# Less半知半解

1. **变量定义**

**@width: 10px; @height: @width + 10px;**

**#header { width: @width; height: @height; }**

编译为：

#header { width: 10px; height: 20px; }

@link-color: #428bca; *// sea blue* @link-color-hover: darken(@link-color, 10%);

*// Variables*

@my-selector: banner;

*// Usage*

.@{my-selector} { font-weight: bold; line-height: 40px; margin: 0 auto; }

编译为：

.banner { font-weight: bold; line-height: 40px; margin: 0 auto; }

#### URLs

*// Variables*@images: "../img";

*// Usage*body {

color: #444;

background: url("@{images}/white-sand.png");

}

#### Import Statements

*// Variables*

@themes: "../../src/themes";

*// Usage*

@import ["@{themes}/tidal-wave.less";](mailto:\"@{themes}/tidal-wave.less\";)

#### Properties

@property: color;

.widget { **@{property}**: #0ee; background-**@{property}**: #999; }

### Variable Variables

@primary: green; @secondary: blue;

.section { @color: primary; .element { color: @@color; } }

### Lazy Evaluation

Variables do not have to be declared before being used.

Valid Less snippet:

.lazy-eval {

width: @var;

}

@var: @a;@a: 9%;

this is valid Less too:

.lazy-eval {

width: @var;

@a: 9%;

}

@var: @a;@a: 100%;

both compile into:

.lazy-eval {

width: 9%;

}

# Mixins 混入

.bordered {

border-top: dotted 1px black;

border-bottom: solid 2px black;

}

// 编译为

#menu a {

color: #111;

.bordered();

}

.post a {

color: red;

.bordered();

}

# Nesting 嵌套

#header { color: black; }

#header .navigation { font-size: 12px; }

#header .logo { width: 300px; }

// Less 写法

#header { color: black;

.navigation { font-size: 12px; }

.logo { width: 300px; }

}

.clearfix {

display: block; zoom: 1;

***&***:after { content: " "; display: block; font-size: 0; height: 0; clear: both; visibility: hidden; }

}

(& represents the current selector parent)

## **Nested At-Rules and Bubbling**

.component {

width: 300px;

@media (min-width: 768px) {

width: 600px;

@media (min-resolution: 192dpi) {

background-image: url();

}

}

@media (min-width: 1280px) { width: 800px; }

}

// 编译为：

.component { width: 300px; }

@media (min-width: 768px) { .component { width: 600px; } }

@media (min-width: 768px) and (min-resolution: 192dpi) {

.component { background-image: url(); }

}

@media (min-width: 1280px) { .component { width: 800px; } }

**五、Operations**

*// numbers are converted into the same units*

@conversion-1: 5cm + 10mm; *// result is 6cm*

@conversion-2: 2 - 3cm - 5mm; *// result is -1.5cm*

*// conversion is impossible*

@incompatible-units: 2 + 5px - 3cm; *// result is 4px*

*// example with variables*

@base: 5%; @filler: @base \* 2; *// result is 10%*

@other: @base + @filler; *// result is 15%*

@color: #224488 / 2; *//results in #112244*

background-color: #112244 + #111; *// result is #223355*

**六、calc() exception**

@var: 50vh/2;

width: calc(50% + (@var - 20px)); *// result is calc(50% + (25vh - 20px))*

# Namespaces and Accessors

#bundle() {

.button {

display: block;

border: 1px solid black;

background-color: grey;

&:hover {

background-color: white;

}

}

.tab { ... }

.citation { ... }

}

Now if we want to mixin the .button class in our #header a, we can do:

#header a {

color: orange;

#bundle.button(); *// can also be written as #bundle > .button*

}

Note: append () to your namespace (e.g. #bundle()) if you don't want it to appear in your CSS output i.e. #bundle .tab.

1. **Maps**

#colors() {

primary: blue;

secondary: green;

}

.button {

color: #colors[primary];

border: 1px solid #colors[secondary];

}

This outputs, as expected:

.button {

color: blue;

border: 1px solid green;

}

# Scope 作用域

@var: red;

#page {

@var: white;

#header { color: @var; *// white* }

}

# Importing

@import "library"; *// library.less*

@import "typo.css";