

## CSS Transforms

CSS transforms을 사용하면 요소를 이동, 회전, 비율 조정 및 기울일 수 있습니다.

translate()  
rotate()  
scaleX()  
scaleY()  
scale()  
skewX()  
skewY()  
skew()  
matrix()

translate()는 현재 위치에서 요소를 이동합니다(X축 및 Y축에 지정된 매개변수에 따라 요소를 이동).

translateX(x)는 X축 값만 사용하여 3D 변환을 정의합니다.

translateY(y)는 Y축 값만 사용하여 3D 변환을 정의합니다.

translateZ(z)는 Z축 값만 사용하여 3D 변환을 정의합니다.

rotate()는 주어진 각도에 따라 요소를 시계 방향 또는 시계 반대 방향으로 회전합니다. 음수 값을 사용하면 요소가 시계 반대 방향으로 회전합니다.

rotateX()메서드는 지정된 각도에서 X축을 중심으로 요소를 회전합니다.

rotateY()메서드는 지정된 각도에서 Y축을 중심으로 요소를 회전합니다.

rotateZ()메서드는 주어진 각도에서 Z축을 중심으로 요소를 회전합니다.

scale()는 요소의 크기를 늘리거나 줄입니다(너비 및 높이에 대해 지정된 매개 변수에 따라 요소의 크기를 늘리거나 줄임).

scaleX()는 요소의 너비를 늘리거나 줄입니다.

scaleY()는 요소의 높이를 늘리거나 줄입니다.

skewX()는 지정된 각도만큼 X축을 따라 요소를 기울입니다.

skewY()는 주어진 각도만큼 Y축을 따라 요소를 기울입니다.

skew()는 지정된 각도만큼 X축과 Y축을 따라 요소를 기울입니다.

matrix()방법은 모든 2D 변환 방법을 하나로 결합합니다.

matrix()는 요소를 회전, 크기 조정, 이동(변환) 및 기울일 수 있는 수학 함수를 포함하는 6개의 매개 변수를 사용합니다.

matrix(scaleX(), skewY(), skewX(), scaleY(), translateX(), translateY());

예)

```
div {  
  transform: matrix(1, -0.3, 0, 1, 0, 0);  
}
```

## CSS Transform Properties

Property	Description
transform	요소에 2D 또는 3D 변환을 적용합니다.
transform-origin	변환된 요소의 위치를 변경할 수 있습니다.
transform-style	중첩된 요소가 3D 공간에서 렌더링되는 방식을 지정합니다.
perspective	3D 요소가 표시되는 방식에 대한 관점을 지정합니다.

## CSS transform-origin 속성

Property Value	Description
x-axis	뷰가 x축에 배치되는 위치를 정의합니다 가능한 값 : left center right length %
y-axis	뷰가 y축에 배치되는 위치를 정의합니다. 가능한 값: top center bottom length %

## CSS transform-style 속성

변환된 자식 요소가 3D 변환을 유지하도록 합니다.

Property Value	Description
flat	하위 요소가 3D 위치를 유지하지 않도록 지정합니다. 기본값입니다
preserve-3d	하위 요소가 3D 위치를 유지하도록 지정합니다.

## CSS Transitions

CSS Transitions을 사용하면 지정된 기간 동안 속성 값을 원활하게 변경할 수 있습니다.

transition

transition-property

transition-duration

transition-timing-function

transition-delay

Property	Description
transition	네 가지 전환 속성을 단일 속성으로 설정하기 위한 속기 속성(property duration timing-function delay)
transition-property	전환 효과에 대한 CSS 속성의 이름을 지정합니다.
transition-duration	전환 효과가 완료되는 데 걸리는 시간(초 또는 밀리초)을 지정합니다.
transition-timing-function	전환 효과의 속도 곡선을 지정합니다.
transition-delay	전환 효과에 대한 지연(초)을 지정합니다.

transition-timing-function속성은 전환 효과의 속도 곡선을 지정합니다.

transition-timing-function 속성은 다음 값을 가질 수 있습니다.

ease - 느린 시작, 빠르게 시작, 느리게 종료하는 전환 효과를 지정합니다(기본값).

linear - 처음부터 끝까지 동일한 속도로 전환 효과를 지정합니다.

ease-in - 느린 시작으로 전환 효과 지정

ease-out - 느린 끝으로 전환 효과를 지정합니다.

ease-in-out - 느린 시작과 끝으로 전환 효과 지정

cubic-bezier(n,n,n,n)- 큐빅-베지어 함수에서 자신만의 값을 정의할 수 있습니다.

## CSS Animations

CSS는 JavaScript 또는 Flash를 사용하지 않고 HTML 요소의 애니메이션을 허용합니다. 애니메이션을 사용하면 요소가 한 스타일에서 다른 스타일로 점진적으로 변경됩니다. 원하는 만큼 CSS 속성을 원하는 만큼 변경할 수 있습니다.

@keyframes

animation-name

animation-duration

animation-timing-function

animation-delay

animation-iteration-count

animation-direction

animation-fill-mode

animation-play-state

animation

### CSS Animation Properties

Property	Description
@keyframes	애니메이션 코드를 지정합니다.
animation-name	@keyframes 애니메이션의 이름을 지정합니다.
animation-duration	애니메이션이 한 주기를 완료하는 데 걸리는 시간을 지정합니다.
animation-timing-function	애니메이션의 속도 곡선을 지정합니다.
animation-delay	애니메이션 시작 지연을 지정합니다.
animation-iteration-count	애니메이션을 재생해야 하는 횟수를 지정합니다.
animation-direction	애니메이션을 앞으로, 뒤로 또는 대체 주기로 재생해야 하는지 여부를 지정합니다.
animation-fill-mode	애니메이션이 재생되지 않을 때(시작 전, 종료 후 또는 둘 다) 요소의 스타일을 지정합니다.
animation-play-state	애니메이션이 실행 중인지 또는 일시 중지되었는지 지정합니다.
animation	모든 애니메이션 속성을 설정하기 위한 속기 속성

CSS 애니메이션을 사용하려면 먼저 애니메이션에 대한 몇 가지 키프레임을 지정해야 합니다. 키프레임은 특정 시간에 요소가 가질 스타일을 유지합니다.

#### @keyframes 규칙

규칙 내부에 CSS 스타일을 지정하면 @keyframes 애니메이션이 특정 시간에 현재 스타일에서 새 스타일로 점진적으로 변경됩니다.

애니메이션이 작동하려면 애니메이션을 요소에 바인딩해야 합니다.

다음 예제는 "예제" 애니메이션을 <div> 요소에 바인딩합니다. 애니메이션은 4초 동안 지속되며 <div> 요소의 배경색이 "빨간색"에서 "노란색"으로 점차 변경됩니다.

예)

```
@keyframes example {  
  from {background-color: red;}  
  to {background-color: yellow;}  
}  
  
div {  
  width: 100px;  
  height: 100px;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
}
```

#### animation-name

animation-name은 @keyframes 애니메이션의 이름을 지정합니다.

#### animation-duration

animation-duration은 애니메이션을 완료하는 데 걸리는 시간을 정의합니다.

animation-duration을 지정하지 않으면 기본값이 0초(0초)이므로 애니메이션이 발생하지 않습니다.

"from" 및 "to"(0%(시작) 및 100%(완료)를 나타냄) 키워드를 사용하여 스타일이 변경되는 시기를 지정했습니다.

퍼센트를 사용하는 것도 가능합니다. 퍼센트를 사용하여 원하는 만큼 스타일 변경을 추가할 수 있습니다.

## animation-timing-function

animation-timing-function 속성은 애니메이션의 속도 곡선을 지정합니다.

animation-timing-function 속성은 다음 값을 가질 수 있습니다.

ease - 천천히 시작한 다음 빠르게, 그 다음에는 느리게 끝나는 애니메이션을 지정합니다(기본값).

linear - 처음부터 끝까지 동일한 속도로 애니메이션을 지정합니다.

ease-in - 느린 시작으로 애니메이션을 지정합니다.

ease-out - 느린 종료로 애니메이션을 지정합니다.

ease-in-out - 느린 시작과 끝으로 애니메이션을 지정합니다.

cubic-bezier(n,n,n,n) - 큐빅-베지어 함수에서 자신만의 값을 정의할 수 있습니다.

## animation-delay

animation-delay 속성은 애니메이션 시작 지연을 지정합니다.

음수 값도 허용됩니다. 음수 값을 사용하면 이미 N 초 동안 재생된 것처럼 애니메이션이 시작됩니다 .

다음 예제에서는 애니메이션이 이미 2초 동안 재생된 것처럼 시작됩니다.

```
div {  
  width: 100px;  
  height: 100px;  
  position: relative;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
  animation-delay: -2s;  
}
```

## animation-iteration-count

animation-iteration-count은 애니메이션이 실행되어야 하는 횟수를 지정합니다.

다음 예제에서는 애니메이션이 중지되기 전에 애니메이션을 3번 실행합니다.

```
div {  
  width: 100px;  
  height: 100px;  
  position: relative;  
  background-color: red;  
  animation-name: example;  
  animation-duration: 4s;  
  animation-iteration-count: 3;  
}
```

"infinite" 값을 사용하여 애니메이션이 영원히 계속되도록 합니다.

## animation-direction

animation-direction속성은 애니메이션을 앞으로, 뒤로 또는 대체 주기로 재생해야 하는지 여부를 지정합니다.

animation-direction 속성은 다음 값을 가질 수 있습니다.

normal - 애니메이션이 정상적으로 재생됩니다(앞으로). 이것은 기본값입니다

reverse - 애니메이션이 역방향(backwards)으로 재생됩니다.

alternate - 애니메이션이 먼저 앞으로 재생된 다음 뒤로 재생됩니다.

alternate-reverse - 애니메이션이 먼저 뒤로 재생된 다음 앞으로 재생됩니다.

## animation-fill-mode

CSS 애니메이션은 첫 번째 키프레임이 재생되기 전이나 마지막 키프레임이 재생된 후에 요소에 영향을 주지 않습니다. animation-fill-mode 속성은 이 동작을 재정의할 수 있습니다.

animation-fill-mode속성은 애니메이션이 재생되지 않을 때(시작 전, 종료 후 또는 둘 다) 대상 요소에 대한 스타일을 지정합니다.

animation-fill-mode 속성

1. none - 기본값. 애니메이션은 실행 전이나 후에 요소에 스타일을 적용하지 않습니다.
2. forwards - 요소는 마지막 키프레임에 의해 설정된 스타일 값을 유지합니다(애니메이션 방향 및 애니메이션 반복 횟수에 따라 다름).
3. backwards - 요소는 첫 번째 키프레임(애니메이션 방향에 따라 다름)에 의해 설정된 스타일 값을 가져오고 애니메이션 지연 기간 동안 이 값을 유지합니다.
4. both - 애니메이션은 정방향 및 역방향 모두에 대한 규칙을 따르며 애니메이션 속성을 양방향으로 확장합니다.

## animation-play-state

animation-play-state속성은 애니메이션이 실행 중인지 또는 일시 중지되었는지 지정합니다.  
속성 값

Value	Description
paused	애니메이션이 일시 중지되었음을 지정합니다.
running	기본값. 애니메이션이 실행 중임을 지정합니다.

CSS Syntax

animation: name duration timing-function delay iteration-count direction fill-mode play-state;

## CSS Multiple Columns

CSS Multiple Columns 레이아웃을 사용하면 신문에서와 같이 텍스트의 여러 열을 쉽게 정의할 수 있습니다.

column-count  
column-gap  
column-rule-style  
column-rule-width  
column-rule-color  
column-rule  
column-span  
column-width

column-count속성은 요소를 나누어야 하는 열 수를 지정합니다.

column-gap속성은 열 사이의 간격을 지정합니다.

column-rule-style속성은 열 사이의 규칙 스타일을 지정합니다.

column-rule-width속성은 열 사이의 규칙 너비를 지정합니다.

column-rule-color속성은 열 사이의 규칙 색상을 지정합니다.

column-rule속성은 위의 모든 column-rule-\* 속성을 설정하기 위한 속기 속성입니다.

예)

```
div {  
    column-rule: 1px solid lightblue;  
}
```

column-span속성은 요소가 걸쳐야 하는 열 수를 지정합니다.

column-width속성은 열에 대해 제안된 최적의 너비를 지정합니다.