

# What's New in CSS Effects and Animations

Session 509

**Jing Jin**

Safari & WebKit

These are confidential sessions—please refrain from streaming, blogging, or taking pictures

# Supported Platforms

# Supported Platforms



# You Should Know...

**HTML**

**CSS**

**JavaScript**

# Benefits of CSS

- Compact
- Scalable
- Optimized
- Easy to edit
- Degrades well
- Searchable

# Demo

**Matthew Delaney**  
Safari & WebKit

# What You'll Learn

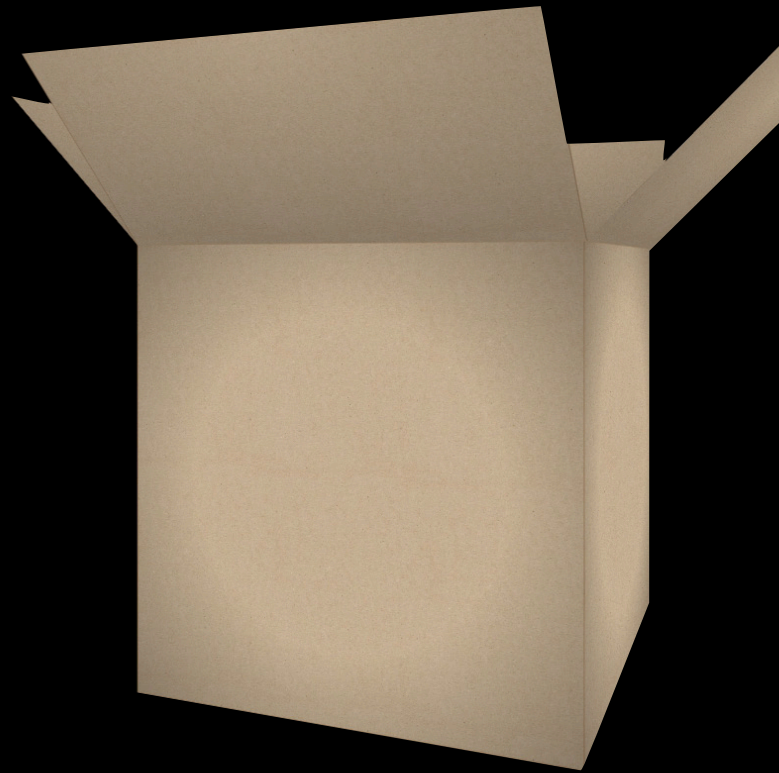
- Add a feel of depth to your content
- Animate with fewer lines of code
- Stay compatible with older browsers

# What You'll Learn

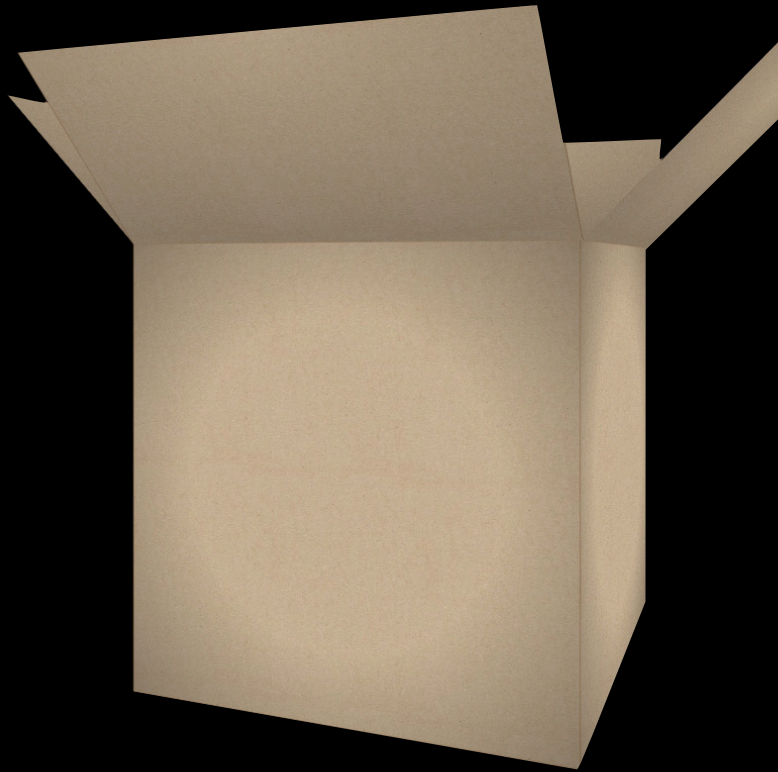
- Add a feel of depth to your content
- Animate with fewer lines of code
- Stay compatible with older browsers



# Conveying Depth with CSS



# Conveying Depth with CSS



**SHIP TO:**

NAME

ADDRESS

ZIP CODE

# Conveying Depth with CSS



# Gradients



```
-webkit-linear-gradient  
-webkit-radial-gradient  
-webkit-repeating-linear-gradient  
-webkit-repeating-radial-gradient
```

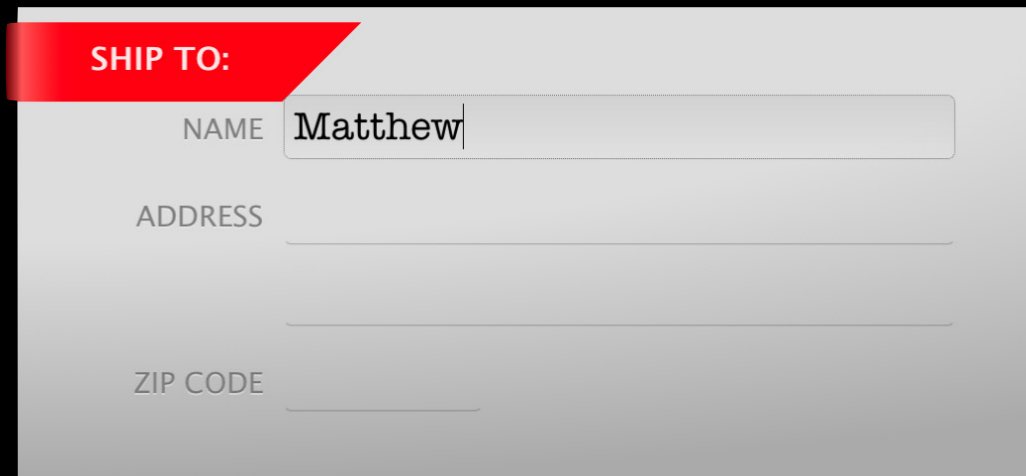
# Gradients

`-webkit-linear-gradient`

`-webkit-radial-gradient`

`-webkit-repeating-linear-gradient`

`-webkit-repeating-radial-gradient`



**SHIP TO:**

NAME

ADDRESS

ZIP CODE

# Linear Gradient

```
.ribbon {  
  background-image:  
    -webkit-linear-gradient(...);  
}
```

# Linear Gradient

```
-webkit-linear-gradient(#900, ← Start color  
red); ← End color
```



# Linear Gradient

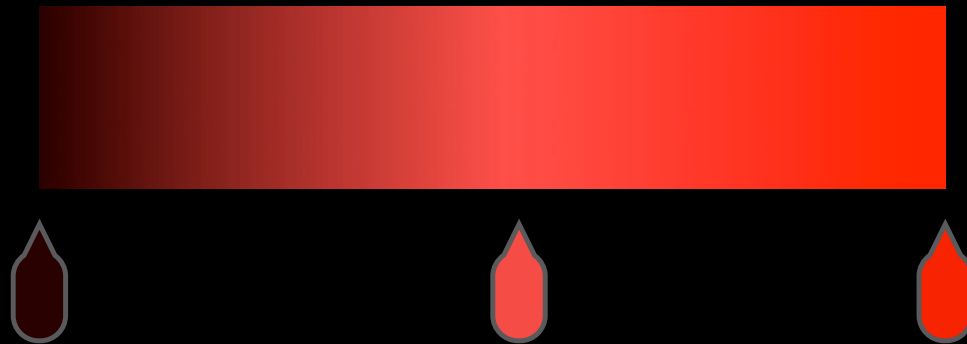
```
-webkit-linear-gradient(left, ← Direction  
    #900,  
    red);
```





# Linear Gradient

```
-webkit-linear-gradient(left,  
    #900,  
    #F33, ← Color Stops  
    red);
```



# Linear Gradient

```
-webkit-linear-gradient(left,  
    #900,  
    #F33 10px, | ← Color Positions  
    red 50px);
```



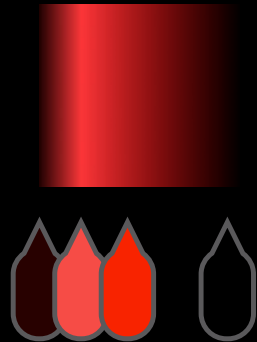
# Linear Gradient

```
.ribbon {  
background-image:  
  -webkit-linear-gradient(  
    left, #900, #F33 10px, red 50px);  
}
```



# Linear Gradient

```
.ribbon {  
background-image:  
  -webkit-linear-gradient(  
    left, #900, #F33 10px, rgba(255, 0, 0, 0) 50px);  
}
```



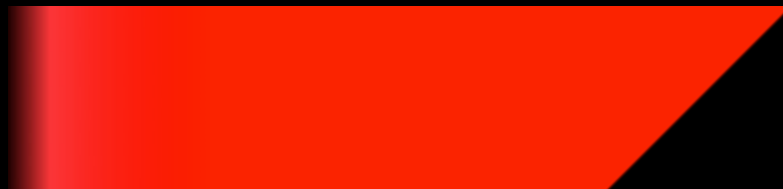
# Linear Gradient

```
.ribbon {  
background-image:  
  -webkit-linear-gradient(  
    left, #900, #F33 10px, rgba(255, 0, 0, 0) 50px),  
  -webkit-linear-gradient( 135deg, transparent 25%, red 25%);  
}
```



# Linear Gradient

```
.ribbon {  
background-image:  
  -webkit-linear-gradient(  
    left, #900, #F33 10px, rgba(255, 0, 0, 0) 50px),  
  -webkit-linear-gradient( 135deg, transparent 25%, red 25%);  
}
```



# Gradients

`-webkit-linear-gradient`

`-webkit-radial-gradient`

`-webkit-repeating-linear-gradient`

`-webkit-repeating-radial-gradient`

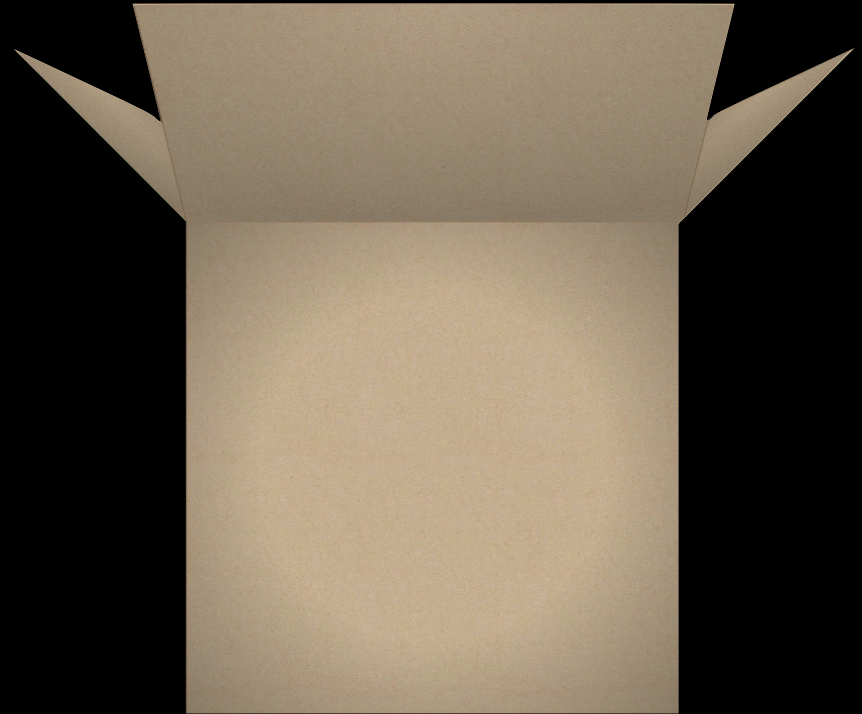
# Gradients

`-webkit-linear-gradient`

`-webkit-radial-gradient`

`-webkit-repeating-linear-gradient`

`-webkit-repeating-radial-gradient`





# Radial Gradient

```
background-image:  
  -webkit-radial-gradient(...);
```

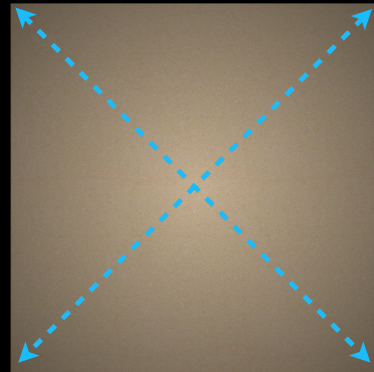
# Radial Gradient

```
background-image: radial-gradient(transparent,  
    -webkit-radial-gradient(circle at (0,0), 0, 0, 0.5));  
url('cardboard.png');
```



# Radial Gradient

```
-webkit-radial-gradient(transparent,  
                        rgba(0, 0, 0, 0.5));
```



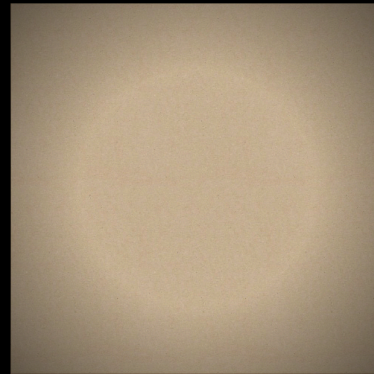
# Radial Gradient

```
-webkit-radial-gradient(transparent 50%,  
                        rgba(0, 0, 0, 0.5));
```

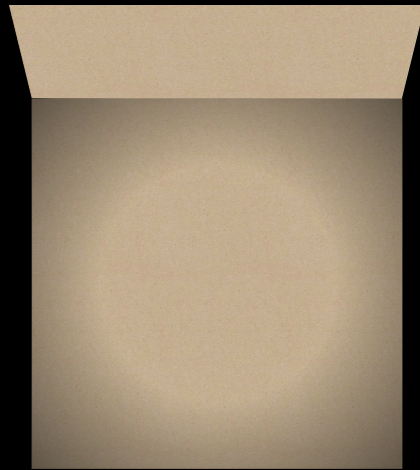


# Radial Gradient

```
-webkit-radial-gradient(transparent 50%,  
                        rgba(0, 0, 0, 0.5));
```

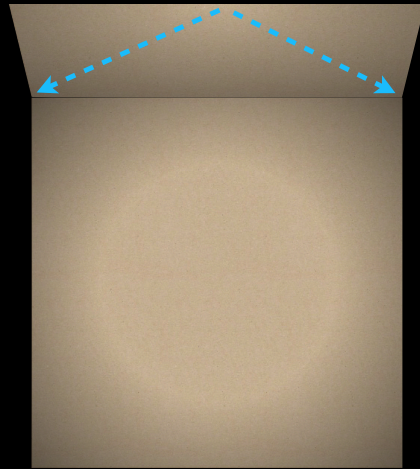


# Radial Gradient



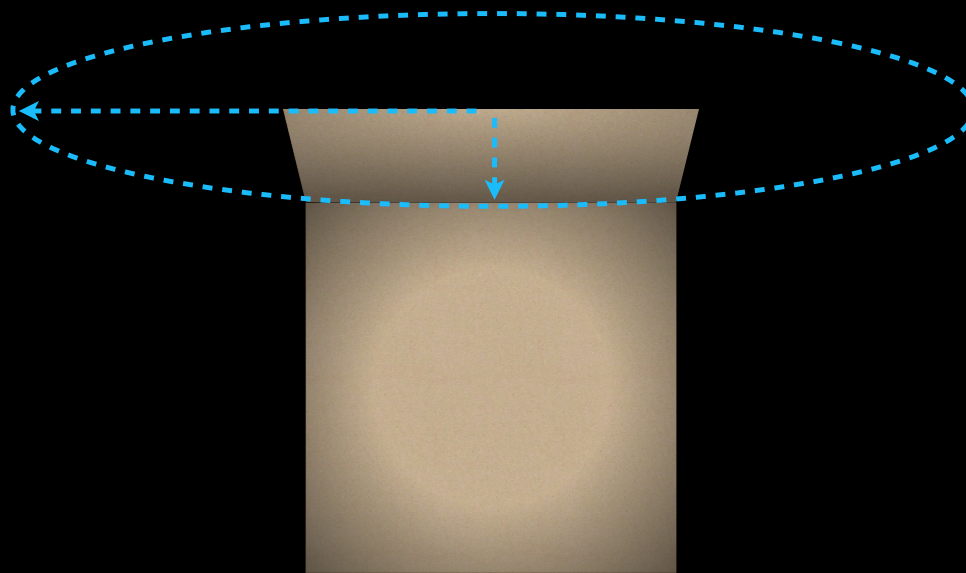
# Radial Gradient

```
-webkit-radial-gradient(top,  
    transparent,  
    rgba(0, 0, 0, 0.5));
```



# Radial Gradient

```
-webkit-radial-gradient(top,  
    100% 100%,  
    transparent,  
    rgba(0, 0, 0, 0.5));
```





# Gradients

`-webkit-linear-gradient`

`-webkit-radial-gradient`

`-webkit-repeating-linear-gradient`

`-webkit-repeating-radial-gradient`

# Gradients

`-webkit-linear-gradient`

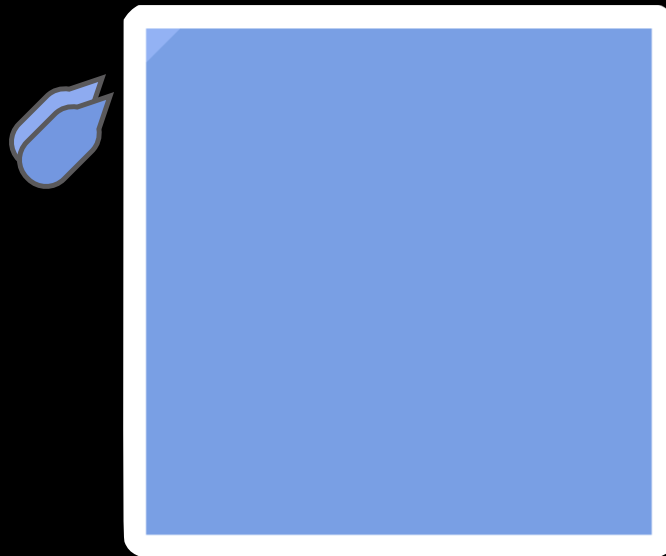
`-webkit-radial-gradient`

`-webkit-repeating-linear-gradient`

`-webkit-repeating-radial-gradient`

# Repeating Linear Gradient

```
-webkit-linear-gradient(top left,  
                        #8EABF0, #8EABF0 5px,  
                        #7397E0 5px, #7397E0 10px)
```



# Repeating Linear Gradient

```
-webkit-repeating-linear-gradient(top left,  
    #8EABF0, #8EABF0 5px,  
    #7397E0 5px, #7397E0 10px)
```



# Repeating Gradients

```
-webkit-radial-gradient(  
    transparent, rgba(0, 0, 0, 0.3) 50px);
```



# Repeating Gradients

```
-webkit-repeating-radial-gradient(  
  transparent, rgba(0, 0, 0, 0.3) 50px);
```

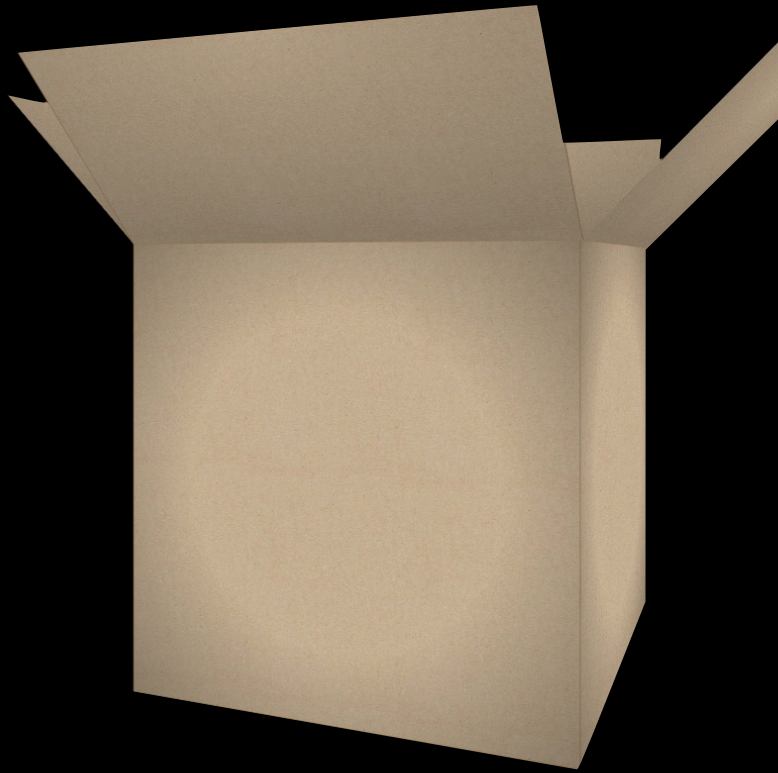


# Gradients



```
-webkit-linear-gradient  
-webkit-radial-gradient  
-webkit-repeating-linear-gradient  
-webkit-repeating-radial-gradient
```

# Conveying Depth with CSS



**SHIP TO:**

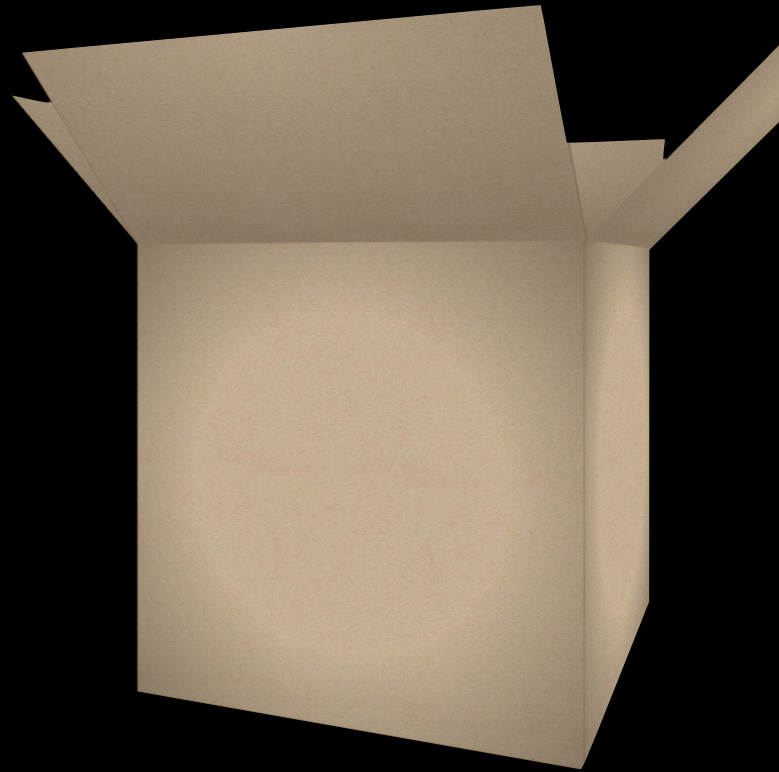
NAME

ADDRESS

ZIP CODE

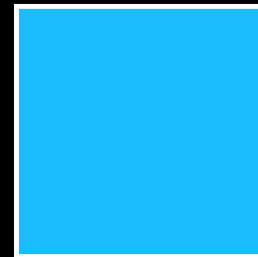
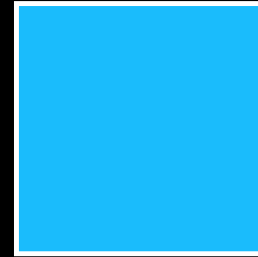


# Conveying Depth with CSS



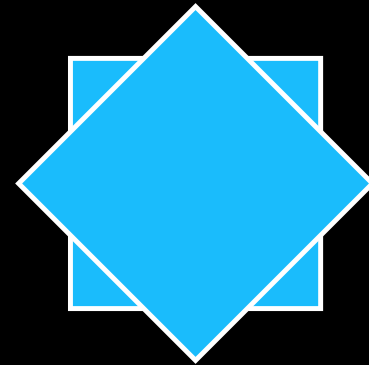
# Transforms

```
-webkit-transform: translateY(200px);
```



# Transforms

```
-webkit-transform: translateY(200px) rotate(45deg);
```



# Building a Box

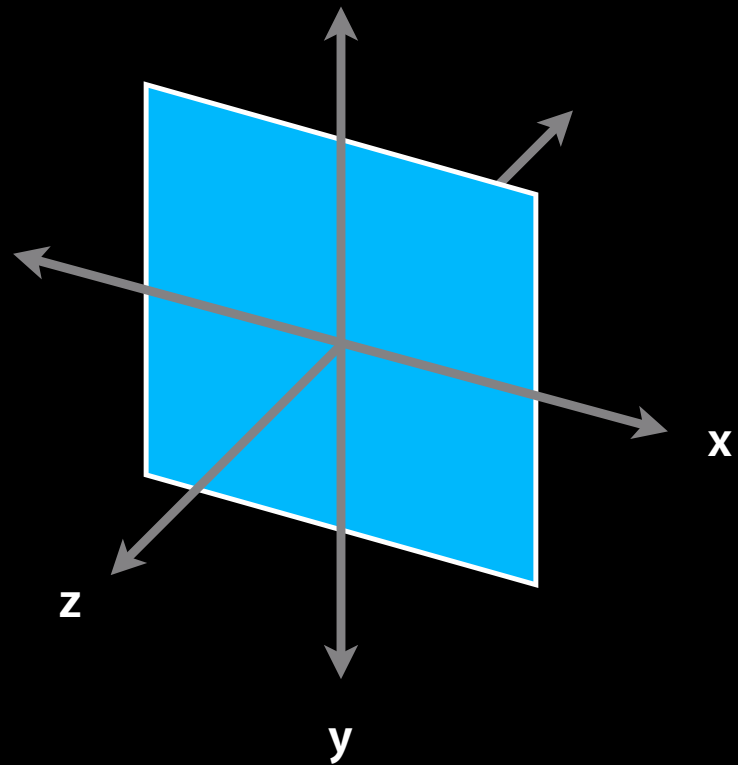
```
<div class="package">  
  <div></div>  
  <div></div>  
  <div></div>  
  <div></div>  
  <div></div>  
</div>
```

# Building a Box

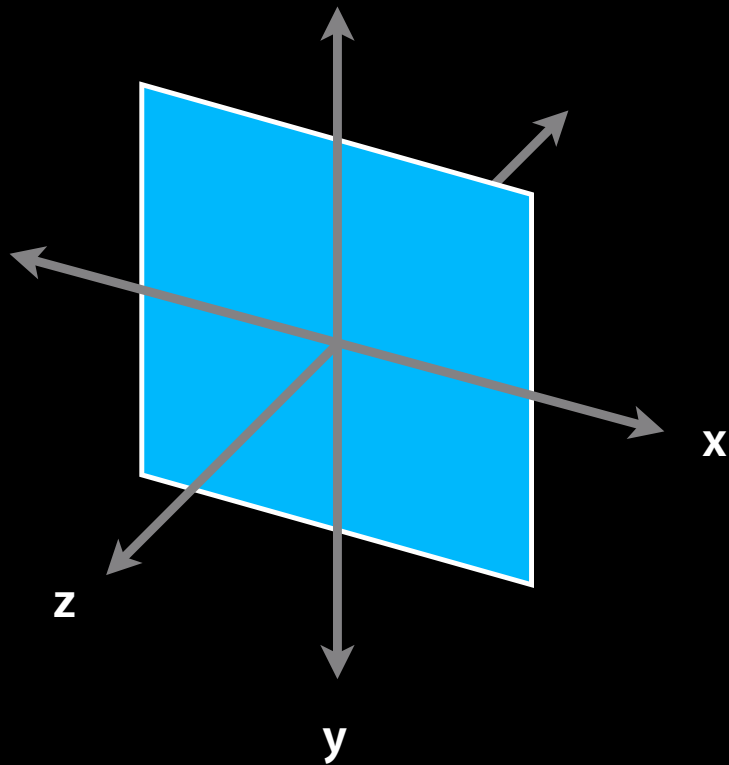


x5

# Building a Box with 3D Transforms

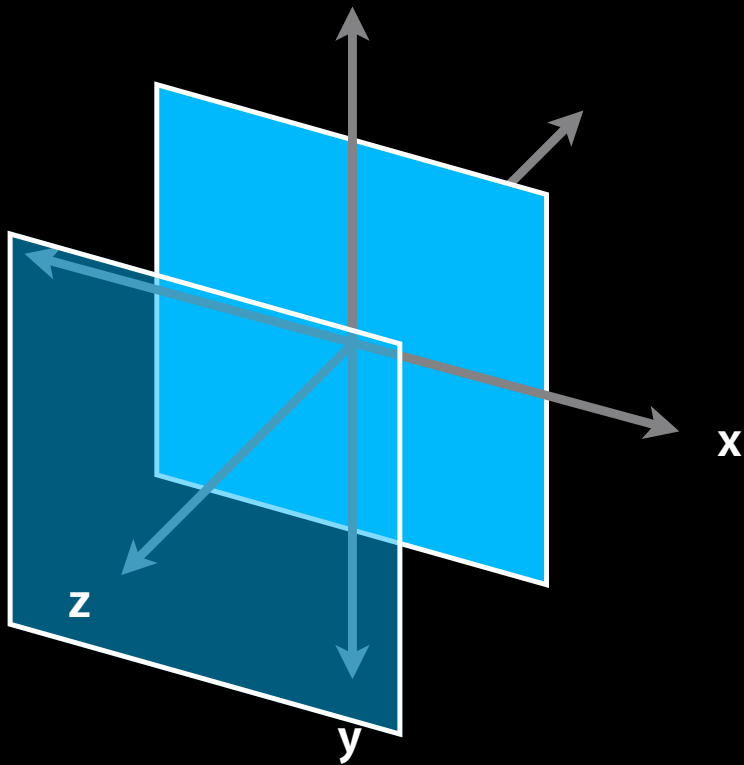


# Building a Box with 3D Transforms



```
.package > div:nth-child(1) {  
  -webkit-transform:  
    translateZ(200px);  
}
```

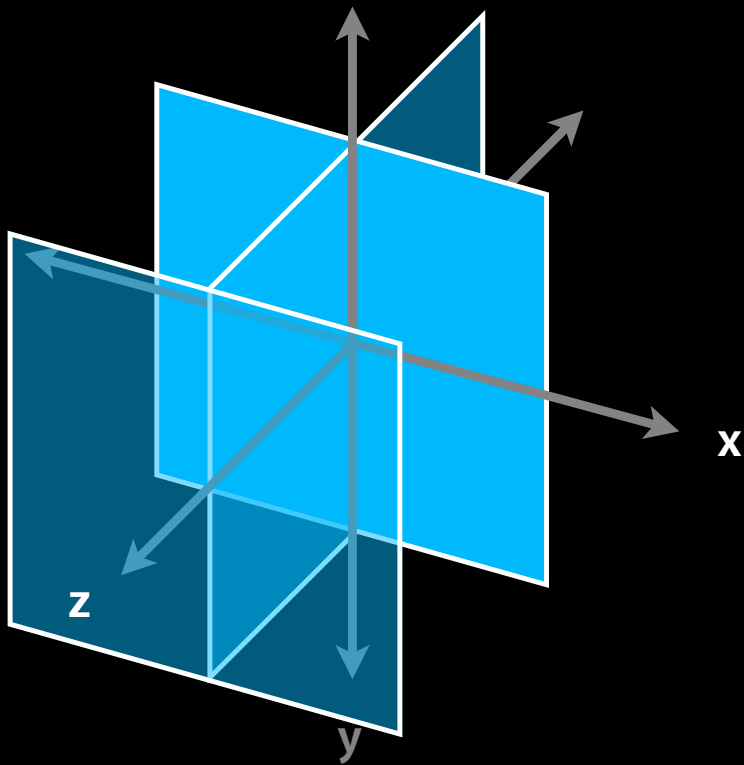
# Building a Box with 3D Transforms



```
.package > div:nth-child(1) {  
  -webkit-transform:  
    translateZ(200px);  
}
```

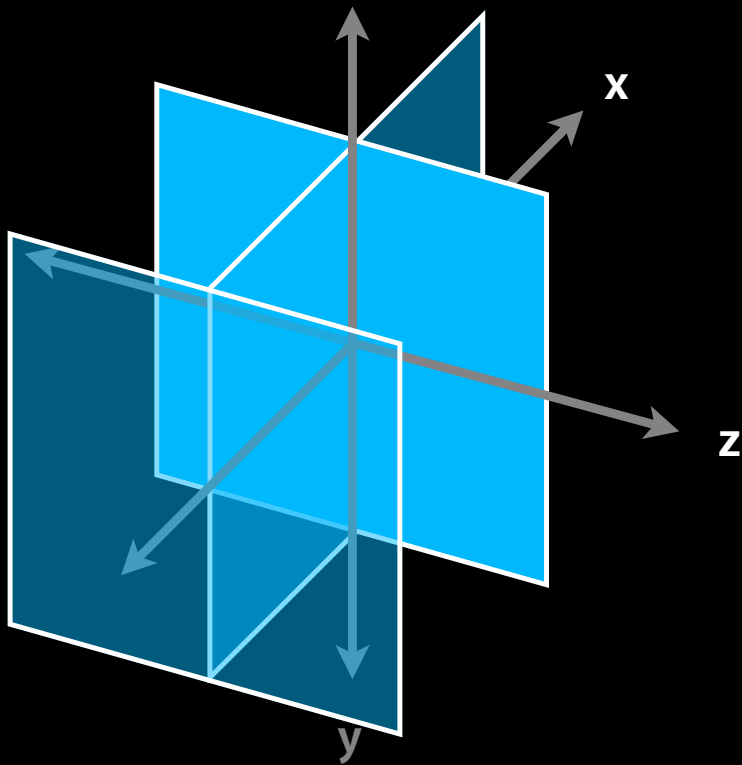


# Building a Box with 3D Transforms



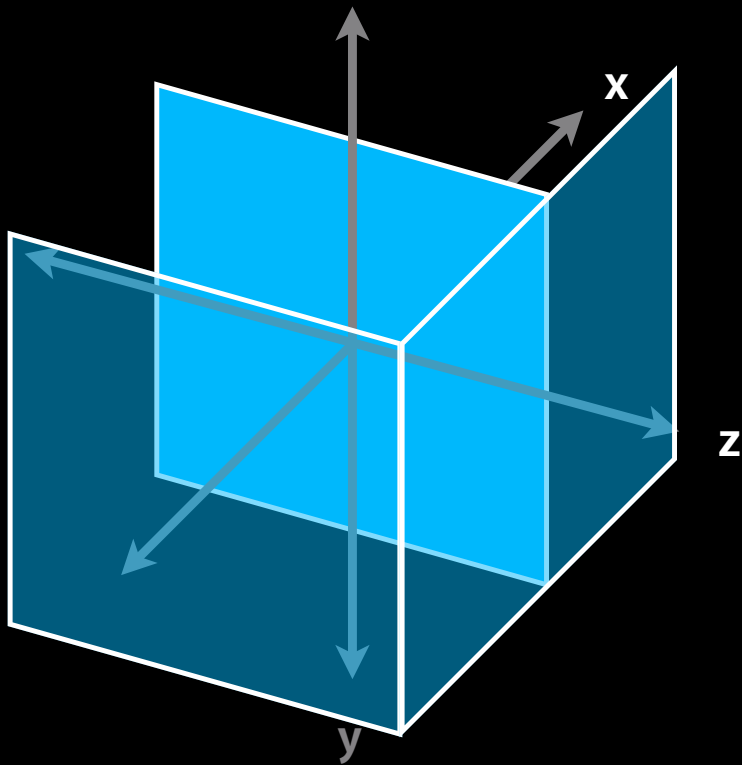
```
.package > div:nth-child(2) {  
  -webkit-transform:  
    rotateY(90deg)  
}
```

# Building a Box with 3D Transforms



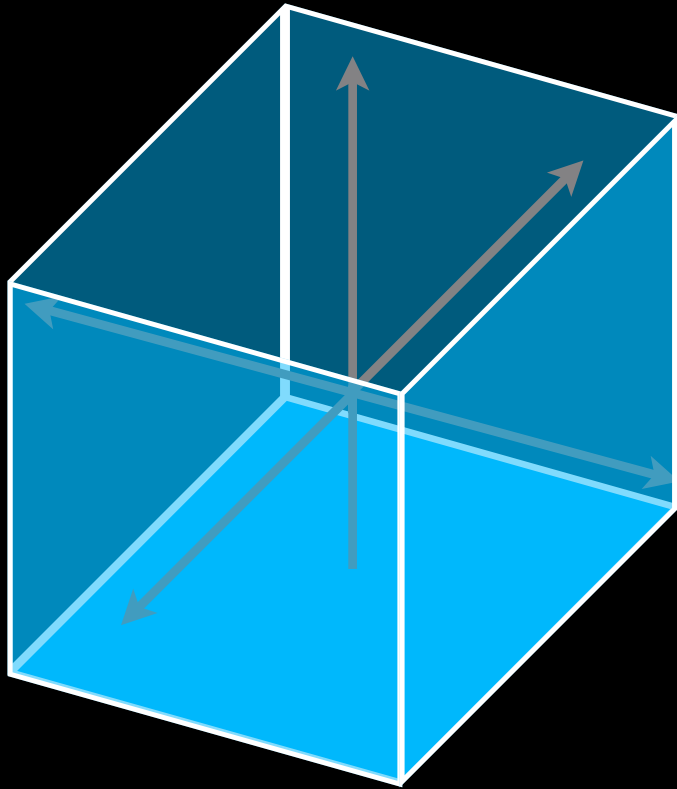
```
.package > div:nth-child(2) {  
  -webkit-transform:  
    rotateY(90deg)  
    translateZ(200px);  
}
```

# Building a Box with 3D Transforms



```
.package > div:nth-child(2) {  
  -webkit-transform:  
    rotateY(90deg)  
    translateZ(200px);  
}
```

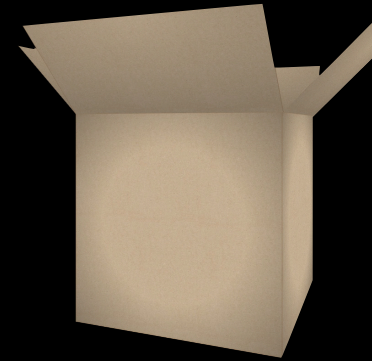
# Building a Box with 3D Transforms



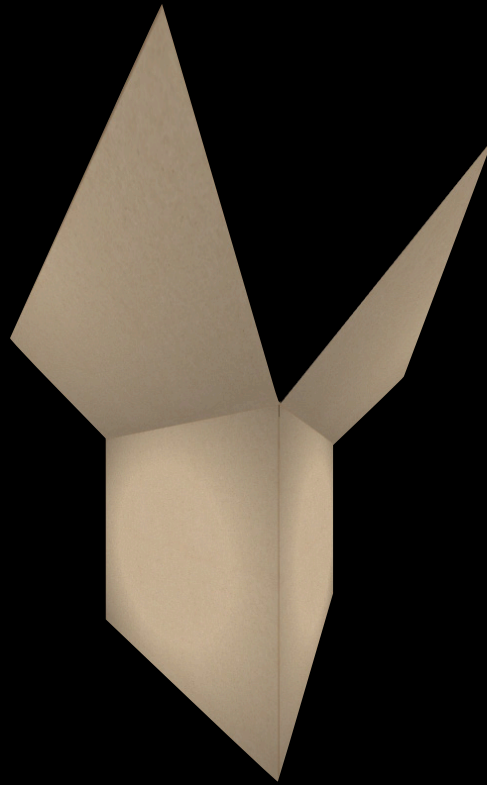
# A 3D Box?



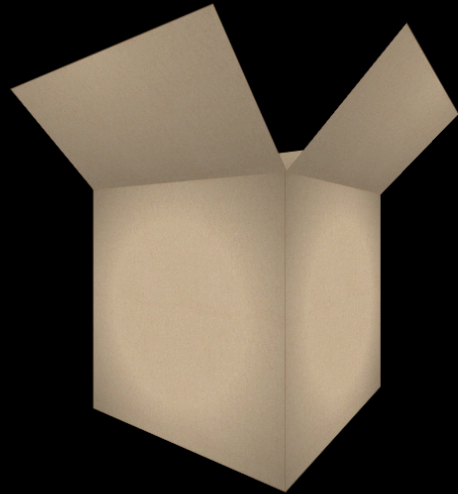
# Perspective



**-webkit-perspective: 400px;**

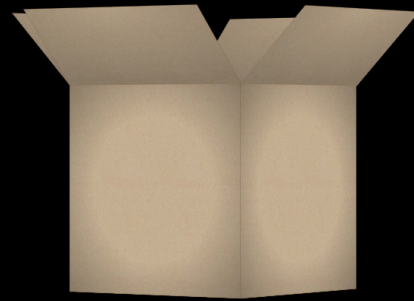


`-webkit-perspective: 800px;`





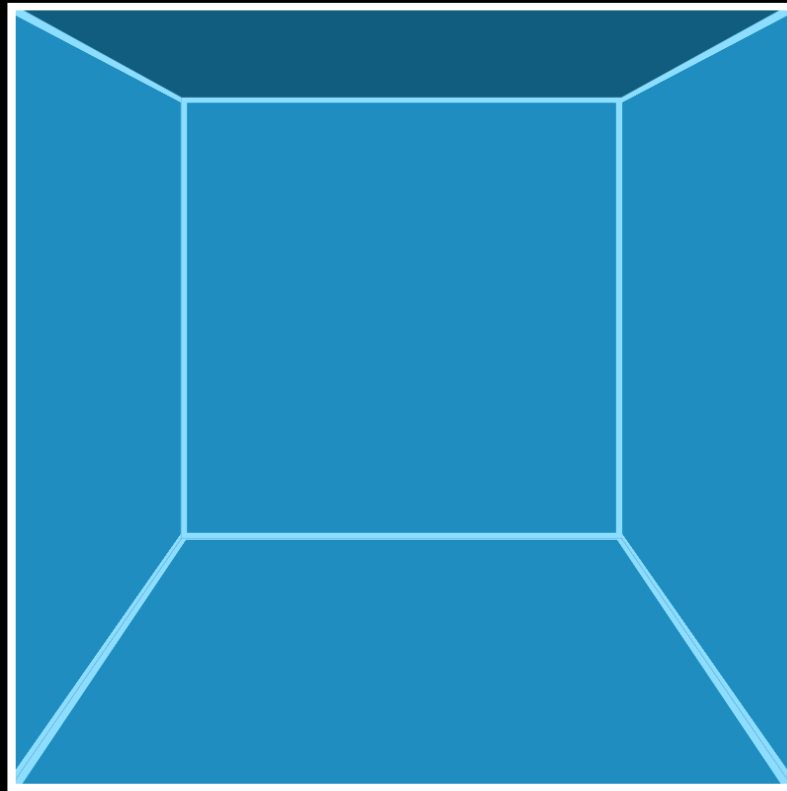
**-webkit-perspective: 8000px;**



# Adding Perspective

```
.package {  
  -webkit-perspective: 800px;  
}
```

# 3D Box with Perspective



# What You'll Learn

- Add a feel of depth to your content
- Animate with fewer lines of code
- Stay compatible with older browsers

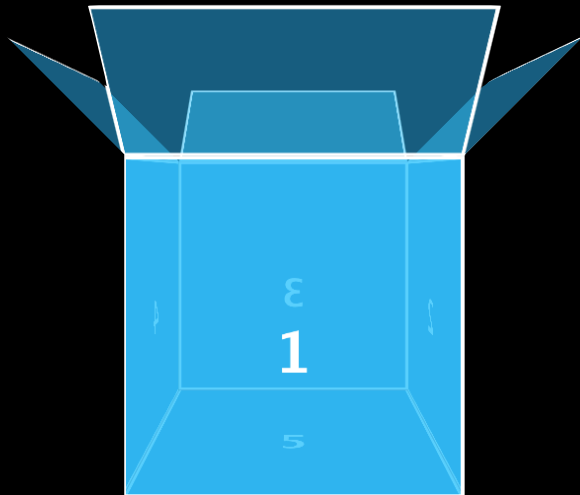
# What You'll Learn

- Add a feel of depth to your content
- Animate with fewer lines of code
- Stay compatible with older browsers

# Animating a Box Rotation

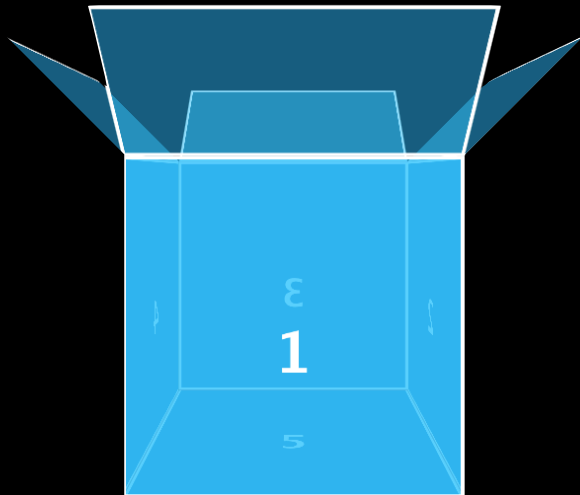


# Rotating a Box



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

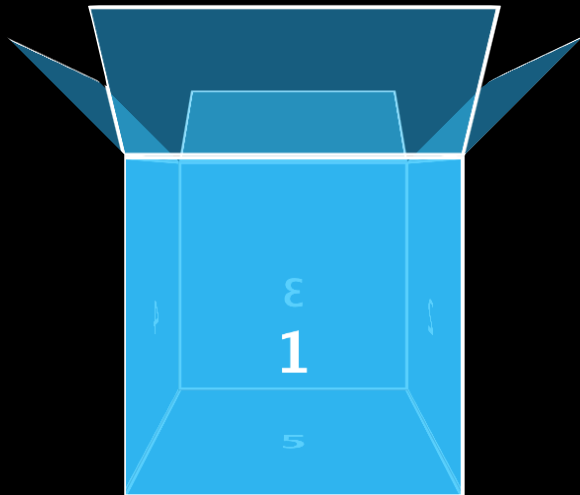
# Rotating a Box



```
<div class="package_container">  
  <div class="package">  
    <div></div>  
    <div></div>  
    <div></div>  
    <div></div>  
    <div></div>  
  </div>  
</div>
```

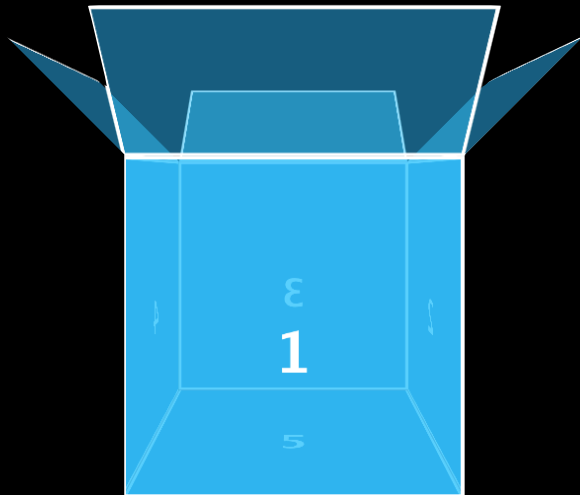


# Rotating a Box



```
.package {  
  -webkit-perspective: 800px;  
}
```

# Rotating a Box



```
.package {
```

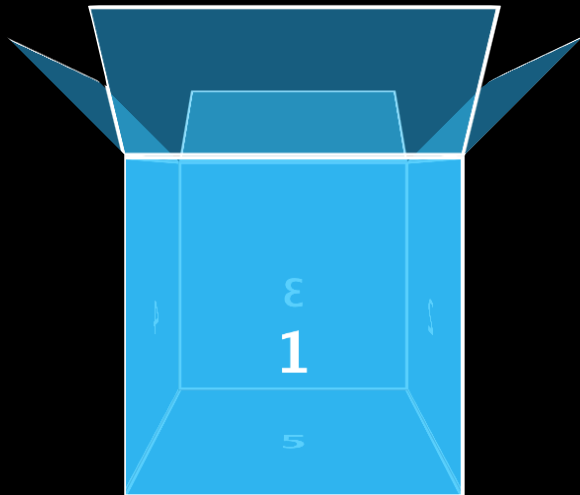
```
}
```

```
.package_container {
```

```
  -webkit-perspective: 800px;
```

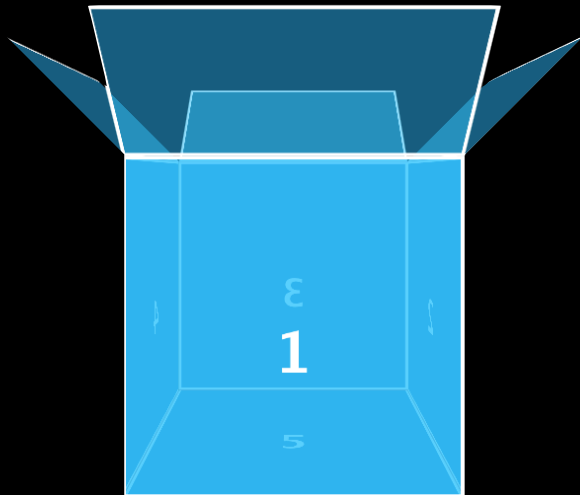
```
}
```

# Rotating a Box



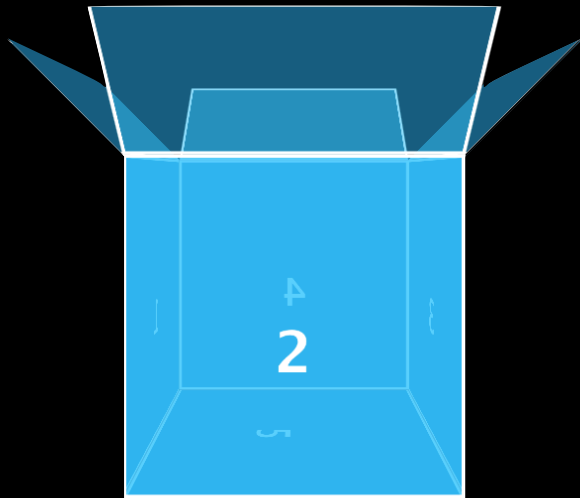
```
.package {  
  -webkit-transform-style: preserve-3d;  
}  
  
.package_container {  
  -webkit-perspective: 800px;  
}
```

# Rotating a Box



```
.package {  
}
```

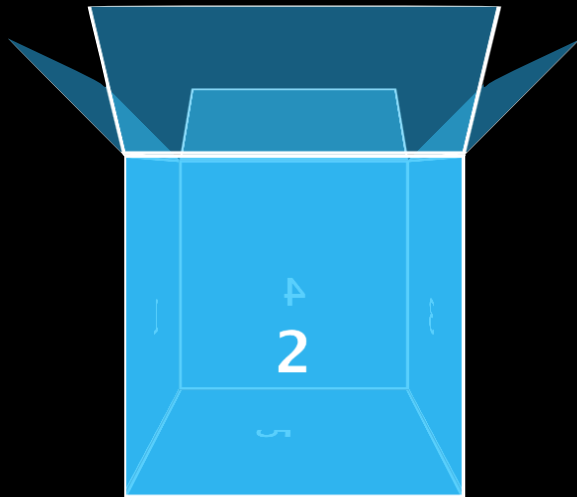
# Rotating a Box



```
.package {  
}  
  
.package.right_side {  
  -webkit-transform: rotateY(-90deg);  
}
```

class="package"right\_side"

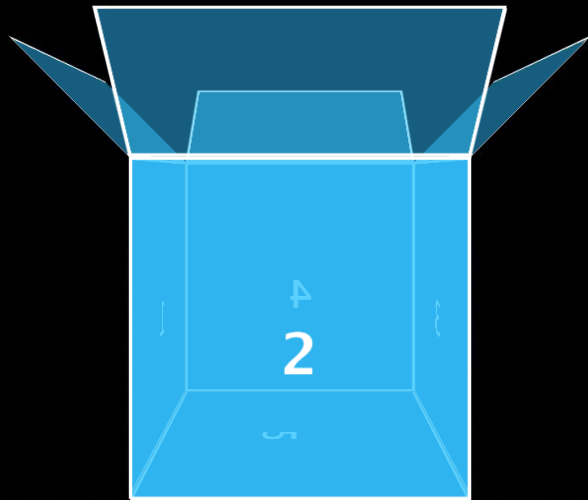
# Rotating a Box



```
.package {  
  -webkit-transition: -webkit-transform  
    1s;  
}  
  
.package.right_side {  
  -webkit-transform: rotateY(-90deg);  
}
```

class="package right\_side"

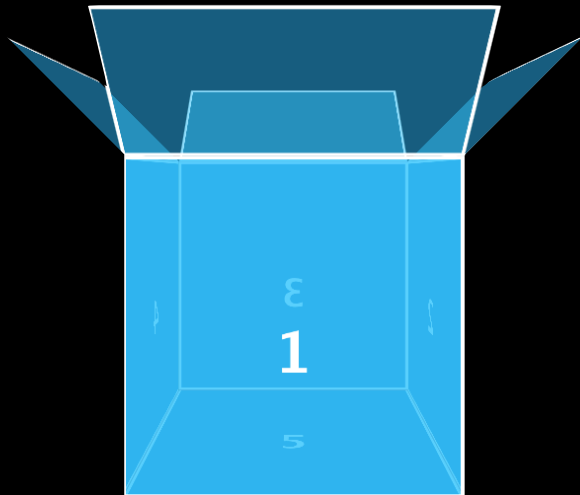
# Rotating a Box



```
.package {  
  -webkit-transition: -webkit-transform  
    1s;  
}  
  
.package.right_side {  
  -webkit-transform: rotateY(-90deg);  
}
```

class="package"right\_side"

# Rotating a Box

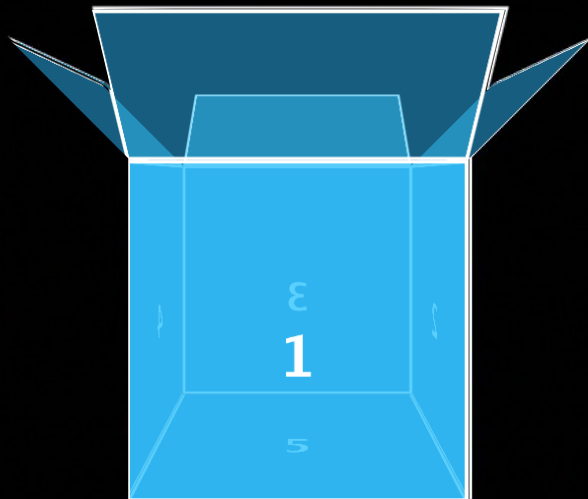


class="package"

```
.package {  
  -webkit-transition: -webkit-transform  
    1s;  
}  
  
.package.inside {  
  -webkit-transform: rotateX(-90deg);  
}
```



# Rotating a Box



class="package"inside"

```
.package {  
  -webkit-transition: -webkit-transform  
    1s;  
}  
  
.package.inside {  
  -webkit-transform: rotateX(-90deg);  
}
```

# Applying a Transition Without Javascript



# Applying a Transition Without Javascript



```
.sticker {  
  background-position: -30px;  
  border-radius: 5px;  
}
```

```
.sticker:hover {  
  background-position: 0;  
  border-bottom-left-radius: 50% 20px;  
  box-shadow: -5px 10px 15px  
             rgba(0, 0, 0, 0.25);  
}
```

# Applying a Transition Without Javascript



```
.sticker {  
  background-position: -30px;  
  border-radius: 5px;  
}
```

```
.sticker:hover {  
  background-position: 0;  
  border-bottom-left-radius: 50% 20px;  
  box-shadow: -5px 10px 15px  
             rgba(0, 0, 0, 0.25);  
}
```

# Transition on Hover



```
.sticker {  
  background-position: -30px;  
  border-radius: 5px;  
  -webkit-transition-property:  
    background-position,  
    border-bottom-left-radius,  
    box-shadow;  
  -webkit-transition-duration: 1s;  
}  
  
.sticker:hover {  
  background-position: 0;  
  border-bottom-left-radius: 50% 20px;
```

# Stamp Animation

When do you want your kittens?



In 5-7 business days

**\$FREE!**



June 10th before 5pm

**\$14.99**



June 9th before noon

**\$24.99**

# Stamp Animation

When do you want your kittens?



In 5-7 business days

**\$FREE!**



June 10th before 5pm

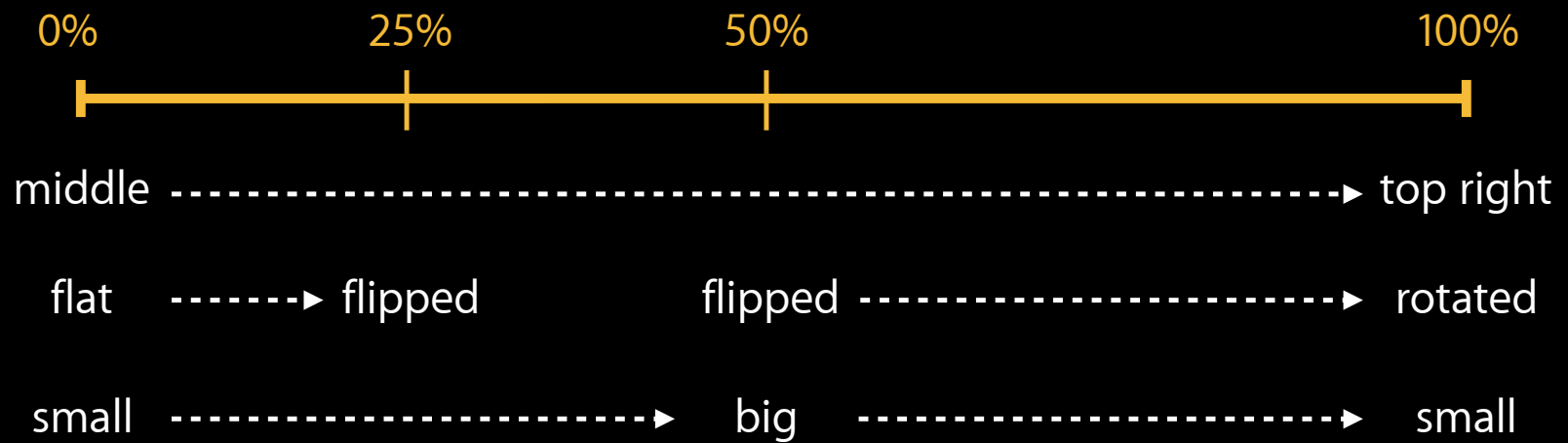
**\$14.99**



June 9th before noon

**\$24.99**

# Stamp Keyframe Animation





# Stamp Keyframe Animation

@-webkit-keyframes

# Stamp Keyframe Animation

```
@-webkit-keyframes stamp-it {
```



Animation name

```
}
```

# Stamp Keyframe Animation

New

```
@-webkit-keyframes stamp-it {  
  25% {  
  
  }  
  50% {  
  
  }  
  100% {  
  
  }  
}
```

# Stamp Keyframe Animation

```
@-webkit-keyframes stamp-it {  
  25% {  
  
  }  
  50% {  
  
  }  
  100% {  
    top: 30px;  
    right: 30px;  
  
  }  
}
```

# Stamp Keyframe Animation

```
@-webkit-keyframes stamp-it {  
  25% {  
    -webkit-transform: translateZ(200px) rotateY(-45deg) rotateX(-30deg);  
  }  
  50% {  
    -webkit-transform: translateZ(300px) rotateY(-45deg) rotateX(-30deg);  
  }  
  100% {  
    top: 30px;  
    right: 30px;  
    -webkit-transform: rotate(-5deg);  
  }  
}
```

# Stamp Keyframe Animation

```
@-webkit-keyframes stamp-it {  
  25% {  
    -webkit-transform: translateZ(200px) rotateY(-45deg) rotateX(-30deg);  
  }  
  50% {  
    -webkit-transform: translateZ(300px) rotateY(-45deg) rotateX(-30deg)  
                      scale(2);  
  }  
  100% {  
    top: 30px;  
    right: 30px;  
    -webkit-transform: rotate(-5deg);  
  }  
}
```

# Stamp Keyframe Animation

```
<style>  
  @-webkit-keyframes stamp-it {...}  
</style>
```

# Stamp Keyframe Animation

```
<style>
  @-webkit-keyframes stamp-it {...}
</style>

<script>
function onClickHandler(event) {
  var stampElement = /* get the element */;
  var stampElement.style.webkitAnimation = "stamp-it 0.5s";
}
</script>
```



# Stamp Keyframe Animation

When do you want your kittens?



In 5-7 business days

**\$FREE!**



June 10th before 5pm

**\$14.99**

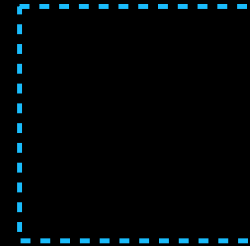


June 9th before noon

**\$24.99**

# Animation Fill Mode

```
@-webkit-keyframes pulse {  
  0% { background-color: orange; }  
  50% { background-color: white; }  
  100% { background-color: orange; }  
}  
  
div.square {  
  width: 100px;  
  height: 100px;  
  -webkit-animation: pulse 2s;  
  -webkit-animation-delay: 1s;  
}
```



# Animation Fill Mode

```
@-webkit-keyframes pulse {  
  0% { background-color: orange; }  
  50% { background-color: white; }  
  100% { background-color: orange; }  
}  
  
div.square {  
  width: 100px;  
  height: 100px;  
  -webkit-animation: pulse 2s;  
  -webkit-animation-delay: 1s;  
}
```

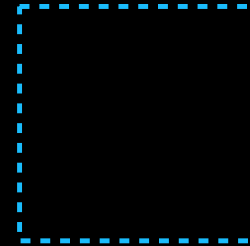
2 second pulse



1 second delay

# Animation Fill Mode

```
@-webkit-keyframes pulse {  
  0% { background-color: orange; }  
  50% { background-color: white; }  
  100% { background-color: orange; }  
}  
  
div.square {  
  width: 100px;  
  height: 100px;  
  -webkit-animation: pulse 2s;  
  -webkit-animation-delay: 1s;  
  -webkit-animation-fill-mode: backwards;  
}
```



# Animation Fill Mode

```
@-webkit-keyframes pulse {  
  0% { background-color: orange; }  
  50% { background-color: white; }  
  100% { background-color: orange; }  
}  
  
div.square {  
  width: 100px;  
  height: 100px;  
  -webkit-animation: pulse 2s;  
  -webkit-animation-delay: 1s;  
  -webkit-animation-fill-mode: backwards;  
}
```

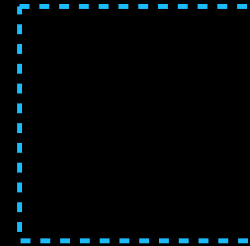
2 second pulse



1 second delay

# Animation Fill Mode

```
@-webkit-keyframes pulse {  
  0% { background-color: orange; }  
  50% { background-color: white; }  
  100% { background-color: orange; }  
}  
  
div.square {  
  width: 100px;  
  height: 100px;  
  -webkit-animation: pulse 2s;  
  -webkit-animation-delay: 1s;  
  -webkit-animation-fill-mode: forwards;  
}
```



# Animation Fill Mode

```
@-webkit-keyframes pulse {  
  0% { background-color: orange; }  
  50% { background-color: white; }  
  100% { background-color: orange; }  
}  
  
div.square {  
  width: 100px;  
  height: 100px;  
  -webkit-animation: pulse 2s;  
  -webkit-animation-delay: 1s;  
  -webkit-animation-fill-mode: forwards;  
  
}
```

2 second pulse



1 second delay

# Adding a Fill Mode

```
<style>
  @-webkit-keyframes stamp-it {...}
</style>

<script>
function onClickHandler(event) {
  var stampElement = /* code to get the element */;
  var stampElement.style.webkitAnimation = "stamp-it 0.5s forwards";
}
</script>
```



# Keyframe Animation with a Fill Mode

When do you want your kittens?



In 5-7 business days

**\$FREE!**



June 10th before 5pm

**\$14.99**



June 9th before noon

**\$24.99**

# Progress Indicator



# Progress Indicator



```
@-webkit-keyframes outer {  
  0% {  
    -webkit-transform: rotate(0);  
  }  
  100% {  
    -webkit-transform: rotate(360deg);  
  }  
}
```

# Progress Indicator



```
@-webkit-keyframes inner {  
  0% {  
    -webkit-transform: rotate(5);  
  }  
  100% {  
    -webkit-transform: rotate(365deg);  
  }  
}
```

# Progress Indicator



```
.outer_paw  
{  
-webkit-animation: outer 5s infinite;  
}
```

```
.inner_paw  
{  
-webkit-animation: inner 5s 0.4s infinite;  
}
```

# Stepped Timing Function

New



```
.outer_paw
{
-webkit-animation: outer 5s infinite;
-webkit-animation-timing-function:
                        steps(10);
}
```

```
.inner_paw
{
-webkit-animation: inner 5s 0.4s infinite;
-webkit-animation-timing-function:
                        steps(10);
}
```

# Transitions vs. Keyframe Animations

# Applying a Transition

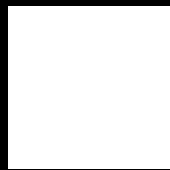


```
.box  
{  
  background-color: white;  
  -webkit-transition: opacity 1s;  
}
```

```
.box:hover  
{  
  background-color: orange;  
}
```



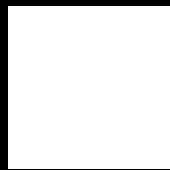
# Applying a Keyframe Animation



```
.box
{
  background-color: white;
  -webkit-animation: fade 1s;
}

@-webkit-keyframes fade
{
  from { background-color: white; }
  from { background-color: orange; }
}
```

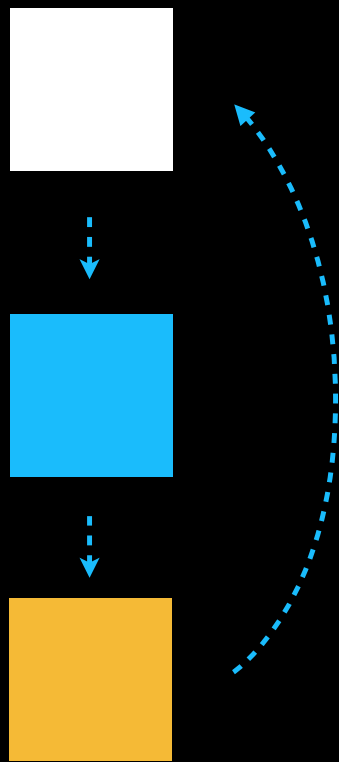
# Keyframe Animation with Multiple Keyframes



```
.box
{
  background-color: white;
  -webkit-animation: fade 1s;
}

@-webkit-keyframes fade
{
  from { background-color: white; }
  50% { background-color: cyan }
  to { background-color: orange; }
}
```

# Repeating Keyframe Animation



```
.box
{
  background-color: white;
  -webkit-animation: fade 1s;
  -webkit-animation-iteration-count:
    infinite;
}

@-webkit-keyframes fade
{
  from { background-color: white; }
  50% { background-color: cyan }
  to { background-color: orange; }
```

# Alternating Keyframe Animation



```
.box
{
  background-color: white;
  -webkit-animation: fade 1s;
  -webkit-animation-iteration-count:
    infinite;
  -webkit-animation-direction:
    alternate;
}

@-webkit-keyframes fade
{
  from { background-color: white; }
```

**Rule of Thumb:  
Use a Transition If Possible**

# Demo

**Matthew Delaney**  
Safari & Webkit

# What You'll Learn

- Add a feel of depth to your content
- Animate with fewer lines of code
- Stay compatible with older browsers

# What You'll Learn

- Add a feel of depth to your content
- Animate with fewer lines of code
- Stay compatible with older browsers



# CSS Media Queries

- Types of media
- Properties of the display area
- Support for CSS properties

# CSS Media Queries

- Types of media
- Properties of the display area
- Support for CSS properties

# Determining the Type of Medium

## Using HTML

```
<link rel="stylesheet" media="screen" href="animated.css" />
```

```
<link rel="stylesheet" media="print" href="static.css" />
```

# Determining the Type of Medium

## Using CSS

```
@media print
{
    div { position: relative !important; }
    .animated_truck { display: none; }
}
```

# Determining the Type of Medium

## Using Javascript

```
if (window.styleMedia && window.styleMedia.type == "screen")  
{  
    // insert code here  
}
```

# CSS Media Queries

- Types of media
- Properties of the display area
- Support for CSS properties

# CSS Media Queries

- Types of media
- Properties of the display area
- Support for CSS properties

# Querying Properties of the Display Area

## Using HTML

```
<link rel="stylesheet"  
      media="screen and (min-width: 800px)"  
      href="columnated.css" />
```



# Querying Properties of the Display Area

## Using CSS

```
@media screen and (min-width:800px)
{
    .article { -webkit-column-count: 2; }
}
```

# Querying Properties of the Display Area

## Using Javascript

```
if (window.styleMedia &&  
    window.styleMedia.matchMedium("screen and (min-width: 800px)"))  
{  
    // insert code here  
}
```

# CSS Media Queries

- Types of media
- Properties of the display area
- Support for CSS properties

# CSS Media Queries

- Types of media
- Properties of the display area
- Support for CSS properties

# Detecting Supported CSS Properties

```
var container = document.getElementById("container");

if (window.styleMedia &&
    window.styleMedia.matchMedium("(-webkit-animation)"))
{
    container.style.webkitAnimation = "fly-out-and-fade 1s";
}
```

# Everything We Covered

- Create depth with gradients and 3D transforms
- Animate with transitions and keyframe animations
- Use CSS media queries for compatibility

# More Information

## Vicki Murley

Safari Technologies Evangelist  
[vicki@apple.com](mailto:vicki@apple.com)

## 2010 CSS Presentations

CSS Effects, Part 1: UI Elements and Navigation

CSS Effects, Part 2: Galleries and 3D Effects

<https://developer.apple.com/videos/wwdc/2010/>

## Safari Dev Center

<http://developer.apple.com/safari/>

# Related Sessions

What's New in Safari Extensions

Marina  
Thursday 2:00PM

Using HTML5 Offline Storage

Marina  
Thursday 3:15PM

Understanding Layout and Gestures in Safari on iOS and Lion

Marina  
Thursday 4:30PM



# Labs

CSS Effects Lab

Internet and Web Lab B  
Thursday 9:00AM

