

# CSS

Lab. 02



부산대학교  
PUSAN NATIONAL UNIVERSITY



Thanks  
to

# CSS

---

- CSS
  - Cascading Style Sheet 의 약자
  - HTML의 Style을 정의하는 문법
- CSS 스타일 적용 순서
  - 1. Author-created style sheets (.css 파일 또는 html 파일 내에 사용자가 직접 정의한 스타일)
  - 2. User style sheets (사용자가 정의한 CSS 스타일을 브라우저에 적용한 것)
  - 3. Browser style sheets (브라우저의 스타일 정의, 기본값)



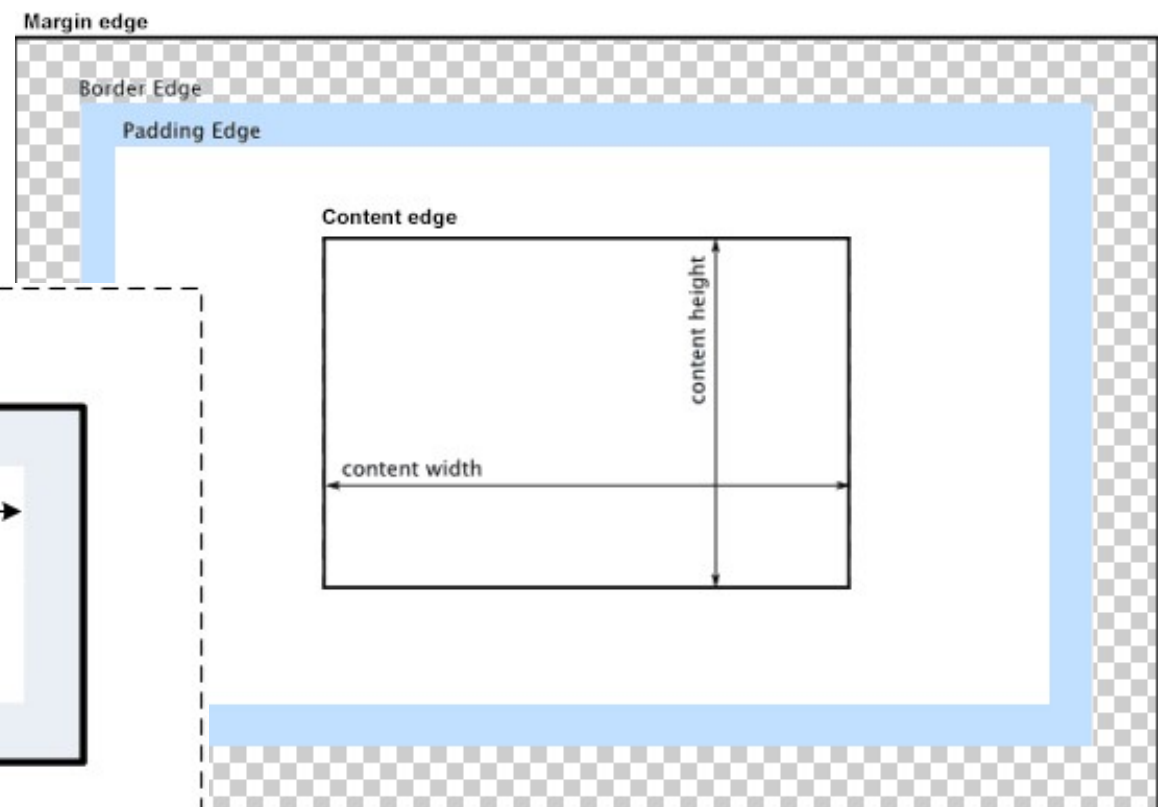
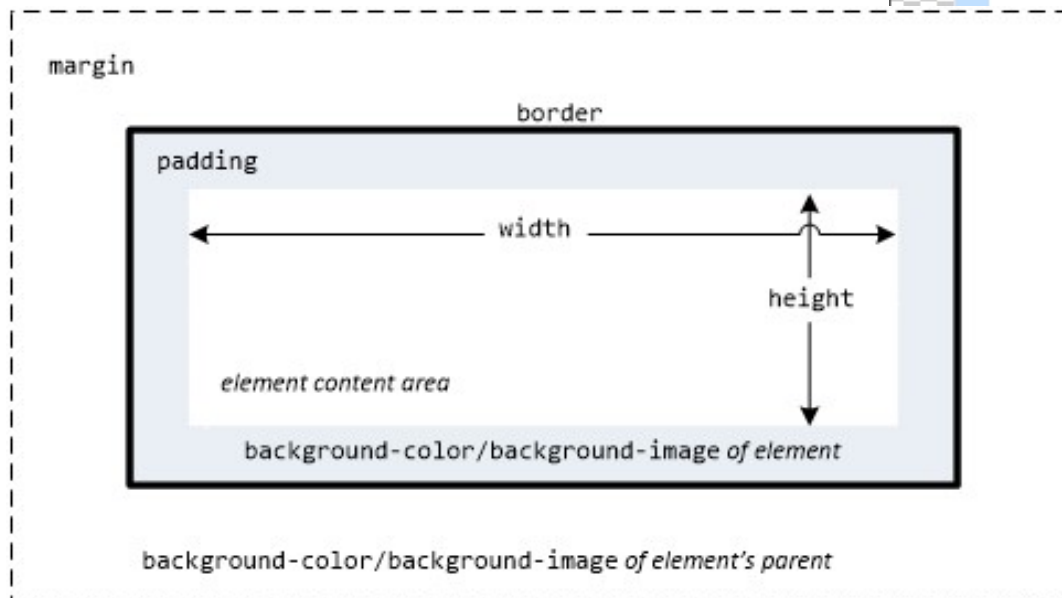
# Selector

- Selector
  - selectors are used to target the HTML elements on our web pages that we want to style (상세설명: [link](#))

Selector	Example	Learn CSS tutorial
<a href="#">Type selector</a>	<code>h1 { }</code>	<a href="#">Type selectors</a>
<a href="#">Universal selector</a>	<code>* { }</code>	<a href="#">The universal selector</a>
<a href="#">Class selector</a>	<code>.box { }</code>	<a href="#">Class selectors</a>
<a href="#">id selector</a>	<code>#unique { }</code>	<a href="#">ID selectors</a>
<a href="#">Attribute selector</a>	<code>a[title] { }</code>	<a href="#">Attribute selectors</a>
<a href="#">Pseudo-class selectors</a>	<code>p:first-child { }</code>	<a href="#">Pseudo-classes</a>
<a href="#">Pseudo-element selectors</a>	<code>p::first-line { }</code>	<a href="#">Pseudo-elements</a>
<a href="#">Descendant combinator</a>	<code>article p</code>	<a href="#">Descendant combinator</a>
<a href="#">Child combinator</a>	<code>article &gt; p</code>	<a href="#">Child combinator</a>
<a href="#">Adjacent sibling combinator</a>	<code>h1 + p</code>	<a href="#">Adjacent sibling</a>
<a href="#">General sibling combinator</a>	<code>h1 ~ p</code>	<a href="#">General sibling</a>

# Box Model

- Box Model
  - CSS는 HTML element를 BOX로 취급함
    - Content area
    - Padding area
    - Border area
    - Margin area
  - [설명 link](#)



# position

- position
  - The **position** property sets how an element is positioned in a document. [Link](#)
  - 문서내에서 HTML element의 위치를 설정하는 CSS 문법

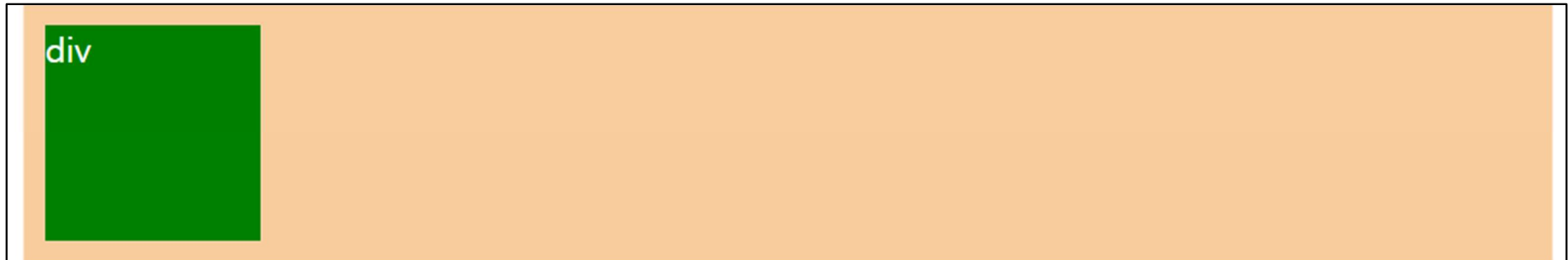
## Syntax

```
position: static;  
position: relative;  
position: absolute;  
position: fixed;  
position: sticky;  
  
/* Global values */  
position: inherit;  
position: initial;  
position: revert;  
position: unset;
```

source: MDN Web Docs

# Layouts in HTML

- `<div>` is block-level layout (한 라인 전체를 차지함)



- `<span>` is inline-level layout (한 라인 내의 일부로 추가됨)



the wheat-colored area is margin of box model

# display

- CSS can change the layout of elements. [link](#)

## Syntax

The CSS `display` property is specified using keyword values.

```
/* legacy values */
display: block;
display: inline;
display: inline-block;
display: flex;
display: inline-flex;
display: grid;
display: inline-grid;
display: flow-root;

/* box generation */
display: none;
display: contents;

/* two-value syntax */
display: block flow;
display: inline flow;
display: inline flow-root;
display: block flex;
display: inline flex;
display: block grid;
display: inline grid;
display: block flow-root;

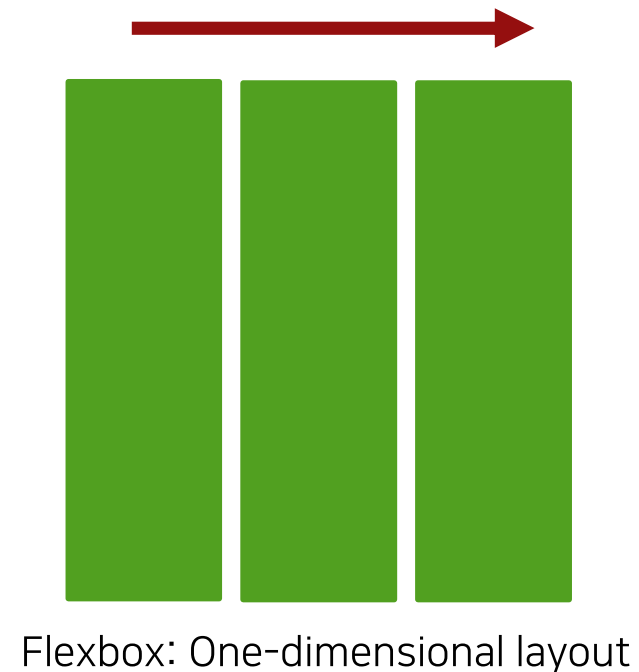
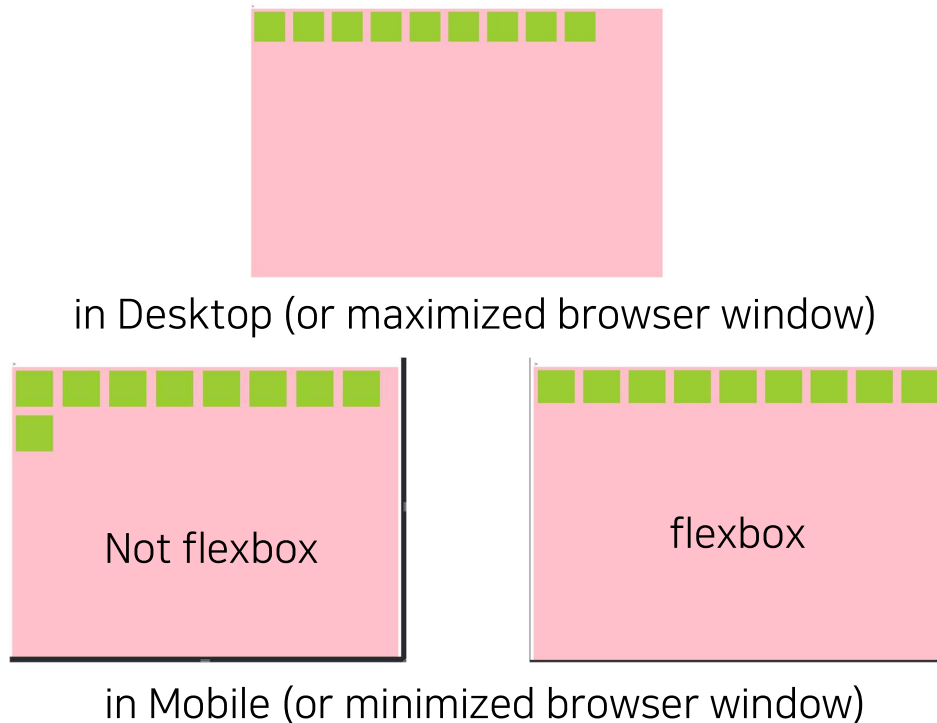
/* other values */
display: table;
display: table-row; /* all table elements have an equivalent CSS display value */
display: list-item;

/* Global values */
display: inherit;
display: initial;
display: revert;
display: unset;
```

source: MDN Web Docs

# display: Flexbox

- flex (flexbox)
  - Flexbox is designed as one-dimensional layout model. [link](#)

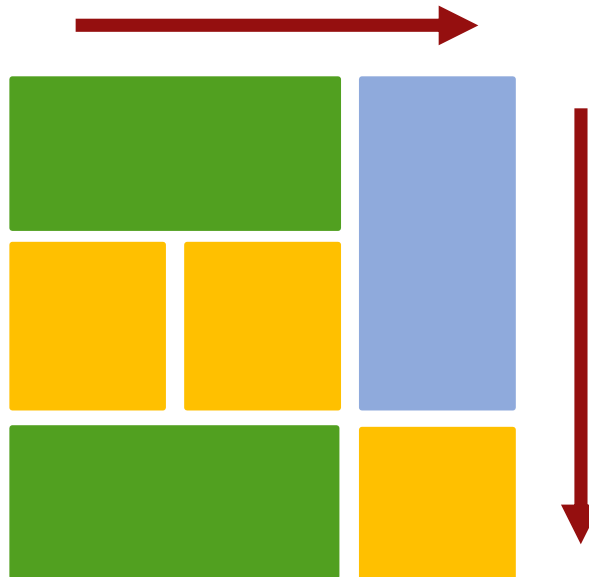




# display: Grid

---

- Grid (CSS Grid)
  - Grid is a CSS layout method developed for two-dimensional layouts.



CSS Grid: Two-dimensional layout

# Examples

- 배포된 “배포 - Examples.zip” 파일 내의 예제 파일을 통해 CSS 스타일 기본 사용법 익히기

 Box-model.html

 clear.html

 Display.html

 float-left.html

 float-menu.html

 layout-with-float-and-clear.html

 overflow.html

 Position.html

 Position-sticky.html

 Visibility.html

 z-index.html

