

HTML + CSS

◆ 플렉스 박스(Flex Box)

정수아

Contents

01 플렉스 박스란?

02 플렉스 컨테이너

03 플렉스 아이템



01

플렉스 박스란?

플렉스 박스가 나오기 이전

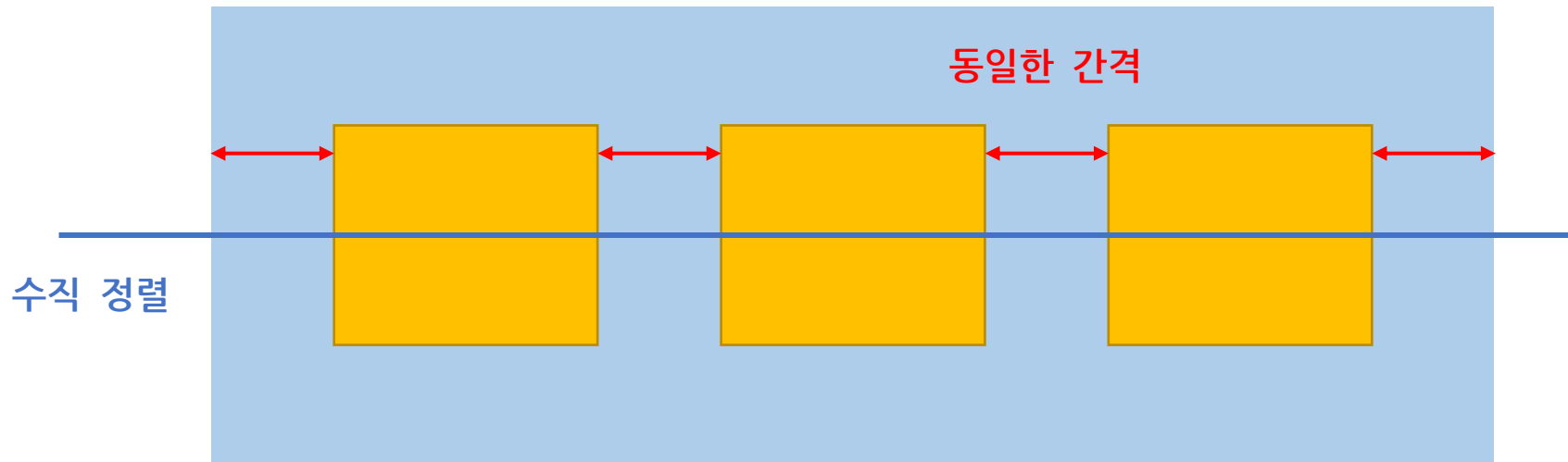
❖ 웹사이트의 레이아웃 만드는 방법

- Position 속성
- Float 속성
- Table 태그

❖ 위와 같은 방법의 단점

- 복잡하고 시간이 오래 걸림
- 박스 안 아이템들의 수직 정렬 등에 제약이 있음

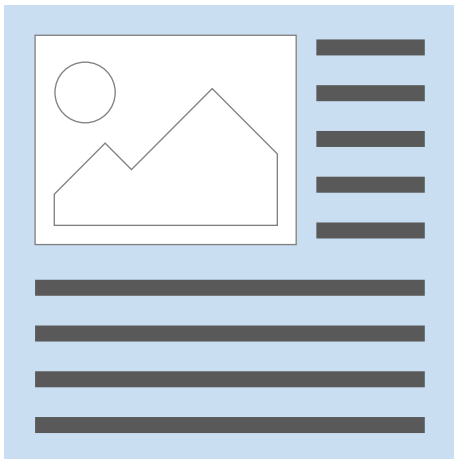
플렉스 박스가 나오기 이전



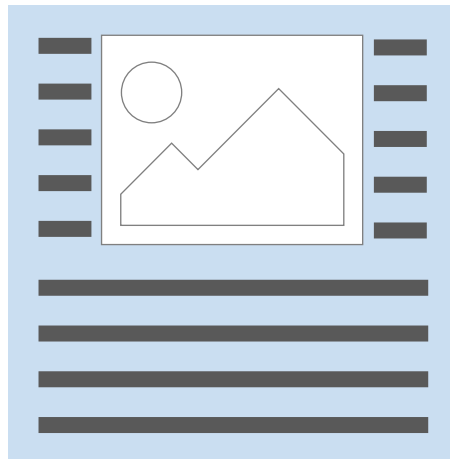
플렉스 박스가 나오기 이전

❖ float의 기본 목적

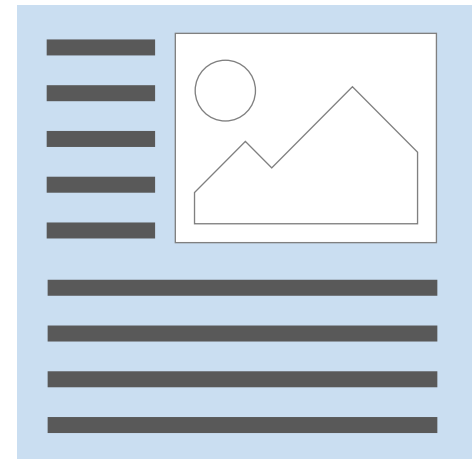
- 이미지와 텍스트를 어떤 방식으로 배치할 것인지를 정하는 css 속성
- ‘배치’와는 전혀 상관 없는 속성으로 박스 배치를 해오고 있었음



float : left



float : center



float : right

플렉스 박스(Flex Box)란?

❖ 플렉스 박스

- 가변적인 박스를 만드는 기술
- 박스 요소들을 다양하고 자유롭게 배치할 수 있음
- 반응형 웹사이트를 제작하기에 최적의 기술



플렉스 박스(Flex Box)란?

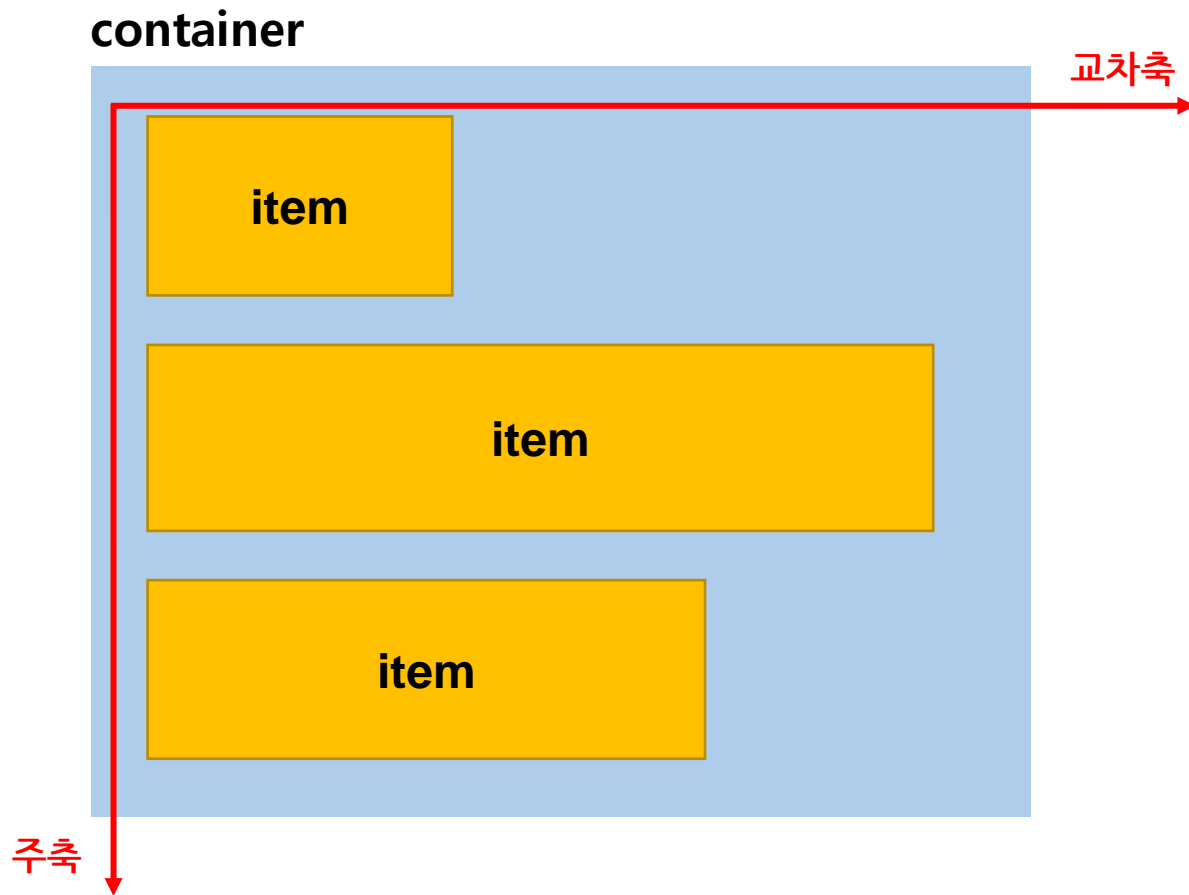
❖ 플렉스 박스의 축

container



축	설명
주축	기본적으로 수평 방향(왼쪽에서 오른쪽으로 배치)
교차축	기본적으로 수직 방향(위에서 아래로 배치)

플렉스 박스(Flex Box)란?





02

플렉스 컨테이너

플렉스 컨테이너

❖ 플렉스 컨테이너(= 부모 박스)

- 가변적인 박스로 작동하기 위한 기본 개념
- 플렉스 컨테이너에 플렉스 박스의 특정 속성값을 적용하면 가변적인 박스로 작동하게 됨
- 플렉스 아이템의 크기, 배치 방향 및 순서, 정렬 방법 등을 제어할 수 있음



플렉스 컨테이너 속성

❖ display 속성

- 플렉스 아이টে을 감싸는 요소로 플렉스 컨테이너를 쓰려고 할 때 사용하는 속성

속성 값	설명
flex	컨테이너 안의 플렉스 아이টে을 블록 레벨 요소로 배치
inline-flex	컨테이너 안의 플렉스 아이টে을 인라인 레벨 요소로 배치

플렉스 컨테이너 속성

❖ flex-direction 속성

- 플렉스 컨테이너 안에서 플렉스 아이টে를 배치하는 주축과 방향을 지정하는 속성

속성 값	설명
row(기본값)	<ul style="list-style-type: none">• 주축을 가로로 지정• 왼쪽에서 오른쪽으로 배치
row-reverse	<ul style="list-style-type: none">• 주축을 가로로 지정• 오른쪽에서 왼쪽으로 배치
column	<ul style="list-style-type: none">• 주축을 세로로 지정• 위쪽에서 아래쪽으로 배치
column-reverse	<ul style="list-style-type: none">• 주축을 세로로 지정• 아래쪽에서 위쪽으로 배치

[실습] flex-direction 속성

main.html

```
<div class="container">
  <div class="item item1"><p>1</p></div>
  <div class="item item2"><p>2</p></div>
  <div class="item item3"><p>3</p></div>
  <div class="item item4"><p>4</p></div>
  <div class="item item5"><p>5</p></div>
</div>
```

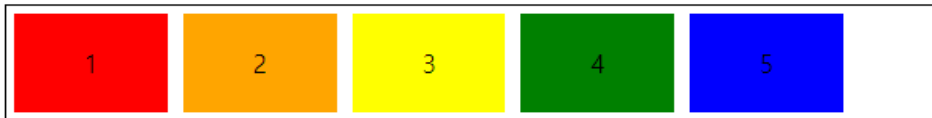
style.css

```
.container {
  display: flex;
  border: 1px solid black;
}

.item {
  padding: 5px 45px;
  margin: 5px;
}

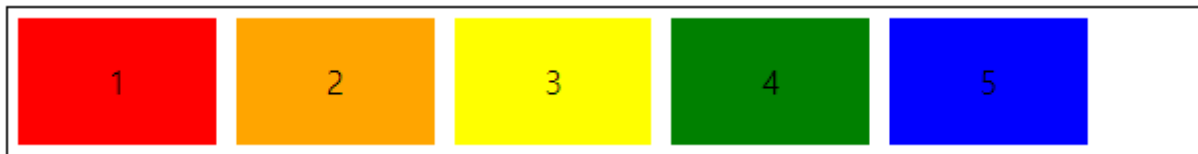
.item1 { background : red; }
.item2 { background : orange; }
.item3 { background : yellow; }
.item4 { background : green; }
.item5 { background : blue; }
```

실행 결과

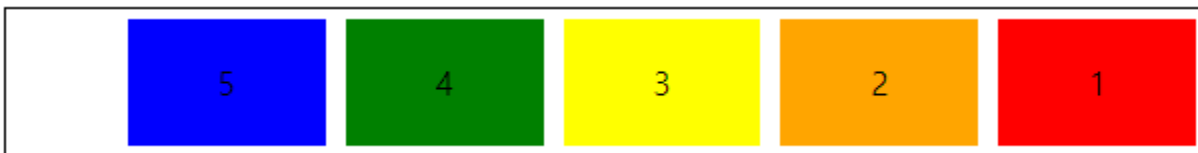


[실습] flex-direction 속성

❖ flex-direction : row;



❖ flex-direction : row-reverse;



[실습] flex-direction 속성

❖ **flex-direction : column;**



❖ **.item { width:100px; }**

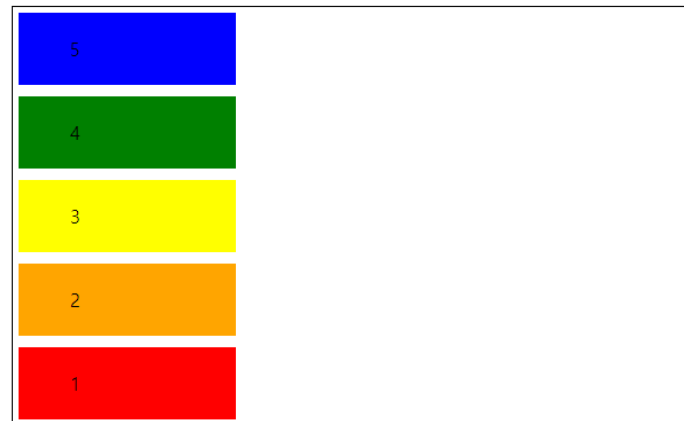


[실습] flex-direction 속성

❖ **flex-direction : column-reverse;**



❖ **.item { width:100px; }**



플렉스 컨테이너 속성

❖ flex-wrap 속성

- 플렉스 아이টে를 여러 줄에 걸쳐 표시하는 속성

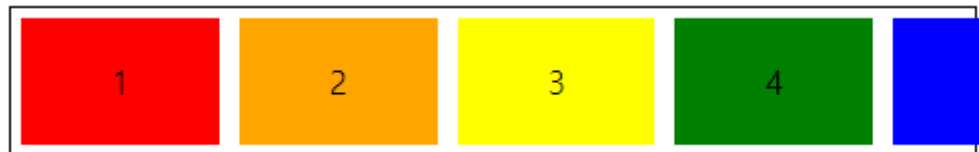
속성 값	설명
nowrap(기본값)	플렉스 아이টে를 한 줄에 표시
wrap	플렉스 아이টে를 여러 줄에 표시
wrap-reverse	플렉스 아이টে를 여러 줄에 표시하되 시작점과 끝점이 바뀜

[실습] flex-wrap 속성

❖ flex-wrap : nowrap;



화면을 늘렸을 때



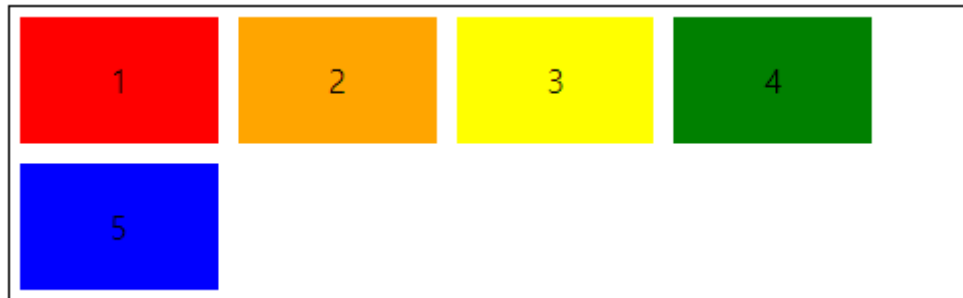
화면을 줄였을 때

[실습] flex-wrap 속성

❖ flex-wrap : wrap;



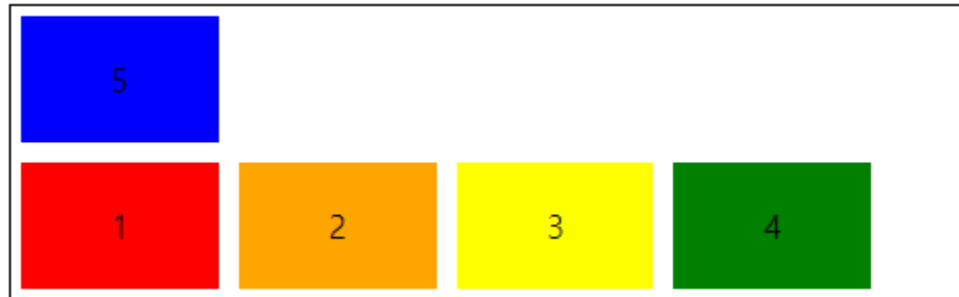
화면을 늘렸을 때



화면을 줄였을 때

[실습] flex-wrap 속성

❖ flex-wrap : wrap-reverse;



플렉스 컨테이너 속성

❖ flex-flow 속성

- flex-direction 속성과 flex-wrap 속성을 한번에 지정하는 속성
- 기본값은 row nowrap
- 작성 방법

```
flex-flow : row nowrap;  
flex-flow : column wrap;
```

플렉스 컨테이너 속성

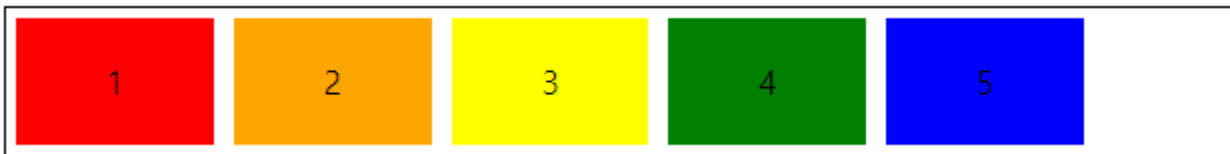
❖ justify-content 속성

- 주축에서 플렉스 아이템 간 간격을 지정하는 속성

속성 값	설명
flex-start(기본값)	아이템들을 시작점에 맞춰 배치
flex-end	아이템들을 끝점에 맞춰 배치
center	아이템들을 중앙에 배치
space-between	첫 번째 아이템과 끝 아이템을 시작점과 끝점에 배치한 후, 나머지 아이템들은 그 사이에 같은 간격으로 배치
space-around	모든 아이템을 같은 간격으로 배치
space-evenly	아이템들의 사이와 양 끝에 균일한 간격을 생성

[실습] justify-content 속성

❖ **justify-content:flex-start;**

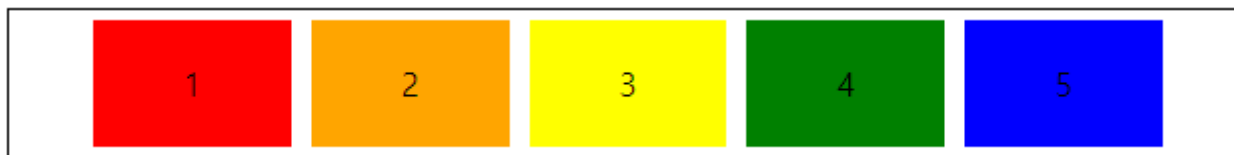


❖ **justify-content:flex-end;**

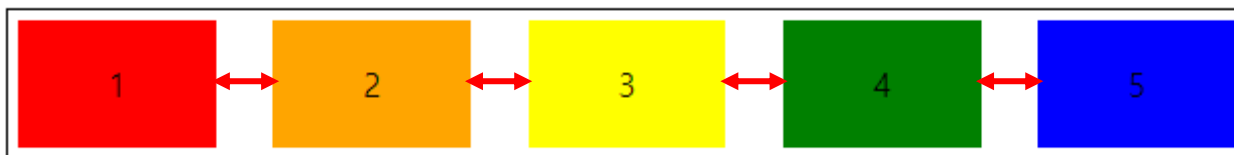


[실습] justify-content 속성

❖ **justify-content:center;**

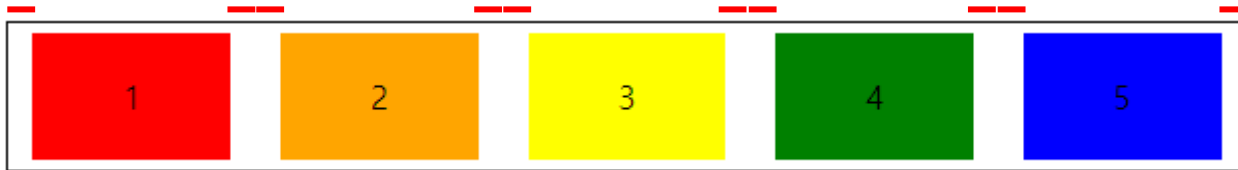


❖ **justify-content:space-between;**

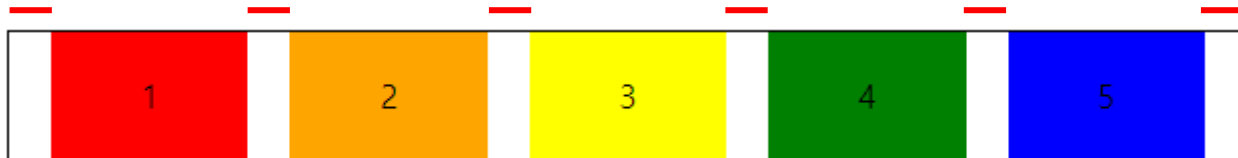


[실습] justify-content 속성

❖ **justify-content:space-around;**



❖ **justify-content:space-evenly;**



플렉스 컨테이너 속성

❖ align-items 속성

- 교차축에서 플렉스 아이템 간 간격을 지정하는 속성

속성 값	설명
flex-start	아이템들을 교차축의 시작점에 배치
flex-end	아이템들을 교차축의 끝점에 배치
center	아이템들을 교차축의 중앙에 배치
baseline	교차축의 시작점에 배치되는 아이템의 글자 베이스라인에 맞춰 다른 아이템들을 배치
stretch(기본값)	아이템들을 확장해서 간격 없이 배치

[실습] align-items 속성

main.html

```
<div class="container">
  <div class="item item1"><p>1</p></div>
  <div class="item item2"><p>2</p></div>
  <div class="item item3"><p>3</p></div>
  <div class="item item4"><p>4</p></div>
  <div class="item item5"><p>5</p></div>
</div>
```

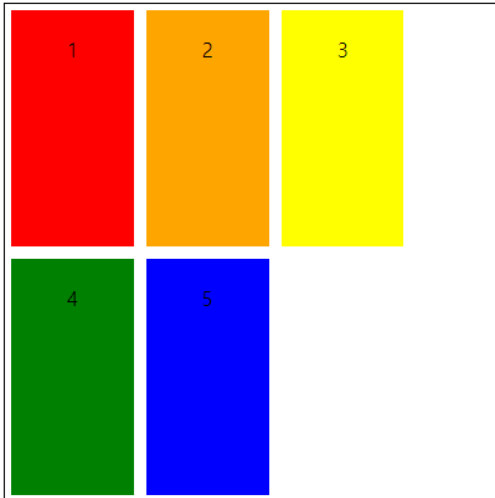
style.css

```
.container {
  display: flex;
  border: 1px solid black;
  width : 400px;
  height : 400px;
  flex-flow : row wrap;
}

.item {
  padding: 5px 45px;
  margin: 5px;
}

.item1 { background : red; }
.item2 { background : orange; }
.item3 { background : yellow; }
.item4 { background : green; }
.item5 { background : blue; }
```

실행 결과



[실습] align-items 속성

main.html

```
<div class="container">
  <div class="item item1"><p>1</p></div>
  <div class="item item2"><p>2</p></div>
  <div class="item item3"><p>3</p></div>
  <div class="item item4"><p>4</p></div>
  <div class="item item5"><p>5</p></div>
</div>
```

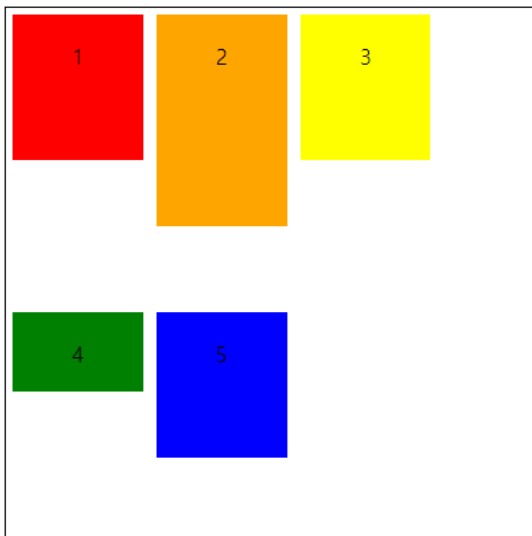
style.css

```
.container {
  display: flex;
  border: 1px solid black;
  width : 400px;
  height : 400px;
  flex-flow : row wrap;
}
```

```
.item {
  padding: 5px 45px;
  margin: 5px;
}
```

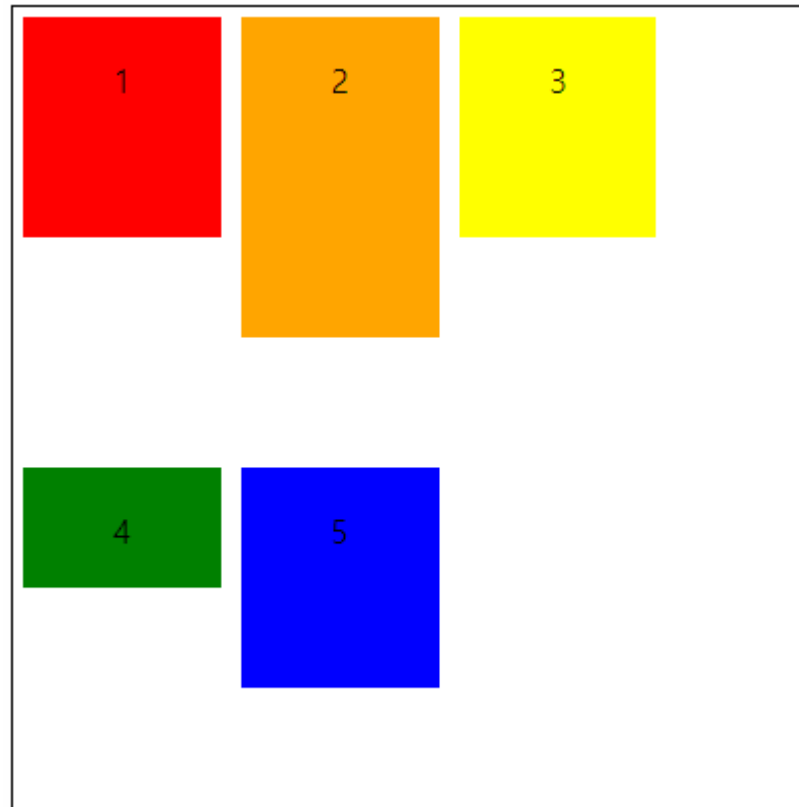
```
.item1 { background : red; height : 100px;}
.item2 { background : orange; height : 150px;}
.item3 { background : yellow; height : 100px;}
.item4 { background : green; height : 50px;}
.item5 { background : blue; height : 100px;}
```

실행 결과



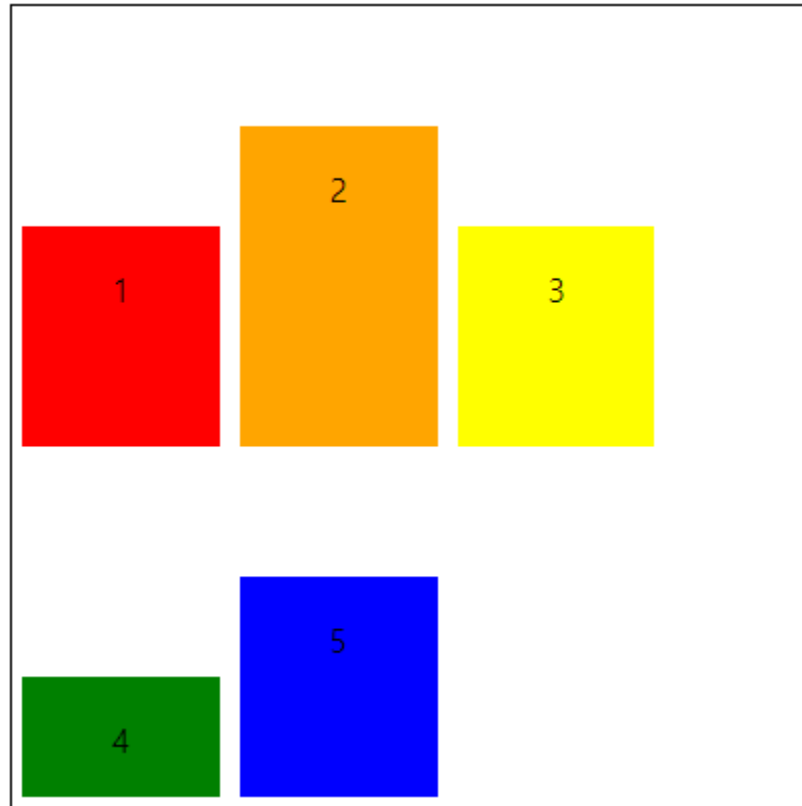
[실습] align-items 속성

❖ align-items:flex-start;



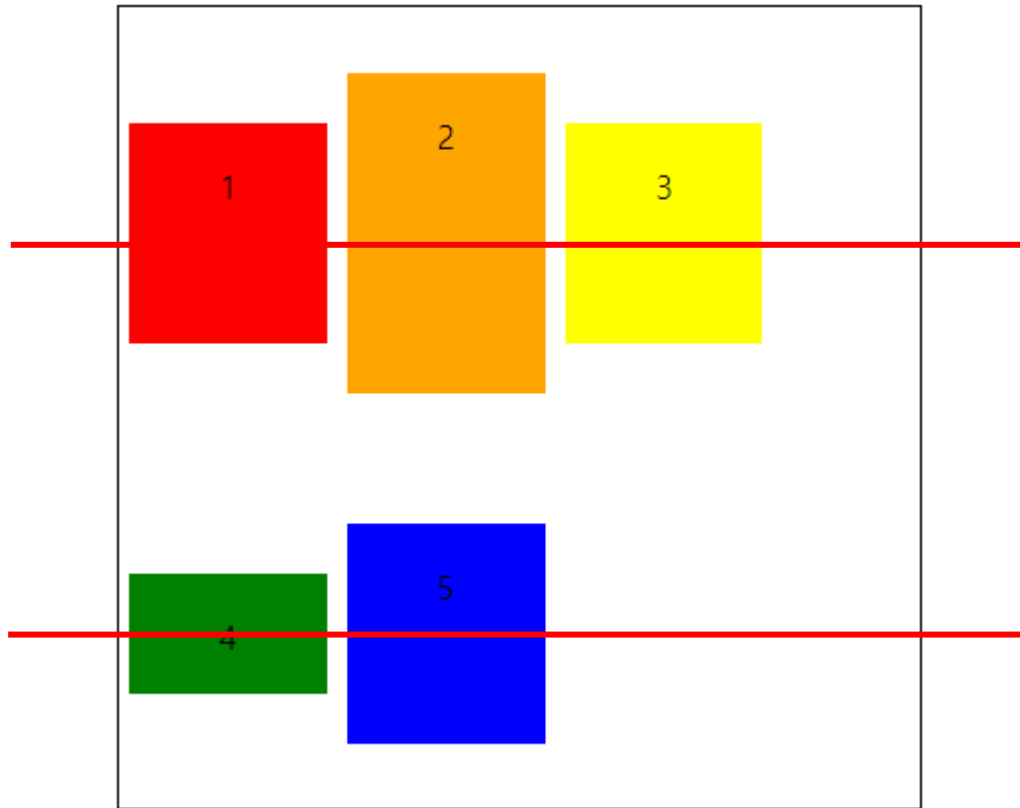
[실습] align-items 속성

❖ align-items:flex-end;



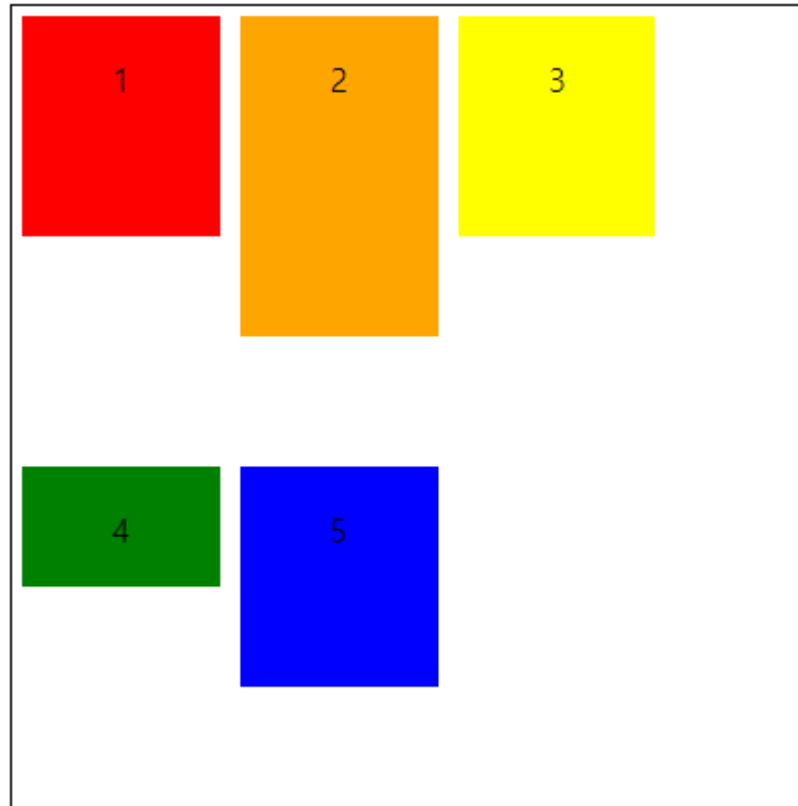
[실습] align-items 속성

❖ align-items:center;



[실습] align-items 속성

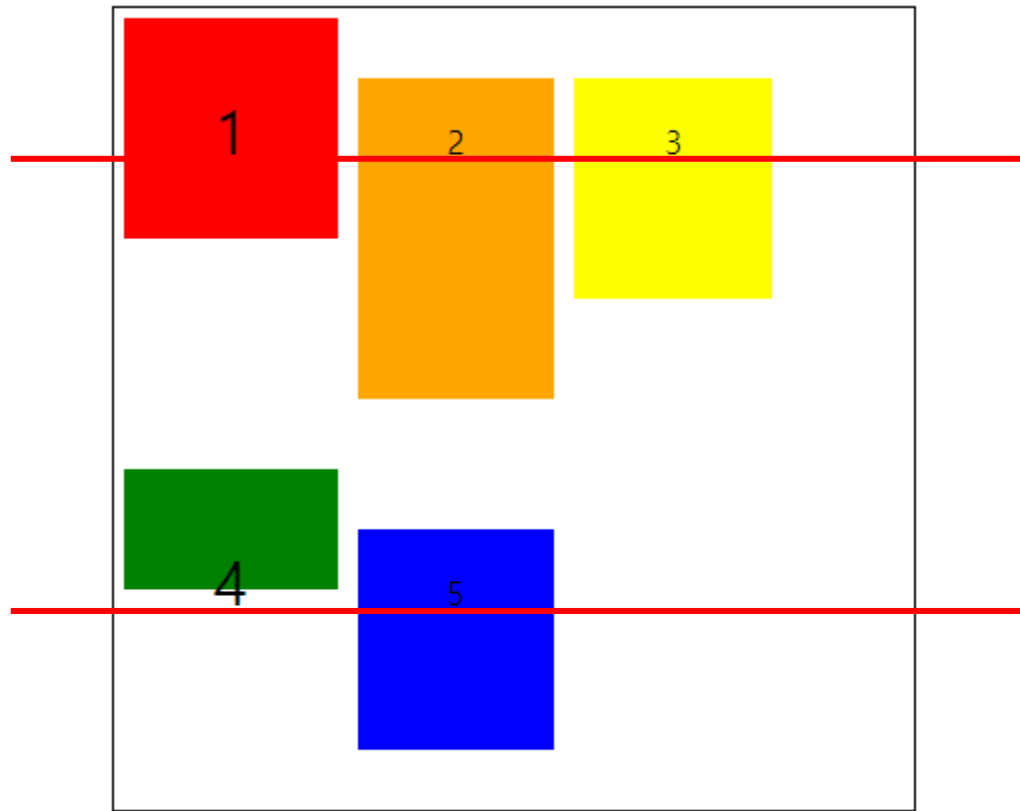
❖ align-items:baseline;



[실습] align-items 속성

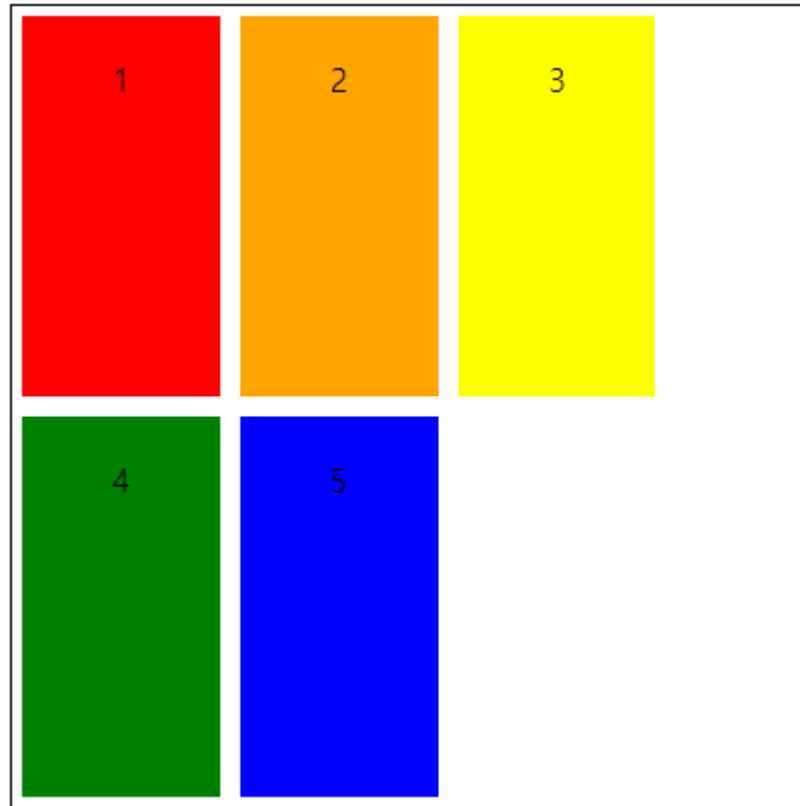
❖ align-items: baseline;

- .item1, .item4의 글자 크기 30px



[실습] align-items 속성

❖ align-items:stretch;



플렉스 컨테이너 속성

❖ align-content 속성

- 교차 축에서 플렉스 아이템 간 간격을 지정하는 속성
- 아이템이 여러 줄 일 때만 적용되는 속성

속성 값	설명
flex-start	아이템들을 시작점에 맞춰 배치
flex-end	아이템들을 끝점에 맞춰 배치
center	아이템들을 중앙에 배치
space-between	첫 번째 아이템과 끝 아이템을 시작점과 끝점에 배치한 후, 나머지 아이템들은 그 사이에 같은 간격으로 배치
space-around	모든 아이템을 같은 간격으로 배치
stretch(기본값)	플렉스 아이템을 늘려서 간격 없이 배치

justify-content
속성

[실습] align-content 속성

main.html

```
<div class="container">
  <div class="item item1"><p>1</p></div>
  <div class="item item2"><p>2</p></div>
  <div class="item item3"><p>3</p></div>
  <div class="item item4"><p>4</p></div>
  <div class="item item5"><p>5</p></div>
</div>
```

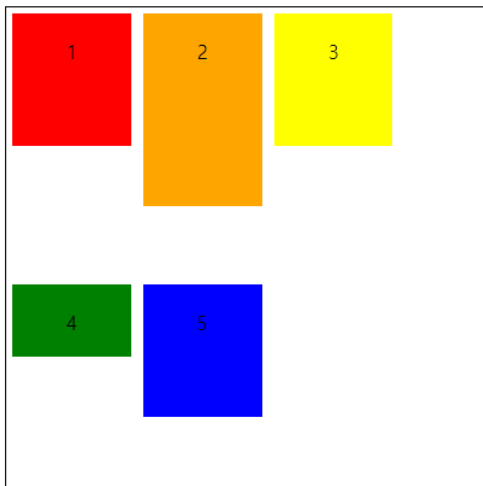
style.css

```
.container {
  display: flex;
  border: 1px solid black;
  width : 400px;
  height : 400px;
  flex-flow : row wrap;
}
```

```
.item {
  padding: 5px 45px;
  margin: 5px;
}
```

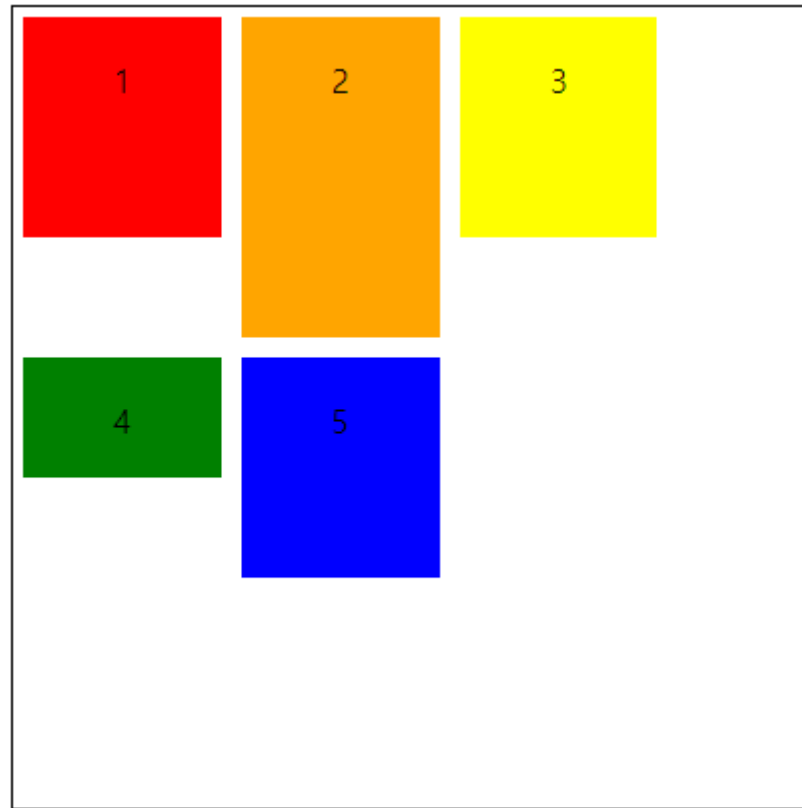
```
.item1 { background : red; height : 100px;}
.item2 { background : orange; height : 150px;}
.item3 { background : yellow; height : 100px;}
.item4 { background : green; height : 50px;}
.item5 { background : blue; height : 100px;}
```

실행 결과



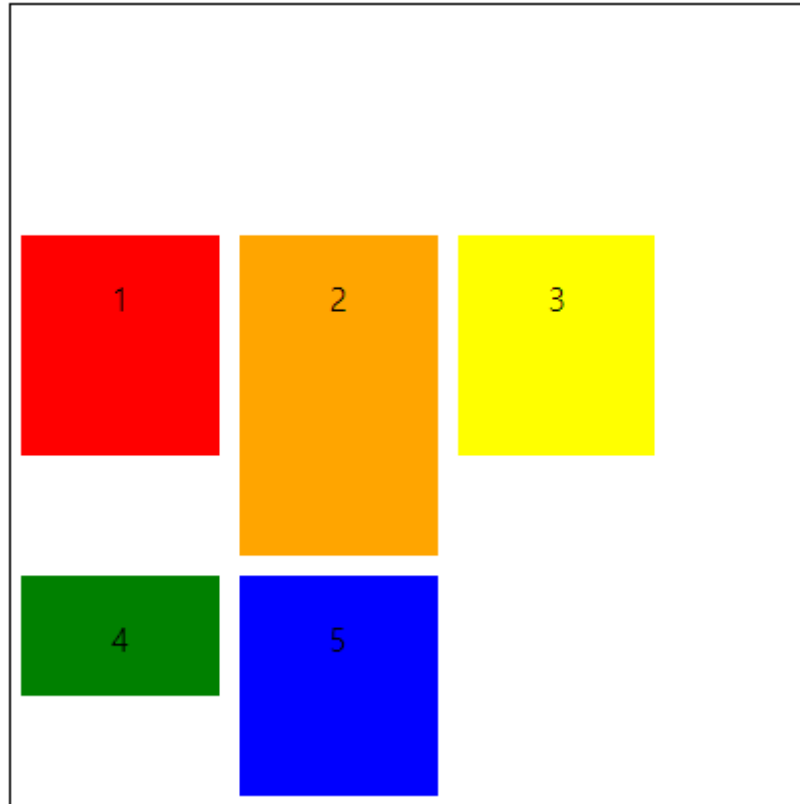
[실습] align-content 속성

❖ align-content:flex-start



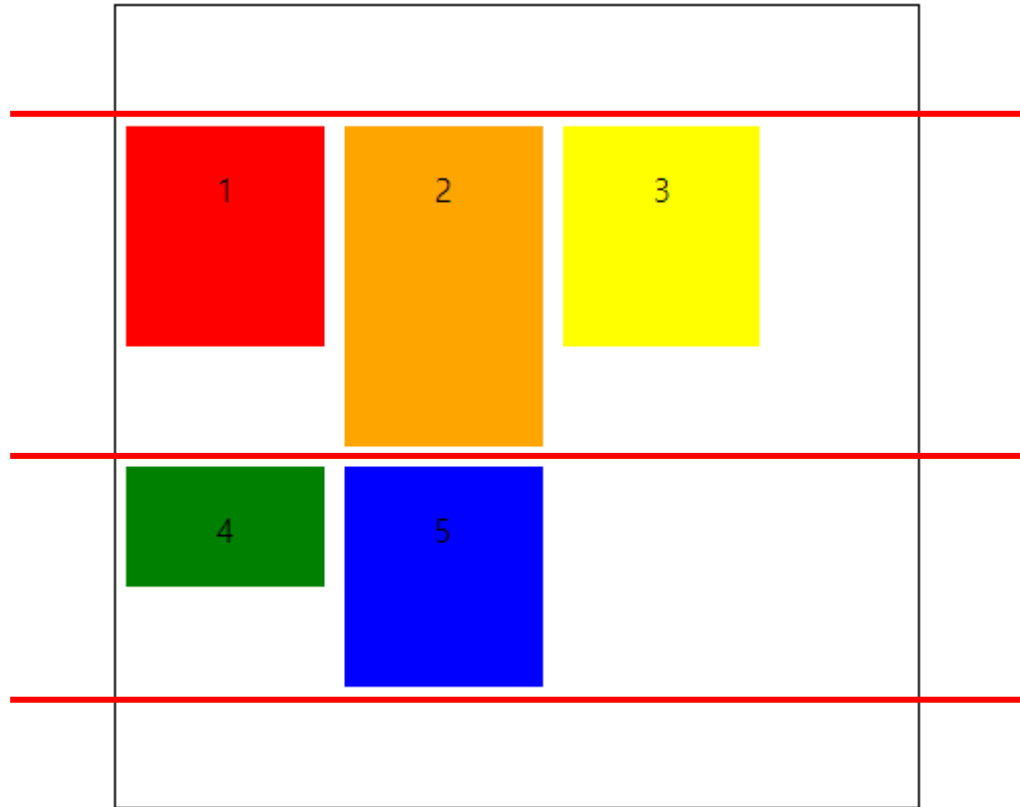
[실습] align-content 속성

❖ align-content: flex-end;



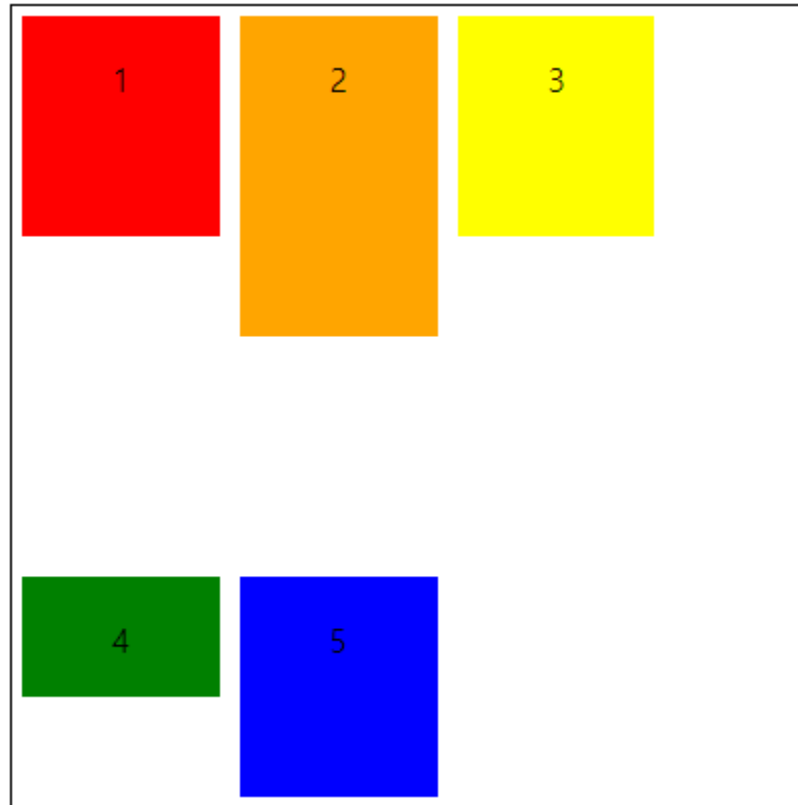
[실습] align-content 속성

❖ align-content:center;



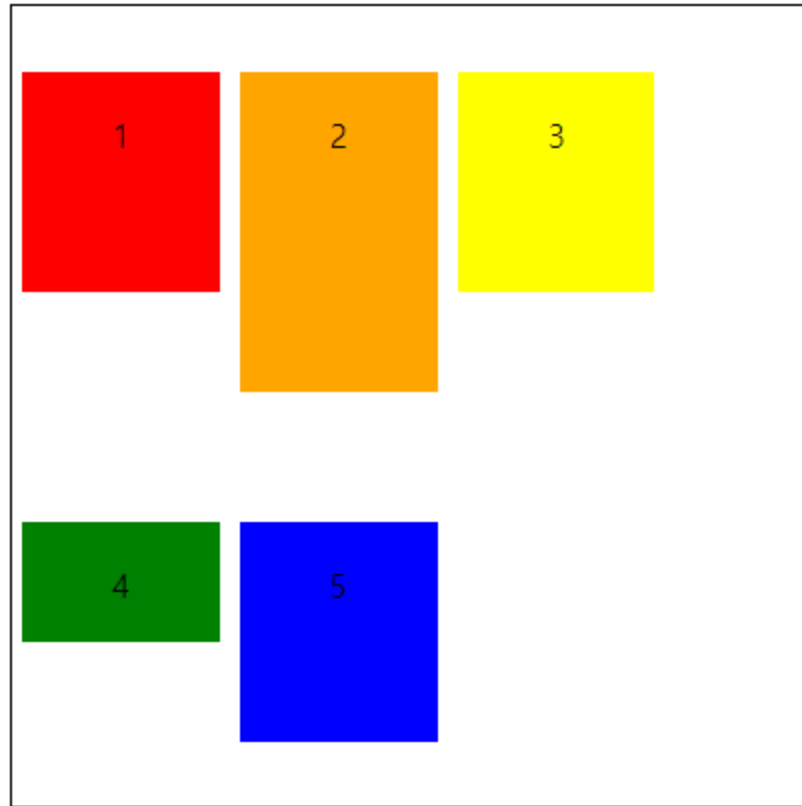
[실습] align-content 속성

❖ align-content:space-between;



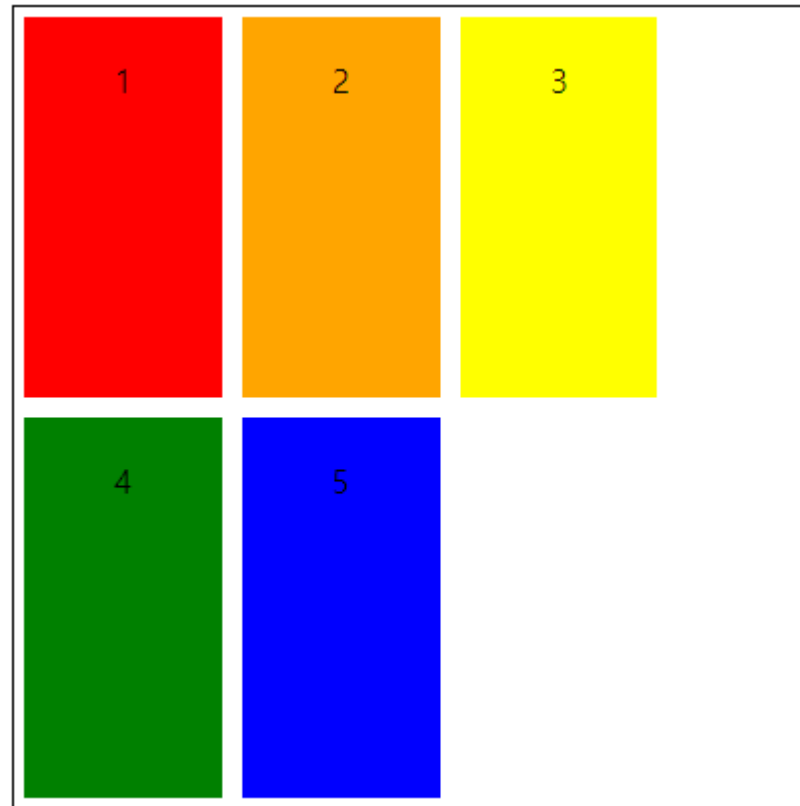
[실습] align-content 속성

❖ align-content:space-around



[실습] align-content 속성

❖ align-content:stretch





03

플렉스 아이템

플렉스 아이템

❖ 플렉스 아이템(=자식 박스)

- 플렉스 컨테이너의 자식 박스
- 플렉스 박스 레이아웃을 적용할 대상



플렉스 아이템 속성

❖ align-self 속성

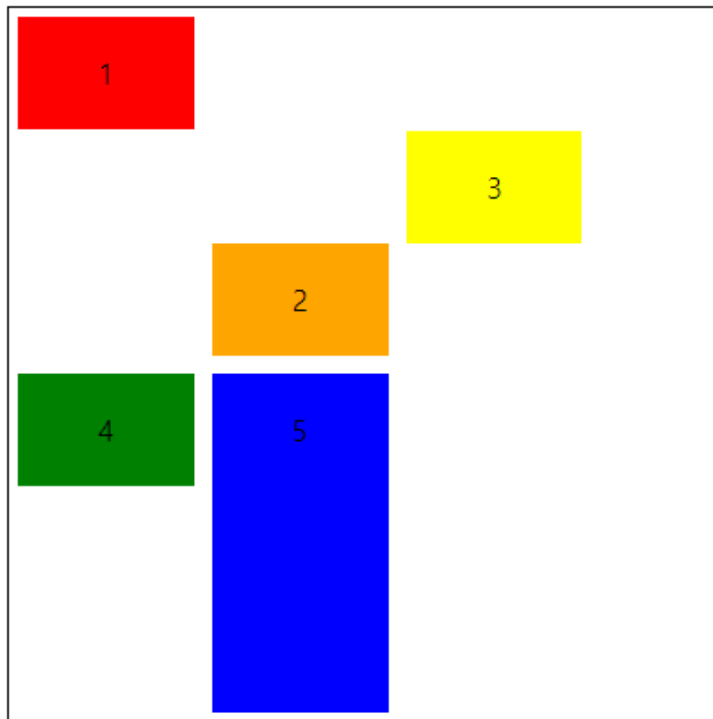
- 플렉스 아이템을 각각 원하는 형태로 배치할 수 있는 속성

속성 값	설명
flex-start	아이템들을 교차축의 시작점에 배치
flex-end	아이템들을 교차축의 끝점에 배치
center	아이템들을 교차축의 중앙에 배치
baseline	교차축의 시작점에 배치되는 아이템의 글자 베이스라인에 맞춰 다른 아이템들을 배치
stretch(기본값)	아이템들을 확장해서 간격 없이 배치

[실습] align-self 속성

```
.item1 { align-self: flex-start; }  
.item2 { align-self: flex-end; }  
.item3 { align-self: center; }  
.item4 { align-self: baseline; }  
.item5 { align-self: stretch; }
```

실행 결과



플렉스 아이템

❖ flex 속성

- 플렉스 아이템의 크기 조절 및 가변적인 박스로 작동하게 하는 속성

속성 명	속성 값
flex	[flex-grow] [flex-shrink] [flex-basis] • 0 1 auto(기본 값)

플렉스 아이템

❖ flex 속성 값

속성 값	설명
flex-grow	<ul style="list-style-type: none">• 플렉스 컨테이너 내부 공간에서 플렉스 아이템의 너비 증가 비율을 설정• 기본값 0• width, flex-basis 속성 값에 따라 늘어나는 크기가 변할 수 있음
flex-shrink	<ul style="list-style-type: none">• 플렉스 컨테이너 내부 공간에서 플렉스 아이템의 너비 감소 비율을 설정• 기본값 1• width, flex-basis 속성 값에 따라 줄어드는 크기가 변할 수 있음
flex-basis	<ul style="list-style-type: none">• 플렉스 아이템의 기본 크기를 설정하기 위한 속성• 기본값 auto• width 속성에서 사용할 수 있는 모든 값을 사용 가능

[실습] flex 속성

main.html

```
<div class="container">
  <div class="item item1"><p>1<p></div>
  <div class="item item2"><p>2</p></div>
  <div class="item item3"><p>3<p></div>
</div>
```

style.css

```
.container {
    display: flex;
    border: 1px solid black;
}

.item {
    margin: 5px;
    text-align: center;
    width: 50px;
    height: 50px;
}
```

```
.item1 {
    background: red;
}

.item2 {
    background: orange;
}

.item3 {
    background: yellow;
}
```

[실습] flex-grow 속성

❖ .item1 { flex-grow : 1 }



❖ .item2 { flex-grow : 1 }



[실습] flex-grow 속성

❖ `.item3 { flex-grow : 1 }`



[실습] flex-shrink 속성

```
.item {  
  margin: 5px;  
  text-align: center;  
  width: 100px;  
  height: 50px;  
}
```

❖ .item1 { flex-shrink : 2 }

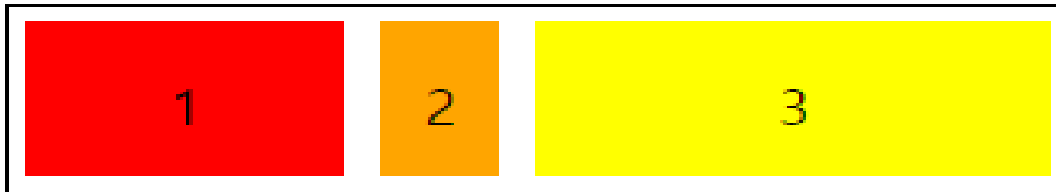


[실습] flex-shrink 속성

❖ .item2 { flex-shrink : 2 }



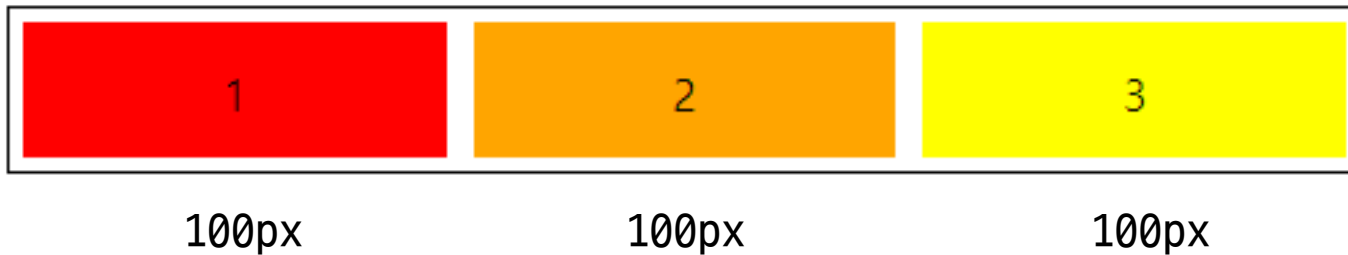
❖ .item2 { flex-shrink : 3 }



[실습] flex-basis 속성

```
.item {  
    margin: 5px;  
    text-align: center;  
    width: 100px;  
    height: 50px;  
}
```

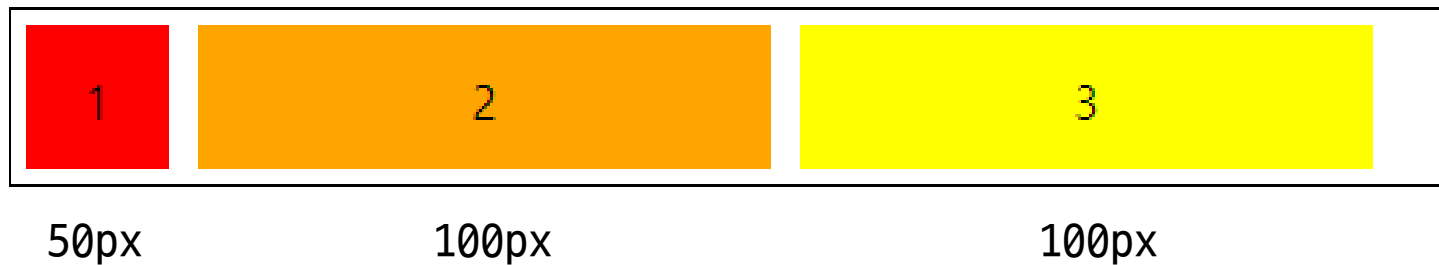
❖ **.item1 { flex-basis : auto }**



[실습] flex-basis 속성

```
.item {  
    margin: 5px;  
    text-align: center;  
    width: 100px;  
    height: 50px;  
}
```

❖ **.item1 { flex-basis : 50px }**



[실습] flex 속성

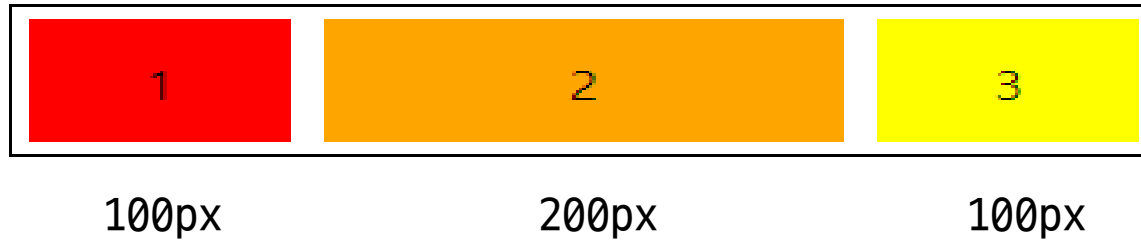
```
.item {  
    margin: 5px;  
    text-align: center;  
    width: 100px;  
    height: 50px;  
}
```

❖ **.item2 { flex : 1 }**



[실습] flex 속성

❖ `.item2 { flex : 1 200px }`

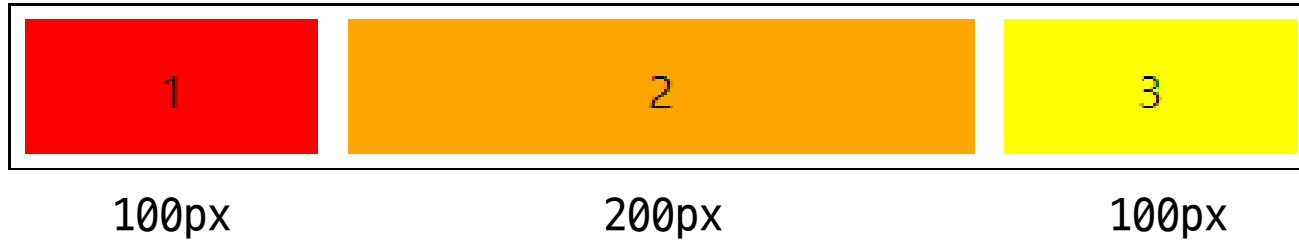


❖ `.item2 { flex : 1 2 }`



[실습] flex 속성

❖ .item2 { flex : 1 2 200px }



Flexbox 연습 사이트

❖ Flexbox froggy

- <https://flexboxfroggy.com/#ko>

❖ Flexbox Defense

- <http://www.flexboxdefense.com/>

THANK 😊 YOU