A BETTER CSS : SASS & LESS

By Frédéric Ghijselinck & Sam Vloeberghs
Competence Center Front-end & UX

A BETTER CSS : SASS & LESS

ORDINA

I. What's wrong with CSS?
II. Introducing Sass
III. Introducing LESS
IV. Hands-on

I) WHAT'S WRONG WITH CSS

ORDINA

- Browser incompatibilities
- No functions / logic
- No variables
- Uncustomizable frameworks
- DRY ..

SASS ; CSS WITH SUPERPOWERS

Sass is the most mature, stable, and powerful professional grade CSS extension language in the world.

II) INTRODUCING SASS

ORDINA

- CSS Compatible
- Feature rich
- Mature
- Large community
- Framework adaption

Examples explained using a Sass variant, Scss

MULTIPLE COMPILERS

- Compass
- Libsass

CSS EXTENSIONS

ORDINA

- Variables
- Nested rules & properties
- Referencing parent selectors
- Functions
- Interpolation
- Variable defaults
- ...

SASS: VARIABLES

ORDINA

```
$success-color:      #43AC6A;
$warning-color:      #f08a24;
$linkcolor:          $warning-color;
$otherlinkcolor:     $success-color;
```

SASS: NESTED RULES & PROPERTIES

ORDINA

```
h1{
    margin-bottom:15px;
    font: {
        size: 1.4em;
        weight: bold;
    }
}
h2{
    margin-bottom:15px;
    font: {
        size: 1.2em;
        weight: bold;
    }
}
```

SASS: REFERENCING PARENT SELECTORS

ORDINA

```
a{
    color:$linkcolor;
    text-decoration:none;
    &:hover{
        border-bottom: 1px solid $linkcolor;
    }
}
```

SASS: FUNCTIONS

ORDINA

```
$p_color: $primary-color;
p{
    content: 'This is a paragraph';
    color: $p_color;
    padding:10px;
    border: 2px solid $p_color;
    &.darker{
        $d_color: darken($color: $p_color, $amount: 10%);
        color: $d_color;
        border-color: $d_color;
    }
    &.lighter{
        $l_color: lighten($amount: 30%, $color: $p_color);
        color: $l_color;
        border-color: $l_color;
    }
}
```

SASS: INTERPOLATION

ORDINA

```
$name: fancy;
$attr: font;

p.#{$name} {
    #{$attr}-color: blue;
}
```

SASS: VARIABLE DEFAULTS

ORDINA

```
$content: "First content";
$content: "Second content?" !default;
$bis_content: "First bis content" !default;

p {
    margin-bottom:10px;
    &.primary:before {
        content: $content;
    }

    &.secundary:before {
        content: $bis_content;
    }
}
```

@-RULES & DIRECTIVES

ORDINA

- Importing & partials
- Media queries
- Extending / inheriting
- Mixins & arguments
- Control directives, logic & expressions

SASS: IMPORTING & PARTIALS

ORDINA

```
@import "reset";
@import "settings";
@import "utils";
@import "styles";

@import "foo.css";
@import "foo" screen;
@import "http://foo.com/bar";
@import url(foo);
```

SASS: MEDIA QUERIES

ORDINA

```
.vierkant{
    height:150px; text-align:center;
    color:#fff;   font-weight:bold;
    span{ display:none; }
    @media #{$small-only} {
        background: blue;
        span.small{ display:inline; }
    }
    @media #{$medium-only} {
        background:pink;
        span.medium{ display:inline; }
    }
    @media #{$large-up} {
        background:green;
        span.large{ display:inline; }
    }
    @media #{$xlarge-up} {
        color:red;
        span.xlarge{ display:inline }
    }
}
```

SASS: EXTENDING / INHERITING

ORDINA

```
.buttonanimation{
    -o-transition:color .3s linear, background .4s linear;
    -ms-transition:color .3s linear, background .4s linear;
    -moz-transition:color .3s linear, background .4s linear;
    -webkit-transition:color .3s linear, background .4s line
    transition:color .3s linear, background .4s linear;
}

button{
    color:#fff;
    background:#198fe6;
    border:1px solid #fff;
    padding:5px 20px;
    text-transform:uppercase;
    font-size:15px;
    @extend .buttonanimation;
    &:hover,
    &:focus,
    &:active{
        background:#fff;
        border:1px solid #198fe6;
        color:#198fe6;
    }
}
```

SASS: CONTROL DIRECTIVES, LOGIC & EXPRESSIONS (0)

ORDINA

- @if
- @for
- @each
- @while

SASS: CONTROL DIRECTIVES, LOGIC & EXPRESSIONS (1)

ORDINA

```
@each $type, $color in
    (info, $info-color),
    (success,$success-color),
    (error, $error-color),
    (warning, $warning-color) {

    .box-#{$type} {
        @extend .boxbasics;
        // COULD BE AVOIDED: MORE CODE GENERATION
        $bordercolor: darken($color, 10%);
        @if($type == 'info') {
            $bgcolor: $color;
        }
        @else{
            $bgcolor: lighten($color, 30%);
        }
        color: $bordercolor;
        @include background($imgpath: "../images/icons/#{$ty
        @include border($size:2px, $color: $bordercolor, $bo
    }

}
```

SASS: MIXINS & ARGUMENTS

ORDINA

```
@mixin border($size: 1px, $color: transparant,
$left: false, $bottom: false, $right:false, $top: false){
    @if ($left)   { border-left: $size solid $color;   }
    @if ($right)  { border-right: $size solid $color;  }
    @if ($top)    { border-top: $size solid $color;    }
    @if ($bottom) { border-bottom: $size solid $color; }
}

@mixin box-sizing($type:border-box) {
    -webkit-box-sizing: $type;  // Android < 2.3, iOS < 4
    -moz-box-sizing: $type;     // Firefox < 29
    box-sizing: $type;          // Chrome, IE 8+, Opera, Saf
}
```

FRAMEWORKS USING SASS

ORDINA

- Zurb Foundation
- Bootstrap
- Bourbon
- Gumby
- .. your own?

II) INTRODUCING LESS

ORDINA

- What is LESS?
- LESS on the client
- LESS on the server
- importing
- variables
- functions
- operations
- mixins
- nested rules

WHAT IS LESS?

ORDINA

- Dynamic style sheet language
- Compiles to CSS (client or server)
- Introduces programming features to css (calculations)
- Looks and feels like CSS (all CSS is valid LESS)

USING LESS ON THE CLIENT

ORDINA

```
<html>
<head>
  <link rel="stylesheet/less" type="text/css" href="css/my.l
  <script src="js/less.js" type="text/javascript"></script>
</head>
</html>
```

USING LESS ON THE SERVER

ORDINA

- Ability to cache it
- Reducing the processing time on the client
- Scale out the HTML
- Server support for LESS
  - Node.js
  - ASP.NET
  - Rails
  - JSP
  - ...

NODE.JS

ORDINA

```
$ npm install less
```

```
$ lessc styles.less

$ lessc styles.less > styles.css
```

```
var less = require('less');

less.render(lessContents,
            function (e, css) {
                console.log(css);
            });
```

LESS BASICS

ORDINA

LESS is meant to feel like CSS but better

- Ability to cache it
- All CSS is valid... really
- Renaming your .css to .less works
- LESS adds to CSS
- Mix of LESS and CSS is no problem, only LESS gets compiled to CSS

LESS BASICS

ORDINA

```
@baseFontSize: 14px;
/* Comments */
// Comments too

#main
{
    h1
    {
        font-size: @baseFontSize;
    }
}
```

IMPORTING

ORDINA

- @import works
- Embeds other .less files in a file
- Allows for simple modularization
- While maintaining merging of CSS
- If import is .css, it preserves the @import statement
- If import is .less, it merges it into file

`init.less`

```
@import "init";
```

VARIABLES

ORDINA

```
@myColor: #ffeedd;

// They are Constants, this doesn't work
@myColor: @myColor + 5%;

@a: Black; // Color
@b: 4px; // Units
@c: 1.0em; // Units
@d: Helvectica, sans serif; // Strings
@e: 1px #000 Solid 0 0; // Complex Type
```

OPERATIONS

ORDINA

```
font-size: 4px + 4; // 8px
font-size: 20px * .8; // 16px;
color: #FFF / 4; // #404040;
width: (100% / 2) + 25%; // 75%
```

COLOR FUNCTIONS

ORDINA

```
color: lighten(@color, 10%);
color: darken(@color, 10%);

color: saturate(@color, 10%);
color: desaturate (@color, 10%);

color: fadein(@color, 10%);
color: fadeout(@color, 10%);
color: fade(@color, 50%); // change transparency

color: spin(@color, 10%);
color: mix(@color, #246);
```

MORE FUNCTIONS

ORDINA

```
@hue: hue(@color);
@sat: saturation(@color);
@light: lightness(@color);
@alpha: alpha(@color);
@color: hsl(20%, 30%, 40%);

// Math
@rnd: round(3.14);
@top: ceil(3.14);
@bot: floor(3.14);
@per: percentage(.14);
```

MIXINS

- Repeatable sections
- Feel like functions
- But insert more than one name/value pair
- Can accept parameters, defaults and overloads

MIXINS

ORDINA

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

#form
{
    .rounded-corners-all(5px);
}
```

MIXINS - DEFAULTS

ORDINA

```
// Default Values
.rounded-corners-all(@size: 5px)
{
  border-radius: @size;
  -webkit-border-radius: @size;
  -moz-border-radius: @size;
}

#form
{
  .rounded-corners-all;
}
```

MIXINS - OVERLOADS

ORDINA

```
// Using overloads
.color(@color)
{
  color: @color;
}

.color(@color, @factor)
{
  color: lighten(@color, @factor);
}

#form
{
  .color(#888, 20%); // Uses 2nd overload
}
```

MIXINS - GUARDS

ORDINA

```
// Using guards
.color(@color) when (alpha(@color) >= 50%)
{
  color: Black;
}

.color(@color) when (alpha(@color) < 50%)
{
  color: transparent;
}

#form
{
  .color(@mainColor); // Uses 1st overload
}
```

NESTED RULES

ORDINA

- Allows you to structure rules in a logical way
- Hierarchies imply the cascading/specificity
- LESS then deconstructs it into CSS for you

NESTED RULES

ORDINA

```
// LESS Nested Rules
nav {
  font-size: 14px;
  font-weight: bold;
  float: right;
  ul {
    // Makes “nav ul {...}”
    list-style-type: none;
    li {
      // Makes “nav ul li {...}”
      float: left;
      margin: 2px;
    }
  }
}
```

NESTED RULES

ORDINA

```
// Use Combinator (&) to mix with parent:
a
{
  text-decoration: none;
  &:hover
  {
    text-decoration: underline;
  }
}

// Results in
a { text-decoration: none; }
a:hover { text-decoration: underline; }
```

DOCUMENTATION ETC.

ORDINA

- Sass Lang website
- Sass Compatibility
- Zurb Foundation
- Official LESS website
- Sass vs LESS

BEDANKT!

Frédéric Ghijselinck & Sam Vloeberghs
Competence Center Front-end & UX