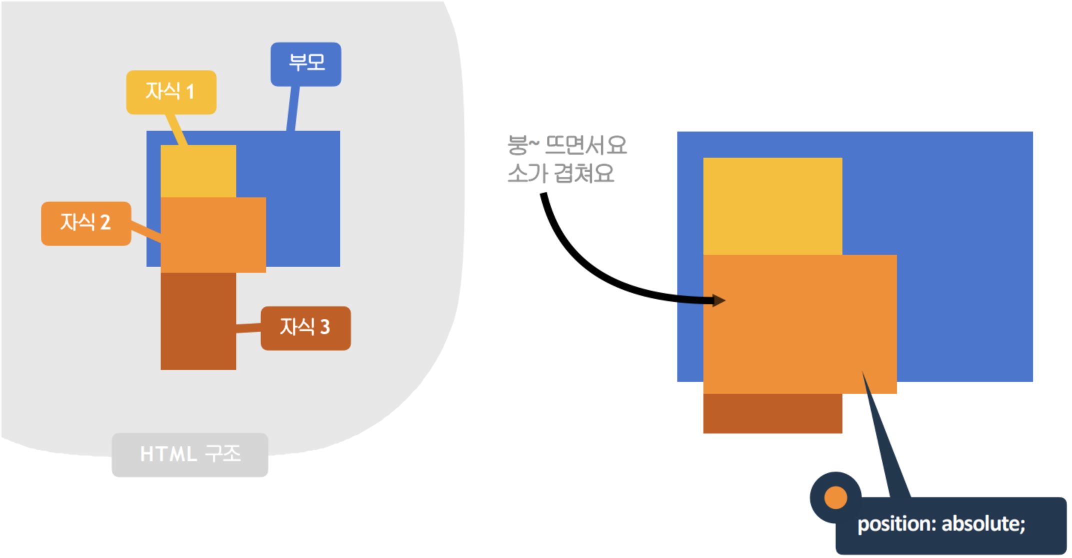
block 1

block 2
position:relative;

block 3

Absolute Positioning

- Located at the top, left, bottom, right in the starting position relative to the entire page
- It is positioned relative to the parent element that has a non-static position



Absolute Positioning

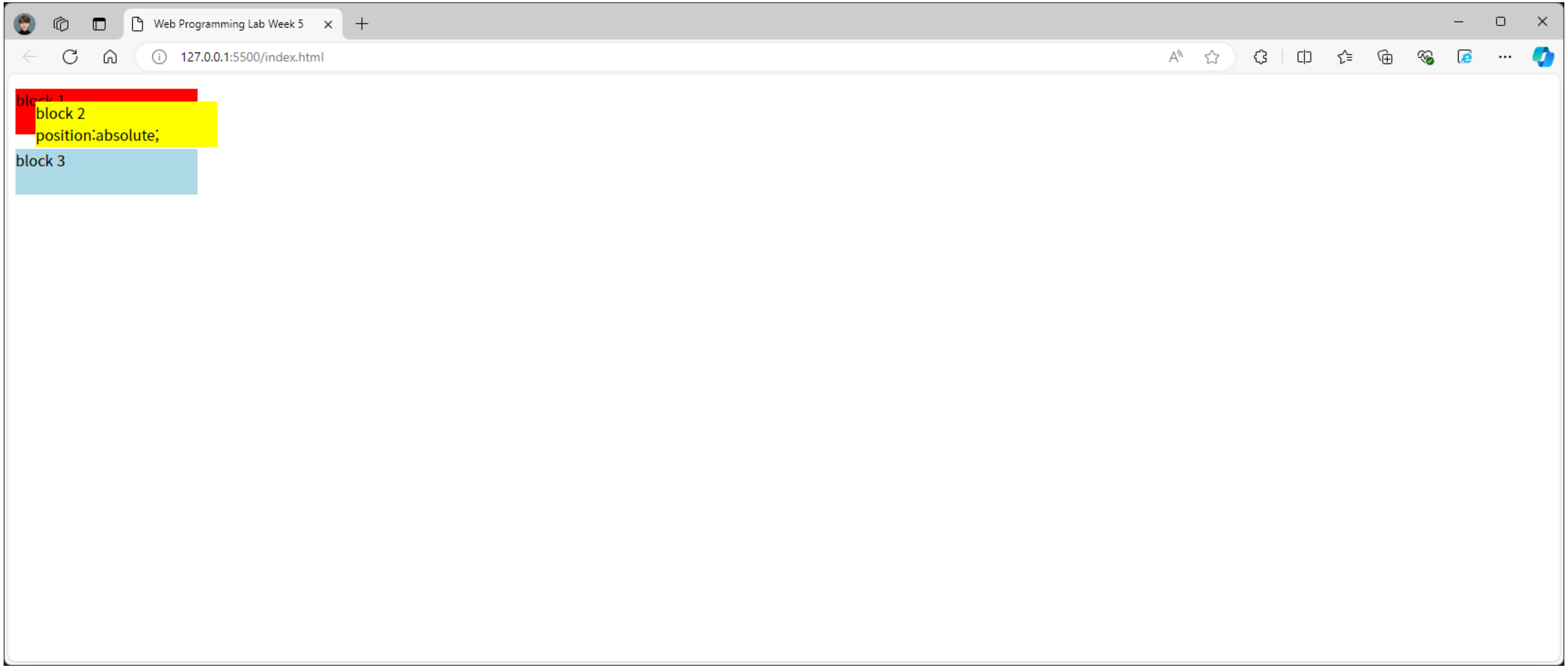
<index.html>

```
<head>
    <link rel="stylesheet"
href="style.css">
</head>
<body>
    <p id="one">block 1</p>
    <div id="two">block 2<br>
position:absolute;</div>
    <p id="three">block 3</p>
</body>
```

<style.css>

```
#one {
    background-color: red;
    width: 200px;
    height: 50px;
}
#two {
    background-color: yellow;
    position: absolute;
    top : 30px;
    left : 30px;
    width: 200px;
    height : 50px;
}
#three {
    background-color: lightblue;
    width: 200px;
    height: 50px;
}
```

Absolute Positioning



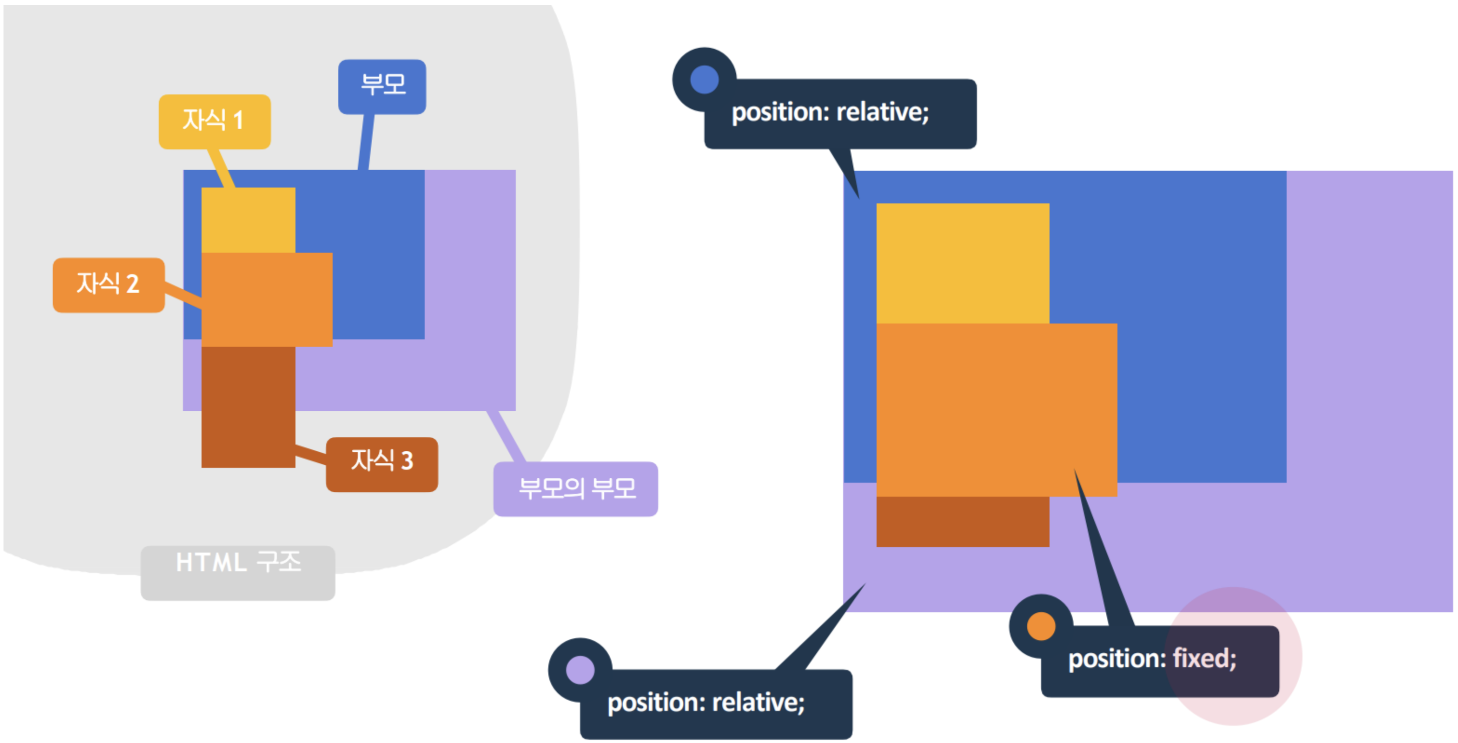
block 1

block 2
position:absolute;

block 3

Fixed Positioning

- Positioning elements relative to the browser window
- Window does not move when scrolling



Fixed Positioning

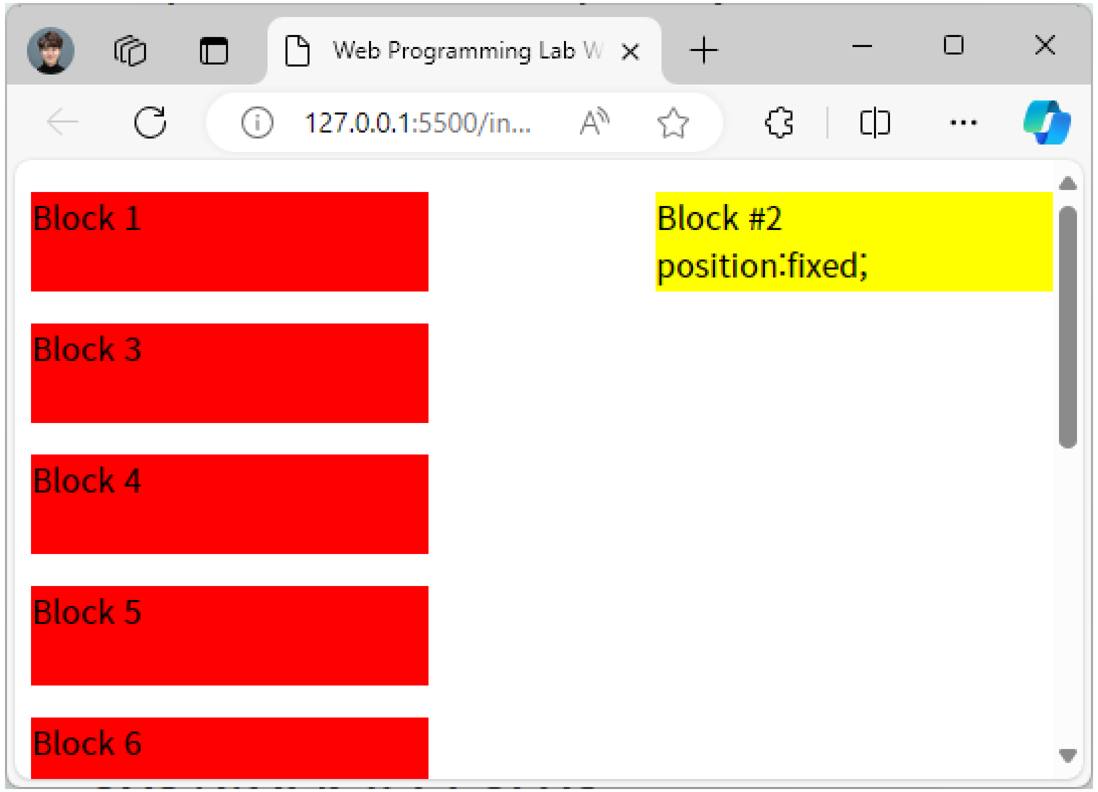
<index.html>

```
<body>
    <p>Block 1</p>
    <p id="two">Block #2<br>position:fixed;</p>
    <p>Block 3</p>
    <p>Block 4</p>
    <p>Block 5</p>
    <p>Block 6</p>
    <p>Block 7</p>
    <p>Block 8</p>
    <p>Block 9</p>
    <p>Block 10</p>
    <p>Block 11</p>
</body>
```

<style.css>

```
p {
    background-color: red;
    width: 200px;
    height: 50px;
}
#two {
    background-color: yellow;
    position: fixed;
    top : 0px;
    right : 0px;
    width: 200px;
    height : 50px;
}
```

Fixed Positioning



Block 1

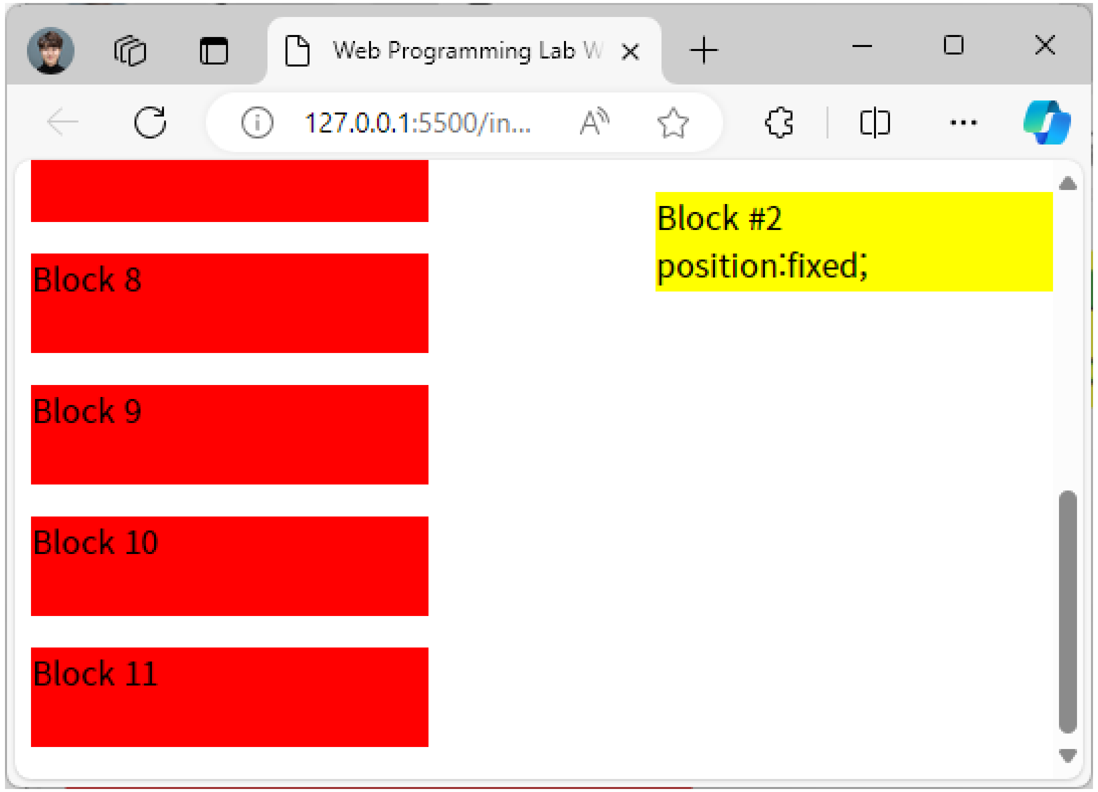
Block #2
position:fixed;

Block 3

Block 4

Block 5

Block 6



Block #2
position:fixed;

Block 8

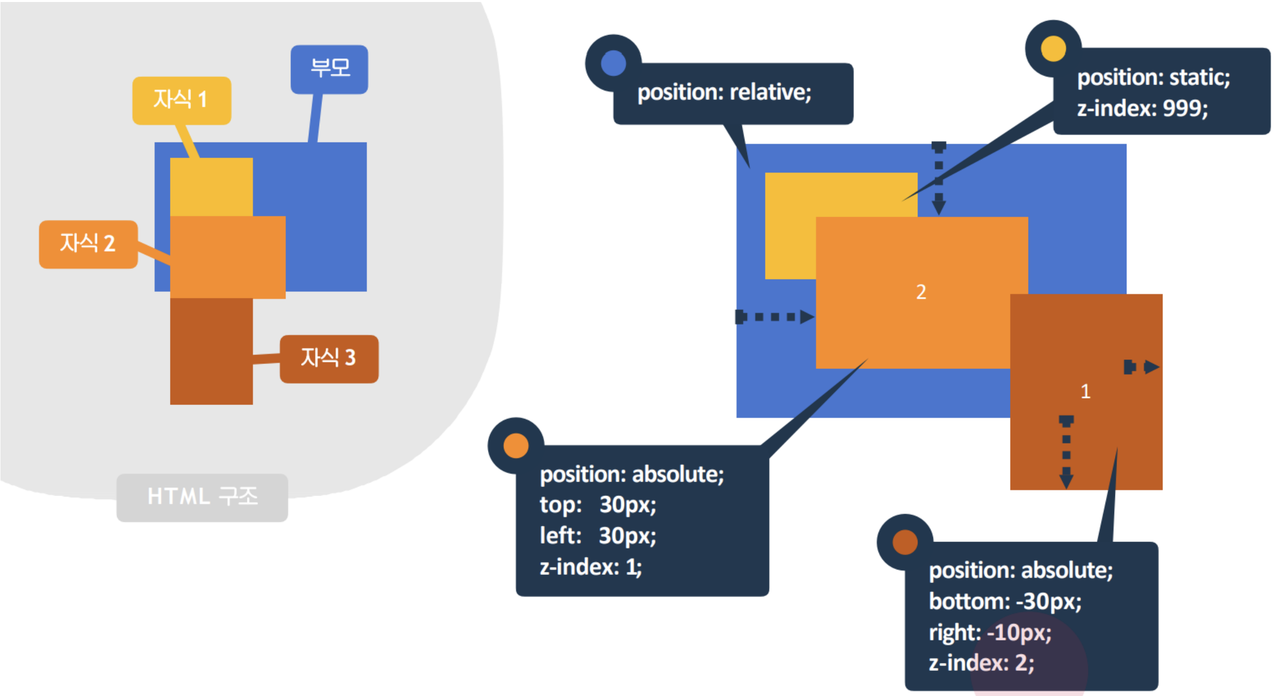
Block 9

Block 10

Block 11

z-index

- Determines the order in which objects appear before and after
- Only works on elements with the position attribute applied



```
position: relative;
```

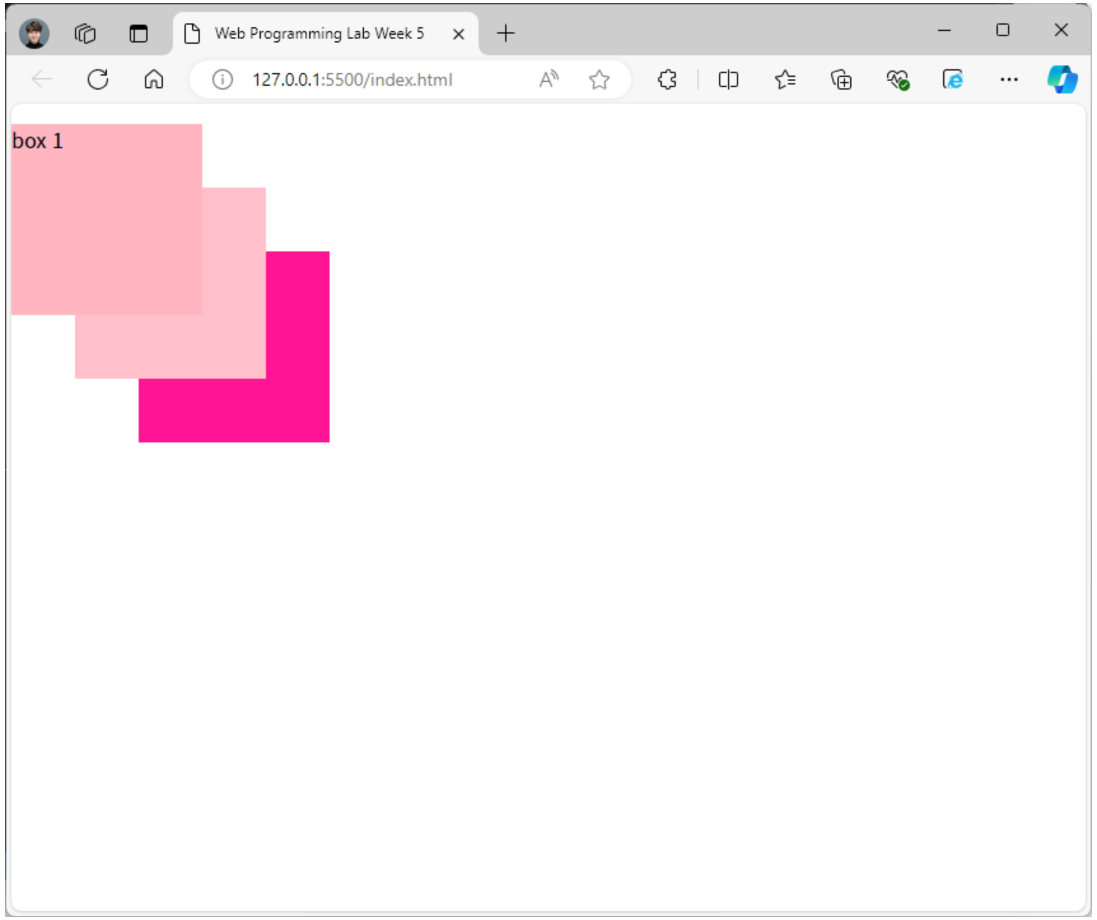
```
position: static;
z-index: 999;
```

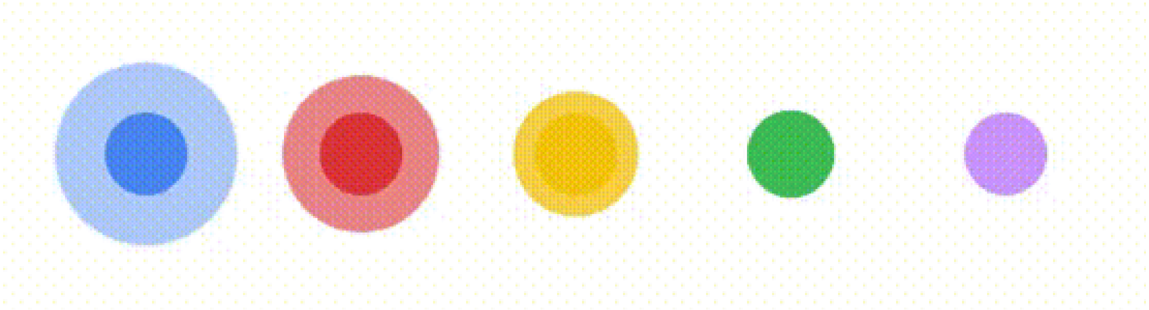
```
position: absolute;
top:  30px;
left:  30px;
z-index: 1;
```

```
position: absolute;
bottom: -30px;
right: -10px;
z-index: 2;
```

z-index

```
#box1 {
            position: absolute; top: 0px; left: 0px;
            width : 150px; height : 150px;
            background : lightpink;
            z-index: 200;
}
#box2 {
            position: absolute; top: 50px; left: 50px;
            width : 150px; height : 150px;
            background : pink;
            z-index: 100;
}
#box3 {
            position: absolute; top: 100px; left: 100px;
            width : 150px; height : 150px;
            background : deeppink;
            z-index: 0;
}
```

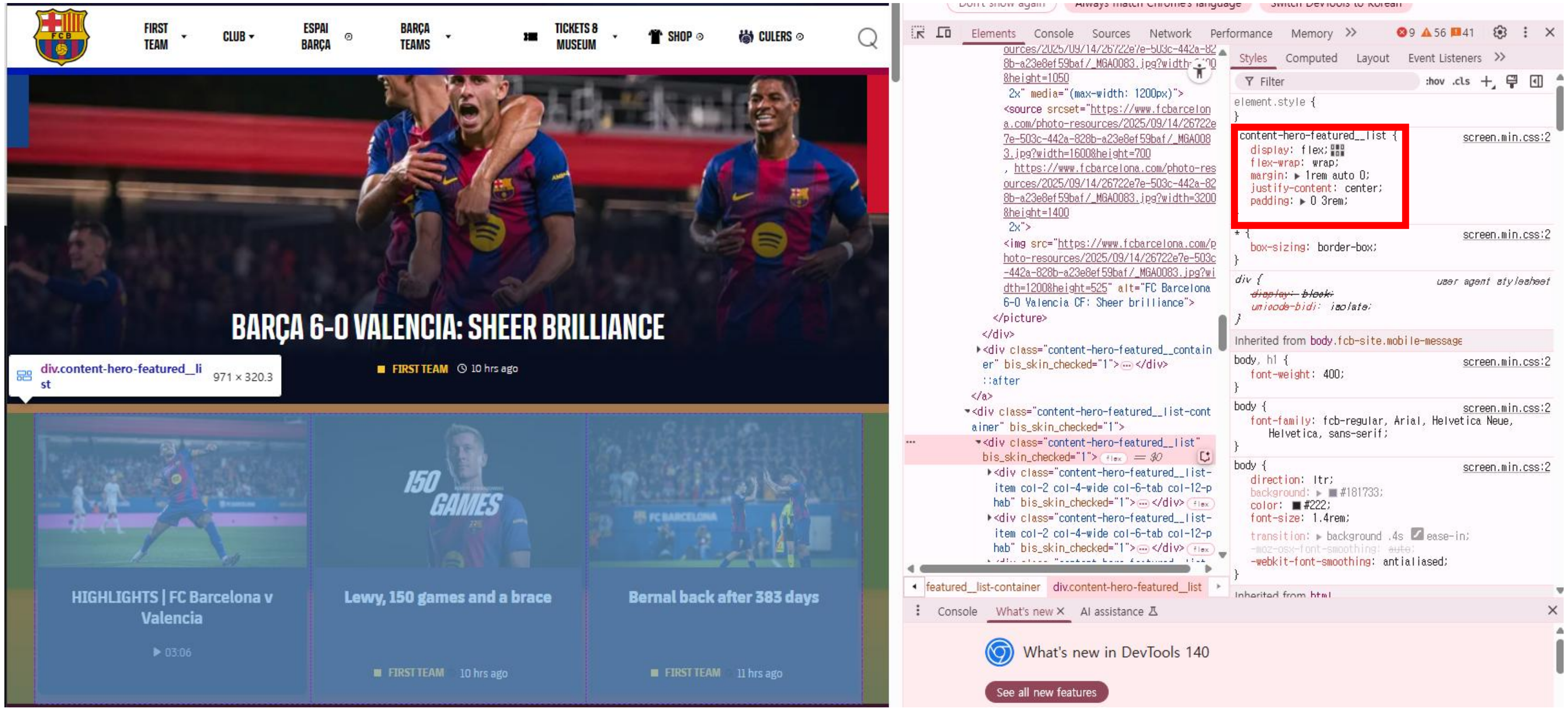




CSS Flexible Box Layout

CSS Flexible Box Layout Module

- Offers powerful features for creating flexible responsive layouts

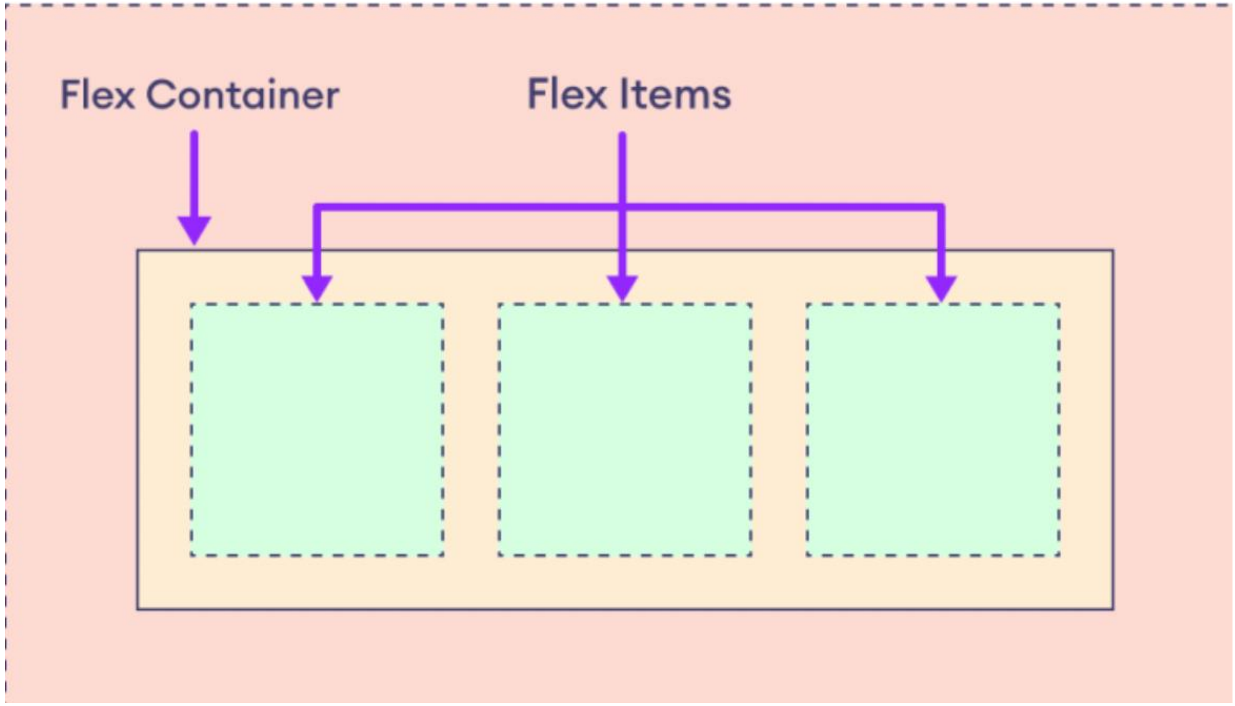


```
ources/2025/09/14/26722e7e-503c-442a-82
8b-a23e8ef59baf/_MGA0083.jpg?width=2400
&height=1050
 2x" media="(max-width: 1200px)">
<source srcset="https://www.fcbarcelon
a.com/photo-resources/2025/09/14/26722e
7e-503c-442a-828b-a23e8ef59baf/_MGA008
3.jpg?width=1600&height=700
, https://www.fcbarcelona.com/photo-res
ources/2025/09/14/26722e7e-503c-442a-82
8b-a23e8ef59baf/_MGA0083.jpg?width=3200
&height=1400
 2x">
<img src="https://www.fcbarcelona.com/p
hoto-resources/2025/09/14/26722e7e-503c
-442a-828b-a23e8ef59baf/_MGA0083.jpg?wi
dth=1200&height=525" alt="FC Barcelona
6-0 Valencia CF: Sheer brilliance">
</picture>
</div>
<div class="content-hero-featured__contain
er" bis_skin_checked="1"> ⋯ </div>
::after
</a>
<div class="content-hero-featured__list-cont
ainer" bis_skin_checked="1">
  <div class="content-hero-featured__list"
  bis_skin_checked="1"> flex == $0
    <div class="content-hero-featured__list-
    item col-2 col-4-wide col-6-tab col-12-p
    hab" bis_skin_checked="1"> ⋯ </div> flex
    <div class="content-hero-featured__list-
    item col-2 col-4-wide col-6-tab col-12-p
    hab" bis_skin_checked="1"> ⋯ </div> flex
```

```
element.style {
}
content-hero-featured__list {                         screen.min.css:2
  display: flex;
  flex-wrap: wrap;
  margin: ▸ 1rem auto 0;
  justify-content: center;
  padding: ▸ 0 3rem;
}
* {                                                   screen.min.css:2
  box-sizing: border-box;
}
div {                                            user agent stylesheet
  ~~display: block;~~
  unicode-bidi: isolate;
}
Inherited from body.fcb-site.mobile-message
body, h1 {                                            screen.min.css:2
  font-weight: 400;
}
body {                                                screen.min.css:2
  font-family: fcb-regular, Arial, Helvetica Neue,
     Helvetica, sans-serif;
}
body {                                                screen.min.css:2
  direction: ltr;
  background: ▸ #181733;
  color: #222;
  font-size: 1.4rem;
  transition: ▸ background .4s ease-in;
  -moz-osx-font-smoothing: auto;
  -webkit-font-smoothing: antialiased;
}
Inherited from html
```

Flexbox Containers and Flexbox Items

- The elements within a flex container are referred to as flex items
- Any element can be set as a flex container using display: flex declaration



Defining a Flexbox Container

- Any element can be set as a flex container using display: flex declaration
- The target HTML element becomes a flexbox container

<index.html>

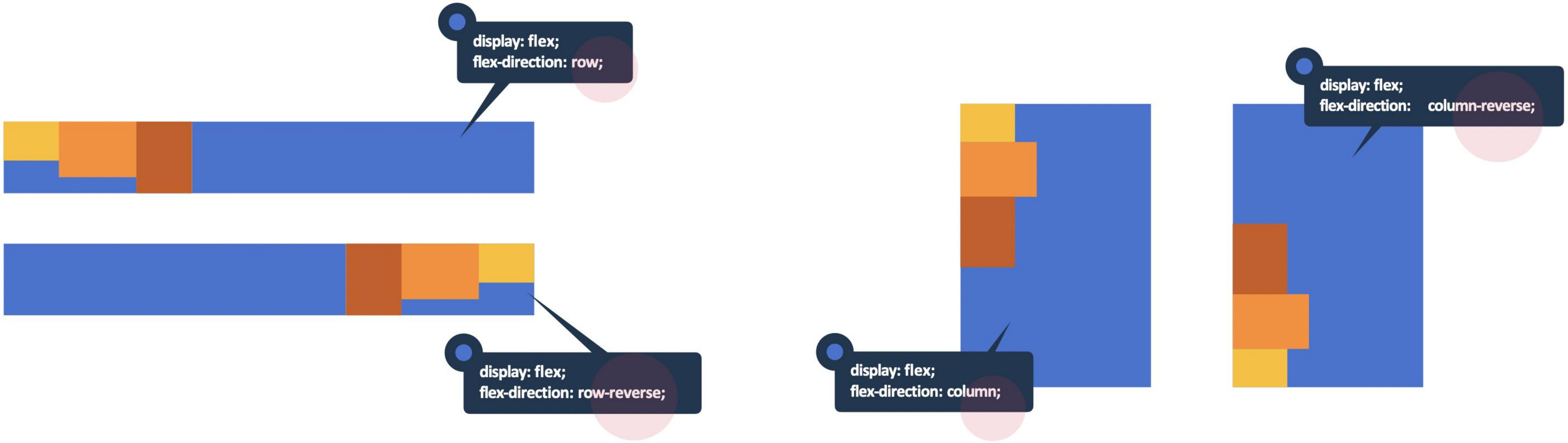
```
<div class=“container”>
    …
</div>
```

<style.css>

```
.container {
    display : flex;
}
```

Setting Flexbox Container Layout

- flex-direction defines the layout orientation of flexbox items
- flex-direction : row / column / row-reverse / column-reverse



```
display: flex;
flex-direction: row;
```

```
display: flex;
flex-direction: row-reverse;
```

```
display: flex;
flex-direction: column;
```

```
display: flex;
flex-direction:    column-reverse;
```

Flexbox Example

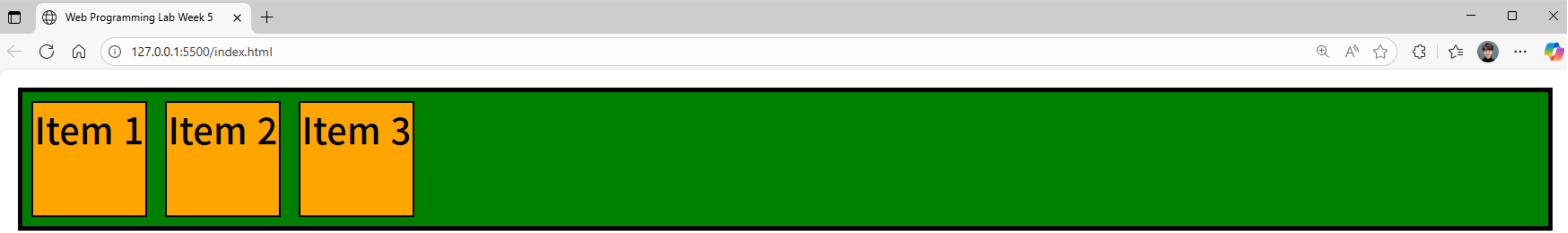
<index.html>

```
<body>
    <div class="container">
        <p id="one">Item 1</p>
        <p id="two">Item 2</p>
        <p id="three">Item 3</p>
    </div>
</body>
```

<style.css>

```
.container {
    display : flex;
    border : 2px solid black;
    background-color : green;
}
p {
    width : 50px;
    height : 50px;
    text-align : center;
    border : 1px solid black;
    background-color : orange;
    margin : 4px;
}
```

Flexbox Example



Item 1 Item 2 Item 3

flex-direction

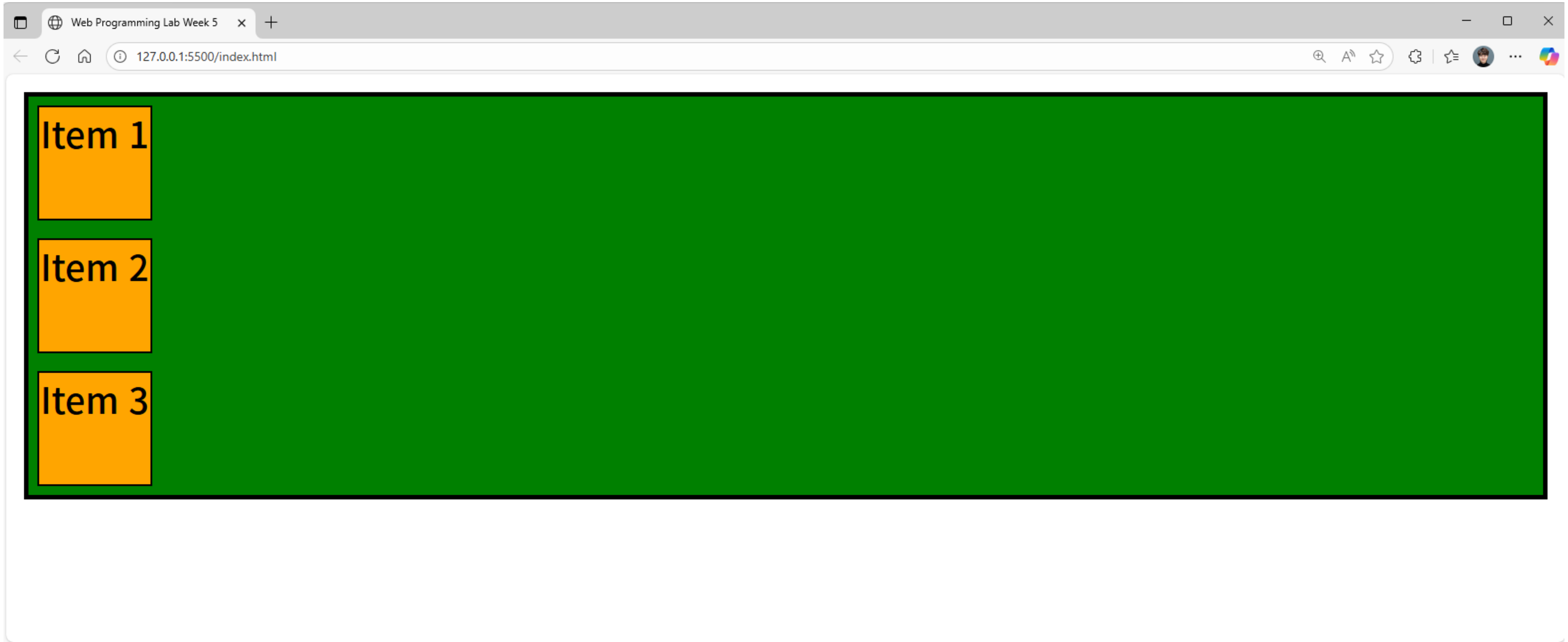
<index.html>

```html
<body>
    <div class="container">
        <p id="one">Item 1</p>
        <p id="two">Item 2</p>
        <p id="three">Item 3</p>
    </div>
</body>
```

<style.css>

```css
.container {
    display : flex;
    flex-direction : column;
    border : 2px solid black;
    background-color : green;
}
p {
    width : 50px;
    height : 50px;
    text-align : center;
    border : 1px solid black;
    background-color : orange;
    margin : 4px;
}
```

flex-direction



flex-wrap / align-content

<index.html>

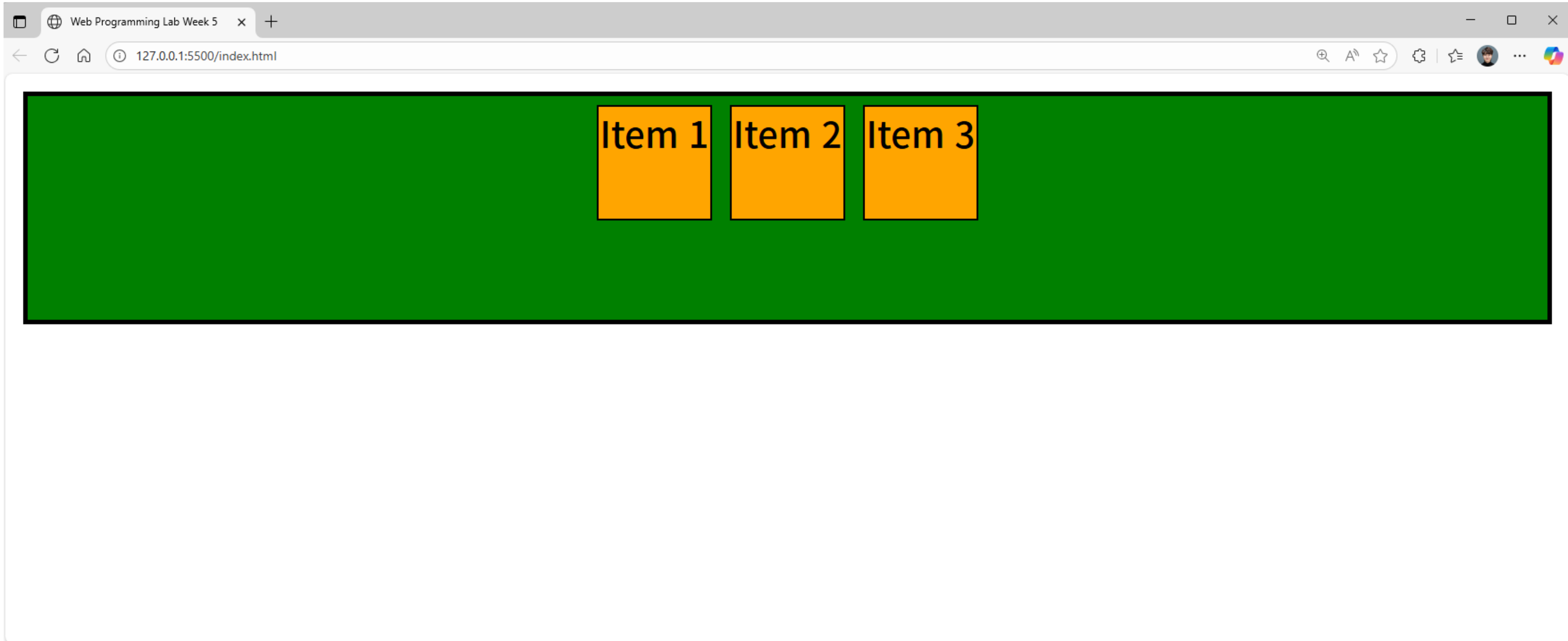
```
<body>
    <div class="container">
        <p id="one">Item 1</p>
        <p id="two">Item 2</p>
        <p id="three">Item 3</p>
    </div>
</body>
```

<style.css>

```
.container {
    display : flex;
    height : 100px;
    flex-direction : column;
    flex-wrap : wrap; align-content: center;
    border : 2px solid black;
    background-color : green;
}

p {
    width : 50px;
    height : 50px;
    text-align : center;
    border : 1px solid black;
    background-color : orange;
    margin : 4px;
}
```

flex-wrap / align-content



align-items

<index.html>

```html
<body>
    <div class="container">
        <p id="one">Item 1</p>
        <p id="two">Item 2</p>
        <p id="three">Item 3</p>
    </div>
</body>
```

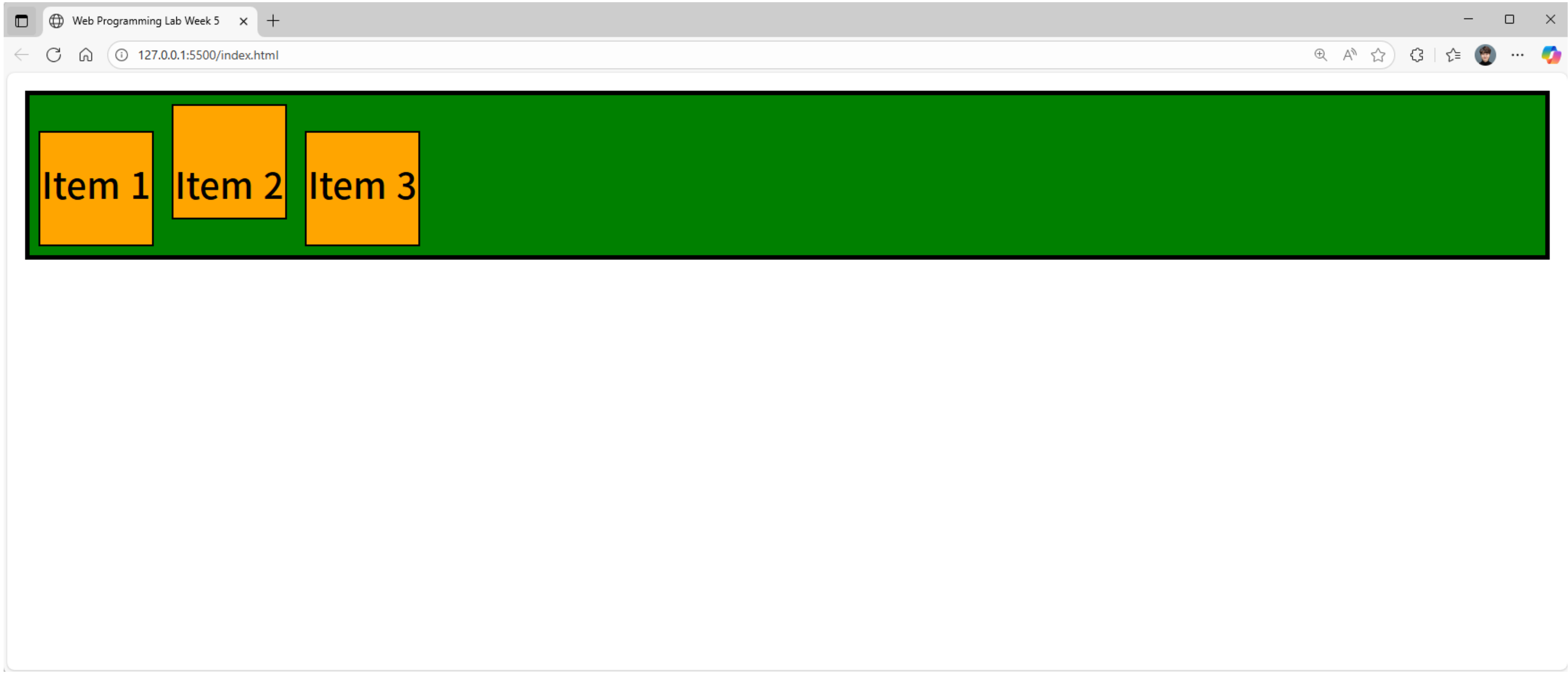
<style.css>

```css
.container {
    display : flex;
    align-items: baseline;
    border : 2px solid black;
    background-color : green;
}

p {
    width : 50px;
    height : 50px;
    text-align : center;
    border : 1px solid black;
    background-color : orange;
    margin : 4px;
    line-height : 46px;
}

#two {
    line-height : 70px;
}
```

align-items



Web Programming Lab Week 5
127.0.0.1:5500/index.html
Item 1
Item 2
Item 3

flex-grow

<index.html>

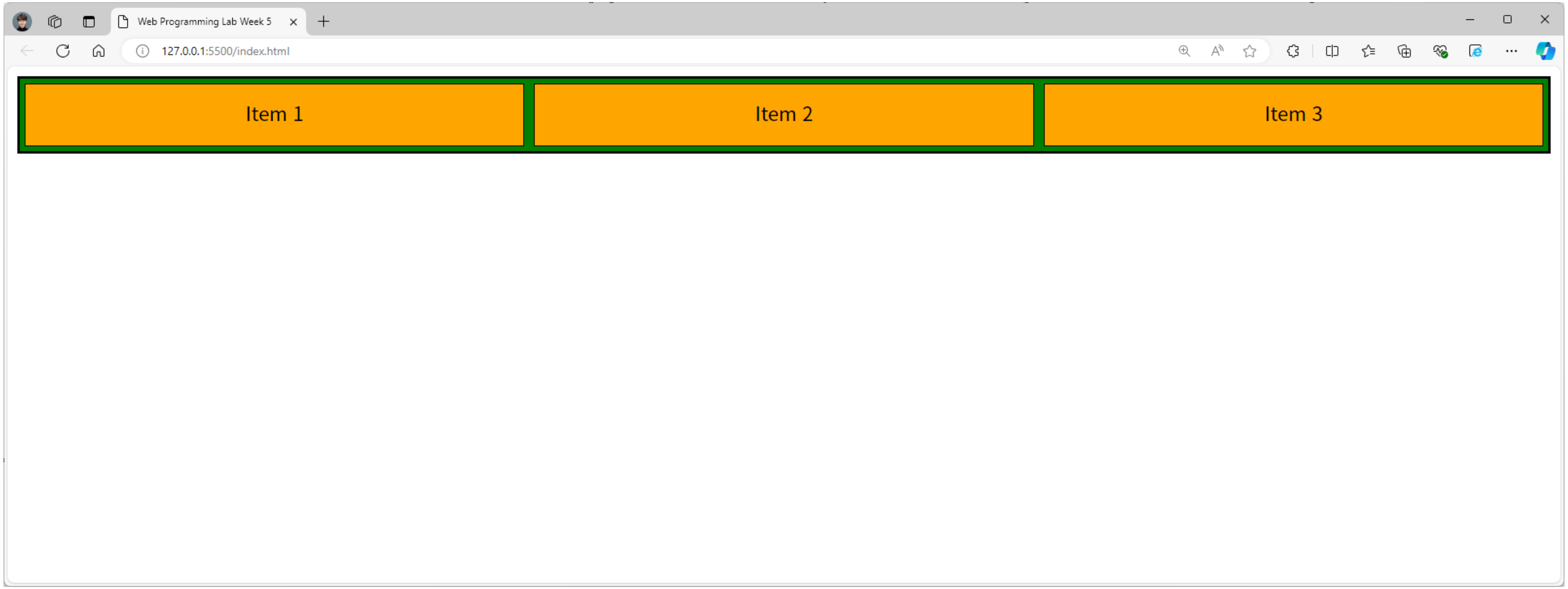
```
<body>
    <div class="container">
        <p id="one">Item 1</p>
        <p id="two">Item 2</p>
        <p id="three">Item 3</p>
    </div>
</body>
```

<style.css>

```
.container {
    display : flex;
    border : 2px solid black;
    background-color : green;
}

p {
    width : 50px;
    height : 50px;
    text-align : center;
    border : 1px solid black;
    background-color : orange;
    margin : 4px;
    flex-grow : 1;
}
```

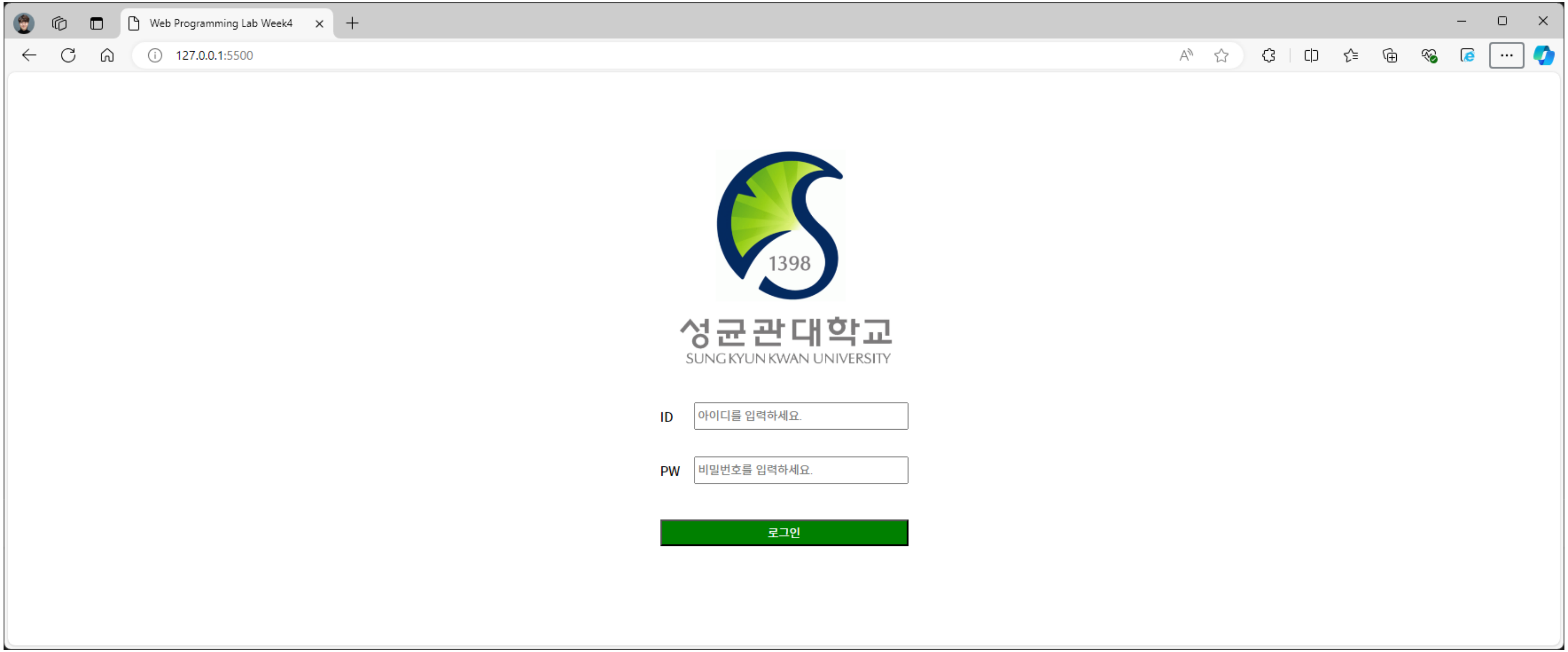
flex-grow


Web Programming Lab Week 5
127.0.0.1:5500/index.html
Item 1
Item 2
Item 3

Modifying Week2 Login Page

- We will not contain HTML & CSS code in this lab material
- Let's implement together

Result



Web Programming Lab Week4
127.0.0.1:5500
1398
성균관대학교
SUNG KYUN KWAN UNIVERSITY
ID
아이디를 입력하세요.
PW
비밀번호를 입력하세요.
로그인