





변형 transform

🕒 작성일시	@2023년 12월 12일 오전 10:05
📁 유형	CSS
☑ 진행	<input type="checkbox"/>

<https://hyejin.tistory.com/138>

transform-origin - CSS: Cascading Style Sheets | MDN
The transform-origin CSS property sets the origin for an element's transformations.

 <https://developer.mozilla.org/en-US/docs/Web/CSS/transform-origin>

 mdn web docs

변형 : 이동, 회전, 크기, 기울기

`transform: none` (기본값)

`transform : translate()` `transform : rotate()` `transform : scale()` `transform : skew()`

- 변형 가능한 요소만 `transform` 을 적용할 수 있습니다. 비대체 인라인 박스, 표 열 박스, 표 열 그룹 박스를 제외한 요소에만 적용할 수 있습니다.



transform(3D)에서 함께 사용할 속성 **perspective**(원근법)과 **backface-visibility**(뒷면가시성)



transform-origin을 함께 사용하여 이동, 회전 등을 변형 시켜 줄 때 요소들의 기준점을 설정할 수 있다.

```
div {  
  transform-origin: center;  
  
  /* x-offset | y-offset */  
  transform-origin: top left;  
  transform-origin: 50px left;  
  
  /* x-offset | y-offset | z-offset */  
  transform-origin: bottom right 60px;  
}
```



이동 : translate()

`translate()` 함수는 transform 속성에 사용할 수 있는 변형 함수 중 하나로 **이동 효과**를 가지고 있다.

`translate()` `translateX()` `translateY()` `translateZ()` `translate3d()`

- translate(x, y)** : 현재의 위치에서 x축과 y축 거리만큼 이동 시킴.

- **translateX(x)** : 현재의 위치에서 x축 거리만큼 이동 시킴.
- **translateY(y)** : 현재의 위치에서 y축 거리만큼 이동 시킴.
- **translateZ(z)** : 현재의 위치에서 z축 거리만큼 이동 시킴.
- **translate3d(x, y, z)** : 하드웨어 가속을 사용하기 때문에 즉, CPU를 사용하기 때문에 CSS 퍼포먼스가 일반적인 `translate()` 보다 더 좋다. 따라서 더 좋은 퍼포먼스를 원한다면 `translate3d` 를 사용하는 것이 좋다.

```
div {
  transform : translate3d(0, 100px, 0);
}
p {
  transform : translateY(100px);
}
/*div와 p는 동일한 결과값을 보여준다. 따라서 의도에 맞게 사용하면 된다.*/

h1 {
  transform : translateX(100%);
  /*h1이 가지는 너비값만큼 x축으로 이동시킴
    h1이 width값을 500px 가지고 있다면 오른쪽으로 500px 이동됨*/
}
```

🔔 회전 : rotate()

`rotate()` 함수는 transform 속성에 사용할 수 있는 변형 함수 중 하나로 회전 효과를 가지고 있다.

`rotate()` `rotateX()` `rotateY()` `rotateZ()` `rotate3d()`

- **rotate(각도)** : 시계방향으로 회전(40deg, 시계방향으로 40도 회전 / deg⇒ degree[도 : 디그리])
- **rotateX(각도)** : x축으로 회전
- **rotateY(각도)** : y축으로 회전
- **rotateZ(각도)** : z축으로 회전
- **rotate3d(x, y, z, 각도)** : x,y,z축으로 회전

```
div {
  transform: rotate3d(2.4, 3, -1.5, 55deg)
}
```

🔔 확대/축소 : scale()

`scale()` 함수는 transform 속성에 사용할 수 있는 변형 함수 중 하나로 확대하거나 축소를 할 수 있다.

함수 값으로 1보다 크면 확대, 1보다 작으면 축소 된다.

`scaleX(1.5)` : x축으로 1.5배 확대

`scaleX(0.5)` : x축으로 0.5배 축소

`scale()` `scaleX()` `scaleY()` `scaleZ()` `scale3d()`

- **scale(x, y)** : x, y축으로 크기 조절
- **scaleX(x)** : x축으로 크기 조절
- **scaleY(y)** : y축으로 크기 조절
- **scaleZ(z)** : z축으로 크기 조절
- **scale3d(x, y, z)** : x,y,z축으로 크기 조절

```
div {
  /*y축으로 1.2배 만큼 확대*/
  transform: scaleY(1.2);
}
```

기울기: skew()

`skew()` 함수는 지정한 각도만큼 요소를 비틀어 왜곡한다.

- skew(x, y) : x, y축 기준으로 기울기 조절
- skewX(x) : x축 기준으로 기울기 조절
- skewY(y) : y축 기준으로 기울기 조절

```
div {
  /*x축 기준으로 45도 기울기*/
  transform: skewX(45deg);
}
```