flex 플렉스

참고

https://css-tricks.com/snippets/css/a-guide-to-flexbox/#aa-justify-content

https://flexboxfroggy.com/#ko

Flex(Flexible Box)라는 속성의 사용으로 float 보다 좀 더 쉽고 flexible하게 레이아웃을 구성할 수 있다.

flex 는 container 와 item의 개념을 사용한다.

container - 상위박스(상위부모)

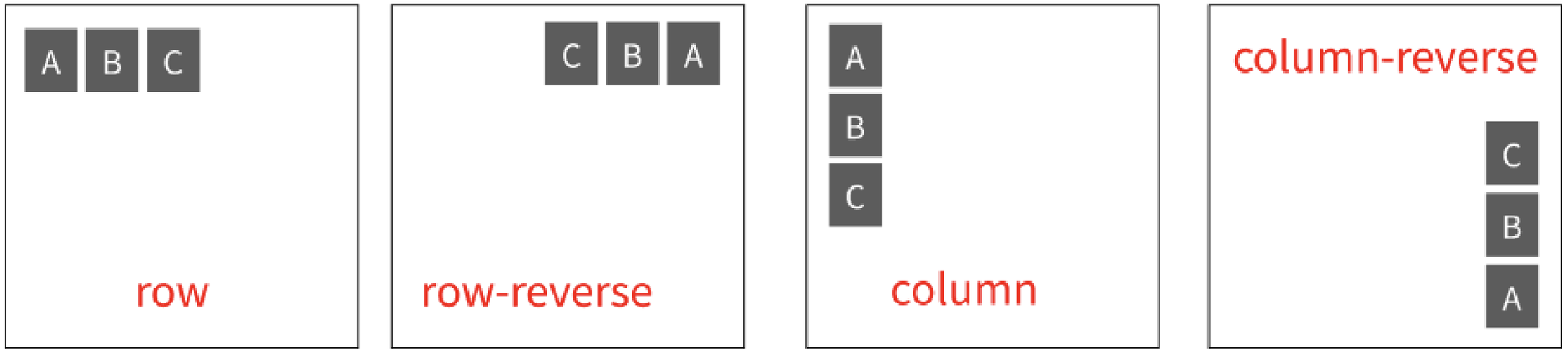
item - 하위요소(하위자식) : 수평, 수직 정렬될 대상 element

각 대상에 적용하는 속성들이 나뉘어져 있다 .

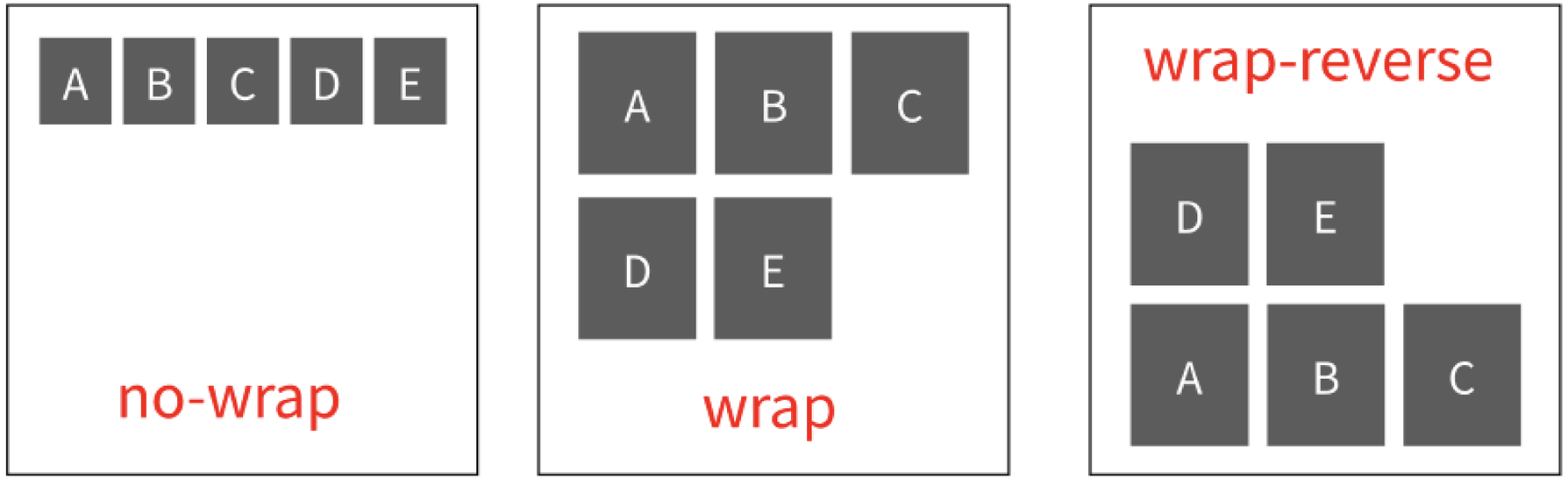
상위부모 컨테이너에 적용하는 속성들

diplay(플렉스컨테이너로설정) : flex / inline-flex

flex-direction 메인축 (방향) : row(기본값수평) /row-column / column (수직)/column-reverse



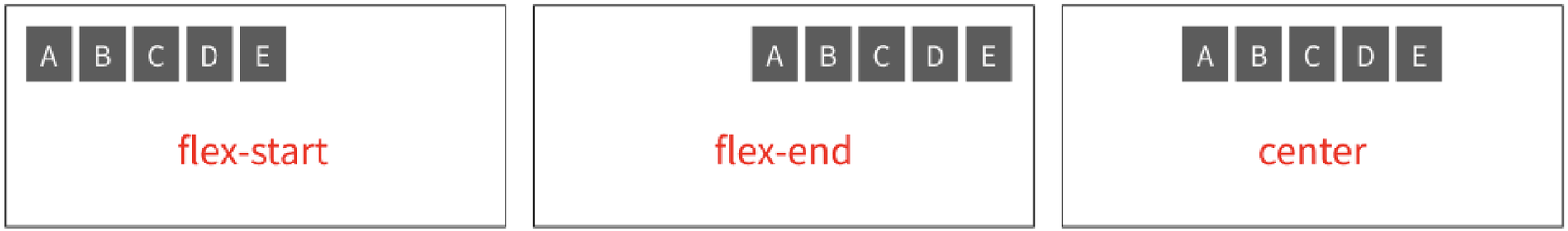
flex-wrap(줄바꿈설정) : wrap/ nowrap/ wrap-reverse



flex-flow(direction 과 wrap 의 축약속성): flex-direction값 flex-wrap값

justify-content(주축의 정렬) :
flex-start/ flex-end /center /space-between/space-around/space-evenly

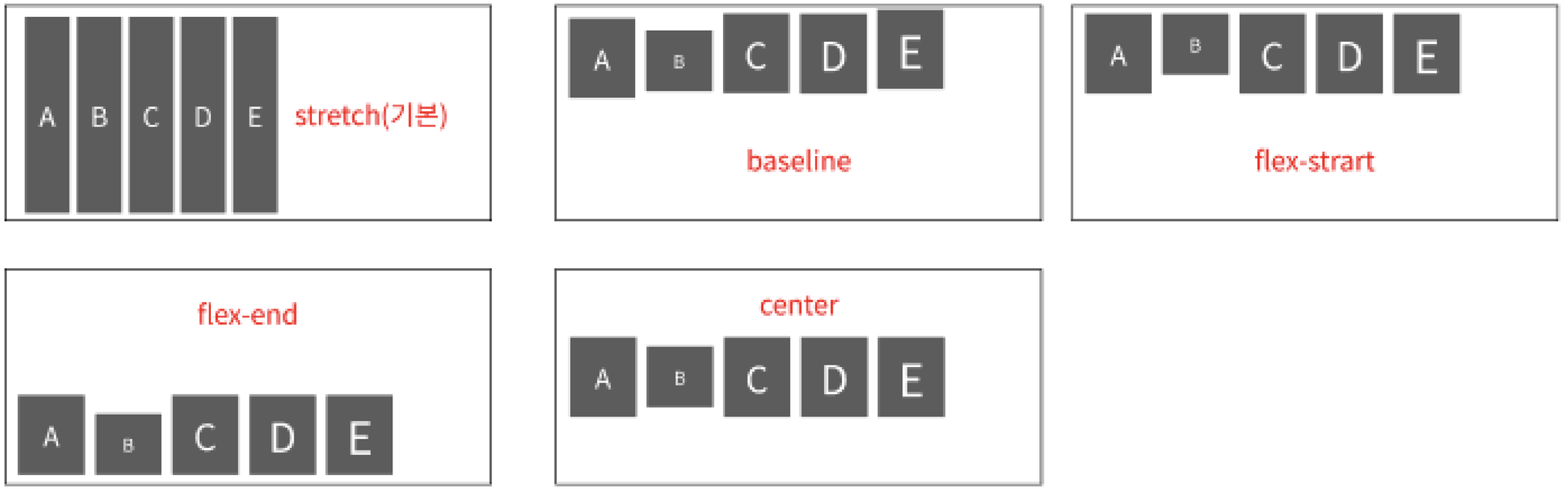
ex : 주축이 row (수평일때)



align-items(교차축의 정렬-한줄안에서) : flex-start , flex-end , center , stretch , baseline ;

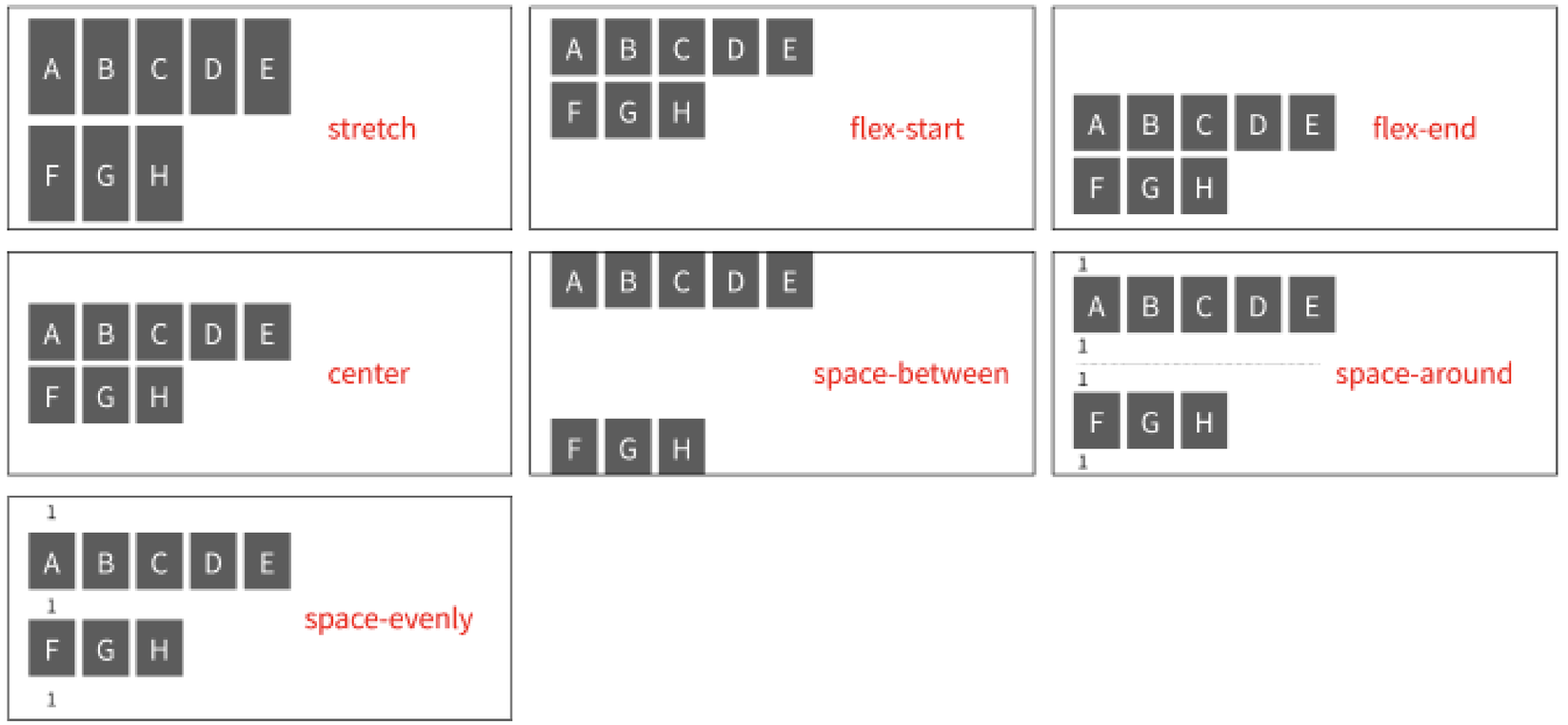
주축이 row (수평일때) 교차축은 column(수직)이 된다.
주축이 column(수직일때) 교차축을 row (수평)이 된다.

ex : 주축이 row 일때 align-items 속성의 값을 지정해서 상하정렬



align-content(교차축의 정렬 flex-wrap 속성을 적용해서 2줄 이상 일때 사용 가능) : flex-start , flex-end ,center , stretch , sapce-between ,space-around ,space-evenly

ex : 주축이 row 일때 align-content속성의 값을 지정해서 상하 줄 정렬

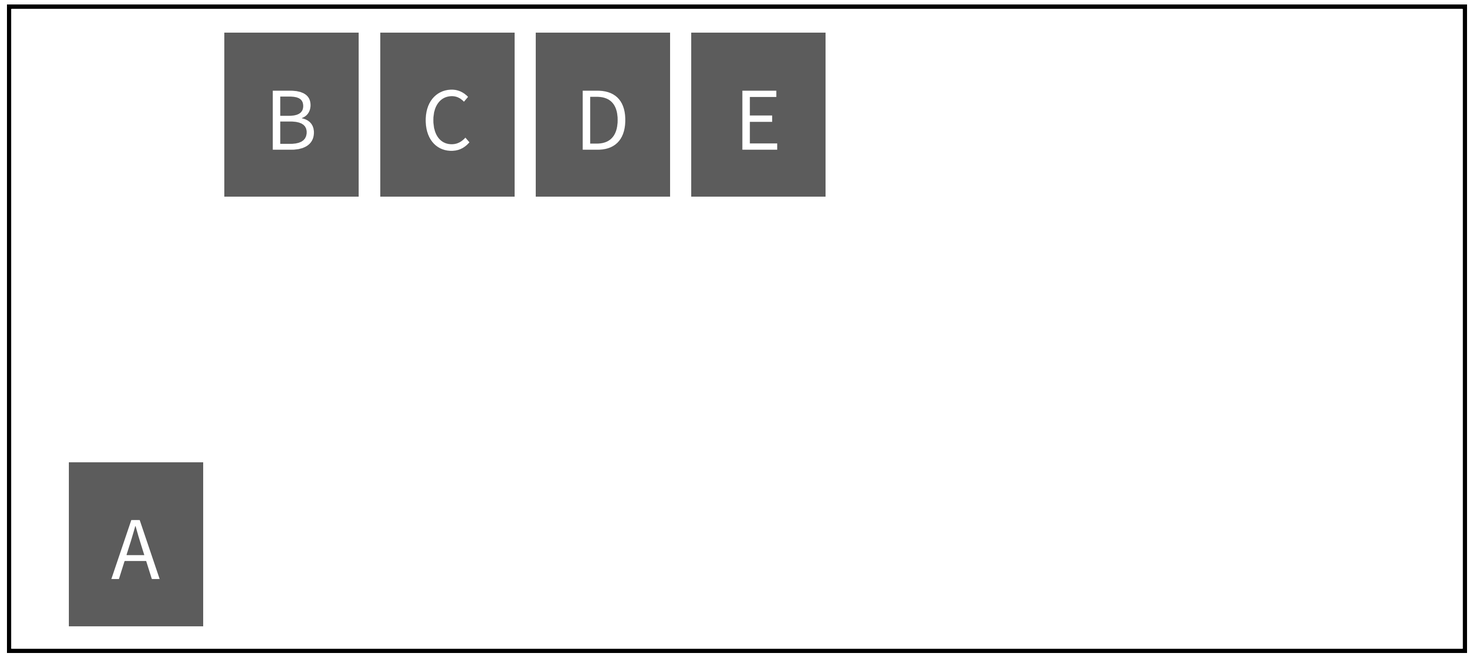


자식 item 에 설정 할 수 있는 속성들

oder 아이템의 순서를 설정 (음수 가능) : -1 / 0 / 1 / 2 .....

align-self (개별아이템 각각의 정렬) : flex-start/ flex-end /tretch, center/baseline /auto (기본값)

ex : A 에 align-self : flex-end를 설정한 경우



flex-grow 아이템의 확대비율: 0 기본

```css
.wrap{
 display: flex;
 width: 600px;height:40px;
 background-color: #666;
 margin-top: 50px;
}

.wrap>div:nth-child(1){background-color: red;}
.wrap>div:nth-child(2){background-color: green;}
.wrap>div:nth-child(3){background-color: blue;}

.wrap1>div{width: 100px;}
.wrap1>div:nth-child(1){flex-grow: 2;}
.wrap1>div:nth-child(2){flex-grow: 1;}
.wrap1>div:nth-child(3){flex-grow: 1;}
```

```html
<div class="wrap1 wrap">
    <div class="box"></div><!--100+ (300/4)*2-->
    <div class="box"></div><!--100+ (300/4)*1-->
    <div class="box"></div><!--100+ (300/4)*1-->
</div>
```

소스설명

전체 wrap의 넓이는 600px 각 아이템의 넓이는 100px입니다. 잉여공간은 300px 이다 .

flex -grow 를 적용하면 각 아이템의 넓이를 확장(grow)해서 잉여공간을 채우게 된다. 그때 남은 잉여공간을 아이템의 grow 비율 때로 나누어서 각 아이템을 확장한다.

100 (원래아이템의사이즈) + (300 잉여공간 / 4 전체grow비율) * 2 각각 아이템의 grow 값

```css
.wrap{
 display: flex;
 width: 600px;height:40px;
 background-color: #666;
 margin-top: 50px;
}

.wrap>div:nth-child(1){background-color: red;}
.wrap>div:nth-child(2){background-color: green;}
.wrap>div:nth-child(3){background-color: blue;}

.wrap2>div{width: 300px;}
.wrap2>div:nth-child(1){flex-shrink: 2;}
.wrap2>div:nth-child(2){flex-shrink: 1;}
.wrap2>div:nth-child(3){flex-shrink: 1;}
```

```html
<div class="wrap2 wrap">
    <div class="box"></div><!--300-((300/4)*2) => 150-->
    <div class="box"></div><!--300-((300/4)*1) => 225-->
    <div class="box"></div><!--300-((300/4)*1) => 255-->
</div>
```

소스설명

전체 wrap의 넓이는 600px 각 아이템의 넓이는 300px입니다. 부족한 공간은 300px 이다 .

flex -shrink 를 적용하면 각 아이템의 넓이를 축소(shrink)해서 넘치는 아이템을 축소해서 wrap 안에 넣게 된다. 그때 부족한 공간(overflow되는)을 아이템의 shirink 비율 때로 나누어서 각 아이템을 축소한다..

300 (원래아이템의사이즈) - (300 부족한공간 넘치는공간 / 4 전체 shirink 비율) * 2 각각 아이템의 shrink 값

flex-basis 아이템의 기본넓이: auto 기본

flex-basis는 flex 아이템의 기본 크기를 설정한다.

flex-basis: auto는 아이템 자신의 크기를 사용한다.

flex-basis: auto 와 width 속성 height 속성을 다 같이 사용한다면 width 와 height 속성의 값이 우선적으로 적용된다.

flex-basis: auto 외의 다른 값을 가지고 있다면 width 속성 height 속성 보다 flex-basis 속성의 값이 더 우선적으로 적용된다.

flex-basis: 0은 아이템의 기본 크기를 0이라고 설정하는 것으로,flex-grow, flex-shrink 설정 비율대로 아이템 넓이를 배분한다.

flex : grow shrink basis : grow shirnk basis 를 한번에 적는 축약속성

flex : 1 1 50px

flex : 0 1 100px

flex : 1 => 1 1 0

flex: 1 50px => 1 1 50px

flex : 1 2; => 1 2 0 ;

flex : 50px => 1 1 50px

flex : none => 0 0 auto

flex : initial => 0 1 auto

flex : auto => 1 1 auto