Animation Essentials

CSS Animations and Transitions

with @RachelNabors & Tuna P. Katz

Get started with “SMASHINGROCKS” at goo.gl/ZL0SVd
It's true.
You will learn...

- CSS Transitions
- CSS Animations
- Sequencing Animations
- How to Design Performant Animations
- And some handy browser tools for working with animations
For your convenience...

All of the exercises use:

• jQuery

• and -prefix-free
Our Teacher’s Assistant Today
CSS Transitions
A transition describes how a property should display change when given a different value.
transition: color 2s;
transition-property: color;
transition-duration: 2s;
Anatomy of a transition

- **transition-property** the property you want to transition. (Only some properties are transitionable, see goo.gl/Ttk1S2)

- **transition-duration** in seconds or milliseconds: 4s or 4000ms

- **transition-timing-function** “cushioning” for the transition, **optional**: defaults to ease

- **transition-delay** the number of milli/seconds to delay the transition before firing it, **optional**
transition: color 2s 100ms;

so this must be the delay

duration always comes first
Transitioning Multiple Properties
transition-property: all;

Don't do it!!
transition-property: color, transform;
transition-duration: 2s, 300ms;
transition-delay: 0, 1s;

redundant, bleh
transition:

color 2s,

transform 300ms 1s;

-> separate sets of values with commas
exercise

Rolling a Ball

Start coding! cdpn.io/fGFDj

Try transitioning its color, too.

Keep up at:
Want to share? Save & tweet codepen link to @rachelnabors
Duration
Three timeframes for user attention

- 100 ms, instantaneous
- 1 second, still connected
Three timeframes for user attention

- 100 ms
- 1 second
- 10 seconds, disconnected
ONLINE WORKSHOPS

* Apr 2
  UIE
  Improve UX with Animation

* Jun 30
  *Smashing Online*
  Animation Essentials: CSS Animations and Transitions

Open Enrollment
Learn with Rachel Nabor
Practical Cartooning for Technical Folk

AVAILABLE FOR TRAINING IN HOUSE AND AT CONFERENCES

rachelnabors.com
“However long your pre-production animation, halve its duration... then halve it again.”

- Studio animation rule of thumb
WEB ANIMATION AND MOTION DESIGN TRAINING AND WORKSHOPS

ONLINE WORKSHOPS

April 2
UIE
Improve UX with Animation

June 30
Smashing Online
Animation Essentials: CSS Animations and Transitions

Open Enrollment
Learn with Rachel Nabors
Practical Cartooning for Technical Folk

AVAILABLE FOR TRAINING IN HOUSE AND AT CONFERENCES

I teach from a place of creative coding and my rich history as an award-winning cartoonist. As a web designer turned front-end developer, I connect the silos of design, UX, and programming wherever I teach. This empathy for all parts of the human equation required for building a better web lets me engage all
250~300 ms: sweet spot for many animations
faster != better
WEB ANIMATION AND MOTION DESIGN
TRAINING AND WORKSHOPS

ONLINE WORKSHOPS

🌟 Apr 20  
UIE
Improve UX with Animation

🌟 Apr 25  
Smashing Online
Animation Essentials: CSS Animations and Transitions

🌟 Open Enrollment — Learn with Rachel Nabors
Practical Cartooning for Technical Folk

AVAILABLE FOR TRAINING IN HOUSE AND AT CONFERENCES

I teach from a place of creative coding and my rich history as an award-winning cartoonist. As a web designer turned front-end developer, I connect the silos of design, UX, and programming wherever I teach. This empathy for all parts of the human equation required for building a better web lets me engage all...
You can slow down transitions with Dev Tools to get a good look at ‘em.
Timing Functions
Easing, also known as “cushioning” in studio animation, describes an animation’s rate of change over time.
Timing Function Values

- linear
- ease-in
- ease-out
- ease-in-out
- steps (more on this later)
Replace “ease” with “slow” to better remember the different timing effects.
What you really want: cubic-bezier

Make your own at cubic-bezier.com or edit with the Chrome/Firefox Dev Tools!
**Easing functions** specify the rate of change of a parameter over time.

Objects in real life don't just start and stop instantly, and almost never move at a constant speed. When we open a drawer, we first move it quickly, and slow it down as it comes out. Drop something on the floor, and it will first accelerate downwards, and then bounce back up after hitting the floor.

This page helps you choose the right easing function.
exercise
Applying Physics to the Ball

Start coding! cdpn.io/LVEEdXg
Experiment with different easing and durations to find the combination that feels “right” to you.

Keep up at:
Want to share? Save & tweet codepen link to @rachelnabors
Any thing but IE 9 and lower & Opera Mini

caniuse.com/#feat=css-transitions
CSS Transitions rock because...

- **Single fire** If you only want something to happen once.
- **Granularity** If you would only animate one or two properties in a given state
Questions?
CSS Animations
.animated-thing {
  animation: black-to-white 1s linear 1;
}

@keyframes black-to-white {
  0% { background: #000; }
  100% { background: #fff; }
}

.animated-thing {
  animation:
    $name
    $duration
    $timing-function (optional)
    $animation-delay (optional)
    $iteration-count;
}
Long form animation properties

- **animation-name:** The name of the keyframe block you want to use.

- **animation-duration:** How long the animations takes to go from 0% to 100%.

- **animation-timing-function:** Like `transition-timing-function`.

- **animation-delay:** The number of seconds to delay the animation instead of playing right away.

- **animation-iteration-count:** The number of times you want to go from 0% to 100%; use `infinite` to never stop. Defaults to 1.
.animated-thing {
  animation:
    $name
    $duration
    $direction
    $fill-mode
    $play-state;
}
**Long form advanced animation properties**

- **animation-direction:** defaults to *normal* but can be set to *alternate, reverse, alternate-reverse*
  
- **animation-fill-mode:** Defaults to *backwards*, reverting to pre-animation. Can be set to *forwards*, assuming the end result of the animation, or *both* (a freakish combination?).

- **animation-play-state:** defaults to *running* but can be set to *paused.*
exercise

Wag the Cat

Start coding! cdpn.io/AfDBF

Play with Tuna’s tail! Change the default advanced CSS animation properties.

Keep up at:
Want to share? Save & tweet codepen link to @rachelnabors
The @keyframes block
@keyframes black-to-white {
  0% {
    background: #000;
    color: #fff;
  }
  100% {
    background: #fff;
    color: #000;
  }
}
+@keyframes black-to-white {
  from {
    background: #000;
    color: #fff;
  }
  to {
    background: #fff;
    color: #000;
  }
}
@keyframes black-to-red {
  0% { color: #000; }
  50% { color: red; }
  100% { color: #000; }
}

@keyframes black-to-red {
    0%, 100% { color: #000; }
    50% { color: red; }
}
@keyframes black-to-red {
  0%, 100% {
    animation-timing-function: linear;
  }
  50% { color: red; }
}
.animated-thing {
  animation:
    black-to-white 1s linear 1,
    black-to-red 2s ease-out infinite 2s;
}
Sprite animation with CSS
Meet `steps()`

- **steps(x)** is a timing function...
- ...splits a block of keyframes into x equal steps, then hops between them.
- The documentation by Tab Atkins
- How it works, plus gotchas: cdpn.io/zeFqy
exercise
Making a Walk-Cycle

The sprite: stash.rachelnabors.com/animation-workshop/sprite_catwalk.png

Start coding! cdpn.io/cdqga

Keep up at:
Want to share? Save & tweet codepen link to @rachelnabors
Browser Support for CSS Animations

Not available on IE 9 and lower & Opera Mini.

Chrome, Safari and Android all require `-webkit-` at this time.

Consult caniuse.com/#feat=css-animation
CSS Animations

- **Looping** Can loop infinitely
- **Self starting** Doesn’t require trigger like transition
- **Repeating** You can set how many time it repeats
- **Alternating** Can alternate between the end state and start state
- **Grouping** Each animation can change a number of properties
Questions?
Sequencing
.sequenced-thing {
  animation:
    sequence1 1s 2,
    sequence2 3s 2s 1 forwards;
}

the 2nd time measurement = delay
exercise

Sitting Tuna Down

Using animation-delay, start a sitting animation after the walking animation.

Start coding! cdpn.io/Djcyxa

Keep up at:
Want to share? Save & tweet codepen link to @rachelnabors
But stacking isn’t the only way, nor the most reliable.
CSS animations and transitions run on the system’s clock.
We’re gonna need a more robust technique.
Chaining Animations with Event Listeners
Four must-have JavaScript event listeners

- animationstart
- animationend
- animationiteration
- transitionend
They will fire for every animation/transition on the element.
Browser prefixes requires (of course)

- `webkitAnimationIteration` for Webkit
- `MSAnimationIteration` for IE 10
- Everyone else is `animationiteration`
exercise

Sitting Tuna Down with event listeners

Using `animationend`, add the `.sit` class after Tuna’s done walking

Start coding! [cdpn.io/pIiqj](http://cdpn.io/pIiqj)

Keep up at:

Want to share? Save & tweet codepen link to @rachelnabors
Questions?
Performance
### Change from default

<table>
<thead>
<tr>
<th>Property</th>
<th>Change from default</th>
<th>Update value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>align-content</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>align-items</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>align-self</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>backface-visibility</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-attachment</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-blend-mode</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-clip</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-color</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-image</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-origin</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-position</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-repeat</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
<tr>
<td><code>background-size</code></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
<td><img src="icons/status.default.png" alt="Status" /> <img src="icons/status.changed.png" alt="Status" /> <img src="icons/status.triggered.png" alt="Status" /></td>
</tr>
</tbody>
</table>

### SELECT A PROPERTY

Choose a property from the list to get more details.
The two CSS properties that don’t trigger layout or repaints: opacity, transform
Handy alternatives

- `transform: scale()` instead of `width` or `height`
- `transform: translate()` instead of `position`
- `opacity` instead of `z-index` or `visibility: hidden`
exercise

Walk the cat... again!

Start coding! cdpn.io/QbwXry

Find a way to walk the cat without using expensive background positioning.

Keep up at:
Want to share? Save & tweet codepen link to @rachelnabors
will-change
Hardware acceleration

AKA “kicking it to the GPU”:

```
.resource-sink {
    transform: translateZ(0);
}
```

Please don’t do this.
Hacks are not future-friendly.
.in-view .animated-thing {
    will-change: transform;
}

.in-view .animated-thing:hover {
    transform: translateY(5%);
}
Don’t slap will-change on everything.

When we will-change everything, we optimize nothing.
**Browser Support for CSS Transitions**

[caniuse.com/#feat=will-change](https://caniuse.com/#feat=will-change)
Chrome Dev Tools
1. Open your Developer Tools.

2. Press the Escape Key to bring up a console window.

3. Choose the Rendering tab.

4. Check the Show FPS meter option.
Paint Rectangles

To enable Chrome's paint rectangles:

1. Open your Developer Tools.
2. Press the Escape Key to bring up a console window.
3. Choose the Rendering tab.
4. Check show paint rectangles under Rendering.
The Timeline Tool

To use Chrome’s timeline tool:

1. Open your Developer Tools.
2. Go to the Timeline tab.
3. Press the “record circle.”
4. Do stuff.
5. Press the “record circle” again to stop.
6. Inspect!
Best perf tip ever:
Don’t animate it if it isn’t visible.
Questions?
the end
Let’s be animation friends!
@RachelNabors & RachelNabors.com WebAnimationWeekly.com