CoffeeScript & Sass

Sprachen über Sprachen

Über mich

- Clemens Helm
- Selbständiger Web Application Developer
- *Ruby on Rails*-Enthusiast

CoffeeScript & Sass

- CoffeeScript: Besseres JavaScript
- Sass: Besseres CSS
- Keine Interpretation zur Laufzeit
- Blog: http://www.rails-troubles.com/

Sass

http://sass-lang.com/

Sass

- Erweitert CSS3 (SCSS)
- RubyGem

```
sass --watch style.scss:style.css
```

Variablen, Berechnungen und Funktionen

```
$background-color: #3bbfce;
$navigation-background-color: lighten($background-color, 10%);

body {
  background: $background-color;
}

#navigation {
  background: $navigation-background-color;
}

#navigation a:hover {
  background: $navigation-background-color + #222;
}
```

Nesting

```
#main-nav {
  float: left;
  width: 20em;
  font: {
    family: sans-serif;
    size: 1.5em;
  }

  a {
    color: #333;

    &:hover {
      color: #666;
    }
  }
}
```

Mixins

```
@mixin border-radius($radius) {
    border-radius: $radius;
    -webkit-border-radius: $radius;
}

@mixin button-style {
    @include border-radius(0.5em);
    background-color: $button-background;
    text-decoration: none;
    border: 1px solid rgba(0,0,0,0.3);
}

.button {
    @include button-style;
    padding: 0.2em 0.5em;
    cursor: pointer;
}
```

Selector Inheritance

```
.button {
    @include button-style;
    padding: 0.2em 0.5em;
    cursor: pointer;
}

.edit-button {
    @extend .button;

    &:before {
        content: "✎ ";
    }
}
```

CoffeeScript

http://jashkenas.github.com/coffee-script/

CoffeeScript

- Kompiliert zu JavaScript (*nicht* kompatibel)
- *Gute Teile* von JavaScript auf einfache Art

```
coffee --watch --compile experimental.coffee
```

Funktionen

```
# Definition
square = (x) -> x * x

# Aufruf
square x

# Anonyme Funktionen (z.B. jQuery)
$('.menu-item').click () ->
  console.log 'you clicked it!'
```

Objekte

```
math =
    root:   Math.sqrt
    square: square
    cube:   (x) -> x * square x
```

Konditionen, Existenz und Instance-Variablen

```
alert ‘Please log in’ unless @current_user?
```

Comprehensions & Ranges

```
eat food for food in ['toast', 'cheese', 'wine']

# Ranges
countdown = (num for num in [10..1])

# Objects
yearsOld = max: 10, ida: 9, tim: 11
ages = for child, age of yearsOld
  "#{child} is #{age}"
```

Classes

```
class Animal
  constructor: (@name) ->
  move: (meters) ->
    alert @name + " moved #{meters}m."

class Snake extends Animal
  move: ->
    alert "Slithering..."
    super 5

class Horse extends Animal
  move: ->
    alert "Galloping..."
    super 45

sam = new Snake "Sammy the Python"
tom = new Horse "Tommy the Palomino"

sam.move()
tom.move()
```

und noch vieles mehr!

CoffeeScript & Sass...

- ... lösen Probleme, die uns schon immer genervt haben
- ... machen Code lesbarer und strukturierter
- ... generieren Best-Practices
- ... steigern den Spaß am Coden

Mehr Infos

- SASS – http://sass-lang.com/
- CoffeeScript – http://jashkenas.github.com/coffee-script/
- Mein Blog – http://www.rails-troubles.com/