

변형/애니메이션

변형

▶ 애니메이션 개요

✓ 변형

단순히 요소가 페이지에 출력만 되는 것이 아니라 사용자의 동작에 따라 크기가 바뀌고 요소가 이동, 회전하는 것

✓ 애니메이션

변형을 부드럽게 연결하는 것

▶ 애니메이션 개요

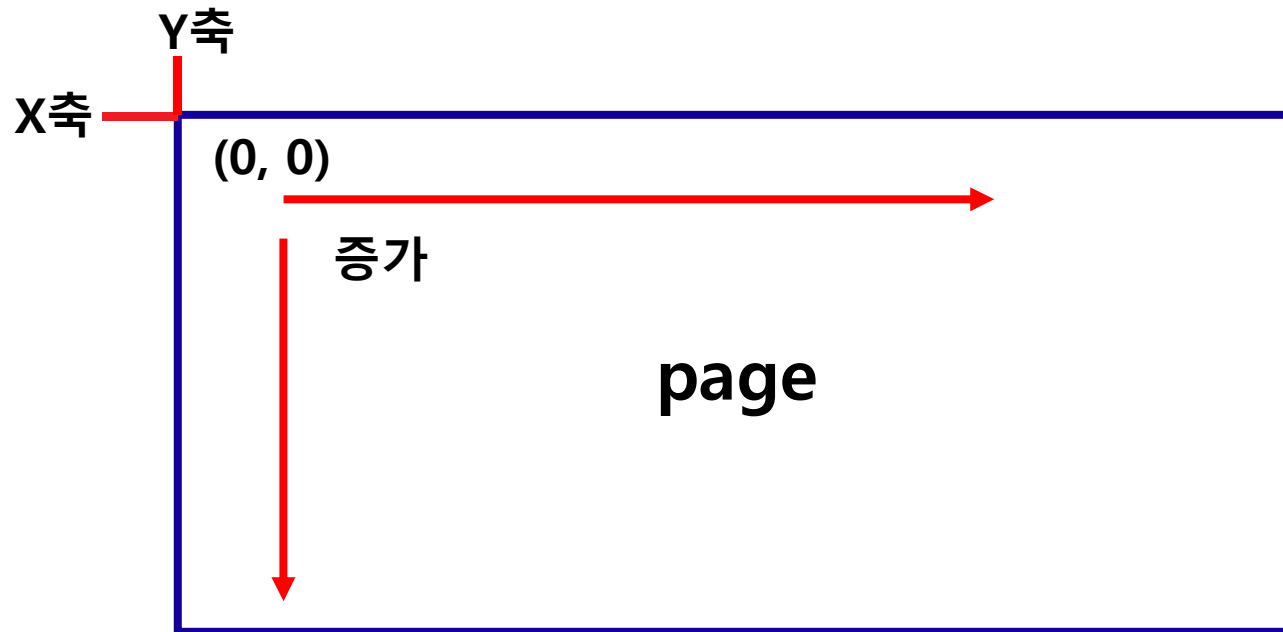
✓ 변형

단순히 요소가 페이지에 출력만 되는 것이 아니라 사용자의 동작에 따라 크기가 바뀌고 요소가 이동, 회전하는 것

▶ 애니메이션 개요

✓ 2차원 변형

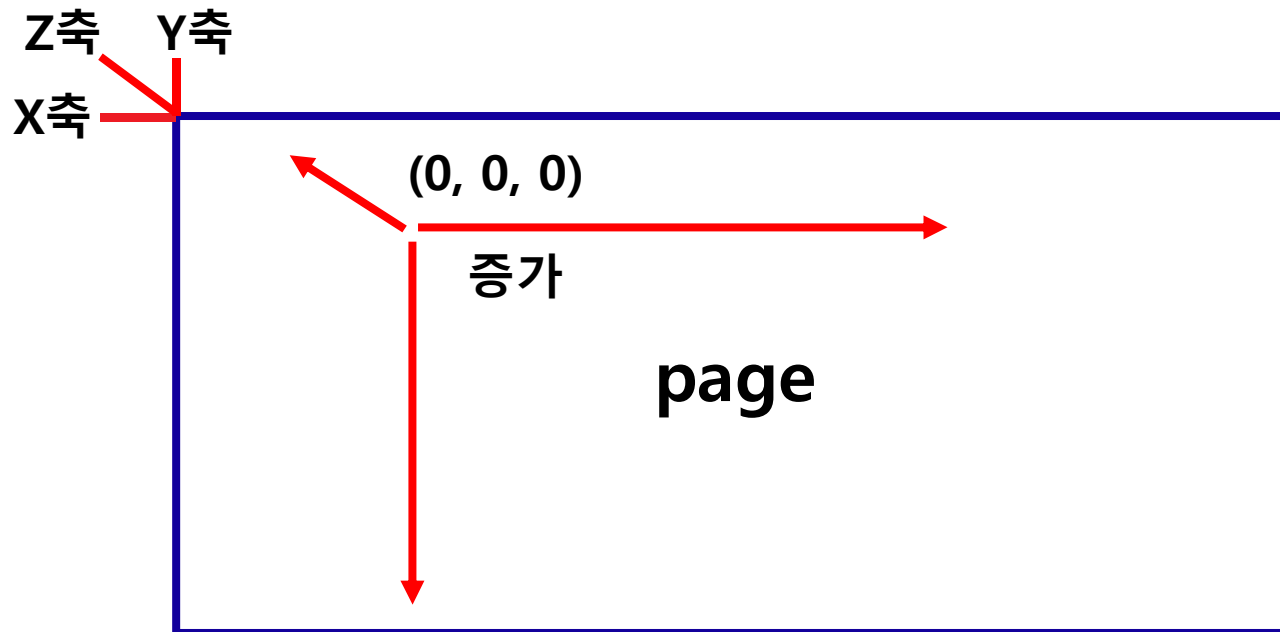
요소가 변형(이동, 회전)할 때 수직, 수평으로 이동하는 것으로
x축, y축으로 나누어 페이지 내에서 이동하는 것



▶ 애니메이션 개요

✓ 3차원 변형

요소가 변형(이동, 회전)할 때 수직, 수평으로 이동뿐만 아니라 화면 상에서 앞으로 이동하거나 뒤로 이동하는 것이 추가된 것으로 X, Y좌표 그리고 Z좌표까지 추가됨



* Z축은 사용자에게 가까울수록 증가

▶ 애니메이션 개요

✓ transform 속성

페이지에서 요소들을 변형시키려면 transform속성과 변형 함수 이용
변형 함수는 2차원 함수와 3차원 함수 구분

선택자{-접두사-transform: 변형함수; transform: 변형함수;}

* 이전 브라우저와 호환을 위해 접두사를 이용한 함수호출

변형 함수

▶ 2차원 변형 함수

구분	내용
translate(x, y)	지정한 크기만큼 x축, y축으로 이동
translateX(tx)	지정한 크기만큼 x축(가로)으로 이동
translateY(ty)	지정한 크기만큼 y축(세로)으로 이동
scale(sx, sy)	지정한 크기만큼 x, y축으로 확대/축소
scaleX(sx)	지정한 크기만큼 x축(가로) 방향으로 sx만큼 확대/축소
scaleY(sy)	지정한 크기만큼 y축(세로) 방향으로 sy만큼 확대/축소
rotate(각도)	지정한 각도만큼 회전
skew(ax, ay)	지정한 각도만큼 x축과 y축으로 요소 변형
skewX(ax)	지정한 각도만큼 x축으로 요소 요소 변형
skewY(ay)	지정한 각도만큼 y축으로 요소 요소 변형

▶ 3차원 변형 함수

구분	내용
matrix3d(n [,n])	4*4행렬로 이동과 확대/축소, 회전 등 변환
translate3d(tx,ty,tz)	지정한 크기만큼
translateZ(tz)	지정한 크기만큼 z축(세로)으로 이동
scale3d(sx, sy, sz)	지정 크기만큼 x, y, z축 방향으로 x, y, z만큼 확대/축소
scaleZ(sz)	지정한 크기만큼 z축 방향으로 z만큼 확대/축소
rotate3d(rx, ry, rz, 각도)	지정한 크기만큼 회전
rotateX(각도)	지정한 각도만큼 x축으로 회전
rotateY(각도)	지정한 각도만큼 y축으로 회전
rotateZ(각도)	지정한 각도만큼 z축으로 회전
perspective(깊이)	입체적으로 보일 수 있는 깊이 값을 지정

▶ translate ? (좌표 값)

요소를 페이지의 일정 좌표로 이동하는 함수

선택자{transform: 함수명(이름);}

* 3D효과를 내기 위해서는 perspective함수를 사용해야 함

구분		인자 값(좌표)
2차원	translate	x, y
	translateX	x
	translateY	y
3차원	translate3d	x, y, z
	translateZ	z

▶ scale ? (좌표 값)

요소를 일정 페이지만큼 확대/축소 시키는 함수 * 양수 : 확대 / 음수 : 축소

선택자{transform: 함수명(이름);}

구분		인자 값(좌표)
2차원	scale	x, y
	scaleX	x
	scaleY	y
3차원	scale3d	x, y, z
	scaleZ	z

▶ rotate ? (좌표 값)

지정 각도만큼 요소를 시계방향(양수)이나 반대방향(음수)으로 회전하는 함수

선택자{transform: 함수명(이름);}

구분		인자 값(좌표)
2차원	rotate	각도(deg)
3차원	rotateX	각도(deg)
	rotateY	각도(deg)
	rotate3d	x, y, z, 각도(deg)
	rotateZ	각도(deg)

▶ skew ? (좌표 값)

지정한 각도만큼 요소를 비틀어 변형하는 함수로 2차원만 가능

선택자{transform: 함수명(이름);}

구분		인자 값(좌표)
2차원	skew	각도(deg), 각도(deg)
	skewX	각도(deg)
	skewY	각도(deg)

기타 변형 속성

▶ transform-origin

기존 변형 기준 x, y, z축이 아닌 특정 지점을 기준으로 변형할 수 있게 하는 속성

선택자{transform-origin: x y z;}

구분	속성 값
x축	x좌표 값 / left center right
y축	y좌표 값 / top center bottom
z축	z좌표 값

▶ perspective

화면에서 원근감을 갖게 하는 속성 * 다른 변형과 적용하려면 먼저 적용되어야 함

선택자{perspective: 숫자(단위);}

▶ transform-style

여러 가지 변형을 동시에 하는 경우

부모 요소에서 적용한 3D변형을 하위 요소에 적용하는 속성

선택자{transform-style: 속성 값;}

구분	속성 값
flat	하위 요소를 평면으로 처리
preserve-3d	하위 요소들에 3D효과 적용

▶ backface-visibility

회전하여 뒷면이 보일 경우

뒷면을 보이게 할 것인지 안 보이게 할 것인지 설정하는 속성

선택자{backface-visibility: 속성 값;}

구분	속성 값
hidden	뒷면이 보이지 않게 설정
visible	뒷면이 보이게 설정

트랜지션

▶ 트랜지션

웹 요소의 스타일이 바뀌는 것 의미

CSS로 애니메이션 같은 효과를 낼 수 있음

구분	속성 값
transition-property	적용할 속성 선택
transition-duration	진행 시간 선택
transition-timing-function	순차적인 속도 선택
transition-delay	트랜지션 시간 지연 선택
transition	속성 값을 한 번에 설정

▶ transition-property

트랜지션을 적용할 속성을 선택하며 여러 개를 선택할 경우 ,로 구분

선택자{transition-property: 속성 값;}

구분	속성 값
all	default, 모든 속성 적용
none	아무 속성도 바꾸지 않음
속성 명	해당하는 속성만 적용 (ex. width, background, translate 등)

▶ transition-duration

트랜지션이 진행되는 시간을 지정하는 속성으로

시간 값은 밀리 초(ms)와 초(s)로 설정

여러 개를 설정할 경우 ,로 구분하고 property값과 1:1로 대응 됨

선택자{transition-duration: 시간;}

▶ transition-timing-function

트랜지션이 진행되는 구간 별로 시간을 지정하는 속성

선택자{transition-timing-function: 속성 값;}

구분	속성 값
linear	
ease	진행 시간 선택
ease-in / ease-out	순차적인 속도 선택
ease-in-out	트랜지션 시간 지연 선택
cubic-bezier(n, n, n, n)	구간 별 속도 지정 (n 값 : 0 ~ 1)

▶ transition-delay

트랜지션이 두 개 이상 있는 경우

하나의 트랜지션이 끝나고 다음 트랜지션이 시작하는 시간

선택자{transition-delay: 시간(s, ms);}

▶ transition

트랜지션 속성 값을 한 번에 지정할 수 있는 속성

* 속성 값을 생략하면 default 값으로 설정 됨

property: all

duration: 0

timing-function: ease

delay: 0

선택자{transition: property duration timing-function delay;}

애니메이션

▶ 애니메이션

시작 스타일과 끝 스타일을 부드럽게 이어주는 기능을 하는 것은
트랜지션과 같지만 애니메이션은 중간 원하는 위치에서
keyframes라는 것을 이용하여 중간 스타일을 넣을 수 있음

▶ 애니메이션

구분	속성 값
@keyframes	애니메이션이 바뀌는 지점 선택
animation-delay	지연 시간 선택
animation-direction	종료 후 처음부터 시작 또는 반대로 시작
animation-duration	실행 시간 선택
animation-fill-mode	종료되었거나 실행되지 않을 때 스타일 선택
animation-iteration-count	반복 회수 선택
animation-name	중간 상태 이름 지정
animation-play-state	멈추었다가 다시 시작
animation-timing-function	순차적인 진행시간 선택
animation	속성을 한 번에 모두 지정

▶ @keyframes

애니메이션의 시작, 끝 지점의 스타일을 정하고 keyframes의 이름 지정
시작 지점과 끝 지점 두 개만 설정할 경우 from to로 설정
중간에 작업이 있으면 %로 구분하여 설정

```
@keyframes 이름{  
    선택자{스타일;}  
}
```

▶ @keyframes

✓ 시작과 끝 스타일만

```
@keyframes 이름{  
    from{ 선택자{스타일;}}  
    to{ 선택자{스타일;}}  
}
```

✓ 중간에 스타일이 있는 경우

```
@keyframes 이름{  
    0%{ 선택자{스타일;}}  
    50%{ 선택자{스타일;}}  
    100%{ 선택자{스타일;}}  
}
```

▶ animation-name

요소에 적용할 애니메이션을 지정하는 속성

keyframes에서 정한 이름을 작성, 적용할 요소 스타일에 작성함

선택자{animation-name: keyframes이름;}

▶ animation-duration

요소에 적용되는 애니메이션의 실행 시간을 지정하는 속성

선택자{animation-duration: 시간(s);}

▶ animation-direction

요소에 적용되는 애니메이션을 반복할 때 처음부터 시작할지 끝에서 역순으로 시작할지 선택하는 속성, animation-iteration-count와 같이 사용

선택자{animation-direction: 속성 값;}

구분	속성 값
normal	처음부터 다시 실행
alternate	끝부터 다시 실행

▶ animation-iterator-count

애니메이션을 얼마나 반복할지 설정

선택자{animation-iterator-count: 속성 값;}

구분	속성 값
숫자	횟수
infinite	무한 반복

▶ animation-timing-function

전체 속도를 지정할 수 있는 속성

선택자{animation-timing-function: 속성 값;}

구분	속성 값
linear	
ease	진행시간 선택
ease-in / ease-out	순차적인 속도 선택
ease-in-out	트랜지션 시간지연 선택
cubic-bezier(n,n,n,n)	구간 별 속도를 지정 (n 값 : 0 ~ 1)

▶ animation

애니메이션 전체 속성을 지정할 수 있는 속성

* , 를 이용하여 두 개 이상의 애니메이션 적용 가능

선택자{animation: name duration timing-function delay iterator-count
direction}