Fall 2017 — ar589.github.io

Week 8 Interactive Design

Considering Animation

Animiation Mechanics

Why would you want to animate something?

Types of Animation

Transitions

Take users from place to place in the information space.

Transition them out of one task into another.

Supplements

Bring information on or off the page, but don’t change the user’s “location” or task.

They generally add or update bits of additional content on the page.

Feedback

Indicates causation between two or more events, often used to connect a user’s interaction with the interface’s reaction.

Demonstrations

Explain how something works or expose its details by showing instead of telling.

Decorations

Do not convey new information and are purely aesthetic.

Considering Animation

Make it Meaningful

- Connect it to your information architecture.
- Connect it to the message.
- Connect it to the brand.

Animation Mechanics

Main Factors of an Animation

Properties

What visual aspects you want to change, like width or color.

Duration

How long the animation should last, often in fractions of a second.

Easing

The rate at which the visual changes occur—for instance, going from slow to fast.

An Abreviated List of Animatable Properties

- all
- background
- background-color
- background-position
- background-size
- border
- border-radius
- border-color
- bottom
- box-shadow
- filter
- font-size
- height
- left
- letter-spacing
- line-height
- margin
- max-height
- max-width
- min-height
- min-width
- opacity
- outline
- outline-color
- outline-width
- padding
- right
- text-shadow
- top
- transform
- width
- z-index

Transform

The best css property to animate!

```
.myDiv {
  transform: translateX(90px);
  transform: skew(15deg);
  transform: rotate(90deg) scale(2);
}
```

Transitions

Transition

Define how an HTML element transitions between states. Must be on the “start state.”

```
.myDiv {
  transition-property: all;
  transition-duration: 1s;
  transition-timing-function: linear;
  transition-delay: 500ms;
  transition: all 1s linear 500ms;
}
```

Transitions animate from a start state to an end state.

Transition Example

The transition property must be set on the start state.

```
/* Start State */
.fade {
  color: magenta;
  transition: all 1s ease-out;
}

/* End State */
.fade:hover {
  color: cyan;
}
```

Animations

CSS Animations

- Can handle more complexity that transitions.
- Based on “keyframes”
- Are defined with the “@keyframes” rule.

@keyframes

Animations are named so you can use them later.

```
@keyframes multi-fade {
  0% {
    color: magenta;
  }
  50% {
    color: cyan;
  }
  100% {
    color: yellow;
  }
}
```

Timing is controlled on the element that uses the animation.

Animation Properties

```
.multi-fade {
  animation-name: multi-fade;
  animation-duration: 2s;
  animation-timing-function: ease-out;
  animation-delay: 400ms;
  animation-iteration-count: 2;
}
```

Events

Animations can run on page load or you can listen for browser events.

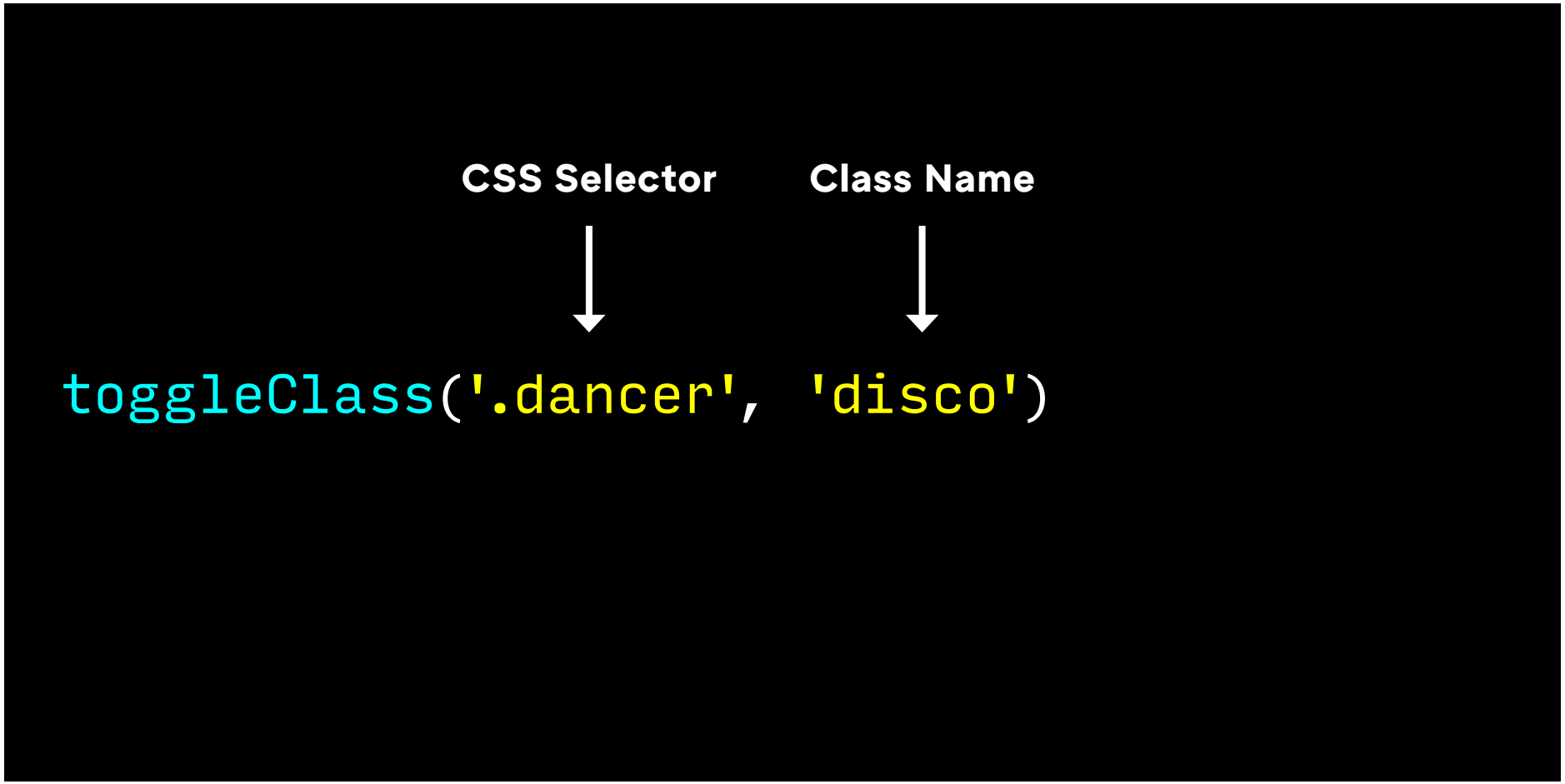
:hover

Run an animation when the selected element is hovered.

```
.jiggle:hover {
  animation-name: jiggle-it;
  animation-duration: 50ms;
  animation-iteration-count: infinite;
  animation-direction: alternate;
}
```

Toggle a class on click

A handy JS function (that I made for you) that will toggle a class of your choosing on the CSS selector of your choosing.



Design Challenge

- Find some good examples of animation in use.
- Create a button with an animated hover state using the transition property.
- Animate a verb using animations. For example, bounce, jump, spin, etc.

Try not to make anyone barf.

Assignment 2 Progress

- Turn your sitemap and content inventory into a functional prototype with actual content in HTML and CSS.