

Week 08

Friday, June 11, 2021 1:56 PM

Transforms

We can manipulate an element's appearance using **transform functions**. The value of the transform property is one or more transform functions (separated by spaces) that will be applied in the order they're provided.

Translation functions allow you to move elements left, right, up, or down

Transforms and Older Browsers

Transforms require vendor prefixing for IE9, Android up to 4.4.3, iOS8, and Blackberry 10. To make the aforementioned code work in IE9 and older mobile WebKit browsers, you would include the following:

```
-webkit-transform: translate(45px,-45px); /* iOS8, Android  
4.4.3, BB10 */  
-ms-transform: translate(45px,-45px); /* IE9 only */  
transform: translate(45px,-45px);
```

Scaling

The **scale(x,y)** function scales an element by the defined factors horizontally then vertically. If only one value is provided, it will be used for both the x and y values, growing or shrinking your element or pseudo-element while maintaining the original aspect ratio. For example, scale(1) would leave the element the same size, scale(2) would double its proportions, scale(0.5) would halve them, and so on. Providing different values will distort the element, as you'd expect:

```
transform: scale(1.5, 0.25);
```

As with translate, you can also use the **scaleX(x)** or **scaleY(y)** functions. These functions will scale only the horizontal dimensions or only the vertical dimensions respectively.

To declare multiple transformations, provide a space-separated list of transform functions. We simply add our scale to the end of that space-separated list.

It's also worth remembering that scaling, like translation, has no impact on the document flow.

Rotation

The rotate() function rotates an element around the point of origin by a specified angle value. As with scale, by default the point of origin is the element's center. Generally, angles are declared in degrees, with positive degrees moving clockwise and negative moving counterclockwise. In addition to degrees, values can be provided in grads, radians, or turns, but we'll just be sticking with degrees.

Skew

The `skew(x,y)` function specifies a skew along the x and y axes. As you'd expect, the x specifies the skew on the x axis, and the y specifies the skew on the y axis. If the second parameter is omitted, the skew will only occur on the x axis:

```
transform: skew(15deg, 4deg);
```

The order of transform functions does matter: if you rotate before translating, your translate direction will be on the rotated axis.

The **transition-timing-function** lets you control the pace of the transition in even more granular detail.

Transition Delay Property

Finally, by using the transition-delay property, it's possible to introduce a delay before the transition begins. Normally a transition begins immediately, so the default is 0. Include the number of milliseconds (ms) or seconds (s) to delay the transition. We don't want our transition to start immediately, because that might be a bad user experience if the user accidentally mouses through our ad on the way from one part of the document to the next. A 50ms delay is enough time to wait to be sure they are intentionally hovering over our advertisement:

```
-webkit-transition-delay: 50ms;  
transition-delay: 50ms;
```

Canvas

With HTML5's Canvas API, we can draw anything we can imagine, all through JavaScript. This can improve the performance of our websites by avoiding the need to download images off the network. With canvas, we can draw shapes and lines, arcs and text, gradients and patterns. In addition, canvas gives us the power to manipulate pixels in images and even video

SVG

SVG stands for **Scalable Vector Graphics**, a specific file format that allows you to describe vector graphics using XML. A major selling point of vector graphics in general is that, unlike bitmap images (such as GIF, JPEG, PNG, and TIFF), vector images preserve their quality even as you blow them up or shrink them down. We can use SVG to do many of the same tasks we can do with canvas, including drawing paths, shapes, text, gradients, and patterns.

