Animation

with Guy Routledge

@guyroutledge | #FEWD

Agenda

- Moving things
- Effects
- Transitions
- Animation

Moving things

Review: position

- `relative`
- `absolute`
- `fixed`

Moving things

We can also move things by transforming them

- `rotate`
- `skew`
- `scale`
- `translate`

Moving things

These values are all applied via the **`transform`** property.

```
transform: rotate(10deg);
transform: skew(-5deg);
transform: scale(1.5);
transform: translate(20px, 100px)
```

Moving things

You can even move things in 3D and when combined with `perspective` you can create some really awesome effects.

http://www.atozcss.com/y

Effects

Effects

A lot of visual styling used to have to be done with images. But now, we can do loads of cool stuff in CSS. This makes making things faster and more flexible, and we can even animate these effects too!

Effects

Some commonly used effects include:

- box-shadow
- text-shadow
- gradients
- filters
- blend modes

Box Shadow

Add a shadow underneath the boundry box of an element

```
.box {
  /* x-offset y-offset blur colour */
  box-shadow: 10px 10px 20px rgba( 0, 0, 0, 0.5 );
}
```

Text Shadow

Add a shadow underneath each letter in a text element

```
.title {
  /* x-offset y-offset blur colour */
  text-shadow: 10px 10px 20px rgba( 0, 0, 0, 0.5 );
}
```

Gradients

Create a gradient which blends from one colour to another

```
.gradient {
  background: linear-gradient( #000, #fff );
}
.gradient {
  /* directions like to left, to right */
  background: linear-gradient( to right, #000, #fff );
}
.gradient {
  /* angles */
  background: linear-gradient( 40deg, #000, #fff );
}
```

Filters

We can filter images (like an image editing package) with the **`filter`** property

```
.filter {
    filter: blur( 20px )
}
```

Filters

There are lots of available filters

```
filter: blur(5px);
filter: brightness(0.4);
filter: contrast(200%);
filter: drop-shadow(16px 16px 20px blue);
filter: grayscale(50%);
filter: hue-rotate(90deg);
filter: invert(75%);
filter: opacity(25%);
filter: saturate(30%);
filter: sepia(60%);

/* Apply multiple filters */
filter: contrast(175%) brightness(3%);
```

Blend Modes

Blend modes may be familiar from graphics packages like Photoshop. We can layer two elements and then change the way they blend together.

These are very new and not very well supported but are very cool :)

Blend Modes

We can either blend background images or individual elements

```
.layers {
    background-blend-mode: multiply;
}
.elements {
    mix-blend-mode: multiply;
}
```

Blend Mode example

Blend modes in more detail including examples **https://css-tricks.com/basics-css-blend-modes/**

Transitions

Transitions

We can style different states using pseudo classes such as

- `:hover`
- `:focus`
- `:active`

Transitions

Instead of a jarring on/off of these different states, we can transition between the changing values to provide a smoother effect.

Transitions

```
.box {
    background:red;
    transition:all 1s ease;
}
.box:hover {
    background:blue;
}
```

Transitions

Transitions are made up of 3 things:

- The properties to transition (all by default)
- The duration of the transition
- The acceleration curve of the transition

Animations

Animations

For a long time, all animation had to be done in Javascript

CSS Animations

Now some types of animation can be done in CSS. This is good because:

- GPU accelerated
- Keeps presentation stuff with other presentation stuff
- No need to learn Javascript ;0)

Animation Screencast

http://www.atozcss.com/css-keyframe-animations/

CSS Animation Syntax

```
.bouncing-ball {
    animation: bounce 2s 500ms infinite ease-in-out;
}
@keyframes bounce {
    0% { top:0; }
    50% { top:100%; }
    100% { top:0; }
}
```

CSS animation syntax

There is also a keyword syntax for keyframes

```
@keyframes moveUp {
    from { top:0; }
    to   { top:100%; }
}
```

CSS Animation properties

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

Vendor prefixes

Because animations (and a lot of the effects we've been looking at today) are new and experimental, they were introduced to browsers with vendor specific prefixes

- `-webkit`
- `-moz`
- `-o`
- `-ms`

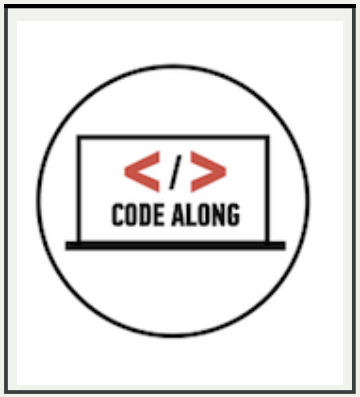
Vendor prefixes

To ensure everything works correctly in every different browser, you'll need to check which prefixes are required and what features are supported where.

The place to go for all that info is **http://www.caniuse.com**

Vendor prefixes

```
@-webkit-keyframes moveUp {
    from { top:0; }
    to   { top:100%; }
}
@keyframes moveUp {
    from { top:0; }
    to   { top:100%; }
}
.something-that-moves {
    -webkit-animation: moveUp 1s infinite;
            animation: moveUp 1s infinite;
}
```



Loading animation