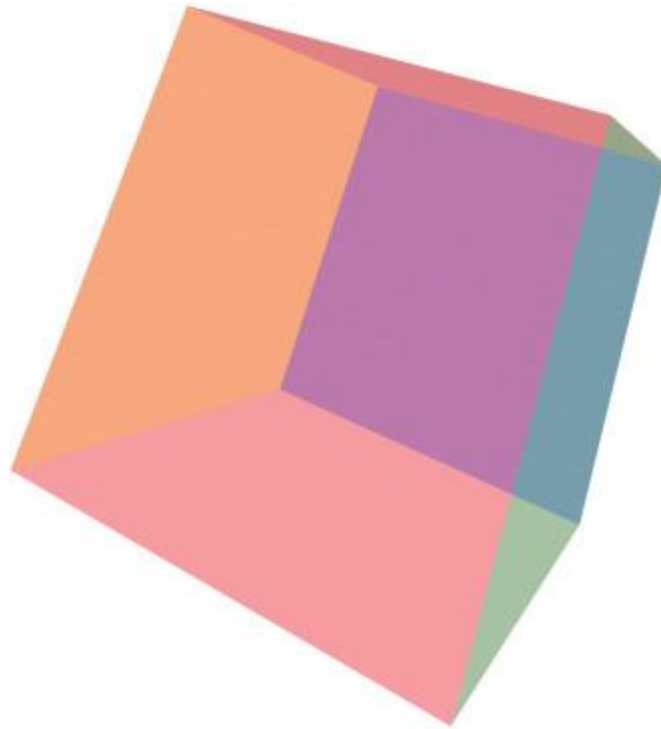




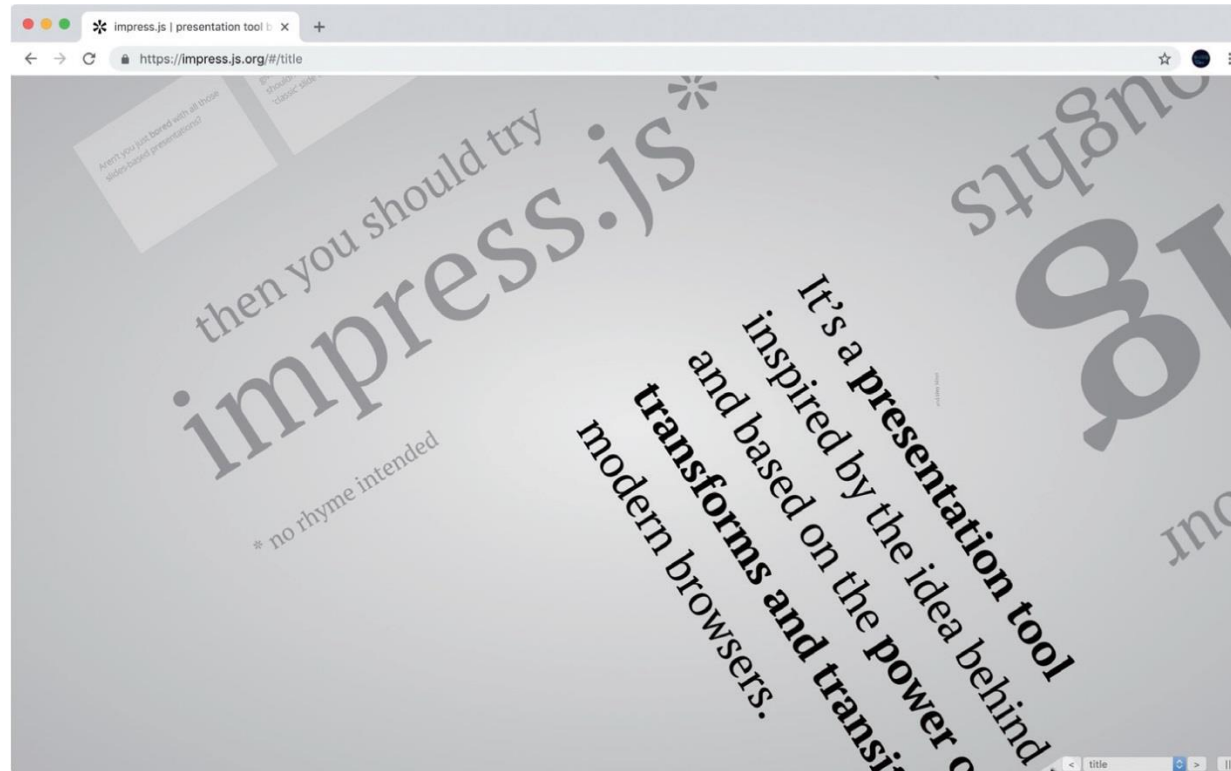
CHAPTER 10 CSS3 변환

CSS3 변환(transform)으로 3D 구현하기



회전하는 정육면체

10.1 변환이란?



- HTML5에서 3차원을 구현하는 방법 2가지: 1. 자바스크립트를 사용한 WebGL / 2. CSS3를 사용한 3차원 변환
- impress.js: HTML을 사용해 프레젠테이션을 만들 수 있게 해주는 플러그인(<https://impress.js.org>)

10.2 2차원 변환 / transform 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Transform Basic</title>
  <style>

    </style>
</head>
<body>
  <section>
    <div></div>
  </section>
</body>
</html>

<style>
  section {
    width: 100px; height: 100px;
    border: 5px solid black;
  }

  div {
    width: 100px; height: 100px;
    background: red;
  }
</style>

div {
  width: 100px; height: 100px;
  background: red;

  transform: rotate(60deg);
}
```



- HTML 페이지 구성
- style 태그 내부에 박스의 너비와 높이를 지정하는 코드 입력
- transform 속성 사용: 코드 실행하면 사각형이 60도 회전

10.2 2차원 변환 / 2차원 변환 함수

변환 함수	설명
<code>translate(translateX, translateY)</code>	특정 크기만큼 이동합니다.
<code>translateX(translateX)</code>	X축으로 특정 크기만큼 이동합니다.
<code>translateY(translateY)</code>	Y축으로 특정 크기만큼 이동합니다.
<code>scale(scaleX, scaleY)</code>	특정 크기만큼 확대 및 축소합니다.
<code>scaleX(scaleX)</code>	X축으로 특정 크기만큼 확대 및 축소합니다.
<code>scaleY(scaleY)</code>	Y축으로 특정 크기만큼 확대 및 축소합니다.
<code>skew(angleX, angleY)</code>	특정 각도만큼 기울입니다.
<code>skewX(angleX)</code>	X축으로 특정 각도만큼 기울입니다.
<code>skewY(angleY)</code>	Y축으로 특정 각도만큼 기울입니다.
<code>rotate(angleZ)</code>	특정 각도만큼 회전합니다.

10.2 2차원 변환 / transform-origin 속성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Transform Basic</title>
  <style>
    section {
      width: 100px;
      height: 100px;
      border: 5px solid black;
    }

    div {
      width: 100px; height: 100px;
      background: red;

      transform: rotate(60deg);
    }
  </style>
</head>
<body>
  <section>
    <div></div>
  </section>
</body>
</html>
```



- transform-origin 속성: 변환 중심을 설정하는 스타일 속성
- transform-origin 속성은 기본적으로 태그 영역의 중심을 변환 중심으로 잡음
- rotate() 변환 함수를 사용했으므로 사각형이 60도 회전: 사각형의 중심을 기준으로 회전
- transform-origin 속성에는 2개의 크기 단위 적용 가능: 각각 변환 중심의 X 좌표와 Y 좌표를 의미
- 객체의 오른쪽 아래가 변환 중심이 되며, 변환 중심을 기준으로 rotate() 함수 실행

10.3 3차원 변환 / 3차원 변환 함수

변환 함수	설명
translate3d (translateX, translateY, translateZ)	특정 크기만큼 이동합니다.
translateX (translateX)	X축으로 특정 크기만큼 이동합니다.
translateY (translateY)	Y축으로 특정 크기만큼 이동합니다.
translateZ (translateZ)	Z축으로 특정 크기만큼 이동합니다.
scale3d (scaleX, scaleY, scaleZ)	특정 크기만큼 확대 및 축소합니다.
scaleX (scaleX)	X축으로 특정 크기만큼 확대 및 축소합니다.
scaleY (scaleY)	Y축으로 특정 크기만큼 확대 및 축소합니다.
scaleZ (scaleZ)	Z축으로 특정 크기만큼 확대 및 축소합니다.
rotate3d (angleX, angleY, angleZ)	특정 각도만큼 회전합니다.
rotateX (angleX)	X축으로 특정 각도만큼 회전합니다.
rotateY (angleY)	Y축으로 특정 각도만큼 회전합니다.
rotateZ (angleZ)	Z축으로 특정 각도만큼 회전합니다.

10.3 3차원 변환 / 3차원 변환 함수

```
<body>
  <section>
    <div></div>
    <div></div>
    <div></div>
    <div></div>
    <div></div>
    <div></div>
  </section>
</body>
```

```
body {
  width: 200px;
  margin: 200px auto;
}
```

```
section {
  width: 200px; height: 200px;
  position: relative;
}
```

```
div {
  width: 200px; height: 200px;
  position: absolute; left: 0; top: 0;
  opacity: 0.3;
}
```

자손의 position 속성에
absolute 키워드를 적용
하면 부모의 position 속
성에 relative 키워드를
적용합니다.

```
/* 옆면 */
div:nth-child(1) { transform: rotateY( 0deg) translate3d(0px, 0px, 100px); }
div:nth-child(2) { transform: rotateY( 90deg) translate3d(0px, 0px, 100px); }
div:nth-child(3) { transform: rotateY(180deg) translate3d(0px, 0px, 100px); }
div:nth-child(4) { transform: rotateY(270deg) translate3d(0px, 0px, 100px); }
```

```
/* 뒷면과 아랫면 */
div:nth-child(5) { transform: rotateX( 90deg) translate3d(0px, 0px, 100px); }
div:nth-child(6) { transform: rotateX(270deg) translate3d(0px, 0px, 100px); }
```

```
div:nth-child(1) {
  transform: rotateY( 0deg) translate3d(0px, 0px, 100px);
  background: red;
}
div:nth-child(2) {
  transform: rotateY( 90deg) translate3d(0px, 0px, 100px);
  background: green;
}

div:nth-child(3) {
  transform: rotateY(180deg) translate3d(0px, 0px, 100px);
  background: blue;
}
div:nth-child(4) {
  transform: rotateY(270deg) translate3d(0px, 0px, 100px);
  background: yellow;
}

div:nth-child(5) {
  transform: rotateX( 90deg) translate3d(0px, 0px, 100px);
  background: brown;
}
div:nth-child(6) {
  transform: rotateX(270deg) translate3d(0px, 0px, 100px);
  background: pink;
}
```

- div 태그 6개를 놓아 면 6개를 만들기
- 이 면 6개를 section 태그로 감싸 하나의 객체로 만들
- div 태그의 position 속성에 absolute 키워드 적용
- left 속성과 top 속성을 사용해 하나로 뭉침
- 각각의 태그를 적당한 각도로 회전시키고 외곽으로 밀기
- 각각의 div 태그에 색상 입력

10.3 3차원 변환 / transform-style 속성

```
section {  
  width: 200px; height: 200px;  
  position: relative;  
  
  animation: rint 3s linear 0s infinite;  
}  
  
@keyframes rint {  
  from {  
    transform: rotateX( 0deg) rotateY( 0deg) rotateZ( 0deg);  
  }  
  
  to {  
    transform: rotateX(360deg) rotateY(360deg) rotateZ(360deg);  
  }  
}  
  
section {  
  width: 200px; height: 200px;  
  position: relative;  
  animation: rint 3s linear 0s infinite;  
  
  transform-style: preserve-3d;  
}
```



- transform-style 속성은 변환을 적용할 때 그 영향력이 자신에게만 적용될지 자손에게도 적용될지 정함
- 자손의 3차원 속성을 유지한 채로 부모를 회전시키고 싶을 때는 부모의 transform-style 속성에 preserve-3d 키워드를 적용해야 함
- section 태그의 transform-style 속성에 preserve-3d 키워드를 적용시키고 코드를 실행하면, 정육면체가 회전

10.3 3차원 변환 / backface-visibility 속성

```
div {  
  width: 200px; height: 200px;  
  position: absolute; left: 0; top: 0;  
  
  backface-visibility: hidden;  
  opacity: 0.3;  
}
```

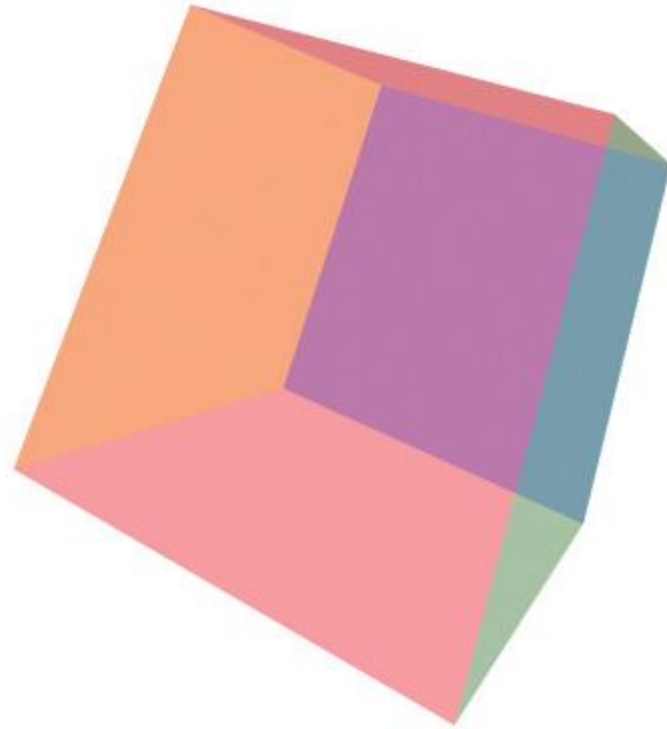


- backface-visibility 속성: 3차원 공간에서 평면의 후면을 보이거나 보이지 않게 만드는 스타일 속성
- div 태그에 backface-visibility 속성에 hidden 키워드를 적용

키워드	설명
visible	후면을 보이게 만듦
hidden	후면을 보이지 않게 만듦

10.4 원근법

```
body {  
  width: 200px;  
  margin: 200px auto;  
  -webkit-perspective: 400;  
}
```



- perspective 속성: 원근법을 지정하며, 화면에 얼마나 많은 3차원 픽셀을 놓을 것인지 정의하는 속성
- 일반적으로 perspective 속성에는 400픽셀에서 2000픽셀 사이의 숫자를 입력함

10.5 회전목마 / body 태그 구성

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Transform Basic</title>
  <style>

  </style>
</head>
<body>

</body>
</html>
```

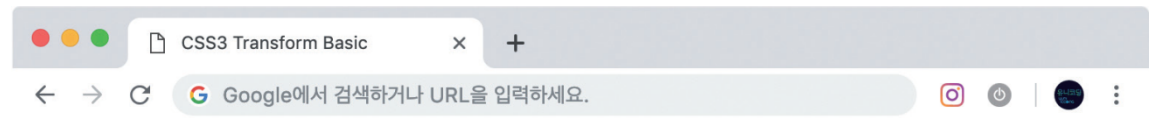
```
<body>
  <h1>HTML5 + CSS3 for Modern Web</h1>
  <h2>CSS3 Transform</h2>
  <p>3D Merry Go Round Gallery</p>
  <hr />
  <div id="canvas">

  </div>
</body>
```

```
<div id="canvas">
  <div id="merry">
    <div class="face"></div>
    <div class="face"></div>
    <div class="face"></div>
    <div class="face"></div>
    <div class="face"></div>
    <div class="face"></div>
```

```
<div class="face"></div>
<div class="face"></div>
<div class="face"></div>
<div class="face"></div>
</div>
</div>
```

```
<div class="face">
  <div class="cell"></div>
  <div class="cell"></div>
  <div class="cell"></div>
  <div class="cell"></div>
</div>
```



HTML5 + CSS3 for Modern Web

CSS3 Transform

3D Merry Go Round Gallery

- HTML 페이지 구성
- 간단한 소개 페이지와 회전 목마를 올려둘 div 태그를 만들고 id 속성값으로 canvas를 입력
- 회전 목마의 열 개수 10개 만들기
- #canvas 태그는 회전 목마 객체를 적당한 위치에 놓기 위해 만든 태그 / #merry 태그는 회전 목마 객체를
- class 속성값으로 face를 부여한 div 태그에 div 태그를 4개씩 놓고 class 속성값으로 cell을 입력

10.5 회전목마 / 스타일 사용

```
.face {
  position: absolute;
  left: 0; top: 0;
  margin-left: -115px;

  backface-visibility: hidden;
  transform-style: preserve-3d;
}

.cell {
  width: 230px; height: 150px;
  margin-bottom: 5px;
  background: url('http://placeholder.it/260x200');
  background-size: 100% 100%;

  transition-duration: 0.5s;
}

.face:nth-child(1) { transform: rotateY( 0deg) translateZ(-370px); }
.face:nth-child(2) { transform: rotateY( 36deg) translateZ(-370px); }
.face:nth-child(3) { transform: rotateY( 72deg) translateZ(-370px); }
.face:nth-child(4) { transform: rotateY(108deg) translateZ(-370px); }
.face:nth-child(5) { transform: rotateY(144deg) translateZ(-370px); }
.face:nth-child(6) { transform: rotateY(180deg) translateZ(-370px); }
.face:nth-child(7) { transform: rotateY(216deg) translateZ(-370px); }
.face:nth-child(8) { transform: rotateY(252deg) translateZ(-370px); }
.face:nth-child(9) { transform: rotateY(288deg) translateZ(-370px); }
.face:nth-child(10){ transform: rotateY(324deg) translateZ(-370px); }

.cell:hover {
  transform: scale(1.2) translateZ(50px);
}
```

- 모든 태그의 margin 속성과 padding 속성을 초기화하고 html 태그와 body 태그의 height 속성도 초기화
- #canvas 태그와 #merry 태그 스타일 사용
- 이어서 회전 목마 내부의 열과 셀의 위치를 잡기

10.5 회전목마 / 애니메이션 적용

```
#merry {  
  position: absolute;  
  left: 50%; top: 30px;  
  
  transform-style: preserve-3d;  
  animation: rint 15s infinite linear;  
}  
  
@keyframes rint  
{  
  from { transform: rotateY(0deg); }  
  to { transform: rotateY(360deg); }  
}
```

- #merry 선택자에 animation 속성을 사용
- 애니메이션 이름은 rint이고 15초 동안 무한 반복
- rint 키 프레임 생성: 키 프레임을 생성하여 15초 동안 한 바퀴를 돌게 만듦

