

# Write LESS CSS

By Nathan Strutz

or

# How to fall in love with CSS

for programmers

Now Talking About Myself...



Photo Copyright © Boeing

Who Are We?

Why Does CSS Hate Us?



**@Neurotic**

Mark Mandel

I really hate CSS. No really, I think it's essentially broken, but we all have no choice but to use it.

19 minutes ago via [TweetDeck](#) ☆ [Favorite](#) ↺ [Retweet](#) ↩ [Reply](#)

CSS Means More

'CSS Explosion'



# The CSS Reinvention Rule

CSS Is Not Zen

CSS is Not DRY

CSS is *Anti-Programmer*

# Knowing When to Use a Framework or Tool

CSS is Low-Level

# Introducing LESS CSS

Leaner CSS

LESS is not...



# LESS is not the Less Framework

"The adaptive CSS grid system"

<http://lessframework.com/>

LESS is not a CSS  
Framework at All

LESS is not the Solution to  
Browser Incompatibilities

# LESS is not SASS

"Syntactically Awesome Style Sheets"

<http://sass-lang.com/>

LESS is Not These Other  
CSS Preprocessors Either

# Other CSS Preprocessors

- `Y`CSS: Cool logo, otherwise uninteresting
- Turbine: Python-style nesting syntax
- CSS Cacheer: Java meets LESS, without features
- CSS Preprocessor: Good start, very similar to LESS, no nesting
- CSSPP: Defunct PHP project
- SmartCSS: PHP and featureful, untouched since 2009
- Mechanical CSS: PHP, good start, untouched since 2009
- DtCSS: PHP, good concepts
- Switch CSS: Python/Apache based, no documentation, untouched since 2006
- CSS DRY: Drupal PHP, nesting and variables
- Dynamic CSS: Drupal PHP, put PHP in your CSS, adds some CSS3 helpers (like LESS Mix-Ins)

# Where Did LESS Come From?

LESS History

LESS is...



LESS is a Language

LESS is CSS

CSS is Lesser LESS

LESS is Less

LESS is a Compiler

# LESS is Ruby

```
gem install less
```

```
lessc style.less style.css -w
```

LESS is Javascript

# LESS is .NET

[www.dotlesscss.org](http://www.dotlesscss.org)



# LESS is PHP

[leafo.net/lessphp](http://leafo.net/lessphp)

LESS is Java

LESS is Constantly Changing!

LESS is for Programmers

# Writing LESS CSS

# Controlled

Laboratory  
Practical, but Not Real Examples

# Nested CSS

# Nested CSS

## LESS

```
.header {  
  background: blue;  
  color: white;  
  a {  
    color: #FF0;  
    &:hover {  
      color: #FF6;  
    }  
  }  
}
```

## CSS Output

```
.header {  
  background: blue;  
  color: white;  
}  
.header a {  
  color: #FF0;  
}  
.header a:hover {  
  color: #FF6;  
}
```



# Less Comments

// How many times have you wanted this?

# LESS Variables

Any CSS Value

# LESS Variables Example

```
@blue: #00B2EE;  
@imagePath: "/path/to/images/";  
@contentBorder: 2px solid @blue;
```

```
.veryprofessionallooking {  
    color: @blue;  
    background:url("@{imagePath}superattractive.png");  
    border: @contentBorder;  
}
```

# LESS Variables CSS output

```
.veryprofessionallooking {  
  color: #00b2ee;  
  background: url("/path/to/images/superattractive.png");  
  border: 2px solid #00b2ee;  
}
```

# LESS Operators

Numbers and Colors

# Less Operators Example

size: 100px + 10%;

size: 110px;

color: #F00 - 25%;

color: #e60000;

width: 100% / 4;

width: 25%;

color: #F00 + #060;

color: #ff6600;

margin: 2px + 4px;

margin: 6px;

font-size: 3em \* 0.25;

font-size: 0.75em;

# Basic LESS Mixins

# LESS Mixins Example

```
.header, .footer {  
  .veryprofessionallooking;  
}
```



# LESS Mixins CSS Output

```
.header, .footer {  
  color: #00b2ee;  
  background: url("/path/to/images/superattractive.png");  
  border: 2px solid #00b2ee;  
}
```

# Advanced LESS Mix-Ins

Think: Functions

# LESS Mixin Function Input

```
.rounded-corners (@radius: 5px) {  
    border-radius: @radius;  
    -webkit-border-radius: @radius;  
    -moz-border-radius: @radius;  
}
```

```
.element1 {  
    .rounded-corners();  
}  
.element2 {  
    .rounded-corners(10px);  
}
```

# LESS Mixin Function Output

```
.element1 {  
  border-radius: 5px;  
  -webkit-border-radius: 5px;  
  -moz-border-radius: 5px;  
}  
.element2 {  
  border-radius: 10px;  
  -webkit-border-radius: 10px;  
  -moz-border-radius: 10px;  
}
```

# LESS Mixin Function Input

```
.foo() {  
  color:#0F0;  
  background:#00F;  
}
```

# LESS Mixin Function Output

# LESS Mixin Function Alternate Input

```
.foo {  
  color:#0F0;  
  background:#00F;  
}
```

# LESS Mixin Function Alternate Input

```
.foo() {  
  color:#0F0;  
  background:#00F;  
}  
.bar {  
  .foo();  
}
```



# LESS Mixin Function Alternate Input

```
#foo() {  
  color:#0F0;  
  background:#00F;  
}  
.foo {  
  #foo();  
}
```

# LESS Mixin Function Alternate Input

```
#foo() {  
  color:#0F0;  
  background:#00F;  
}  
.foo {  
  #foo;  
}
```

Mixin @arguments

# Mixin @Arguments

```
.prettybox(@top, @right, @bottom, @left) {  
  border-width:@arguments;  
}
```

```
.uglyBox {  
  .prettybox(3px, 2px, 4px, 1px);  
}
```

# LESS Color Functions

# LESS Color Functions

lighten(@color, @amount);

darken(@color, @amount);

saturate(@color, @amount);

desaturate(@color, @amount);

fadein(@color, @amount);

fadeout(@color, @amount);

spin(@color, @amount);

# LESS Scope

Variable Precedence in Nested  
Rules and Mixins

# Namespaces

For Bundling Mixins  
`#myNamespace > .myMixin;`



# @Import-ing

LESS or Just CSS

# Escaping

~"Don't LESS Me, Bro!";

# Javascript Evaluation

Are you sure this is a good idea?

That's the Language

# IDE Support

NetBeans, TextMate, LESS.App  
or, any CSS editor

LESS in the Real World

# Implementing LESS

Ruby? Really?



Node.js? Really?

Javascript? Really?

Java? Really?

.NET or PHP? Really?

# LESS in ColdFusion

Really.

Experiments...

# A Simple Suggestion

# The Better Workflow



# A Simple Switch

```
if (dev) { pure.less; } else { compiled.css }
```

Development: Javascript

Test: Build With Ant

# Production: Build and Minify With Ant

# Summary

visit [lesscss.org](http://lesscss.org)  
bookmark [dopefly.com](http://dopefly.com)  
& [github.com/NathanStrutz](https://github.com/NathanStrutz)  
start writing LESS CSS now!

Questions?

Thank you!

