Agenda

1. Grading + rubric
2. Better understanding CSS
   - Box model
   - Units
3. Discussion of reading and Layout Land
4. Sass + Sassy activities
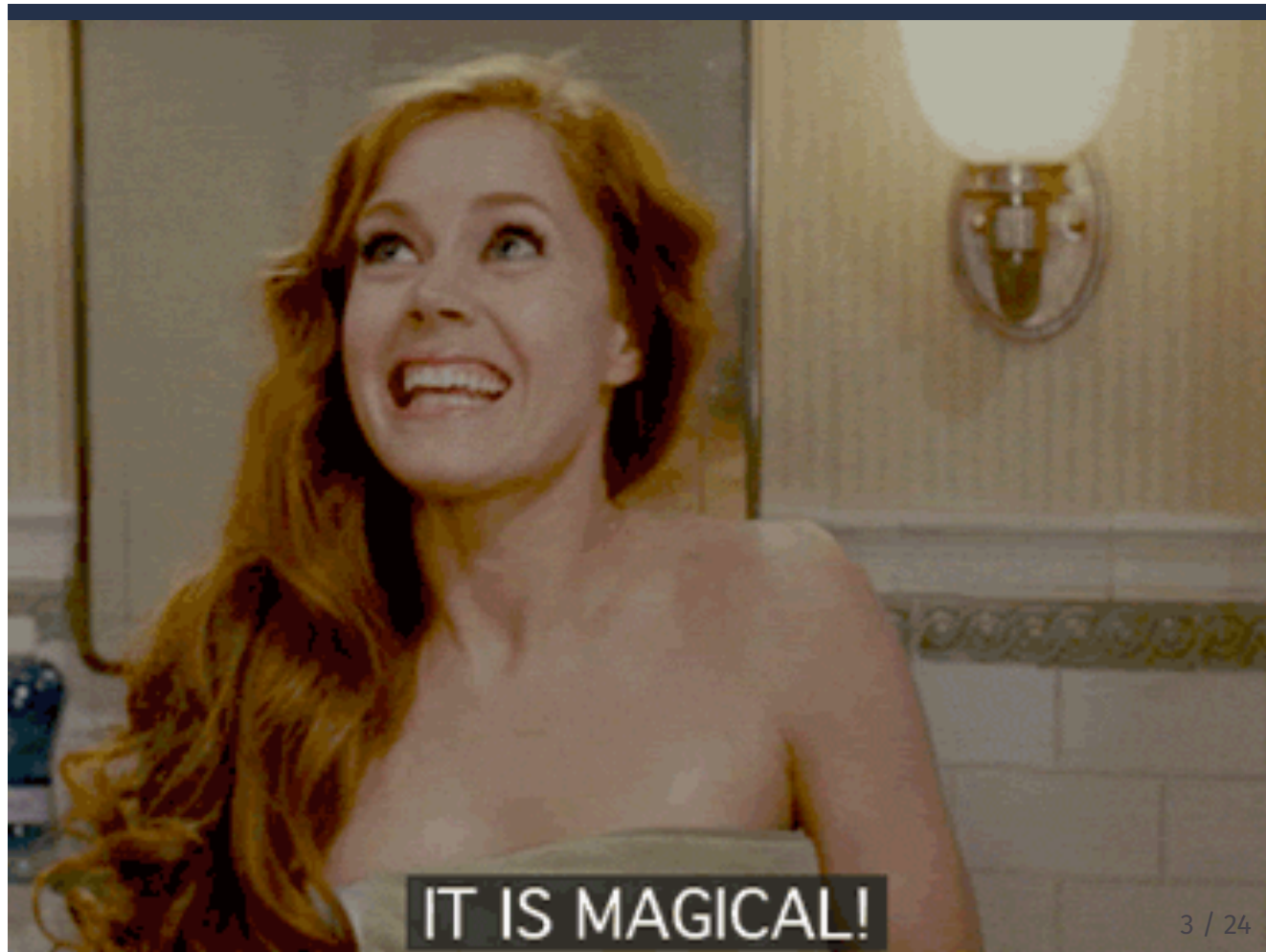
What Is

Sass

Sass is a CSS preprocessor

originally written in Ruby

that compiles into CSS

2 syntaxes

- *.sass

- *.scss

Why do I use

Sass

Sass magic: Variables

```scss
$variable: value;
```

```scss
$font-mono: 'Fira Mono', monospace;
```

```scss
.remark-code { font-family: $font-mono; }
```

```scss
.remark-code { font-family: 'Fira Mono',
monospace; }
```

```scss
$color-grey: #889094;
$color-text: $color-grey;
```

```scss
p { color: $color-text; }
```

CSS Variables

Sass

```scss
$color-grey: #889094;
$color-text: $color-grey;

p { color: $color-text; }
```

CSS

```css
:root {
  --color-grey: #889094;
  --color-text: var(--color-grey);
}

p { color: var(--color-text); }
```

CSS Variables

CSS Variables (Custom Properties) ⧉

Permits the declaration and usage of cascading variables in stylesheets.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	15	60	67	11	10.3			4.4	5
10	16	61	68	11.1	11.2			4.4.4	6.2
11	17	62	69	12	11.4	all	69	67	7.2
	18	63	70	TP	12				

✓ ✗ Partial Support

Global: 86.73% + 0.06% = 86.79%

Data from caniuse.com | Embed from caniuse.bitsofco.de

Enable accessible colours

Sass magic

Use variables when:

- the value is repeated at least twice

- the value is likely to be updated at least once

- all occurrences of the value are tied to the variable

Nesting

Sass magic: Nesting

Sass

```scss
.ul {
  padding: 0;

  li { color: green; }
}
```

```scss
.ul { padding: 0; }
.ul li { color: green; }
```

Sass magic: Nesting

```html
<header class="site-header">
  <nav class="nav-bar">
    <ul class="nav-menu">
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
      <li class="nav-
menu__item"></li>
    </ul>
  </nav>
</header>
```

Don't do this

CSS

```css
    header.site-header
nav.nav-bar ul.nav-menu
li.nav-menu__item {}
```

Sass

```scss
header.site-header {
  nav.nav-bar {
    ul.nav-menu {
      li.nav-menu__item {}
    }
  }
}
```

Inception rule
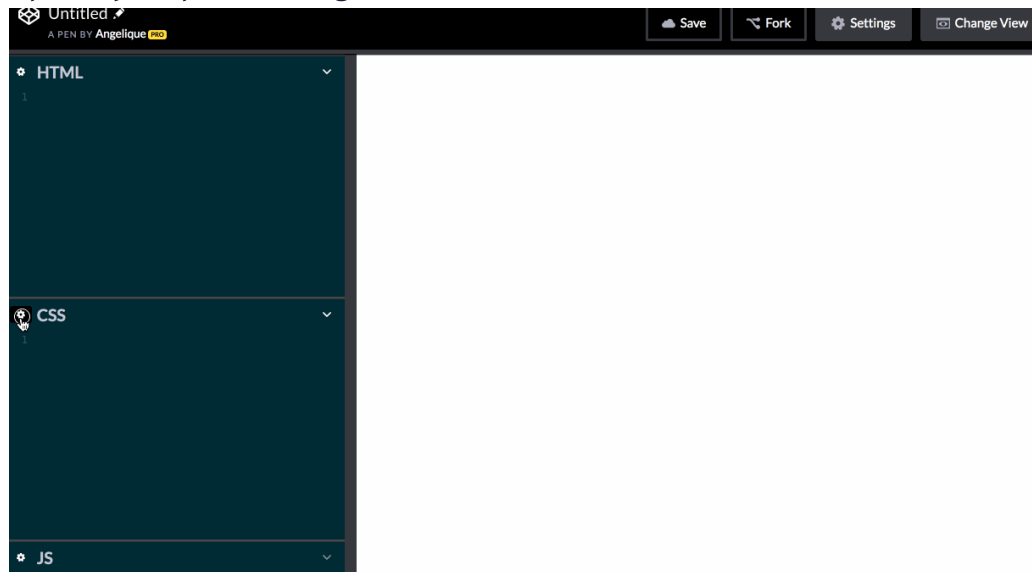
Don't go more than 3 levels deep

Sass magic: Nesting

```
.element {
  @media screen only and (min-width:600px) {
    // media queries here
  }
}
```
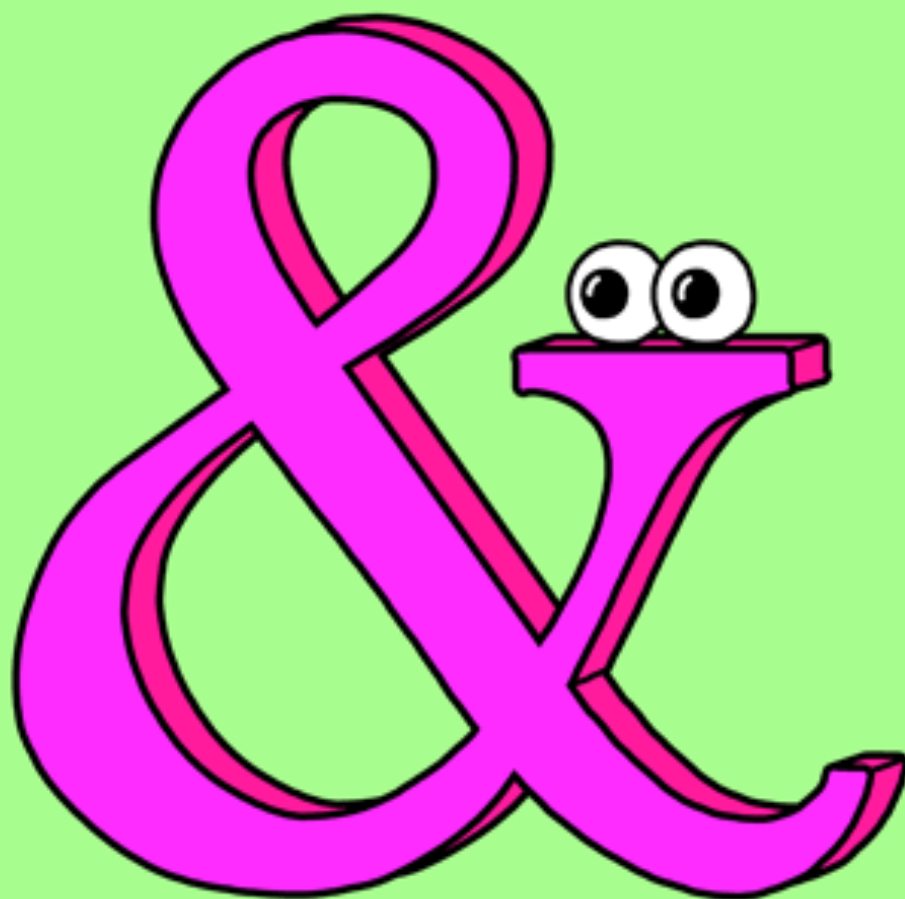
Q&A

Q&A

1. Fork your resume CodePen
2. Update your pen's settings to use Sass (SCSS)

3. Update your now Sass-y CodePen to use at least one variable and, if appropriate, nest a selector
   - Remember the *inception* rule

Sass magic: Nesting + Ampersand

SCSS

```scss
.some-class {
 &.another-class { }
}
```

Sass magic: Nesting + Ampersand

Sass

```
.class1 {
  &.class2 {}
}

.class1 {
  .class2 {}
}
```

CSS

```
.class1.class2 {}

.class1 .class2 {}
```

Sass magic: Nesting + Ampersand

```scss
.class1 {
  .class2 & {
    .class3 { color: red; }
  }
}
```

```css
.class2 .class1 .class3 { color: red; }
```

Sass magic: Ampersand + Pseudo classes

```
a {
  &:link {}
  &:visited {}
  &:hover, &:focus {}
  &:active {}
}
```

Sass magic: Ampersand + Pseudo elements

```
a {
  &::before {}
  &::after {}
}
```

Sass magic: Ampersand & BEM

CSS

```
.photo {}
.photo__img {}
.photo__caption {}

.photo--featured {}
.photo--featured .photo__img
{}
.photo--featured
.photo__caption {}
```

Sass

```
.photo {
  &__img {}
  &__caption {}
  &--featured {
    .photo__img {}
    .photo__caption {}
  }
}
```

Sass magic: Mixins

```scss
@mixin size($width, $height) {
  height: $height;
  width: $width;
}
```

```scss
img {
  @include size(25px, 100%);
}
```

Sass magic: Mixins

```
img {
  @include size(25px, 100%);
}
```

```
img {
  height: 100%;
  width: 25px;
}
```

Sass magic: Mixins

```scss
@mixin size($width, $height:
$width) {
  height: $height;
  width: $width;
}
```

```scss
img {
  @include size(25px);
}
```

Sass magic: Functions

```
lighten($color, $percent)
darken($color, $percent)
rgba($color, $alpha)
complement($color)
invert($color)
```

```
p { color: rgba($color-text, 0.5); }
p { color: rgba(green, 0.5); }
```