

# 1. 불투명도 속성

예제 8-1 마우스를 올리면 선명하게 보이게 설정하기

ch08/01\_opacity1.html

```
<head>
  <style>
    a:link {
      opacity: 0.5;
    }
    a:hover {
      opacity: 1.0;
    }
    img {
      opacity: 0.2;
    }
    img:hover {
      opacity: 1.0;
    }
  </style>
</head>
<body>
  <h3>마우스를 올리면 선명하게 보입니다.</h3>
  <div>
    <a href="http://www.google.com">구글 웹 사이트</a>
  </div>
  <p></p>
  <div>
    
  </div>
</body>
```

마우스를 올리면 선명하게 보입니다.

구글 웹 사이트

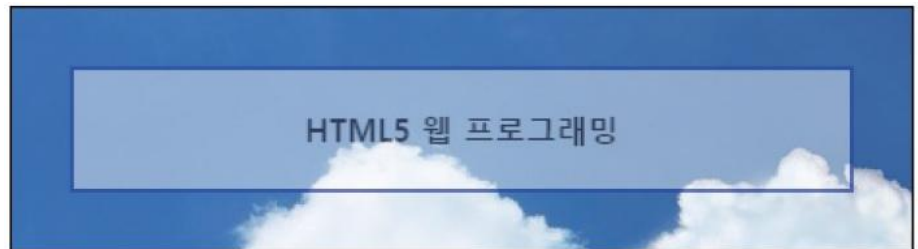


# 1. 불투명도 속성

예제 8-2 텍스트 상자 안의 배경 이미지를 반투명하게 처리하기

ch08/02\_opacity2.html

```
<head>
  <style>
    div.background {
      background: url(sky.jpg) repeat;
      border: 1px solid black;
    }
    div.box {
      margin: 30px;
      background-color: #ffffff;
      border: 2px solid blue;
      opacity: 0.5;
    }
    div.box p {
      margin: 5%;
      font-weight: bold;
      color: #000000;
      text-align: center;
    }
  </style>
</head>
<body>
  <div class="background">
    <div class="box">
      <p>HTML5 웹 프로그래밍</p>
    </div>
  </div>
</body>
```



# 1. 불투명도 속성

예제 8-3 마우스를 올리면 힌트와 정답 보여주기

ch08/03\_opacity3.html

```
<head>
  <style>
    div.tip {
      opacity: 0.2;
    }
    div.ans {
      opacity: 0.0;
    }
    div.tip:hover {
      opacity: 1.0;
      color: red;
    }
    div.ans:hover {
      opacity: 1.0;
      color: blue;
    }
  </style>
</head>
```

```
<body>
  <p>[ 문제 ] CSS3에서 불투명도를 적용하기 위한 속성은?</p>
  <p>① border</p>
  <p>② opacity</p>
  <p>③ transparency</p>
  <p>④ visible</p>
  [ 힌트 ] <div class="tip">'불투명'을 뜻하는 영문 단어를 찾아보세요.</div>
  <p></p>
  [정답] <div class="ans">정답은 ②번입니다.</div>
</body>
```

[문제] CSS3에서 불투명도를 적용하기 위한 속성은?

- ① border
- ② opacity
- ③ transparency
- ④ visible

[힌트]

'불투명'을 뜻하는 영문 단어를 찾아보세요.

[정답]

[문제] CSS3에서 불투명도를 적용하기 위한 속성은?

- ① border
- ② opacity
- ③ transparency
- ④ visible

[힌트]

'불투명'을 뜻하는 영문 단어를 찾아보세요.

[정답]

정답은 ②번입니다.

## 2. 가시성 속성

- 가시성 속성

- 어떤 요소를 보이게 하거나 반대로 보이지 않게 할 때 사용

- 디스플레이 속성

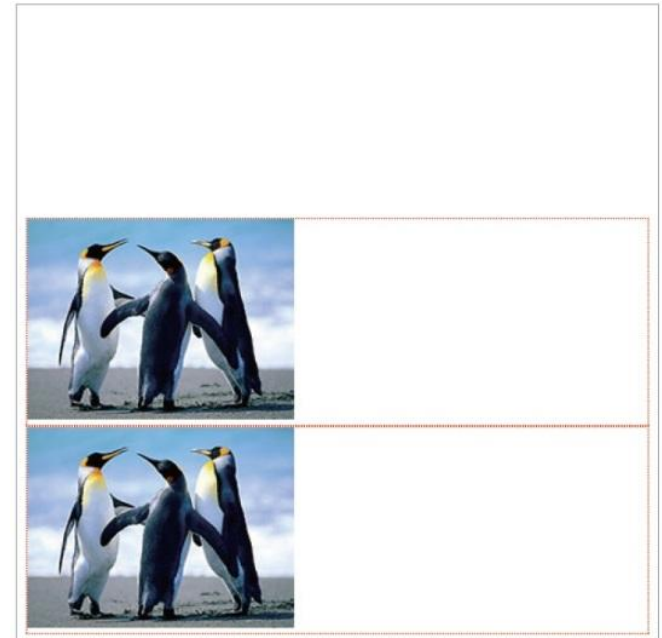
- 가시성 속성과 반대로 요소가 차지하는 공간도 사라짐

## 2. 가시성 속성

예제 8-4 가시성 속성과 디스플레이 속성 비교하기

ch08/04\_visibility.html

```
<head>
  <style>
    .v1 {
      visibility: hidden;
      border: 1px dotted red;
    }
    .v2 {
      visibility: visible;
      border: 1px dotted red;
    }
    .v3 {
      display: none;
      border: 1px dotted red;
    }
  </style>
</head>
<body>
  <div class="v1"> <!-- 보이지 않도록 설정: 공간 있음 -->
    
  </div>
  <div class="v2">
    
  </div>
  <div class="v3"> <!-- 보이지 않도록 설정: 공간 삭제 -->
    
  </div>
  <div class="v2">
    
  </div>
</body>
```



### 3. 형식 변환 속성

- **블록 형식**

- 다음 요소가 항상 새로운 행에서 시작되며 화면의 최대 너비만큼 차지

- **인라인 형식**

- 다음 요소가 이전 요소 바로 뒤에 배치되며 최소한의 너비만 가짐

### 3. 형식 변환 속성

예제 8-5 인라인 형식을 블록 형식으로 변환하기

ch08/05\_display1.htmlhtml

```
<head>
  <style>
    p strong {
      color: blue;
      border: 1px dotted red;
    }
    p.bk strong {
      display: block;
      color: blue;
      border: 1px dotted red;
    }
  </style>
</head>
<body>
  <h3>[인라인 형식]</h3>
  <p> 세계적인 IT 기업에는 <strong>Google</strong> <strong>Apple</strong> <strong>Oracle</strong> 등이 있습니다.</p>
  <h3>[블록 형식으로 변환한 후]</h3>
  <p class="bk"> 세계적인 IT 기업에는 <strong>Google</strong> <strong>Apple</strong> <strong>Oracle</strong> 등이 있습니다.</p>
</body>
```

#### [인라인 형식]

세계적인 IT 기업에는 Google Apple Oracle 등이 있습니다.

#### [블록 형식으로 변환한 후]

세계적인 IT 기업에는

Google

Apple

Oracle

등이 있습니다.

### 3. 형식 변환 속성

예제 8-6 블록 형식을 인라인 형식으로 변환하기

ch08/06\_display2.html

```
<head>
  <style>
    ul.in li {
      display: inline;
      background-color: yellow;
      border: 1px solid;
      border-color: blue;
      margin: 3px;
      padding: 5px;
    }
  </style>
</head>
<body>
  <h4>[블록 형식]</h4>
  <ul>
    <li><a href="http://www.google.com">Google</a></li>
    <li><a href="http://www.apple.com">Apple</a></li>
    <li><a href="http://www.oracle.com">Oracle</a></li>
  </ul>
  <h4>[인라인 형식으로 변환한 후]</h4>
  <ul class="in">
    <li><a href="http://www.google.com">Google</a></li>
    <li><a href="http://www.apple.com">Apple</a></li>
    <li><a href="http://www.oracle.com">Oracle</a></li>
  </ul>
</body>
```

#### [블록 형식]

- [Google](http://www.google.com)
- [Apple](http://www.apple.com)
- [Oracle](http://www.oracle.com)

#### [인라인 형식으로 변환한 후]





## 4. 백그라운드 속성(그레디언트 효과)

### ● 그레디언트 효과

- background나 background-image 속성을 이용
- 선형과 원형 두 가지 형태가 있음

### ● 그레디언트 효과에서 색상 번짐 방향 설정

**표 8-1** direction에 지정할 수 있는 값

방향	각도
to top	0deg, 360deg
to right	90deg
to bottom	180deg
to left	270deg

## 4. 백그라운드 속성(그레디언트 효과)

### 예제 8-7 선형 그레디언트 효과 적용하기

ch08/07\_linear.html

```

<head>
  <style>
    #grad1 {
      height: 70px;
      background: red;
      background: linear-gradient(270deg, red, yellow);
    }
    #grad2 {
      height: 70px;
      background: red;
      background: linear-gradient(red, yellow, green);
    }
    #grad3 {
      height: 70px;
      background: red;
      background: linear-gradient(red, orange, yellow, green, blue, indigo, violet);
    }
    #grad4 {
      height: 70px;
      background: red;
      background: linear-gradient(to right, red, orange, yellow, green, blue, indigo, violet);
    }
  </style>
</head>
<body>
  <h4>2색 선형 그레디언트</h4>
  <div id="grad1"></div>
  <h4>3색 선형 그레디언트</h4>
  <div id="grad2"></div>
  <h4>7색 선형 그레디언트</h4>
  <div id="grad3"></div>
  <h4>2색 선형 그레디언트(to right)</h4>
  <div id="grad4"></div>
  <p><strong>참고:</strong> 최신 브라우저를 사용해주시기 바랍니다.</p>
</body>

```

2색 선형 그레디언트



3색 선형 그레디언트



7색 선형 그레디언트



2색 선형 그레디언트(to right)



참고: 최신 브라우저를 사용해주시기 바랍니다.

## 4. 백그라운드 속성(그레디언트 효과)

### 예제 8-8 원형 그레디언트 효과 적용하기

ch08/08\_radial.html

```

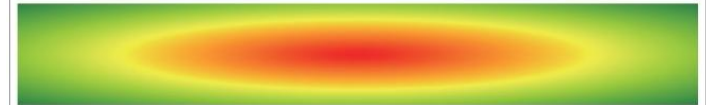
<head>
  <style>
    #grad1 {
      height: 70px;
      background: red;
      background: radial-gradient(yellow, green); }
    #grad2 {
      height: 70px;
      background: red;
      background: radial-gradient(red, yellow, green); }
    #grad3 {
      height: 70px;
      background: red;
      background: radial-gradient(red 5%, yellow 15%, green 60%) }
    #grad4 {
      height: 100px;
      width: 150px;
      background: red;
      background: radial-gradient(circle, red, yellow, green); }
  </style>
</head>
<body>
  <h4>기본 2색 원형 그레디언트</h4>
  <div id="grad1"></div>
  <h4>3색 원형 그레디언트 </h4>
  <div id="grad2"></div>
  <h4>색상 영역 지정(% 단위)</h4>
  <div id="grad3"></div>
  <h4>원형 모양 지정(ellipse, circle)</h4>
  <div id="grad4"></div>
  <p><strong>참고: </strong>최신 브라우저를 사용해주시기 바랍니다.</p>
</body>

```

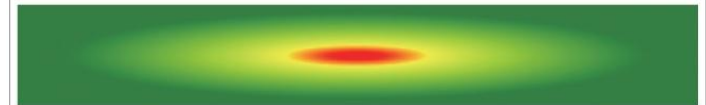
기본 2색 원형 그레디언트



3색 원형 그레디언트



색상 영역 지정(% 단위)



원형 모양 지정(ellipse, circle)



참고: 최신 브라우저를 사용해주시기 바랍니다.

# 1. 2차원 변환 함수

**표 8-2** 2차원 변환 함수의 종류

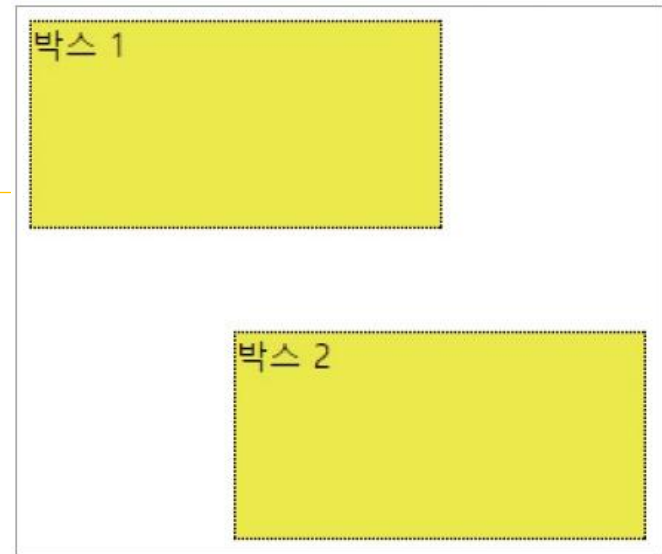
함수	설명	사용 예
translate( )	평행 이동 변환	transform: translate(50px, 100px);
rotate( )	회전 변환	transform: rotate(20deg);
scale( )	크기 변환	transform: scale(2, 3);
skewX( )	X축 기울기 변환	transform: skewX(20deg);
skewY( )	Y축 기울기 변환	transform: skewY(20deg);
skew( )	X, Y축 기울기 변환	transform: skew(20deg, 10deg);
matrix( )	2차원 행렬 구조 변환	transform: matrix(1, -0.3, 0, 1, 0, 0);

## 2. 평행 이동 변환

### 예제 8-9 평행 이동 변환하기

ch08/09\_translate.htmlial.html

```
<head>
  <style>
    div {
      width: 200px;
      height: 100px;
      border: 1px dotted black;
      background-color: yellow;
    }
    div#box2 {
      transform: translate(100px, 50px);
    }
  </style>
</head>
<body>
  <div id="box1">박스 1</div>
  <div id="box2">박스 2</div>
</body>
```



### 3. 회전 변환

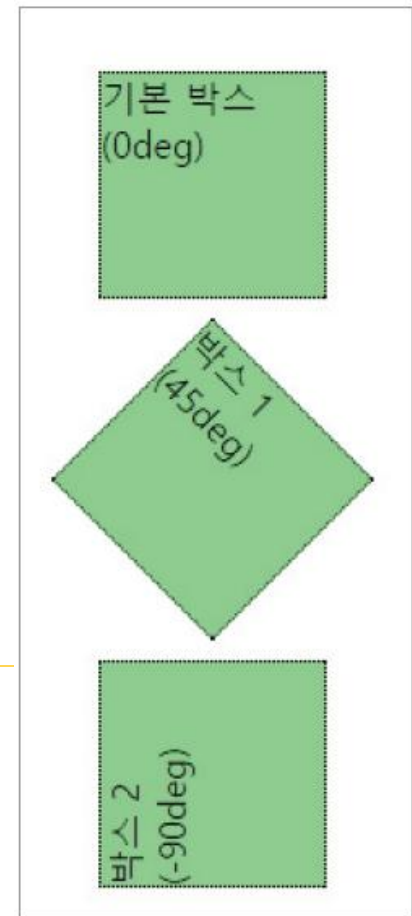
예제 8-10 회전 변환하기

ch08/10\_rotate.html

```

<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      border: 1px dotted black;
      background-color: lightgreen;
      margin: 30px;
    }
    div#box1 {
      transform: rotate(45deg);
    }
    div#box2 {
      transform: rotate(-90deg);
    }
  </style>
</head>
<body>
  <div>기본 박스 (0deg)</div>
  <div id="box1">박스 1 (45deg)</div>
  <div id="box2">박스 2 (-90deg)</div>
</body>

```



## 4. 크기 변환

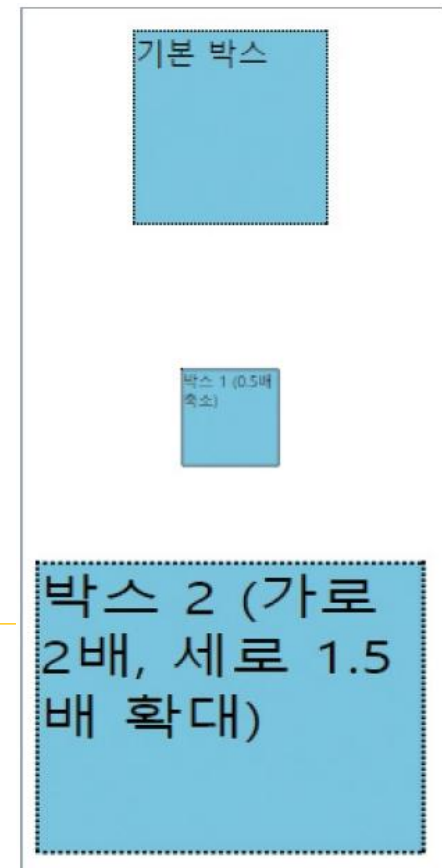
예제 8-11 크기 변환하기

ch08/11\_scale.html

```

<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      border: 1px dotted black;
      background-color: skyblue;
      margin: 50px;
    }
    div#box1 {
      transform: scale(0.5, 0.5);
    }
    div#box2 {
      transform: scale(2, 1.5);
    }
  </style>
</head>
<body>
  <div>기본 박스</div>
  <div id="box1">박스 1 (0.5배 축소)</div>
  <div id="box2">박스 2 (가로 2배, 세로 1.5배 확대)</div>
</body>

```



## 5. 기울기 변환

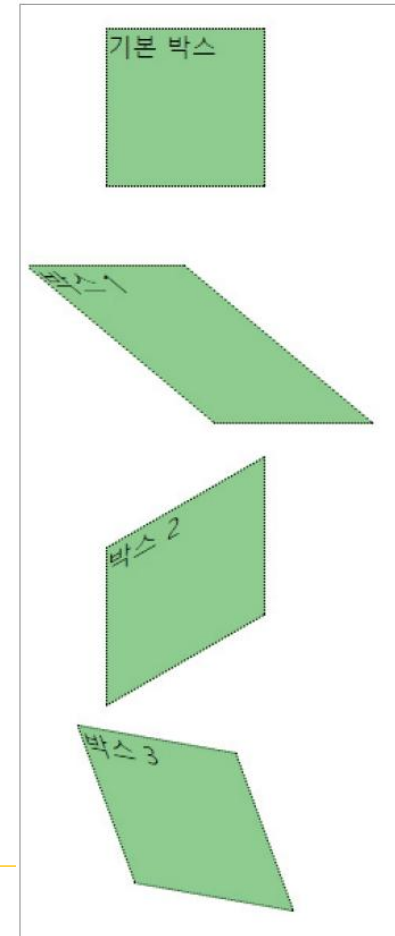
예제 8-12 기울기 변환하기

ch08/12\_skew.html

```

<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      border: 1px dotted black;
      background-color: lightgreen;
      margin: 50px;
    }
    div#box1 {
      transform: skewX(50deg);
    }
    div#box2 {
      transform: skewY(-30deg);
    }
    div#box3 {
      transform: skew(20deg, 10deg);
    }
  </style>
</head>
<body>
  <div>기본 박스</div>
  <div id="box1">박스 1</div>
  <div id="box2">박스 2</div>
  <div id="box3">박스 3</div>
</body>

```



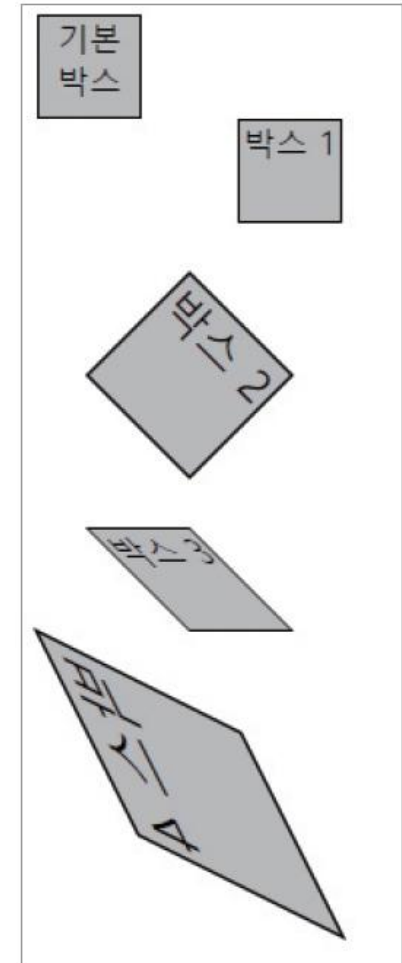


## 6. 2차원 행렬 구조 변환

예제 8-13 2차원 행렬 구조 변환하기

ch08/13\_matrix.html

```
<head>
  <style>
    div {
      width: 50px;
      height: 50px;
      background-color: silver;
      border: 1px solid black;
      text-align: center;
    }
    div#box1 {
      transform: matrix(1, 0, 0, 1, 100, 0);
    }
    div#box2 {
      transform: matrix(1, 1, -1, 1, 50, 50);
    }
    div#box3 {
      transform: matrix(1, 0, 1, 1, 50, 100);
    }
    div#box4 {
      transform: matrix(1, 2, 2, 1, 50, 150);
    }
  </style>
</head>
<body>
  <div>기본 박스</div>
  <div id="box1">박스 1</div>
  <div id="box2">박스 2</div>
  <div id="box3">박스 3</div>
  <div id="box4">박스 4</div>
</body>
```



# 7. 혼합 변환

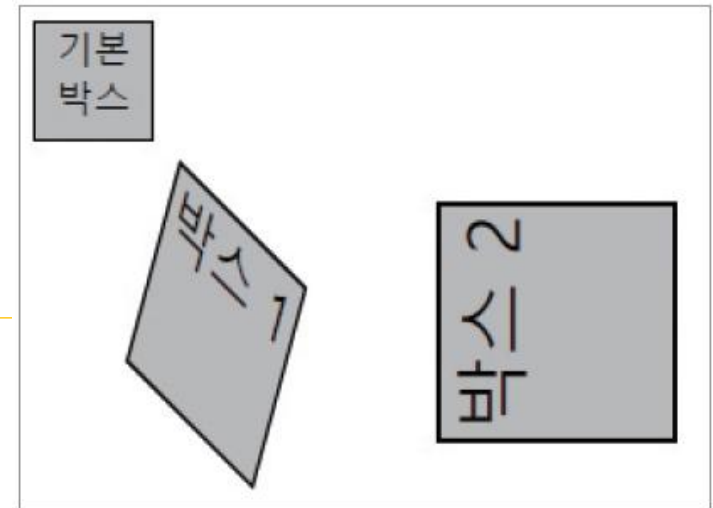
예제 8-14 혼합 변환하기

ch08/14\_compound.html

```

<head>
  <style>
    div {
      width: 50px;
      height: 50px;
      background-color: silver;
      border: 1px solid black;
      text-align: center;
    }
    div#box1 {
      transform: rotate(45deg) scale(1.5) skew(30deg) translate(50px);
    }
    div#box2 {
      transform: translate(200px) rotate(-90deg) scale(2);
    }
  </style>
</head>
<body>
  <div>기본 박스</div>
  <div id="box1">박스 1</div>
  <div id="box2">박스 2</div>
</body>

```



# 7. 혼합 변환

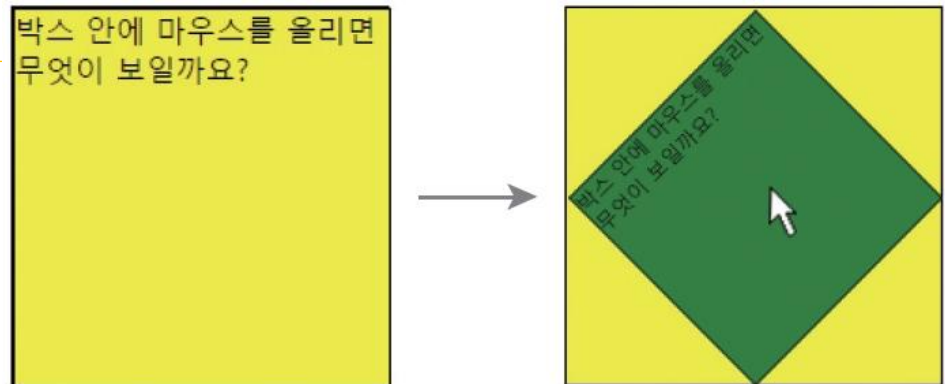
예제 8-15 2차원 변환 효과 응용하기

ch08/15\_apply.html

```

<head>
  <style>
    div {
      width: 200px;
      height: 200px;
      border: 1px solid black;
      background: yellow;
    }
    .c1:hover {
      transform-origin: 50% 50% 0px;
      transform: translate(0px, 0px) rotate(-45deg) scale(0.7);
      background: green;
    }
  </style>
</head>
<body>
  <div>
    <div class="c1">박스 안에 마우스를 올리면 무엇이 보일까요?</div>
  </div>
</body>

```



# 1. 3차원 변환 함수

표 8-3 3차원 변환 함수의 종류

함수	설명	사용 예
perspective(depth)	3차원 관점 변환	transform: perspective(100px);
translate3d(x, y, z)	3차원 평행 이동 변환	transform: translate3d(50px, 100px, 150px);
translateX(x)	3차원 평행 이동(x축)	transform: translateX(50px);
translateY(y)	3차원 평행 이동(y축)	transform: translateY(50px);
translateZ(z)	3차원 평행 이동(z축)	transform: translateZ(50px);
scale3d(x, y, z)	3차원 크기 변환	transform: scale3d(2, 2, 2);
scaleX(x)	3차원 크기 변환(x축)	transform: scaleX(2);
scaleY(y)	3차원 크기 변환(y축)	transform: scaleY(2);
scaleZ(z)	3차원 크기 변환(z축)	transform: scaleZ(2);
rotate3d(x, y, z, angle)	3차원 회전 변환	transform: rotate3d(1, 1, 1, 45deg);
rotateX(angle)	3차원 회전 변환(x축)	transform: rotateX(30deg);
rotateY(angle)	3차원 회전 변환(y축)	transform: rotateY(30deg);
rotateZ(angle)	3차원 회전 변환(z축)	transform: rotateZ(30deg);

# 1. 3차원 변환 함수

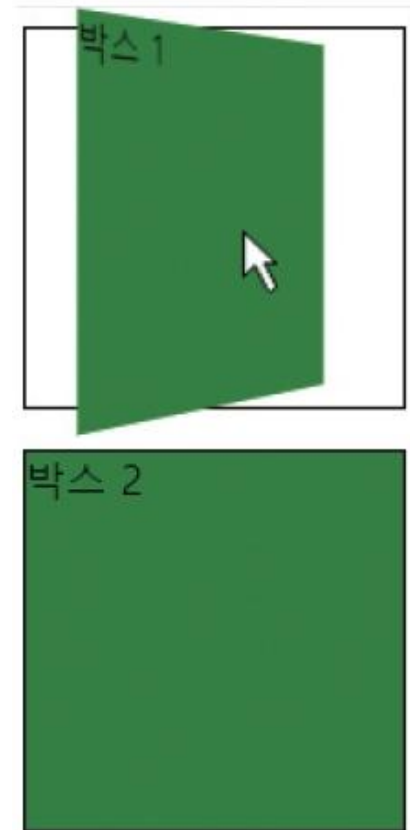
예제 8-16 3차원 회전하기

ch08/16\_3d.html

```

<head>
  <style>
    div {
      background-color: green;
      height: 150px;
      width: 150px;
    }
    .container {
      background-color: white;
      border: 1px solid black;
    }
    .transformed:hover {
      backface-visibility: visible;
      transform-origin: 50% 42%;
      transform: perspective(500px) rotateY(50deg) rotateX(0deg);
    }
    .transformed2:hover {
      backface-visibility: visible;
      transform-origin: 50% 42%;
      transform: perspective(500px) rotateY(0deg) rotateX(45deg);
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="transformed">박스 1</div>
  </div>
  <p></p>
  <div class="container">
    <div class="transformed2">박스 2</div>
  </div>
</body>

```



# 1. 변화 속성

## ● 변화 속성

- 효과가 적용되는 과정을 좀 더 부드럽게 보여주거나, 그 과정을 시간적으로 조정할 수 있도록 해주는 속성

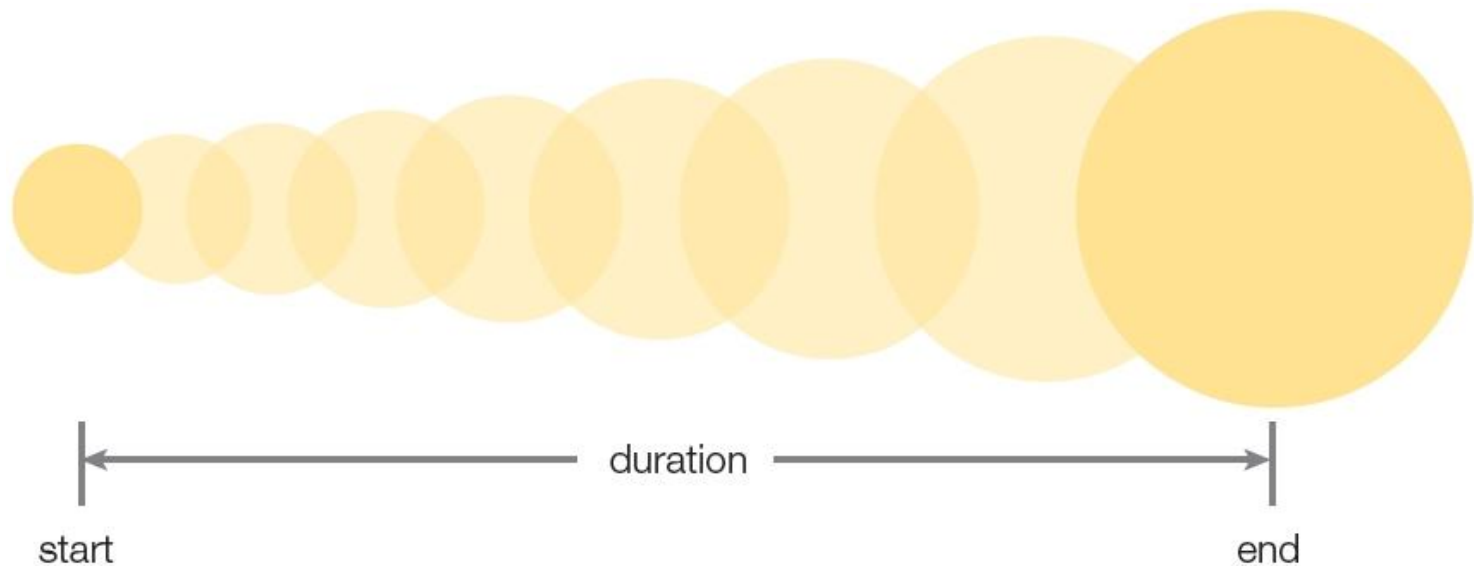


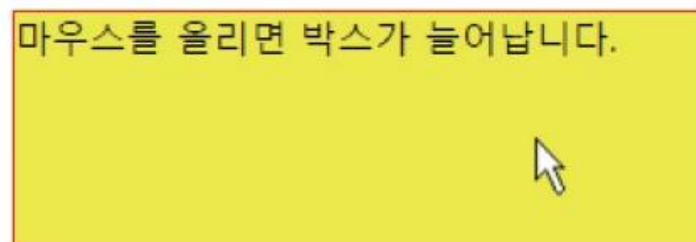
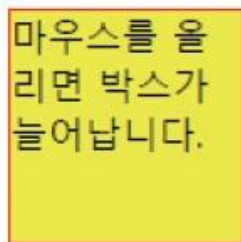
그림 8-2 변화 효과의 원리

# 1. 변화 속성

**예제 8-17** 박스 가로 길이 늘리기

ch08/17\_transition1.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: yellow;
      border: 1px solid red;
      transition: width 2s;
    }
    div:hover {
      width: 300px;
    }
  </style>
</head>
<body>
  <div>마우스를 올리면 박스가 늘어납니다.</div>
</body>
```



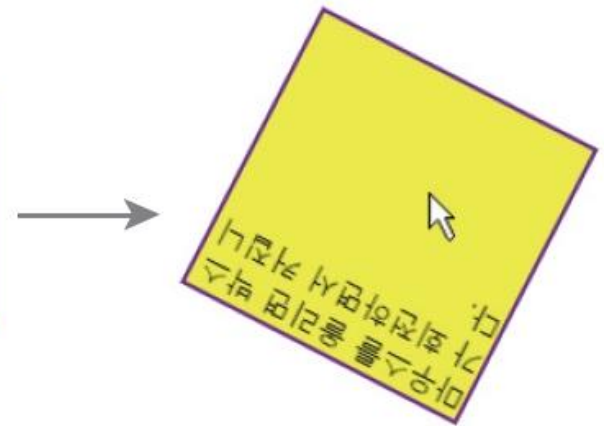
# 1. 변화 속성

예제 8-18 박스를 회전시키면서 크기와 테두리 색상 변경하기

ch08/18\_transition2.html

```
<head>
  <style>
    div {
      margin: 50px;
      width: 100px;
      height: 100px;
      background: yellow;
      border: 1px solid red;
      transition: width 3s, height 3s, border 3s, transform 3s;
    }
    div:hover {
      width: 200px;
      height: 200px;
      border: 3px solid blue;
      transform: rotate(360deg);
    }
  </style>
</head>
<body>
  <div>마우스를 올리면 박스가 회전하면서 커집니다.</div>
</body>
```

마우스를 올리면 박스가 회전하면서 커집니다.





# 1. 변화 속성

## ● 변화 속성

- transition-property : 변화 효과를 적용할 속성들 나열
- transition-duration : 변화가 지속되는 시간 지정
- transition-timing-function : 변화의 시작과 끝 타이밍 지정
- transition-delay : 변화 효과가 지연되는 시간 지정

# 1. 변화 속성

## ● 변화 속성 작성 방식

### - 단축형

```
div { transition: property duration timing-function delay; }
```

```
div {  
  transition: background 2s ease 1s,  
             padding 1s linear 2s;  
}
```

### - 기본형

```
div {  
  transition-property: width, color;  
  transition-duration: 1s;  
  transition-timing-function: ease;  
  transition-delay: 3s;  
}
```

### - 확장 기본형

```
div {  
  transition-property: width, height, border-width, color;  
  transition-duration: 1s, 2s, 1s, 3s;  
  transition-timing-function: ease, ease-in, ease-out, linear;  
  transition-delay: 3s, 1s, 1s, 2s;  
}
```

## 2. transition-property 속성

예제 8-19 변화 효과 대상 지정하기

ch08/19\_tproperty.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: orange;
      transition-property: width, background, color;
    }
    div:hover {
      width: 400px;
      background: blue;
      color: white;
    }
  </style>
</head>
<body>
  <div>마우스를 올리면 여러 속성이 변합니다.</div>
</body>
```

마우스를 올리면 여러 속성이 변합니다.



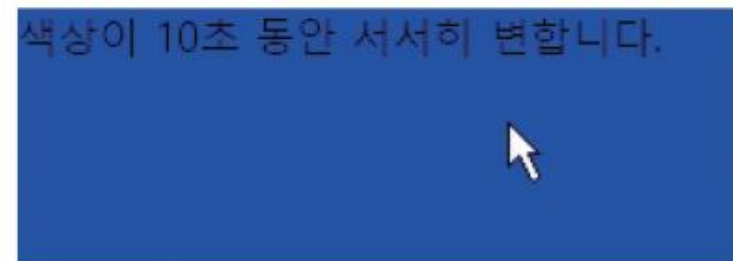
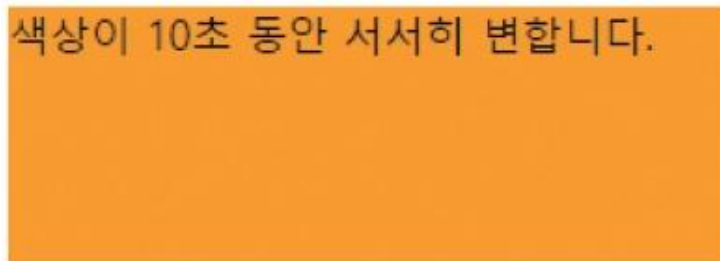
마우스를 올리면 여러 속성이 변합니다.

### 3. transition-duration 속성

예제 8-20 변화 효과의 지속 시간 설정하기

ch08/20\_tduration.html

```
<head>
  <style>
    div {
      width: 280px;
      height: 100px;
      background: orange;
      transition: background;
      transition-duration: 10s;
    }
    div:hover {
      background: blue;
    }
  </style>
</head>
<body>
  <div>색상이 10초 동안 서서히 변합니다.</div>
</body>
```



## 4. transition-timing-function 속성

### ● 속성값

- linear : 처음부터 끝까지 같은 속도
- ease : 느리게 시작하여 점점 빨라졌다가 느리게 끝남
- ease-in : 느리게 시작하여 점점 빨라지다가 일정한 속도에 다다르면 같은 속도를 유지
- ease-out : 일정한 속도의 등속 변화로 시작해서 점점 느려지면서 끝남
- ease-in-out : 느리게 시작하여 느리게 끝남
- cubic-bezier(n, n, n, n) : 처음과 끝의 속도를 설정

## 4. transition-timing-function 속성

### ● 큐빅 베이지 타이밍 함수

- 원하는 대로 타이밍을 조정할 때 사용
- 기본값은 cubic-bezier(0.25, 0.1, 0.25, 1.0)

**표 8-4** 속성값과 큐빅 베이지 함수값의 매칭

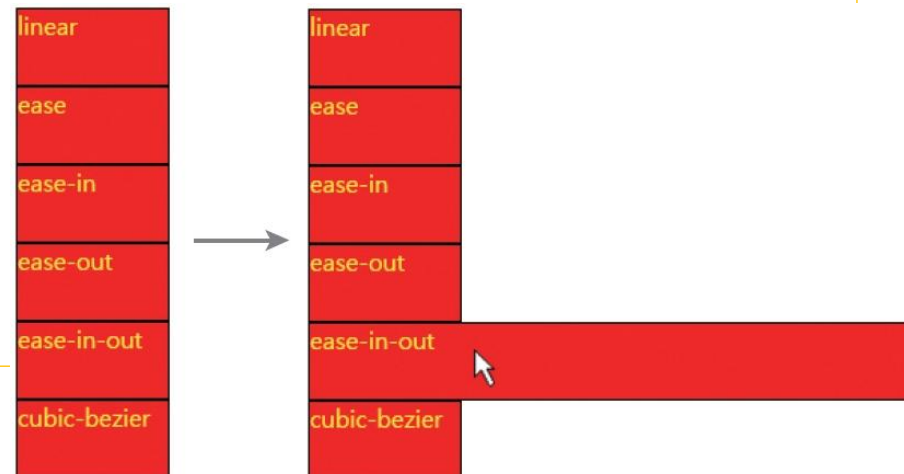
속성값 이름	큐빅 베이지 타이밍 설정값
Ease	cubic-bezier(0.25, 0.1, 0.25, 1)
Linear	cubic-bezier(0, 0, 1, 1)
ease-in	cubic-bezier(0.42, 0, 1, 1)
ease-out	cubic-bezier(0, 0, 0.58, 1)
ease-in-out	cubic-bezier(0.42, 0, 0.58, 1)

## 4. transition-timing-function 속성

예제 8-21 변화 효과의 타이밍 설정하기

ch08/21\_tfunction.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 50px;
      background: red;
      color: yellow;
      border: 1px solid black;
      transition: width 3s;
    }
    #div1 { transition-timing-function: linear; }
    #div2 { transition-timing-function: ease; }
    #div3 { transition-timing-function: ease-in; }
    #div4 { transition-timing-function: ease-out; }
    #div5 { transition-timing-function: ease-in-out; }
    #div6 { transition-timing-function: cubic-bezier(0.1, 0.0, 0.1, 1.0); }
    div:hover { width: 400px; }
  </style>
</head>
<body>
  <div id="div1" style="top: 100px">linear</div>
  <div id="div2" style="top: 150px">ease</div>
  <div id="div3" style="top: 200px">ease-in</div>
  <div id="div4" style="top: 250px">ease-out</div>
  <div id="div5" style="top: 300px">ease-in-out</div>
  <div id="div5" style="top: 350px">cubic-bezier</div>
</body>
```

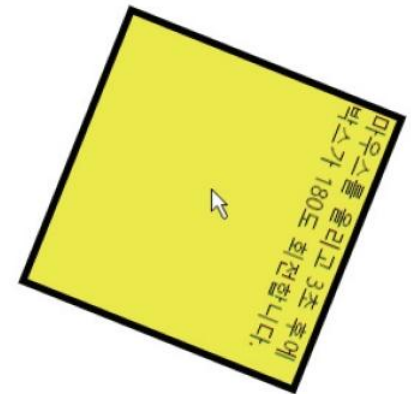
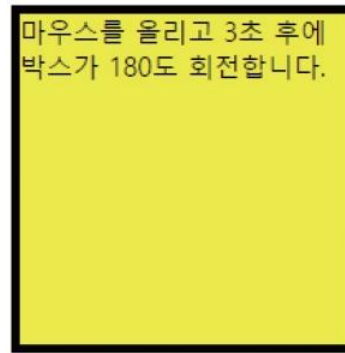


## 5. transition-delay 속성

예제 8-22 변화 효과의 지연 시간 설정하기

ch08/21\_tdelay.html

```
<head>
  <style>
    div {
      margin: 50px;
      width: 200px;
      height: 200px;
      background: yellow;
      border: 5px solid black;
      transition-duration: 5s;
      transition-delay: 3s;
    }
    div:hover {
      transform: rotate(180deg);
    }
  </style>
</head>
<body>
  <div>마우스를 올리고 3초 후에 박스가 180도 회전합니다.</div>
</body>
```



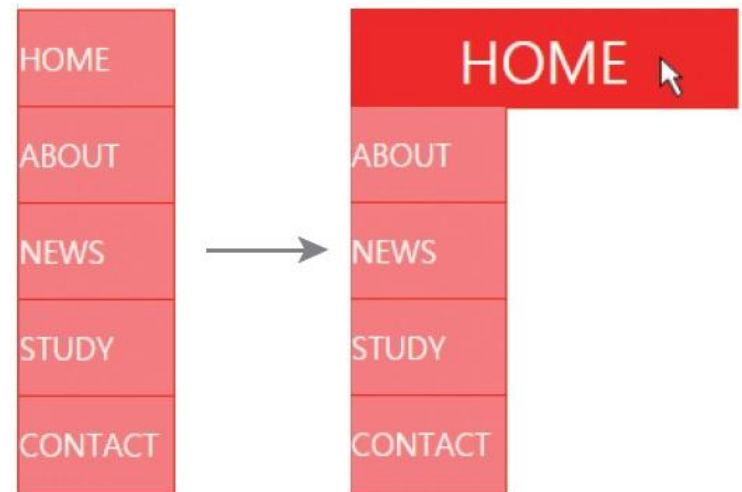


## 5. transition-delay 속성

예제 8-23 홈페이지 메뉴 만들기

ch08/23\_tmenu.html

```
<head>
  <style>
    a {
      text-decoration: none;
      color: white;
    }
    div {
      position: absolute;
      left: 0px;
      width: 80px;
      height: 50px;
      background: #ff8080;
      border: 1px solid red;
      transition-property: width background;
      transition-duration: 2s; 2s;
      line-height: 50px;
    }
    div:hover {
      width: 200px;
      transition-timing-function: linear;
      background: #ff0000;
      color: white;
      text-align: center;
      font-size: 30px;
    }
  </style>
</head>
<body>
  <div style="top:50px"><a href="#" target="new">HOME</a></div>
  <div style="top:100px"><a href="#" target="new">ABOUT</a></div>
  <div style="top:150px"><a href="#" target="new">NEWS</a></div>
  <div style="top:200px"><a href="#" target="new">STUDY</a></div>
  <div style="top:250px"><a href="#" target="new">CONTACT</a></div>
</body>
```



# 1. 애니메이션의 원리

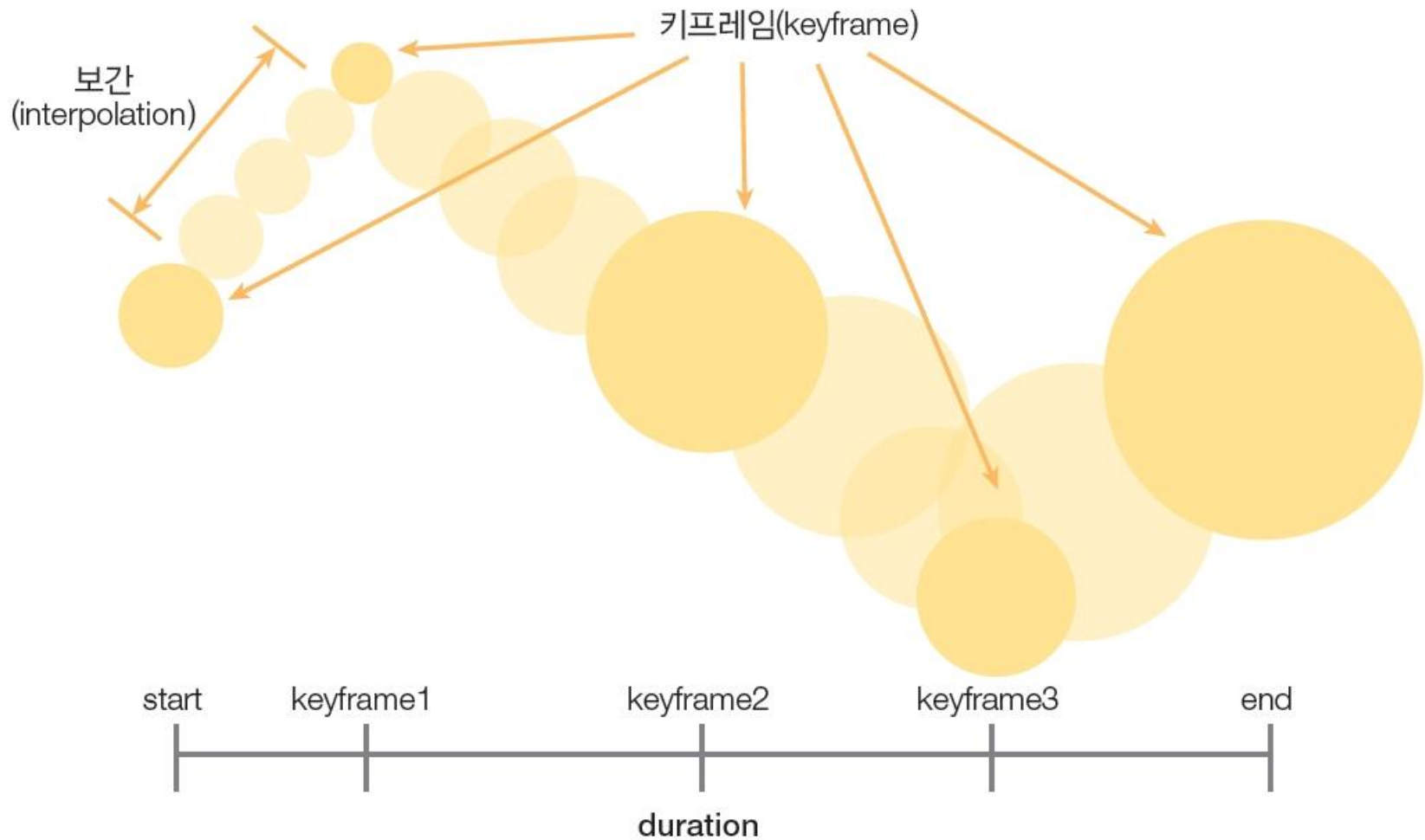


그림 8-3 애니메이션의 원리

# 1. 애니메이션의 원리

## ● 키프레임 정의

```
@keyframes animationname {
  keyframes-selector { css-styles; }
}
```

## ● 위에서 아래로 움직이는 키프레임 설정

```
@keyframes box_animation {
  from { top: 0px; }
  to { top: 200px; }
}
```

## ● 키프레임 안에 퍼센트 단위로 애니메이션 설정

```
@keyframes animationname {
  keyframes-selector { css@keyframes 애니메이션이름{
    0% { 시작값 } /* from */
    25% { 변경값 }
    50% { 변경값 }
    75% { 변경값 }
    100% { 종료값 } /* to */
  }
}
```

## 2. 애니메이션 속성

### ● 애니메이션 속성의 종류

animation-name	@keyframes 애니메이션의 이름 지정
animation-duration	애니메이션의 지속 시간을 초 단위로 설정
animation-timing-function	애니메이션의 시작과 끝 타이밍 지정
animation-delay	애니메이션 시작을 지연시키는 시간을 초 단위로 설정
animation-iteration-count	애니메이션이 반복 재생되는 횟수 설정
animation-direction	애니메이션의 방향 설정
animation-fill-mode	애니메이션을 재생하고 있지 않을 때 속성값 설정
animation-play-state	애니메이션 재생 상태 설정

## 2. 애니메이션 속성

예제 8-24 무한 반복하며 좌우로 이동하는 박스 만들기

ch08/24\_animation.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: red;
      position: relative;
      animation: boxmove 5s linear infinite alternate;
    }
    @keyframes boxmove {
      from { left: 0px; }
      to { left: 300px; }
    }
  </style>
</head>
<body>
  <div>애니메이션 박스</div>
  <p><strong>참고: </strong>IE9 이하 혹은 낮은 버전에서는 지원하지 않습니다.</p>
</body>
```

애니메이션  
박스

참고: IE9 이하 혹은 낮은 버전에서는 지원하지 않습니다.

### 3. animation-delay 속성

예제 8-25 웹 문서가 로드된 후 일정 시간 후에 애니메이션 시작하기

ch08/25\_adelay.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 50px;
      background: red;
      position: relative;
      animation: boxmove 5s linear infinite alternate;
    }
    #box1 {
      animation-delay: 3s;
    }
    #box2 {
      animation-delay: 5s;
    }
    @keyframes boxmove {
      from { left: 0px; }
      to { left: 300px; }
    }
  </style>
</head>
<body>
  <div id="box1">애니메이션 박스 1</div>
  <div id="box2">애니메이션 박스 2</div>
  <p><strong>참고: </strong>IE9 이하 혹은 낮은 버전에서 지원하지 않습니다.</p>
</body>
```



## 4. animation-direction 속성

표 8-5 animation-direction 속성값의 종류

속성값	설명
normal	기본 설정값이다. 애니메이션이 순방향으로 재생된다.
reverse	애니메이션이 역방향으로 재생된다.
alternate	애니메이션이 양방향으로 재생된다. - 홀수 : 순방향으로 재생된다. - 짝수 : 역방향으로 재생된다.
alternate-reverse	애니메이션이 양방향으로 재생된다. - 홀수 : 역방향으로 재생된다. - 짝수 : 순방향으로 재생된다.

## 4. animation-direction 속성

예제 8-26 애니메이션의 진행 방향 설정하기

ch08/26\_adirection.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 50px;
      background: red;
      position: relative;
      animation: boxmove 5s linear infinite;
    }
    #box1 {
      animation-delay: 3s;
      animation-direction: reverse;
    }
    #box2 {
      animation-delay: 5s;
      animation-direction: alternate-reverse;
    }
    @keyframes boxmove {
      from { left: 0px; }
      to { left: 300px; }
    }
  </style>
</head>
<body>
  <div id="box1">애니메이션 박스 1</div>
  <div id="box2">애니메이션 박스 2</div>
  <p><strong>참고: </strong>IE9 이하 혹은 낮은 버전에서 지원하지 않습니다.</p>
</body>
```

애니메이션  
박스 1

애니메이션  
박스 2

참고: IE9 이하 혹은 낮은 버전에서 지원하지 않습니다.



## 5. animation-iteration-count 속성

예제 8-27 애니메이션의 반복 횟수 설정하기

ch08/27\_acount.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 50px;
      background: red;
      position: relative;
      animation: boxmove 5s;
    }
    #box1 {
      animation-delay: 3s;
      animation-direction: reverse;
      animation-iteration-count: 2;
    }
    #box2 {
      animation-delay: 5s;
      animation-direction: alternate-reverse;
      animation-iteration-count: 5;
    }
    @keyframes boxmove {
      from { left: 0px; }
      to { left: 300px; }
    }
  </style>
</head>
<body>
  <div id="box1">애니메이션 박스 1</div>
  <div id="box2">애니메이션 박스 2</div>
  <p><strong>참고: </strong>IE9 이하 혹은 낮은 버전에서 지원하지 않습니다.</p>
</body>
```

애니메이션  
박스 1

애니메이션  
박스 2

참고: IE9 이하 혹은 낮은 버전에서 지원하지 않습니다.

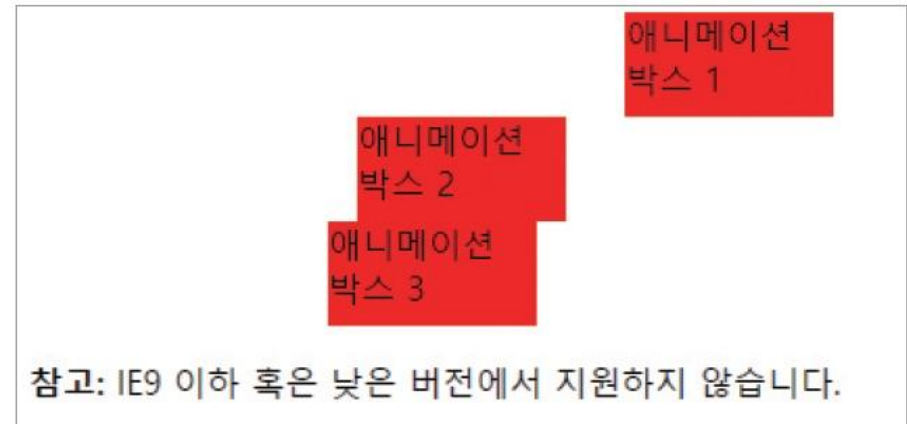
## 6. animation-timing-function 속성

예제 8-28 애니메이션의 타이밍 설정하기

ch08/28\_afunction.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 50px;
      background: red;
      position: relative;
      animation: boxmove 5s alternate;
    }
    #box1 {
      animation-delay: 1s;
      animation-iteration-count: 3;
      animation-timing-function: ease;
    }
    #box2 {
      animation-delay: 2s;
      animation-iteration-count: 3;
      animation-timing-function: linear;
    }
    #box3 {
      animation-delay: 3s;
      animation-iteration-count: 3;
      animation-timing-function: ease-out;
    }
    @keyframes boxmove {
      from { left: 0px; }
      to { left: 300px; }
    }
  </style>
</head>
```

```
<body>
  <div id="box1">애니메이션 박스 1</div>
  <div id="box2">애니메이션 박스 2</div>
  <div id="box3">애니메이션 박스 3</div>
  <p><strong>참고: </strong>IE9 이하 혹은 낮은 버전에서 지
    원하지 않습니다.</p>
</body>
```



# 7 animation-play-state 속성

표 8-6 animation-play-state 속성값의 종류

속성값	설명
paused	애니메이션을 일시 정지한다.
running	애니메이션을 재생한다.

## 예제 8-29 마우스를 올리면 멈추게 하기

ch08/29\_astate.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 50px;
      background: red;
      position: relative;
      animation: boxmove 5s infinite alternate;
    }
    #box1 {
      animation-delay: 1s;
      animation-timing-function: ease;
    }
    #box2 {
      animation-delay: 2s;
      animation-timing-function: linear;
    }
  </style>
</head>
```

▶ 소스코드 뒷 페이지 계속

## 7 animation-play-state 속성

```
#box3 {
  animation-delay: 3s;
  animation-timing-function: ease-out;
}
@keyframes boxmove {
  from { left: 0px; }
  to { left: 300px; }
}
div:hover {
  animation-play-state: paused;
}
</style>
</head>
<body>
  <div id="box1">애니메이션 박스 1</div>
  <div id="box2">애니메이션 박스 2</div>
  <div id="box3">애니메이션 박스 3</div>
  <p><strong>참고: </strong>IE9 이하 혹은 낮은 버전에서 지원하지 않습니다.</p>
</body>
```



참고: IE9 이하 혹은 낮은 버전에서 지원하지 않습니다.

# 7 animation-play-state 속성

예제 8-30 커튼을 치고 걷어내는 듯한 효과 만들기

ch08/30\_effect1.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      position: absolute;
      animation-duration: 5s;
      animation-timing-function: linear;
      animation-delay: 2s;
      animation-iteration-count: infinite;
      animation-play-state: running;
    }
    #div1 {
      background-color: blue;
      animation-direction: normal;
      animation-name: L-box;
    }
    #div2 {
      background-color: yellow;
      animation-direction: reverse;
      animation-name: R-box;
    }
    @keyframes L-box {
      0% { left: 0px; }
      50% { left: 200px; }
      100% { left: 0px; }
    }
```

```
@keyframes R-box {
  0% { left: 400px; }
  50% { left: 200px; }
  100% { left: 400px; }
}
</style>
</head>
<body>
  <div id="div1">왼쪽 박스</div>
  <div id="div2">오른쪽 박스</div>
</body>
```

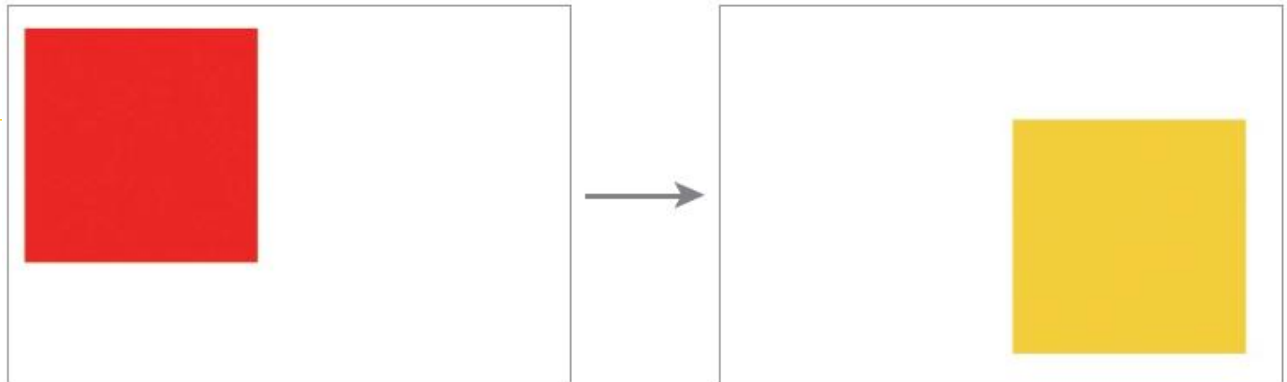


# 7 animation-play-state 속성

예제 8-31 상하좌우로 움직이면서 색상 변경하기

ch08/31\_effect2.html

```
<head>
  <style>
    div {
      width: 100px;
      height: 100px;
      background: red;
      position: relative;
      animation: colorbox 5s infinite;
      animation-direction: alternate;
    }
    @keyframes colorbox {
      from { background: red; left: 0px; top: 0px; }
      25% { background: orange; left: 300px; top: 0px; }
      50% { background: yellow; left: 300px; top: 300px; }
      75% { background: green; left: 0px; top: 300px; }
      to { background: red; left: 0px; top: 0px; }
    }
  </style>
</head>
<body>
  <div></div>
</body>
```



# 7 animation-play-state 속성

예제 8-32 점핑볼 만들기

ch08/32\_effect3.html

```

<head>
  <style>
    @keyframes bounce {
      from, to {
        bottom: 0px;
        animation-timing-function: ease-out;
      }
      50% {
        bottom: 200px;
        animation-timing-function: ease-in;
      }
    }
    div {
      position: absolute;
      width: 20px;
      height: 20px;
      border-radius: 10px;
      animation-name: bounce;
      animation-iteration-count: infinite;
    }
    #b1 {
      left: 10px;
      background: red;
      animation-duration: 5s;
    }
    #b2 {
      left: 50px;
      background: blue;
      animation-duration: 10s;
    }
  
```

```

    #b3 {
      left: 90px;
      background: green;
      animation-duration: 3s;
    }
    #b4 {
      left: 130px;
      background: silver;
      animation-duration: 8s;
    }
    #b5 {
      left: 170px;
      background: black;
      animation-duration: 1s;
    }
  </style>
</head>
<body>
  <div id="b1"></div>
  <div id="b2"></div>
  <div id="b3"></div>
  <div id="b4"></div>
  <div id="b5"></div>
</body>

```

