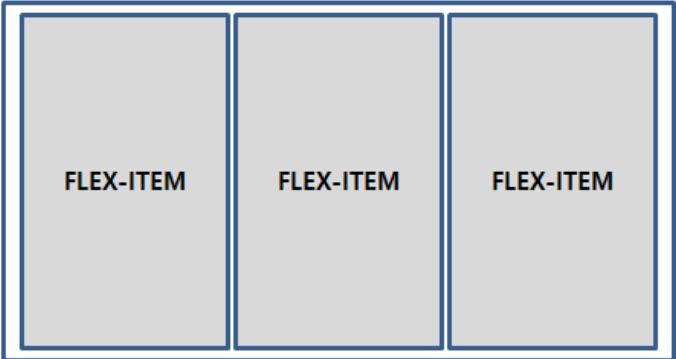


1. 플렉서블 박스

플렉서블 박스에서 부모박스는 가변적인 박스로 작동하기 위한 기본 개념 **wrap**처럼 모든 요소를 감싸고 있는 존재) 이 부모 박스에 플렉서블 박스에 특정 속성값을 적용하여 가변적인 박스로 작동하도록 설정

아래와 같이 부모요소에 **display:flex**를 설정하면 특별한 설정없이 자식요소들이 3등분으로 가로 배열이 된다.



```
<style>
    .parent { display: flex; }
    .parent div {background:silver;}
</style>

<div class="parent">
    <div>flex-item</div>
    <div>flex-item</div>
    <div>flex-item</div>
</div>
```

Basic

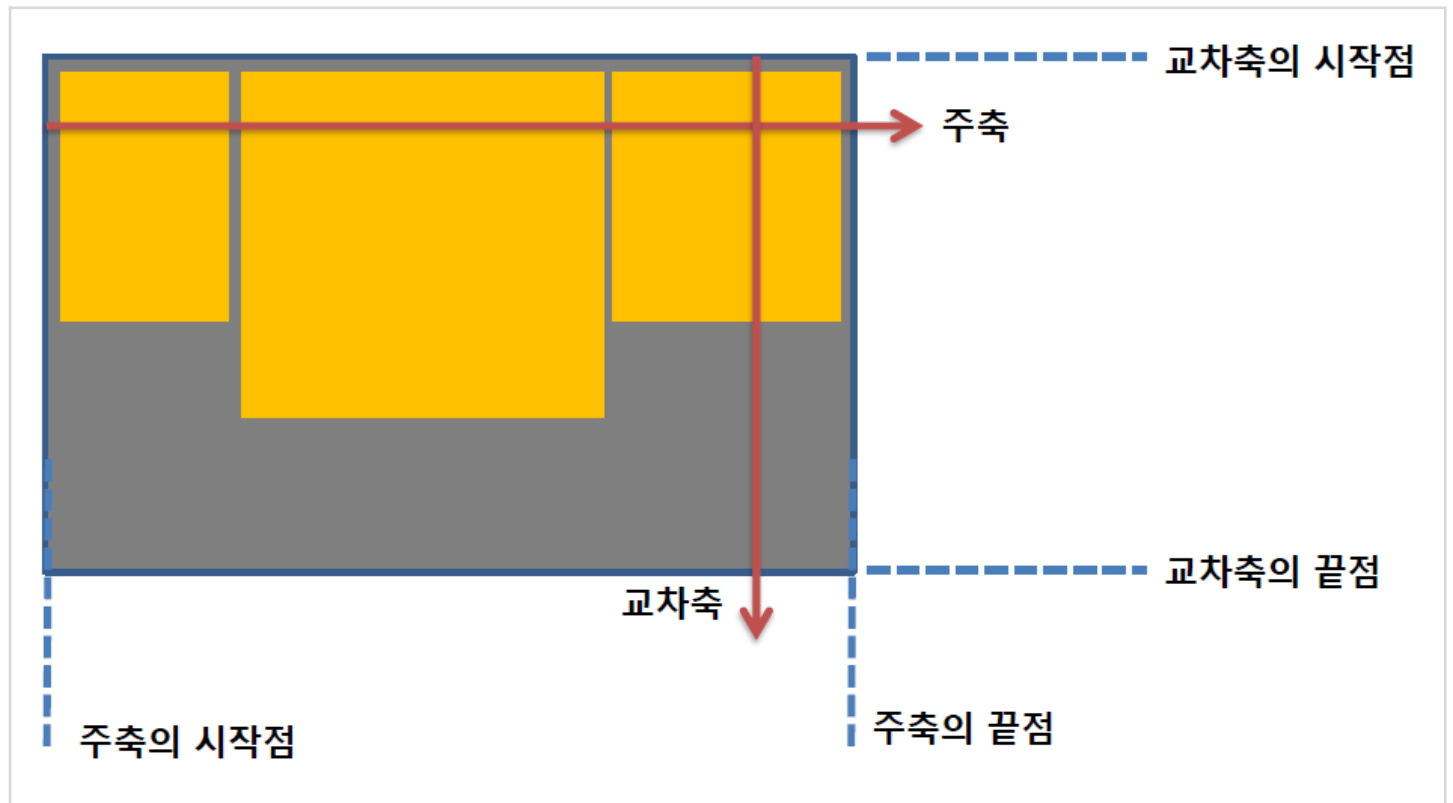
<https://codepen.io/alikerock/pen/WNNEaom>

부모요소를 플렉스 박스, 자식요소를 플렉스 아이템이라고 한다.

플렉서블 레이아웃 속성들

플렉스 박스	플렉스 요소
display	order
flex-direction	align-self
justify-content	flex
align-items	
flex-wrap	
flow-flow	
align-content	

2. 플렉서블 박스의 기본개념



기본값 : 주축은 가로이다. 요소들이 가로로 배치된다.

기본값 : 교차축은 세로이다. 요소들은 위에서 부터 배치된다.

3. 플렉서블 박스의 기본개념

속성명 : 속성값	설명
<code>display: flex;</code>	자식요소를 블록 수준의 플렉스 아이টে으로 작동하게 한다. (부모 요소는 block 요소로 변경된다)
<code>display: inline-flex;</code>	부모 요소를 인라인- flex 수준(inline-block) 글씨 +flex (부모 요소가 inline-block 요소로 변경된다)

기본적으로 컨텐츠의 너비만큼 너비가 잡힌다.

4. 플렉스 아이টে의 배치 방향 설정하기(주축 변경)

속성명 : 속성값	설명
<code>flex-direction:row(기본값)</code>	<ul style="list-style-type: none"> - 기본값으로 주축의 방향을 가로로 설정 - 요소를 주축의 시작점에서 끝점 방향으로 배치
<code>flex-direction:row-reverse</code>	<ul style="list-style-type: none"> - 요소를 주축의 끝점에서 시작점 방향으로 배치
<code>flex-direction:column</code>	<ul style="list-style-type: none"> - 주축의 방향을 세로로 설정 - 요소를 주축의 시작점에서 끝점 방향으로 배치
<code>flex-direction:column-reverse</code>	<ul style="list-style-type: none"> - 주축의 방향을 세로로 설정 - 요소를 주축의 끝점에서 시작점 방향으로 배치

[CSS]

```
.wrap{
  display:flex;
  flex-direction:row;
  height:500px;
  margin:20px auto;
}
```

[화면]



참조 : <https://codepen.io/alikerock/pen/RVOWJo>

5. 플렉스 아이টে을 여러 줄로 배치하기

속성명 : 속성값	설명
<code>flex-wrap:nowrap(기본값)</code>	박스를 한줄로 배치, 기본값 / 아래줄로 넘기지 말아라.(기본값)
<code>flex-wrap:wrap</code>	박스를 여러 줄로 배치 / 넘치면 아래줄로 넘겨라.
<code>flex-wrap:wrap-reverse</code>	박스를 여러줄로 배치하되 역방향으로 배치, 주축이 가로일 때는 아래에서 위쪽으로,

주축이 세로일 때는 오른쪽에서 왼쪽으로 배치.

css에서 wrap의 자식요소의 너비를 100%로 했지만 아래로 배열되지 않는다. 기본값이 아래로 넘기지 말아라(nowrap)이기 때문이다.

[CSS]

```
.wrap{
  display:flex;
  flex-direction:row;
  height:500px;
  margin:20px auto;
}

.wrap div{
  width:100%;
}
```

[화면]



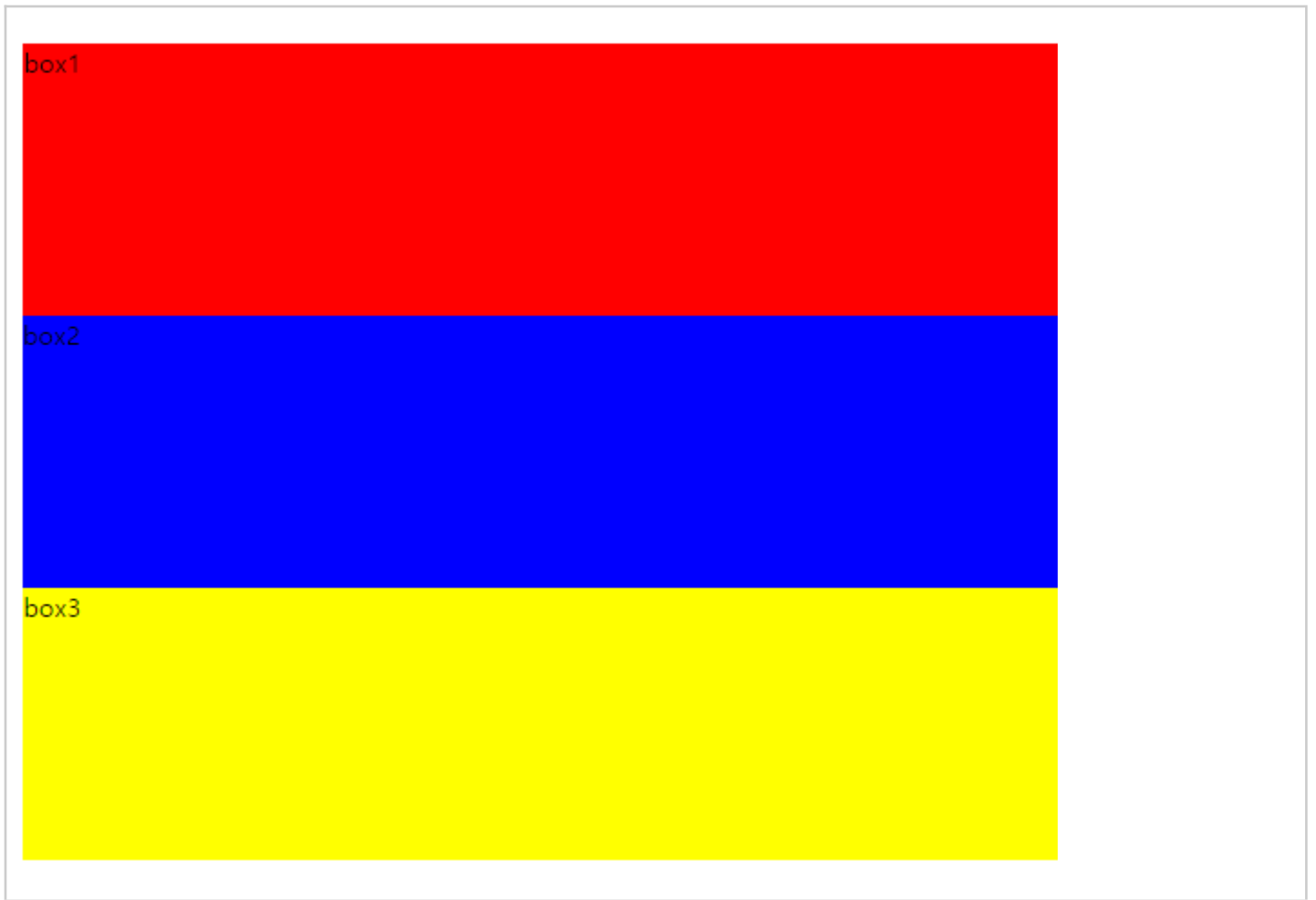
flex-wrap의 속성을 wrap으로 설정하면 자식요소의 너비가 넘치면 다음줄로 넘겨준다.

[CSS]

```
.wrap{
  display:flex;
  flex-direction:row;
  flex-wrap:wrap;
  height:500px;
  margin:20px auto;
}

.wrap div{
  width:100%;
}
```

[화면]



참조 : <https://codepen.io/alikerock/pen/GmLpwE>

6. 플렉스 아이템의 방향과 배치를 한번에 설정하기

속성명 : 속성값

flex-flow: [flex-direction] [flex-wrap]

설명

`flex-flow: row wrap`

주축의 방향을 가로로 즉 자식요소들을 가로로
배열하고 자식요소들의 너비가 커서
부모요소너비에 들어오지 않으면 아래로 내려라.

자식요소들을 가로로 배열하되 넘치면 다음줄로 넘긴다. 그런데 순서를 반대로 한다.

[CSS]





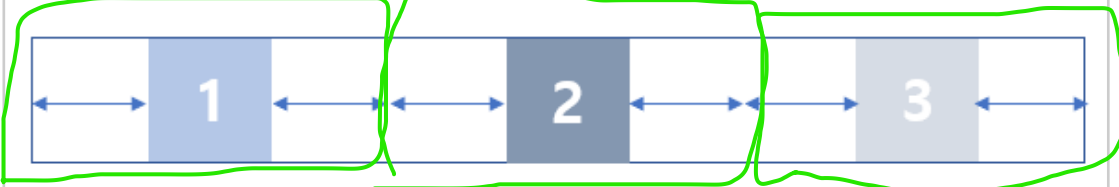

```
.wrap{  
  display:flex;  
  flex-flow:row wrap-reverse;  
  height:500px;  
  margin:20px auto;  
}
```

[화면]



참 조 : <https://codepen.io/alikerock/pen/aWxvgO>

7. 주축방향으로 다양하게 배치하기(justify-content) (하나, 하나이상의 요소를)

속성명 : 속성값	설명
justify-content: flex-start	<p>요소들을 주축으로 시작점을 정렬, 기본값</p> 
justify-content: flex-end	<p>요소들을 주축의 끝점부터 정렬</p> 
justify-content: center	<p>요소들을 중앙으로 정렬</p> 
justify-content: space-between	<p>첫번째 요소 마지막 요소는 양쪽끝으로 붙이고 나머지 요소는 동일한 간격으로 정렬</p> 
justify-content: space-around	<p>요소들 마다의 좌우 공간을 동일하게 배치</p> 
justify-content: space-evenly (IE 11 이하에서 불가)	<p>요소들 간의 간격을 동일하게 배치</p> 

[CSS]

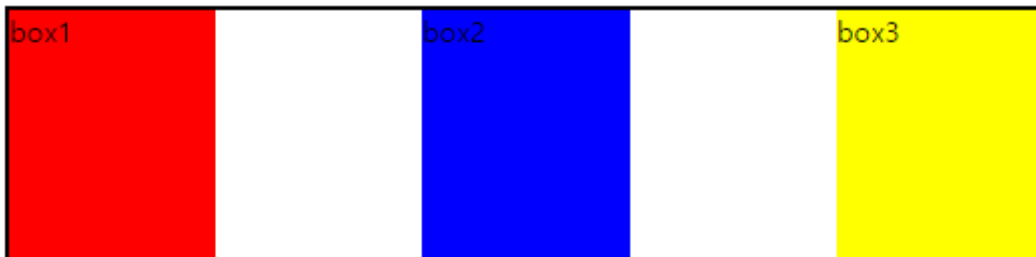
```

.wrap{
  display:flex;
  justify-content:space-between;
  width:90%;
  height:500px;
  margin:20px auto;
  border:2px solid;
}

.wrap div{
  width:20%;
}

```

[화면]



참조 : <https://codepen.io/alikerock/pen/MmRKwM>

8. 교차축 방향으로 다양하게 플렉스 아이템 배치하기

속성명 : 속성값	설명
align-items: stretch	박스를 확장해서 배치(기본값)

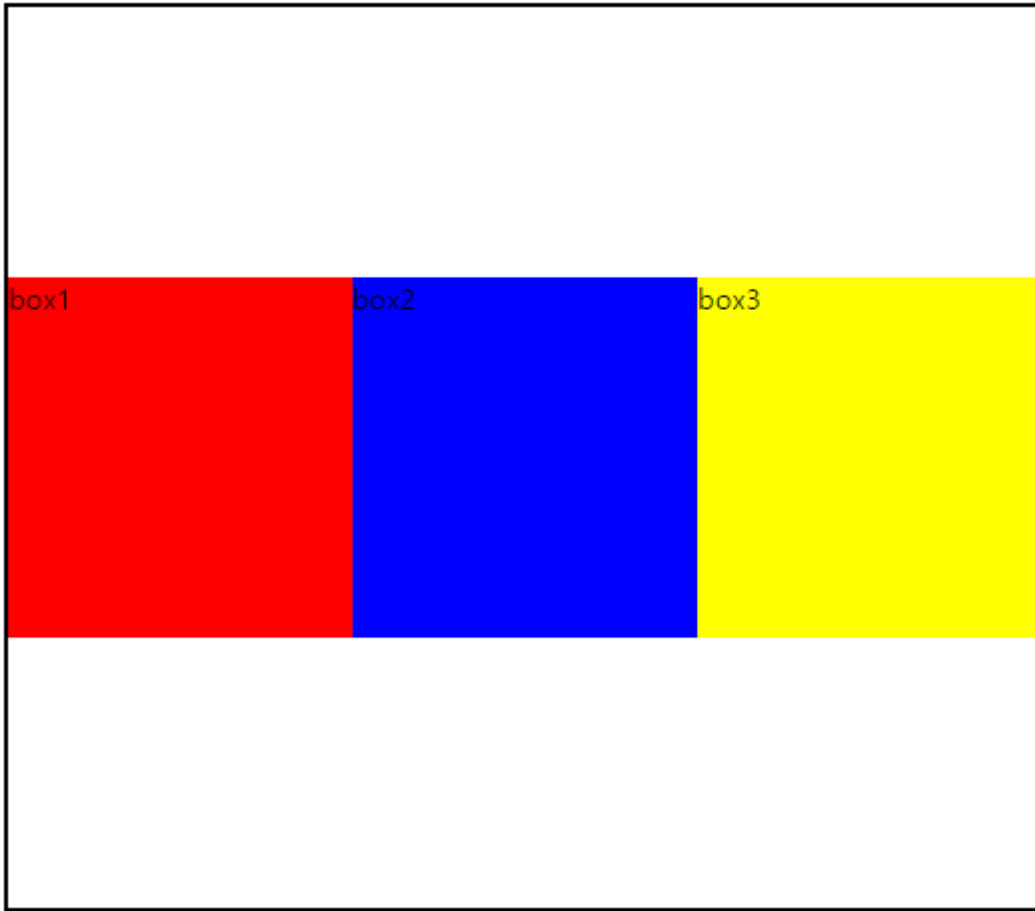
[CSS]

```
.wrap{
  display:flex;
  align-items:center;
  width:90%;
  height:500px;
  margin:20px auto;
  border:2px solid;
}
```

```
.wrap div{
  width:100%;
```

```
} height:200px;
```

[화면]



참조 : <https://codepen.io/alikerock/pen/EmJPNP>

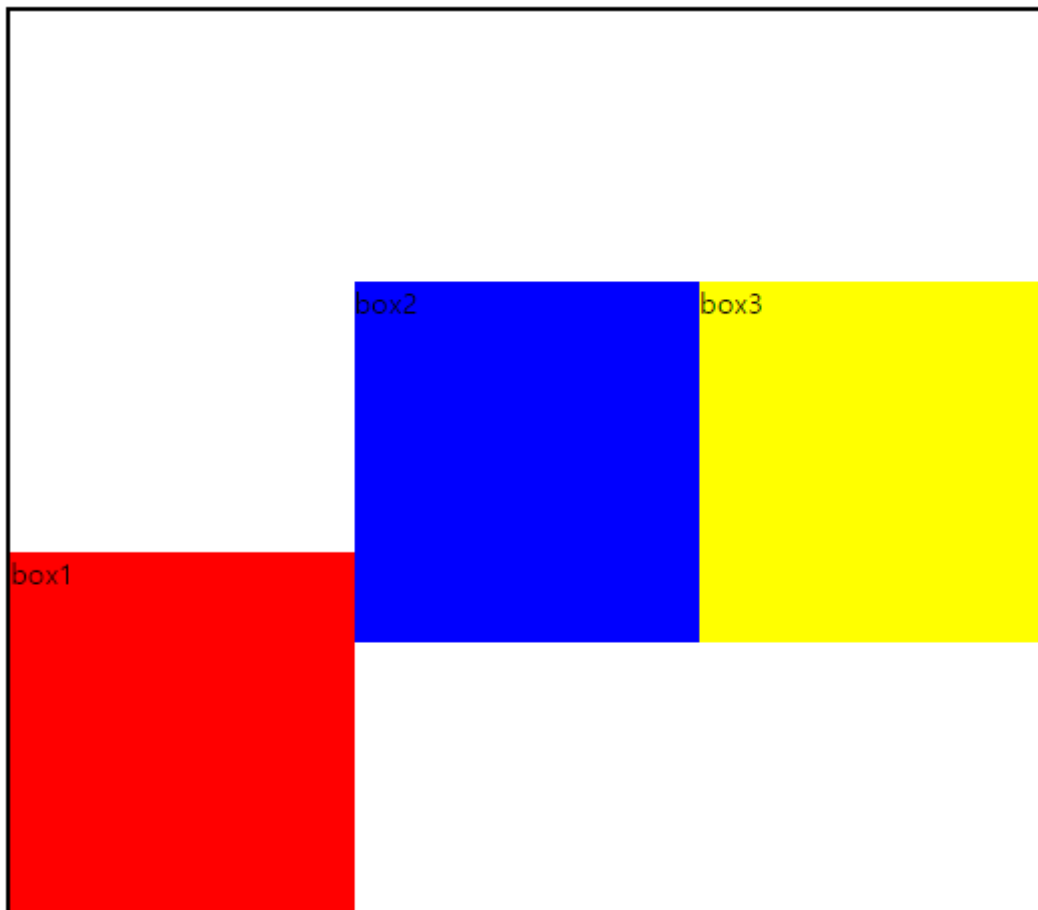
9. 교차축 방향으로 플렉스 아이টে을 개별적으로 배치하기

속성명	속성값
align-self	auto, stretch, flex-start, flex-end, center, baseline

[CSS]

```
.wrap div:first-child{  
  background:red;  
  align-self:flex-end;  
}sp
```

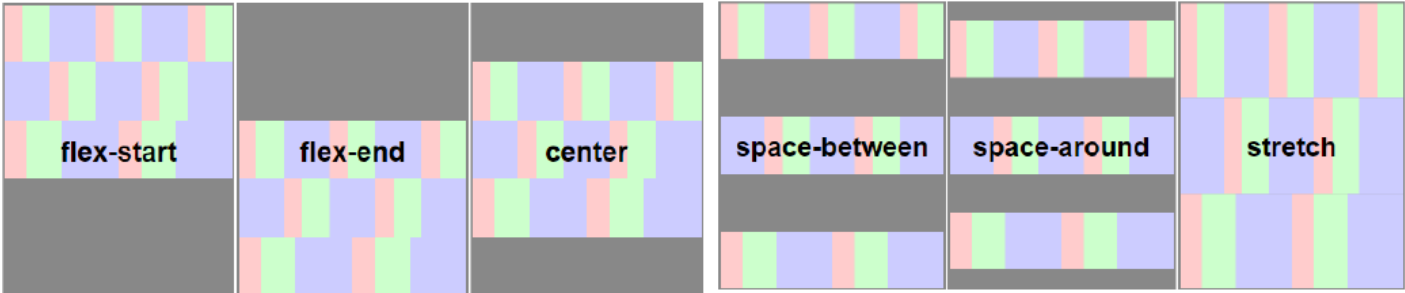
[화면]



참조 : <https://codepen.io/alikerock/pen/rmbxGO>

10. 여러줄인 플렉스 아이টে을 교차축 방향으로 다양하게 배치

속성명	속성값
align-content	Stretch , flex-start, flex-end, center, space-between, space-around, space-evenly



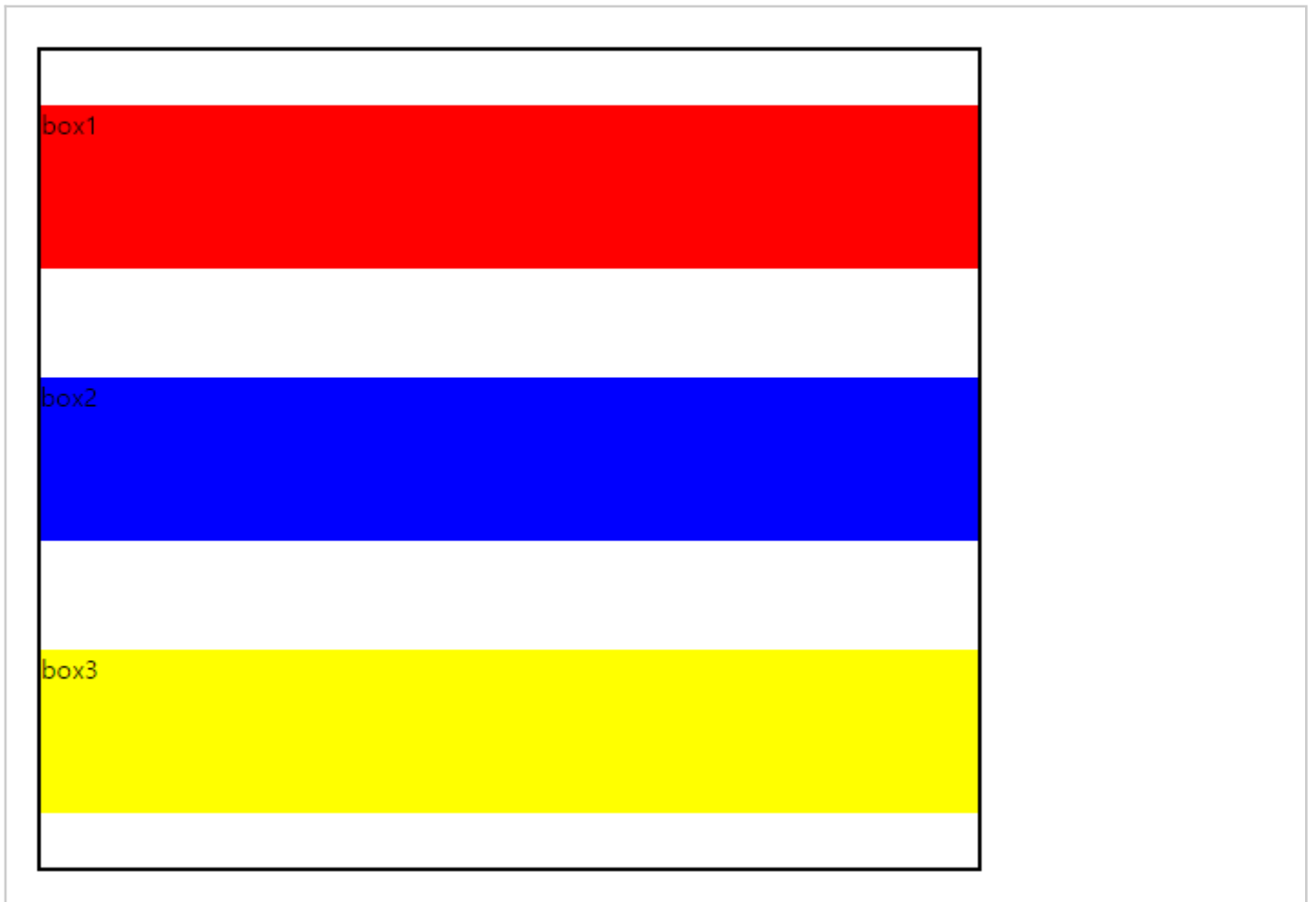
align-content:space-evenly;

[CSS]

```
.wrap{
  display:flex;
  flex-wrap:wrap;/* 아래로 내려올수 있도록 */
  align-content:space-around;
  width:90%;
  height:500px;
  margin:20px auto;
  border:2px solid;
}

.wrap div{
  width:100%;
  height:100px;
}
```

[화면]



참조 : <https://codepen.io/alikerock/pen/zwXrWK>

11. 플렉스 아이템의 배치 순서 바꾸기

속성명	속성값
order	0(기본값), 정수값

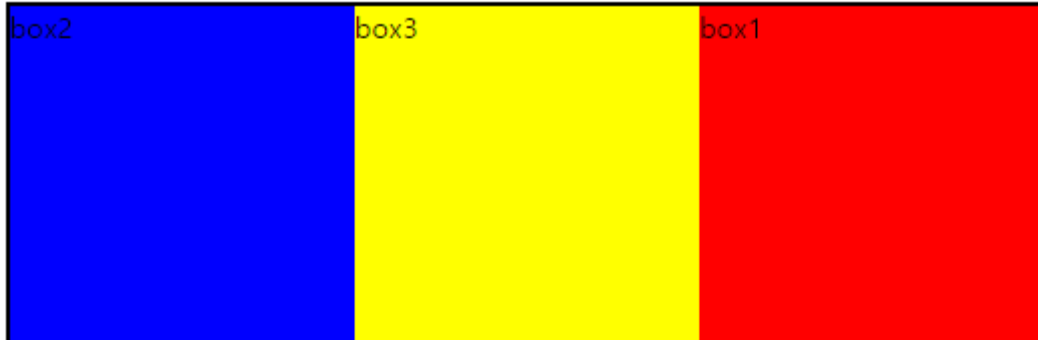
정수값으로는 양수, 음수 모두 사용할 수 있다.
같은 값을 지정하면 HTML태그 순서에 따라 배치된다.
값이 작을수록 앞으로 배치된다.

[CSS]

```
.wrap div:first-child{
  order:3;
  background:red;
}
.wrap div:nth-child(2){
  background:blue;
  order:1;
}
```

```
.wrap div:nth-child(3){
  background:yellow;
  order:2;
}
```

[화면]



참조 : <https://codepen.io/alikerock/pen/MmRKBa>

12. 플렉스 아이템의 크기 변경1

플렉스 아이템이 플렉서블 박스보다 작거나 클 경우 여백이 생기거나 넘칠수 있다.
다음 속성을 사용하여 늘이거나, 줄이거나, 또는 가변적인 박스로 설정 가능하다.

속성명	속성값
flex	[flex-grow] [flex-shrink] [flex-basis] 1 1 0 0 1 auto (기본값) 0 auto(0 1 auto와 같음) initial(0 1 auto와 같음) auto(1 1 auto와 같음) none(0 0 auto와 같음)

flex-grow : 플렉서블 박스에 여백이 있을 때 플렉스 아이템의 크기를 늘일 수 있는 속성.
속성값을 비율로 설정. width, flex-basis 속성값에 따라 늘어가는 크기가 변할 수 있다.

예) 플렉서블 박스 안에 자식 박스가 세 개 있는 상황. 각각 flex-grow 속성값을 1, 1, 2로 설정하면 남은 공간을 4등분하여 각각 1/4, 1/4, 2/4 크기 만큼을 기본 크기에 더한다.

flex-shrink : 플렉서블 박스 안의 플렉스 아이템의 크기가 넘칠 경우 크기를 줄일 수 있는 속성

예) 플렉서블 박스안에 자식 박스가 세개 있는 상황. 각각 flex-shrink 속성을 1,1,2로 설정하면 넘치는 공간을 4등분하여, 각각 1/4, 1/4, 2/4 크기 만큼을 넘치는 양에서 계산후 축소한다.

[참조 링크](#)

flex-basis : 플렉스 아이템의 초기 크기를 지정

예)

속성값으로 0을 설정할 경우: 플렉스 아이템에 **flex-grow**, **flex-shrink** 속성값으로 설정한 비유일 그대로 적용된다.

속성값으로 **auto**를 설정할 경우: 플렉스 아이템이 기본적으로 가지고 있는 크기를 기준으로 **flex-grow**, **flex-shrink** 속성값에서 설정한 비율이 적용된다.

flex-grow : 0.5; 모자른 부분을 채우는 값을 50%

flex-grow : 1; 모자른 부분을 채우는 값을 100%;

flex-basis:0(1)(auto); 기본값으로 콘텐츠(내용)의 너비만큼만 인식.

flex-basis:1; 콘텐츠(내용)의 너비만큼만 인식.

flex-basis:200px; 너비가 200px

flex-basis:50%; 부모가 만들어준 공간에서 50%;

[flex-grow] [flex-shrink] [flex-basis]

1 1 0

모자르면 늘리는 속성 100%, 넘치면 줄이는 속성 100%, 콘텐츠의 너비 맞춰서.

[CSS]

```
.wrap div:first-child{
    flex:1 1 0;
    background:red;
}
.wrap div:nth-child(2){
    background:blue;
    flex:1 1 0;
}
.wrap div:nth-child(3){
    background:yellow;
    flex:2 2 0;
}
```

첫번째 박스와 두번째 박스에는 크기를 늘이는 속성인 **flex-grow** 속성의 값을 1로,

> 모자란 크기의 4/1을 더해줘서 **grow**(크게)해준다.

flex-shrink 값을 1로 설정.

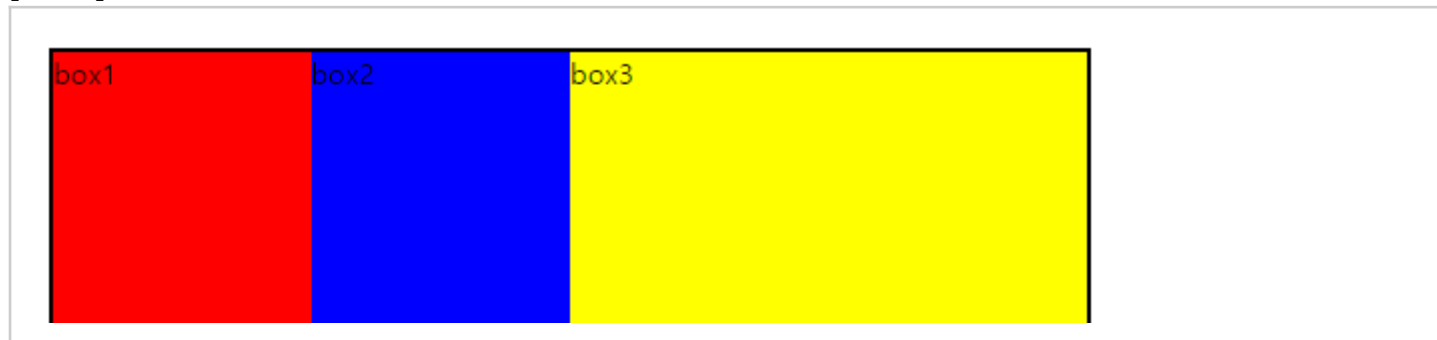
> 넘치는 크기의 4/1을 빼주어 **shrink**(줄어들게)해준다.

flex-basis 값을 0으로 설정

> **grow,shrink**에서 설정한 비율을 그래도 적용함)

마지막 박스에는 **flex-grow** 속성의 값을 2로,
> 모자란 크기의 4/2를 더해준다.
flex-shrink 값을 2로 설정.
> 넘치는 크기의 4/2를 빼준다.
flex-basis 값을 0으로 설정
> grow,shrink에서 설정한 비율을 그래도 적용함)

[화면]



참조 : <https://codepen.io/alikerock/pen/YVMwoj>

13. 플렉스 아이템의 크기 변경2

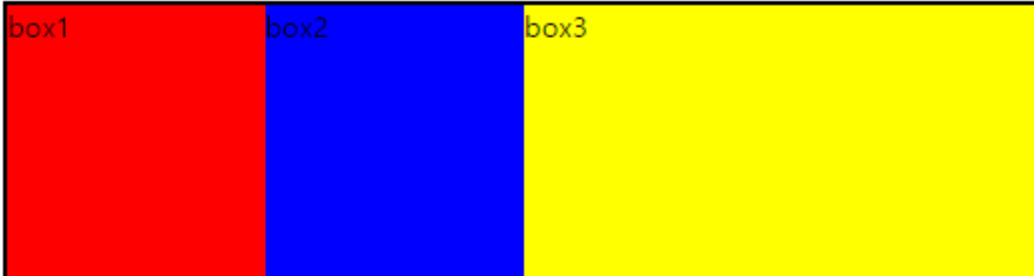
기본 크기만 지정해도 너비를 변경할 수 있다.

1/4, 1/2, 2/4 을 사용하도록 한다.

[CSS]

```
.wrap div:first-child{
  flex:1;
  background:red;
}
.wrap div:nth-child(2){
  background:blue;
  flex:1;
}
.wrap div:nth-child(3){
  background:yellow;
  flex:2;
}
```

[화면]



참조 : <https://codepen.io/alikerock/pen/ybrOyJ>

flex-basis

참조 : <https://codepen.io/alikerock/pen/rNxJqzL>

https://www.w3schools.com/cssref/tryit.asp?filename=trycss3_flex-basis

14. 플렉스 아이템의 크기 변경 애니메이션

참조 : <https://codepen.io/alikerock/pen/mddBBKK>

종합 참조

참조 : <https://codepen.io/alikerock/pen/GEKXEJ>

Flex-shrink

Flex-shrink속성은 넘치는 플렉스 아이템의 크기를 줄여줍니다. 예제부분에서 클래스명 **shrink**의 자식요소는 너비가 **200px**로 설정되어 있어 3개 요소의 합은 부모 요소의 너비보다 큰 상태이고 현재는 **flex-shrink**속성이 설정되어 있지 않지만 기본값인 1이 적용된 상태로 각각 조금씩 축소되어 넘치지 않고 배치되어 있는 상태입니다.

그림 A-246 각각 축소되어 있는 플렉스 아이템

현재 화면을 개발자도구의 **Elements** 부분에서 **box3**에 마우스를 올려보면 해당 요소의 크기가 **200px**이 아니라 **133.33px**로 확인되고 있습니다. 각각의 플렉스 아이템은 부모의 너비 **400px** 자식요소전체의 너비는 **600px**이기 때문에 총 **200px**이 넘쳐서 표현이 되어야 하지만 **flex-shrink**의 기본값인 1이 설정되어 있어 **flex-shrink**의 값이 1:1:1의 비율로 설정된 값에서

각각 1/3의 비율로 축소된 상태입니다. 즉 200px의 1/3인 33.333%의 값인 66.6666px이 축소되어 133.33px로 크기가 되어 있습니다.

클래스명 **shrink**의 자식요소중 마지막 요소에 **flex-shrink**의 값을 2로 설정합니다. 그러면 1:1:2의 비율이 되기 때문에 첫번째와 두번째는 1/4의 비율 즉 25%이며, 넘치는 200px의 값에서 25%인 즉 50px이 줄어서 150px의 너비로 보일 것입니다. 그리고 마지막 요소는 2/4의 비율 즉 50%이며 100px이 줄어서 100px로 보일 것입니다. CSS를 작성하고 브라우저 화면을 확인해봅니다.

코드 A-204 [58_flex.html]

중략...

```
.shrink div {  
  
    width: 200px;  
  
}  
  
.shrink div:last-child {  
  
    flex-shrink: 2;  
  
}  
  
</style>
```

중략...

그림 A-247 200px의 25%가 축소된 플렉스 아이템

그림 A-248 200px의 20%가 축소된 플렉스 아이템