

10. CSS display / position

배치의 비밀.



“

DISPLAY



display

inline | **block** | list-item | **inline-block** | table |
inline-table | table-row-group | table-header-group |
table-footer-group | table-row | table-column-group
| table-column | table-cell | table-caption |
none | inherit

display

Lv3에서 새로 추가된 값들:

run-in | flow | **flow-root** | **flex** | **grid** | ruby | ruby-base
| ruby-text | ruby-base-container | ruby-text-
container | **contents** | inline-table | inline-flex |
inline-grid

“

CHANGED DISPLAY



CHANGED DISPLAY

position: **absolute** | **fixed**

float: **left** | **right**

↓ changed display

display: **block**

“

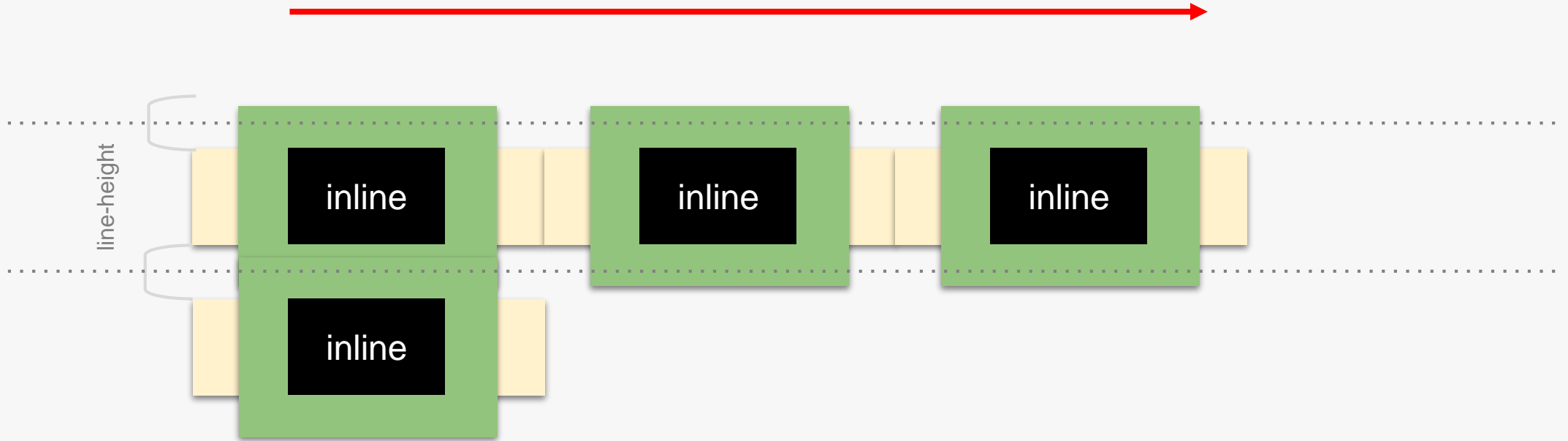
DISPLAY : INLINE



display : inline

흐름 방향	수평		
너 비	X	높 이	X
수평 마진	O	수직 마진	X
수평 패딩	O	수직 패딩	O(X)

display : inline



“

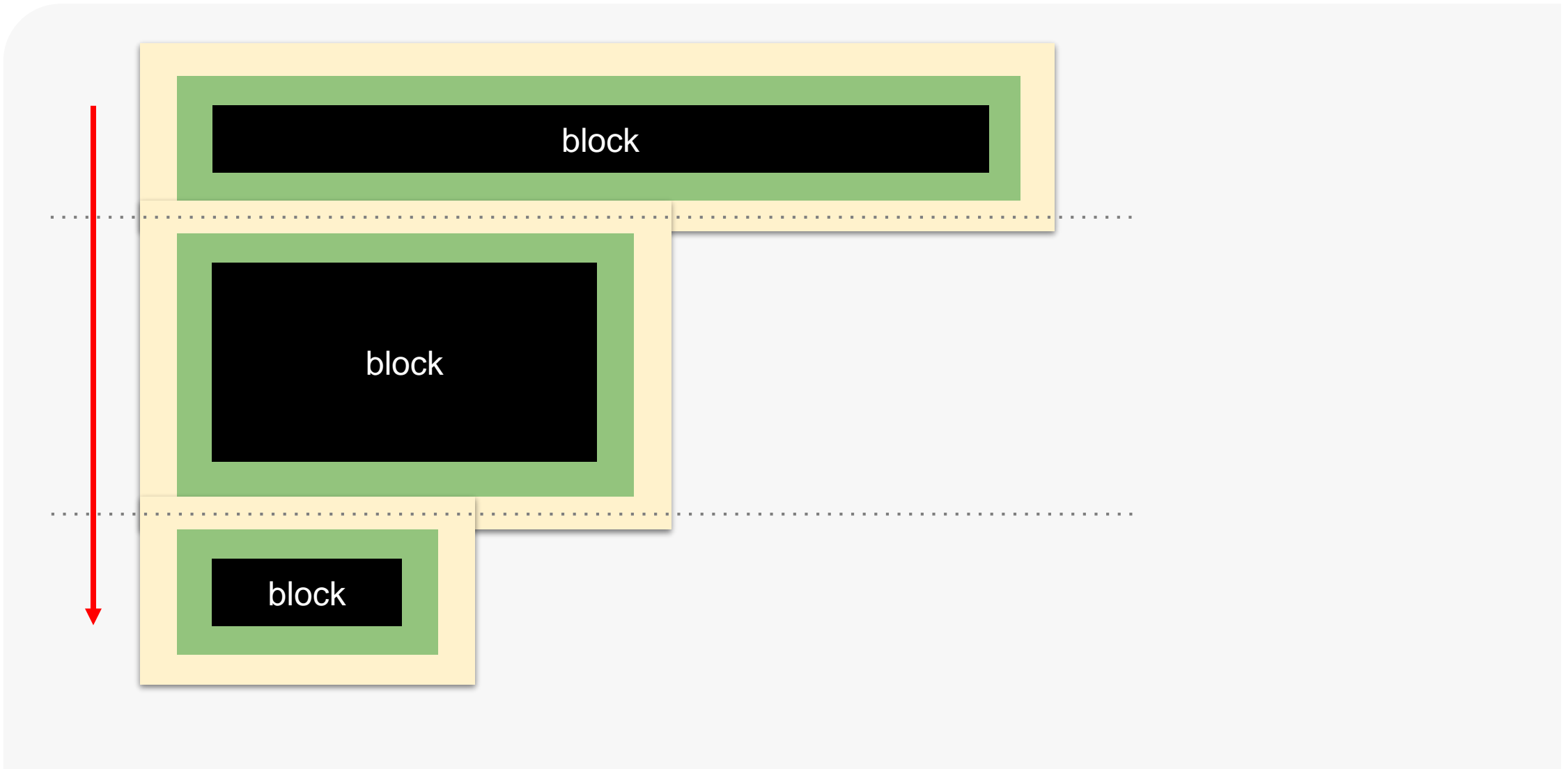
DISPLAY : BLOCK



display : block

흐름 방향	수직		
너 비	O	높 이	O
수평 마진	O	수직 마진	O(중첩)
수평 패딩	O	수직 패딩	O

display : block



“

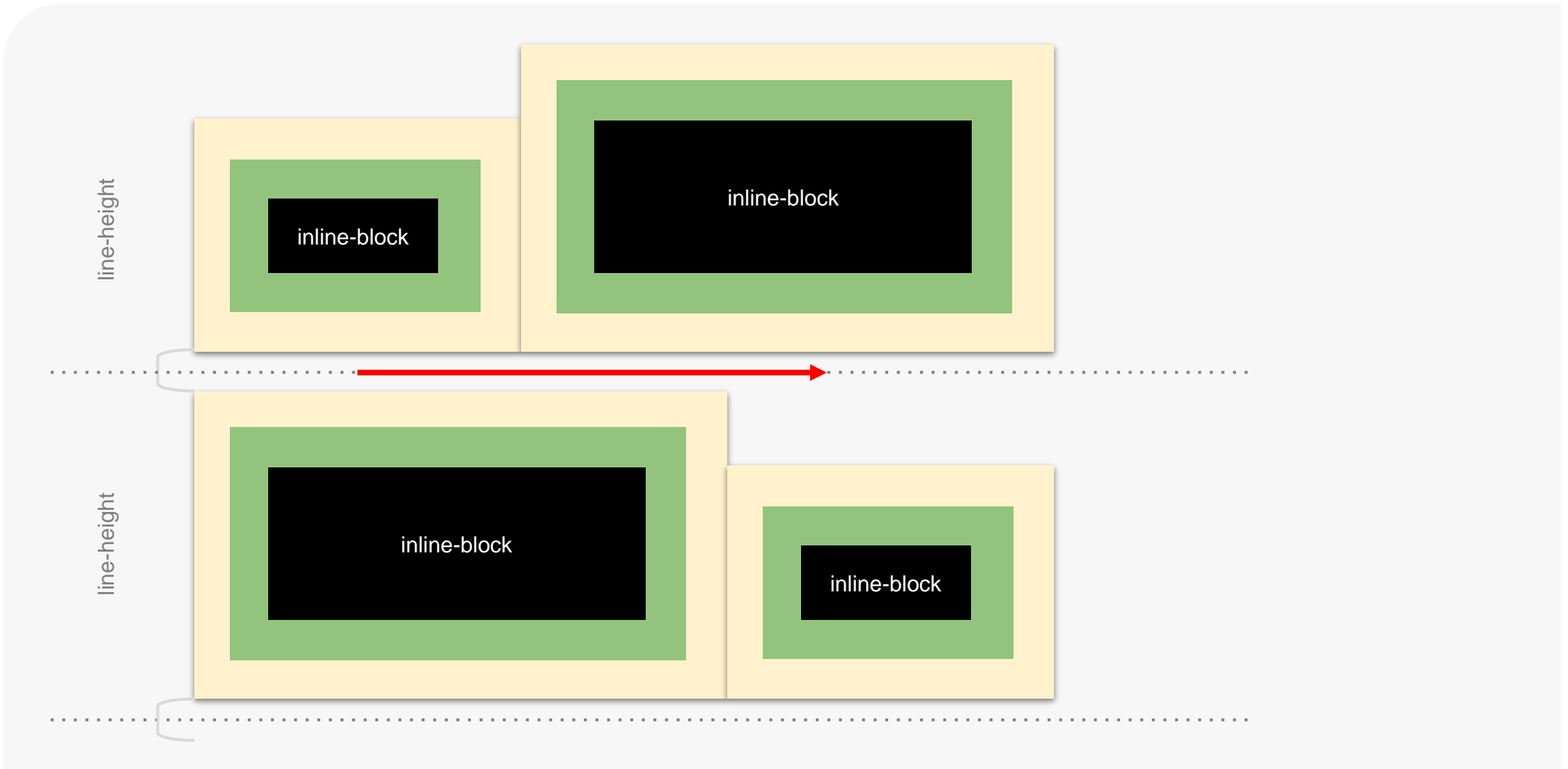
DISPLAY : INLINE-BLOCK



display : inline-block

흐름 방향	수평		
너 비	O	높 이	O
수평 마진	O	수직 마진	O(중첩X)
수평 패딩	O	수직 패딩	O

display : inline-block



“

DISPLAY : NONE



display : none

어떤 장치도 표시하거나 접근할 수 없음.

- 화면
- 인쇄
- 보조공학기기
- 마우스
- 키보드
- ...

display : none vs [hidden]

```
.hidden { display: none; }
```

==

```
<p class="desc" hidden>
```

“

Display Module Lv3



display

Lv3에서 새로 추가된 값들:

run-in | flow | **flow-root** | **flex** | **grid** | ruby | ruby-base
| ruby-text | ruby-base-container | ruby-text-
container | **contents** | inline-table | inline-flex | inline-
grid

display : flow-root

블록 컨테이너가 된다.

포함 콘텐츠는 새 블록 형식 문맥(block formatting context)이 된다.

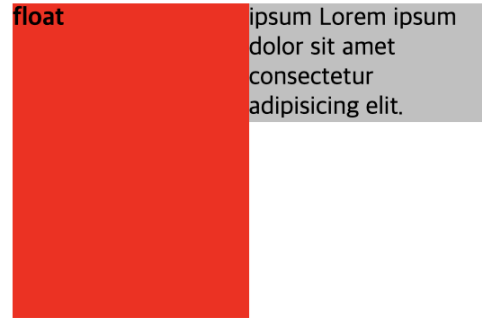
float, margin 속성을 다르게 처리.

- 블록이 된다.
- 포함한 float 요소는 컨테이너 끝에서 clear 된다.
- 부모-자식 요소의 수직 마진을 병합하지 않는다.

display : flow-root

float is cleared

block



flow-root



vertical margin not collapsed

block



flow-root



inline becomes block

```
span {display:flow-root}
```

display : flex

- flex 컨테이너 박스를 생성한다.
- flex 형식 문맥을 설정한다.
- 포함 아이টে을 1차원 기반으로 배치.

자세한 내용은 **13. CSS flex** 강의 영상 참고.

display : grid

- grid 컨테이너 박스를 생성한다.
- grid 형식 문맥을 설정한다.
- 포함 아이টে를 2차원 기반으로 배치.

자세한 내용은 **14. CSS grid** 강의 영상 참고.

CSS 배치 코드 간결함 비교

Comparison of CSS layout code brevity.

display: inline-block

Minified CSS: 234 bytes



.item--a .item--b

float: left

Minified CSS: 171 bytes



.item--a .item--b

display: flex

Minified CSS: 95 bytes



.item--a .item--b

display: grid

Minified CSS: 47 bytes



.item--a .item--b

The shortest code is the grid.(Minified CSS: 47 bytes)

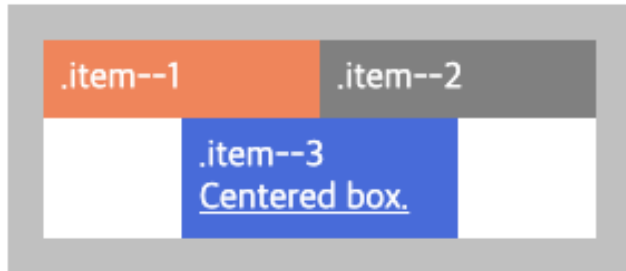
```
.container--4 {  
  display: grid;  
  grid: auto / 120px 1fr;  
}
```

Demo - <https://t.ly/OMSy>

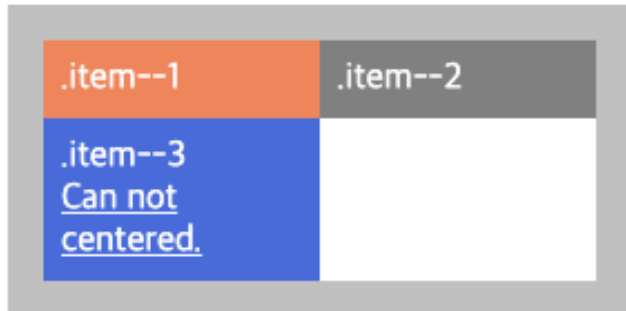
flex만 가능한 배치 (격자에 구애 받지 않음)

A 'flex' demo that 'grid' cannot do.

display: flex



display: grid



grid만 가능한 배치 (셀의 병합 가능)

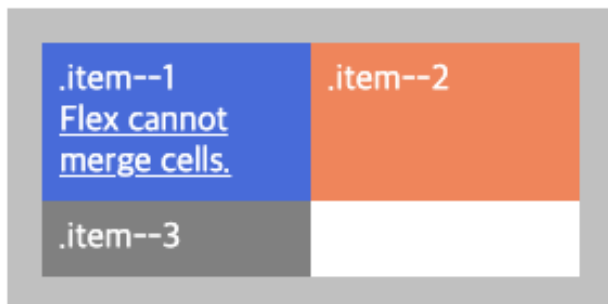
A 'grid' demo that 'flex' cannot do.

Condition: The container's height is not fixable.

display: grid



display: flex



“

POSITION



position : static

left, right, top, bottom, z-index ✕

배치 기준 없음. 흐름에 따라 배치.

position : relative

left, right, top, bottom, z-index, inset ○

박스의 현재 위치가 배치의 기준.

배치를 변경할 때 다른 박스의 흐름을 깨지 않음.

자식 또는 자손 요소의 absolute 배치 기준이 됨.

position : absolute

left, right, top, bottom, z-index, inset ○

일반적인 흐름에서 완전히 이탈.

부모, 형제의 크기나 위치에 전혀 영향을 미치지 않
음.

조상 박스가 relative, absolute, fixed, transform 일
때,

조상 기준으로 배치.

position : fixed

left, right, top, bottom, z-index, inset ○

뷰포트가 배치 기준.

조상 요소에 transform 속성이 있으면 transform 속성이 있는 요소가 배치 기준.

position : sticky

left, right, top, bottom, z-index, inset ○

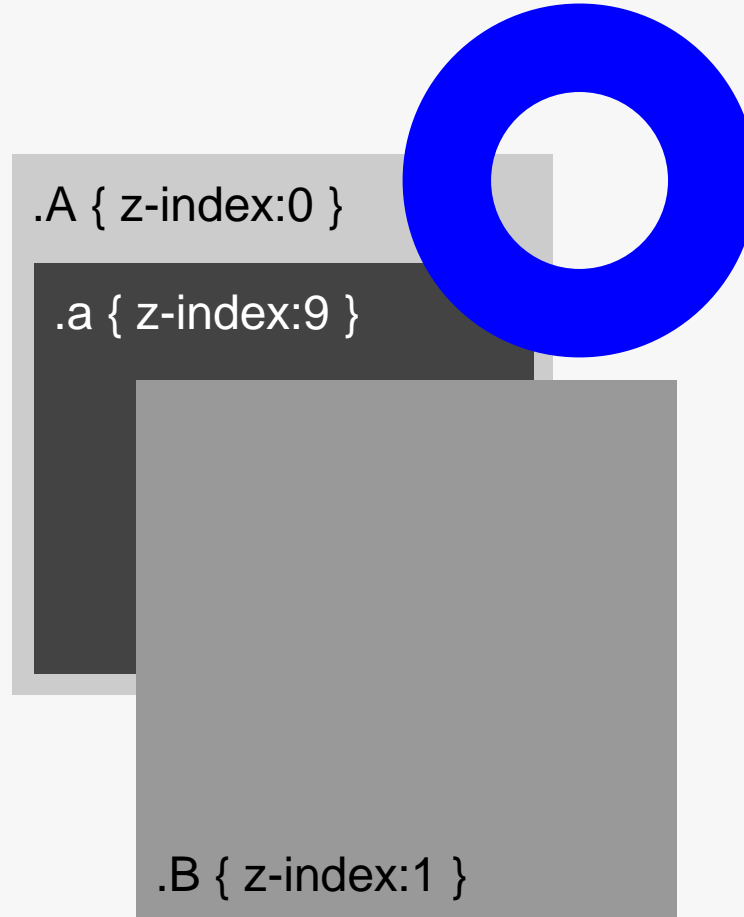
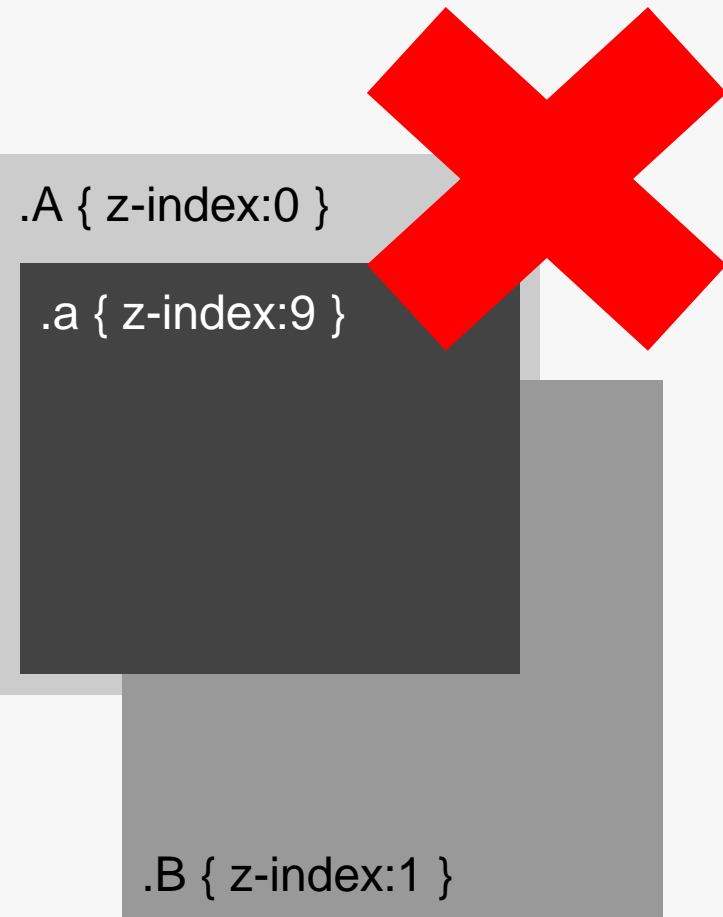
스크롤 포트가 배치 기준.

부모 요소가 스크롤 포트에 보이는 동안 스크롤 포트

기준으로 고정.

부모 요소가 스크롤 밖으로 이탈하면 고정을 멈춤.

z-index



Thank you !

