

LESS

Brian Love

Webucator, Inc.



What is LESS?

- ✦ A programming language
- ✦ Compiles to CSS3 -- a CSS preprocessor
- ✦ Similar syntax to CSS
- ✦ Often referred to as “dynamic CSS”

Why use LESS?

- ✦ Save time
- ✦ Reduce mistakes
- ✦ Reduce repetition
- ✦ Freakin' cool

Installation on Mac

- ✦ I use homebrew
- ✦ `brew install git`
- ✦ `brew install node`
- ✦ `curl https://npmjs.org/install.sh | sh`
- ✦ `npm install less --global`
- ✦ `lessc -v`

Installation on Windows

✦ winless.org

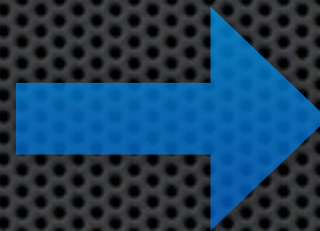
OK. What can it do?

Variables

- ✦ Start with the @ symbol
- ✦ Hexadecimal colors: #333 or #333333
- ✦ Strings: “Webucator, Inc.”
- ✦ Sizes: 10px or 10%

LESS

```
@webucatorOrange: #ff7800;  
p { color: @webucatorOrange }
```



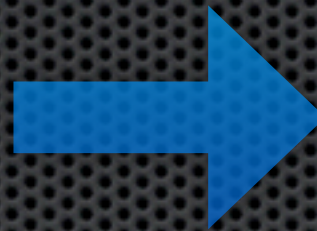
CSS

```
p { color: #ff7800; }
```


LESS

```
@company: "Webucator"  
footer { content: @company }
```

```
@company: "Webucator"  
@varName: "company"  
footer { content: @@var }
```



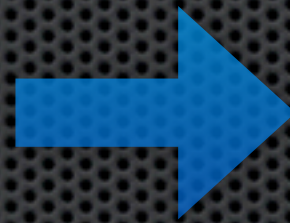
CSS

```
footer { content: "Webucator" }
```

```
footer { content: "Webucator" }
```


LESS

```
@padding: 5px;  
p { padding: @padding }
```



CSS

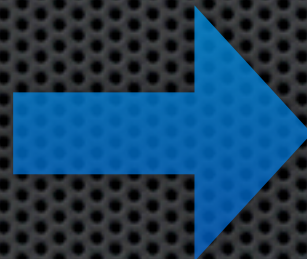
```
p { padding: 5px; }
```


Mixins

- ✦ Include properties from one ruleset to another
- ✦ Reuse code
- ✦ Can accept parameters
- ✦ Can define default values for parameters
- ✦ `@arguments variable = argumentsCollection`

LESS

```
.border-radius(@radius: 2px) {  
  -webkit-border-radius: @radius;  
  -moz-border-radius: @radius;  
  border-radius: @radius;  
}  
@gray: #333;  
header > nav {  
  border: 1px solid @gray;  
  .border-radius;  
}  
aside > nav {  
  .border-radius(5px);  
}
```

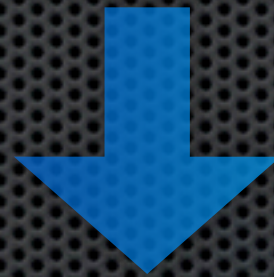


CSS

```
header > nav {  
  border: 1px solid #333;  
  -webkit-border-radius: 2px;  
  -moz-border-radius: 2px;  
  border-radius: 2px;  
}  
aside > nav {  
  -webkit-border-radius: 5px;  
  -moz-border-radius: 5px;  
  border-radius: 5px;  
}
```


LESS

```
.box-shadow(@x: 0, @y: 0, @radius: 4px, @spread: 0, @color: rgba(0,0,0,0.5))
{
  -webkit-box-shadow: @arguments;
  -moz-box-shadow: @arguments;
  box-shadow: @arguments;
}
#instructions {
  .box-shadow(2px, 2px, 4px);
}
```



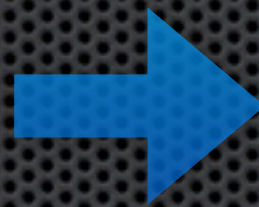
CSS

```
#instructions {
  -webkit-box-shadow: 2px 2px 4px 0 rgba(0,0,0,0.5);
  -moz-box-shadow: 2px 2px 4px 0 rgba(0,0,0,0.5);
  box-shadow: 2px 2px 4px 0 rgba(0,0,0,0.5);
}
```


Cascading + Nesting

- ✦ Nest rulesets in place of cascading
- ✦ Can be used in combination with cascading
- ✦ Mimics your DOM structure


```
header {
  color: white;
}
header .logo {
  width:300px;
  float:left;
}
header > #searchcontainer {
  width:300px;
  float:right;
}
header > #searchcontainer input {
  width:250px;
  margin-right:6px;
  color:@darkGray;
}
header > #searchcontainer input:focus {
  color:@lightGray;
}
```



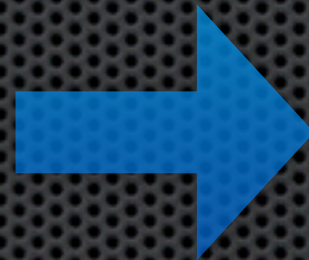
```
header {
  color: white;
  .logo {
    width:300px;
    float:left;
  }
  #searchcontainer {
    width:300px;
    float:right;
    input {
      width:250px;
      margin-right:6px;
      color:@darkGray;
      &:focus {
        color:@lightGray;
      }
    }
  }
}
```


&combinator

- ✦ Nested selector is concatenated to the parent selector
- ✦ Useful for pseudo-classes
 - ✦ `&:hover`
 - ✦ `&:focus`

LESS

```
@gray: #333;
img {
  border:none;
  &:focus {
    border:1px solid @gray;
  }
  @.inline {
    display:inline;
  }
}
```



CSS

```
img {
  border:none;
}
img:focus {
  border:1px solid #333;
}
img.inline {
  display:inline;
}
```


Operations

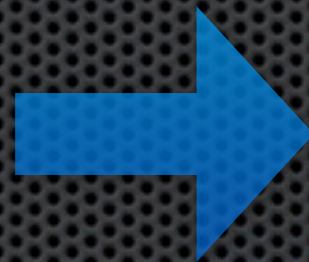
- ✦ Any number, color or variable can be operated upon
- ✦ Math functions
- ✦ Math operators
 - ✦ +
 - ✦ -
 - ✦ *
 - ✦ /
- ✦ Color functions

Math Functions

- ✧ floor()
- ✧ ceiling()
- ✧ round()
- ✧ percentage()

LESS

```
@padding: 2px;
figure {
  padding:@padding;
  img {
    padding:@padding * 2;
  }
  figcaption {
    padding:@padding + 4px;
  }
}
```



CSS

```
figure {
  padding:2px;
}
figure img {
  padding:4px;
}
figure figcaption {
  padding: 6px;
}
```


Color Functions

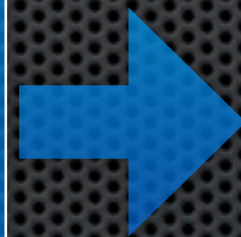
- `darken(@color, 10%);` //returns color 10% darker
- `lighten(@color, 10%);` //returns color 10% lighter
- `fadein(@color, 10%);` //returns color 10% more transparent
- `fadeout(@color, 10%);` //returns color 10% less transparent
- `fade(@color, 80%);` //returns color with 80% transparency
- and more...

Color Functions

- ✦ `hue(@color);` //hue channel of color
- ✦ `saturation(@color);` //saturation channel of color
- ✦ `lightness(@color);` //lightness channel of color
- ✦ `alpha(@color);` //alpha channel of color

LESS

```
@red: #891101;
.nav {
  background-color: fadeout(darken(@red, 10%), 20%);
  li {
    a {
      color: lighten(@red, 80%);
    }
  }
}
```



CSS

```
.nav {
  background-color: rgba(86, 11, 1, 0.8);
}
.nav li a {
  color: #fe988b;
}
```


Color Operators

- ✦ `color: @black + #111; //adds one hexadecimal unit to @black color variable`
- ✦ `color: @white - #111; //subtracts one hexadecimal unit from @white color variable`

Comments

- ✦ `/*` These comments are preserved into your compiled CSS `/*`
- ✦ `//` These comments are silent

@import

- ✦ @import will compile and copy result into single file
- ✦ All variables and mixins are available to main file or files imported after declarations
- ✦ Order matters
- ✦ Can include/ignore .less extension
- ✦ Can import “classic” css using .css extension
- ✦ You can break out files logically, but avoid the (terrible) @import() statement in traditional CSS


```
// import normalize for CSS resets
@import "normalize";
// same as @import "normalize.less";

// import mixins
@import "mixins";

// base for mobile devices
@import "base";

//tables and small laptops
@media only screen and (min-width: 768px) {
    @import "768up";
}

//desktop
@media only screen and (min-width: 1030px) {
    @import "1030up";
}
```


String Interpolation

- ✦ Use `@{name}` construct
- ✦ For embedding variable values within declarations


```
@baseUrl: "http://www.webucator.com/";  
@imageUri: "images/";  
background-image: url('@{baseUrl}@{imageUri}bg.png');
```



```
background-image: url('http://www.webucator.com/images/bg.png');
```


Escaping

- ✦ If you need to output CSS that is not valid CSS syntax
- ✦ Proprietary syntax not recognized by LESS
- ✦ If not used, LESS compiler will throw an error
- ✦ Simple prefix with ~ symbol and put in quotes (string)


```
.css-gradient(@from: #dfdfff, @to: #f8f8f8) {  
    background-color: @to;  
    background-image: -webkit-gradient(linear, left top, left bottom, from(@from), to(@to));  
    background-image: -webkit-linear-gradient(top, @from, @to);  
    background-image: -moz-linear-gradient(top, @from, @to);  
    background-image: -o-linear-gradient(top, @from, @to);  
    background-image: linear-gradient(to bottom, @from, @to);  
}  
header > nav {  
    .css-gradient;  
    filter: ~"progid:DXImageTransform.Microsoft.gradient(startColorstr='#dfdfff',  
endColorstr='#f8f8f8')";  
}
```



```
header > nav {  
    background-color: #f8f8f8;  
    background-image: -webkit-gradient(linear, left top, left bottom, from(#dfdfff), to(#f8f8f8));  
    background-image: -webkit-linear-gradient(top, #dfdfff, #f8f8f8);  
    background-image: -moz-linear-gradient(top, #dfdfff, #f8f8f8);  
    background-image: -o-linear-gradient(top, #dfdfff, #f8f8f8);  
    background-image: linear-gradient(to bottom, #dfdfff, #f8f8f8);  
    filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#dfdfff',  
endColorstr='#f8f8f8');  
}
```


Cool. How do we use it?

Pre-compile

- ✦ Compile LESS to CSS3, so only the single CSS file is loaded by your application.

```
$ lessc -v  
lessc 1.3.0 (LESS Compiler) [JavaScript]  
  
$ lessc style.less > ../css/style.css
```


Post-compile

- You can use a less.js file, and include your less files in your HTML.

```
<!---LESS stylesheets first-->
<link rel="stylesheet/less" type="text/css"
      href="styles.less"/>

<!---Post-compile LESS to CSS3-->
<script src="less.js" type="text/javascript">
</script>
<script type="text/javascript">
    less.watch();
</script>
```


Pre versus Post-compile

- ✦ Pre-compile is easier during development
- ✦ Post-compile should NEVER be used in production

LESS Elements

- ✦ lesselements.com contains commonly used mixins, such as:
 - ✦ `.gradient`
 - ✦ `.rounded`
 - ✦ `.opacity`
 - ✦ `.box-shadow`
 - ✦ `.inner-shadow`

Clearing floats

```
.clearfix {  
  &:before,  
  &:after {  
    content: "";  
    display: table;  
  }  
  &:after {  
    clear: both;  
  }  
  /* for IE */  
  zoom: 1;  
}
```


Minification and Compression

- ✦ `lessc -x styles.less > ../css/styles.css`
- ✦ `lessc -x --yui-compress styles.less > ../css/styles.css`

Alternatives to LESS

- ✦ SASS: Syntactically Awesome StyleSheets
- ✦ SCSS: v2 of SASS
- ✦ Stylus

Not included

- ✦ Use JavaScript within LESS
- ✦ Namespaces and accessors
- ✦ Read about these at lesscss.org

Now go use LESS!