

ASSEMBLING SASS

2

PROPERTY OF:
CODE SCHOOL

PROJECT VERSION:
PART TWO

STANDARD
INSPECTION : NO. 18



This product shall be subject to inspection by
Chief Engineer, Machine & Plant
and shall be approved by the
Executive Director, Project, at the time of
acceptance. The acceptance shall be the
basis of the acceptance of the
product. The acceptance shall be the basis of
the acceptance of the product.

S

| MORE SASS



Agenda

1. Sass Wrap-Up

2. Compass

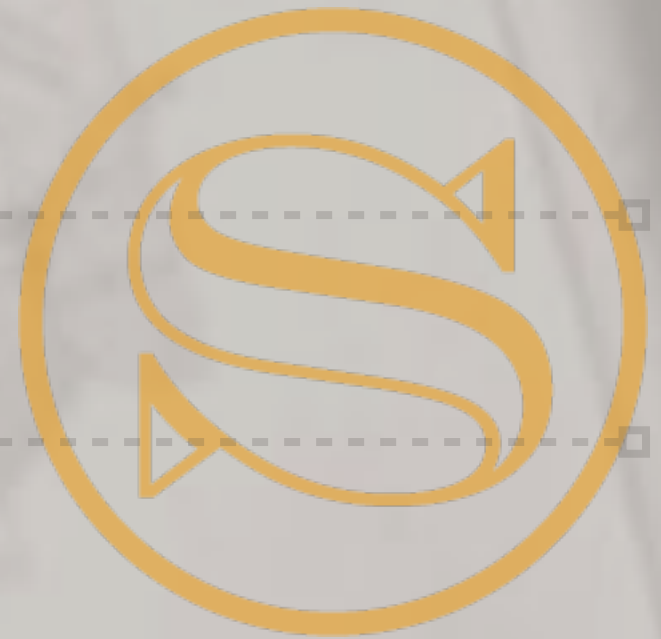


1.1 Indented Syntax

1.2 More Lists

1.3 More Utilities

1.4 More Placeholders



The Other Syntax

Indented syntax (.sass) relies on whitespace to simplify:

application.scss

```
.content {  
  border: 1px solid #ccc;  
  padding: 20px;  
  h2 {  
    font-size: 3em;  
    margin: 20px 0;  
  }  
}
```



application.sass

```
.content  
  border: 1px solid #ccc  
  padding: 20px  
  h2  
    font-size: 3em  
    margin: 20px 0
```

1.1 Indented Syntax



Indented Syntax + Mixins

Mixin declaration and use (optionally) become more efficient:

application.scss

```
@mixin box-sizing($x) {  
  -webkit-box-sizing: $x;  
  -moz-box-sizing: $x;  
  box-sizing: $x;  
}  
.content {  
  @include box-sizing(border-box);  
}
```



application.sass

```
=box-sizing($x)  
  -webkit-box-sizing: $x  
  -moz-box-sizing: $x  
  box-sizing: $x  
  
.content  
  +box-sizing(border-box)
```

1.1 Indented Syntax



Indented Syntax + This Course

- ⦿ Examples in this course will use the indented syntax
- ⦿ **Either** can be used on the challenges

1.1 Indented Syntax



Assembly Tip

The indented syntax **will**
feel strange at first, but give
it a fair shot*

* It took a while for us to warm up to it



List Refresher

A comma- **or** space-separated group of values:

`application.sass`

```
$authors: nick aimee dan drew  
$designers: jason, justin, tim
```

1.2 More Lists



application.sass

```
$authors: nick aimee dan drew  
  
@each $a in $authors  
  .author-#{ $a }  
    background: url(#{ $a }.jpg)
```

application.css

```
.author-nick {  
  background: url(nick.jpg)  
}  
.author-aimee {  
  background: url(aimee.jpg)  
}  
.author-dan {  
  background: url(dan.jpg)  
}  
.author-drew {  
  background: url(drew.jpg)  
}
```



1.2 More Lists



Length - number of items in a list:

```
$authors: nick aimee dan drew  
length($authors) // 4
```

Append - adds a value to the end of a list:

```
$authors: nick aimee dan drew  
append($authors, tim) // nick aimee dan drew tim
```

1.2 More Lists



Join - combines two lists together:

```
$authors1: nick aimee  
$authors2: dan drew  
join($authors1, $authors2) // nick aimee dan drew
```

Index - returns a value's list position or false:

```
$authors: nick aimee dan drew  
index($authors, dan) // 3  
index($authors, tim) // false
```

← starts at 1, not 0

1.2 More Lists



Nth - return the item at list position n:

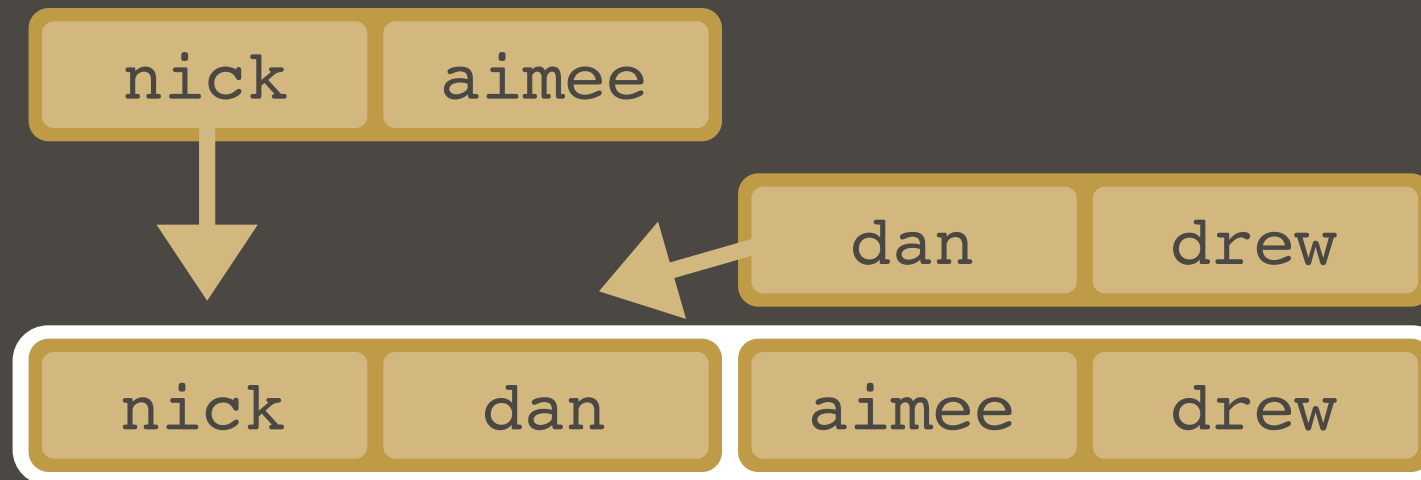
```
$authors: nick aimee dan drew  
nth($authors, 3) // dan
```

Zip - combines lists in comma-separated pairs:

```
$authors: nick aimee
```

```
$co-authors: dan drew
```

```
zip($authors, $co-authors)
```



1.2 More Lists



Using List Helpers

Pairing separate lists to create related pairs:

application.sass

```
$authors: nick aimee dan drew  
$colors: green blue red yellow  
  
$author-style: zip($authors, $colors)
```

nick

green

aimee

blue

dan

red

drew

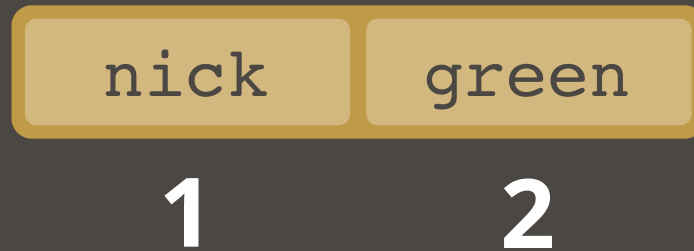
yellow

1.2 More Lists



application.sass

```
$authors: nick aimee dan drew  
$colors: green blue red yellow  
  
$author-style: zip($authors, $colors)  
  
@each $a in $author-style  
  .author-#{nth($a, 1)}  
    background: nth($a, 2)
```



application.css

```
.author-nick {  
  background: green;  
}  
.author-aimee {  
  background: blue;  
}  
.author-dan {  
  background: red;  
}  
.author-drew {  
  background: yellow;  
}
```

1.2 More Lists



List Length

Adding to the array requires a manual update to the `width`:

application.sass

```
$authors: nick aimee dan drew  
  
.author  
  float: left  
  width: 25%
```



application.sass

```
$authors: nick aimee dan drew tim  
  
.author  
  float: left  
  width: 25%
```

1.2 More Lists



Using `length()` keeps the `width` updated for us:

application.sass

```
$authors: nick aimee dan drew  
  
.author  
  float: left  
  width: 100% / length($authors)
```



application.css

```
.author {  
  float: left;  
  width: 25%;  
}
```

1.2 More Lists

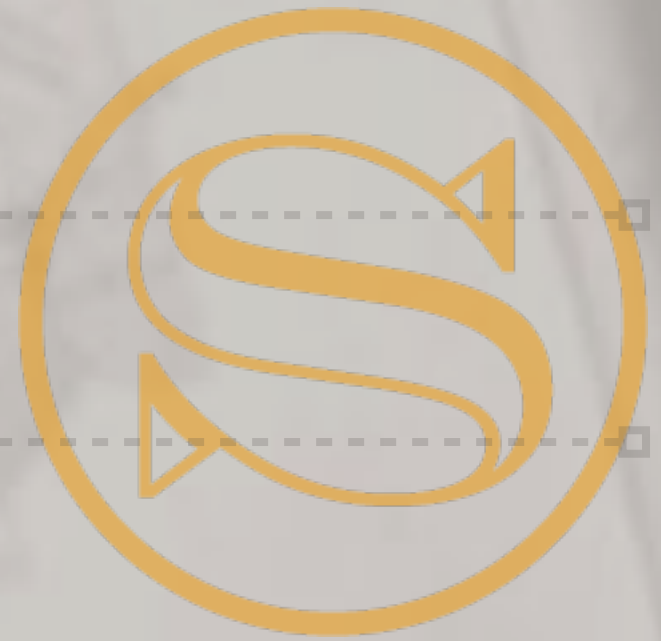


1.1 Indented Syntax

1.2 More Lists

1.3 More Utilities

1.4 More Placeholders



Scaling Color

lighten and darken adjustments are linear:

application.sass

```
.content  
  color: lighten(#eee, 7%)
```



application.css

```
.content {  
  color: white;  
}
```

light colors reach
white at low percentages

1.3 More Utilities



Scaling Color

The `scale_color` function changes one or more color aspects **relative** to the start value:

- `$red`
- `$green`
- `$blue`
- `$saturation`
- `$lightness`
- `$alpha`

1.3 More Utilities



Scaling Color

Add keyword arguments for any aspects to be scaled:

application.sass

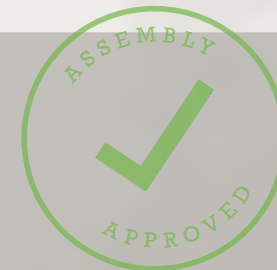
```
.content
  color: scale_color(#eee, $lightness: 7%)
```

application.css



```
.content {
  color: #efefef;
}
```

← 7% between #eee and white



1.3 More Utilities



Simple Conditionals

Part 1 included how to create an `if / else` chain in Sass:

application.sass

```
$theme: light

header
  @if $theme == dark
    background: #000
  @else
    background: #fff
```



application.css

```
header {
  background: white;
}
```

1.3 More Utilities



Single-line if - similar to a ternary operator:

```
if(condition, "true value", "false value")
```

1.3 More Utilities



application.sass

```
$theme: light

header
  @if $theme == dark
    background: #000
  @else
    background: #fff
```



application.sass

```
$theme: light

header
  background: if($theme == dark, #000, #fff)
```



1.3 More Utilities



Placeholder Selector Refresher

Can be extended; doesn't compile to a selector on its own:

application.sass

```
%btn
  border: 1px solid #ccc
  font-size: 1em
  text-transform: uppercase
```

1.4 More Placeholders



application.sass

```
%btn
  border: 1px solid #ccc
  font-size: 1em
  text-transform: uppercase

.btn-a
  @extend %btn
  background: #777

.btn-b
  @extend %btn
  background: #ff0
```

application.css

```
.btn-a, .btn-b {
  border: 1px solid #cccccc;
  font-size: 1em;
  text-transform: uppercase;
}
.btn-a {
  background: #777777;
}
.btn-b {
  background: yellow;
}
```



```
<a class="%btn" href="about.html">About</a>
```



1.4 More Placeholders



application.sass

```
%btn
.btn
  border: 1px solid #ccc
  font-size: 1em
  text-transform: uppercase

.btn-a
  @extend %btn
  background: #777

.btn-b
  @extend %btn
  background: #ff0
```

application.css

```
.btn-a, .btn-b,
.btn
  border: 1px solid #cccccc;
  font-size: 1em;
  text-transform: uppercase;
}
.btn-a {
  background: #777777;
}
.btn-b {
  background: yellow;
}
```



```
<a class="btn" href="about.html">About</a>
```

1.4 More Placeholders



application.sass

```
%btn
  border: 1px solid #ccc
  font-size: 1em
  text-transform: uppercase

.btn-a
  @extend %btn
  background: #777

.btn-b
  @extend %btn
  background: #ff0
```

index.html

```
<a href="#" class="btn-a">
  <b>Visit</b> Us
</a>
```



what if we need a nested
style on all %btn elements?

1.4 More Placeholders



application.sass

```
%btn
  border: 1px solid #ccc
  font-size: 1em
  text-transform: uppercase
  b
    color: #000

.btn-a
  @extend %btn
  background: #777
.btn-b
  @extend %btn
  background: #ff0
```

application.css

```
.btn-a, .btn-b {
  border: 1px solid #cccccc;
  font-size: 1em;
  text-transform: uppercase;
}
.btn-a b, .btn-b b {
  color: black;
}
.btn-a {
  background: #777777;
}
.btn-b {
  background: yellow;
}
```

1.4 More Placeholders



2 COMPASS



2.1 Extending Sass

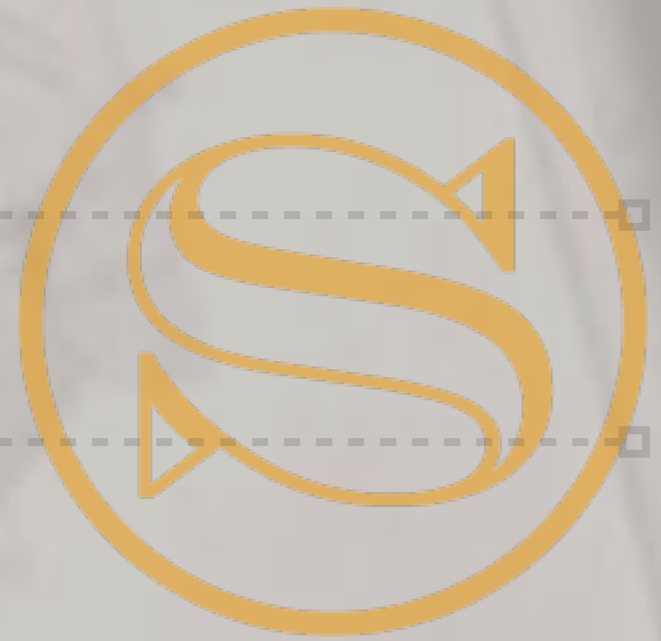
2.2 History

2.3 Importing Compass

2.4 The Reset Module

2.5 Prefixed No More

2.6 Gradients





Compass

- Open-source CSS Authoring Framework
- Adds reusable patterns, utilities, and modules like vertical rhythm and spriting
- We'll focus on the high points

2.1 Extending Sass



- Created by Chris Eppstein as charityware
- **Other core team members:**
Scott Davis, Eric Meyer, Brandon Mathis, and Anthony Short
- Source: <http://github.com/chriseppstein/compass>

2.2 History



2.1 Extending Sass

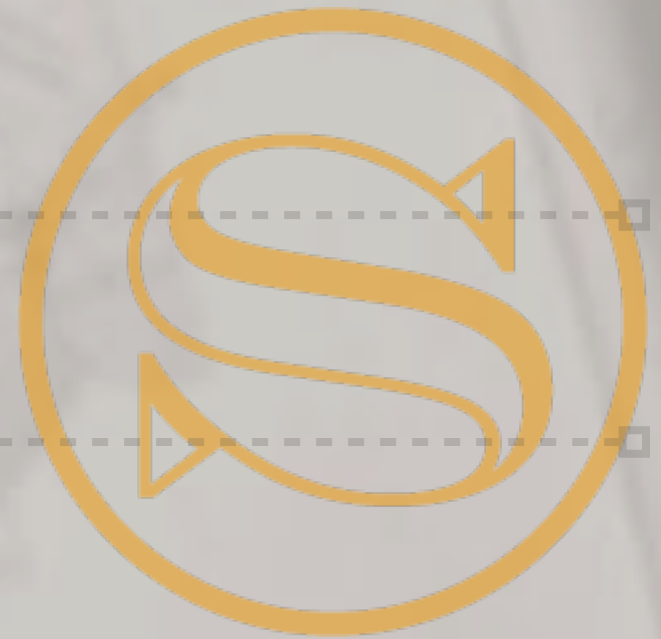
2.2 History

2.3 Importing Compass

2.4 The Reset Module

2.5 Prefixed No More

2.6 Gradients



Modules

Compass comprises **five** main modules:

- `utilities`
- `typography`
- `css3`
- `layout`
- `reset`

`reset` is the only module
that automatically outputs some
cSS when imported



2.3 Importing Compass



Importing

Once installed, use `@import` to make Compass available:

`application.sass`

```
@import "compass" ← utilities, typography, and css3  
@import "compass/layout"
```

2.3 Importing Compass



Assembly Tip

It's easiest to lead your
main Sass file off with
these imports*

* Unless restricting which features are available



Importing Reset

Adds a set of rules based on Eric Meyer's reset after compiling:

application.sass

```
@import "compass/reset"
```



- **Not ideal** in production: unable to make tweaks to fit your site's specific needs
- Avoid if you plan on using Normalize.css

2.4 The Reset Module



2.1 Extending Sass

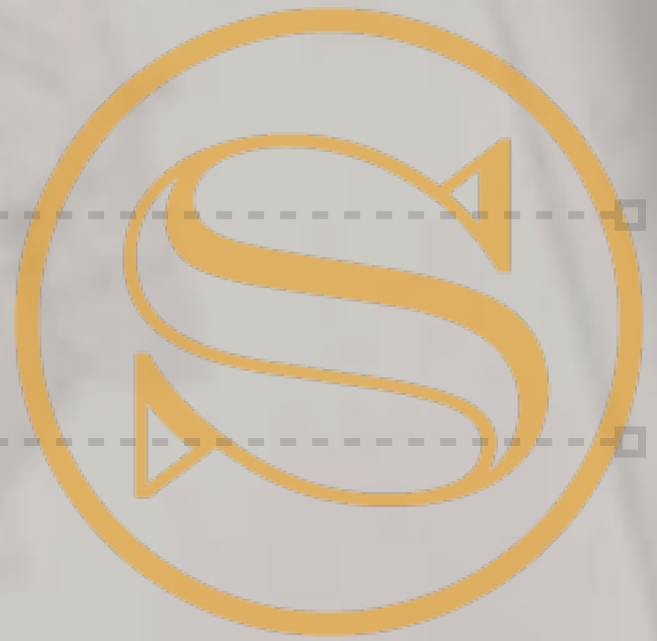
2.2 History

2.3 Importing Compass

2.4 The Reset Module

2.5 Prefixed No More

2.6 Gradients



Part 1 showed how mixins can simplify prefixed properties:

application.sass

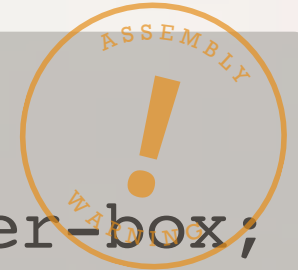
```
=box-sizing($x)
  -webkit-box-sizing: $x
  -moz-box-sizing: $x
  box-sizing: $x

.content
  +box-sizing(border-box)
```

application.css

```
.content {
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
}
```

-o-? -ms-?



2.5 Prefixed No More



application.sass

```
@import "compass"
```

```
.content  
  +box-sizing(border-box)
```

application.css

```
.content {  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}
```

(the extra prefixes
weren't needed)

2.5 Prefixed No More



Useful CSS3 Module Mixins*

- ⦿ background (gradients)
- ⦿ border-radius
- ⦿ box-shadow
- ⦿ columns
- ⦿ transform
- ⦿ transition

* Check the browsers you intend to support to see if providing prefixes is still necessary

2.5 Prefixed No More



The Background Mixin

Prefixed CSS3 gradients are a bit... verbose:

application.css

```
.content {  
  background: -webkit-gradient(linear, 50% 0%, 50% 100%,  
color-stop(0%, #9b9592), color-stop(100%, #3c3733));  
  background: -webkit-linear-gradient(top, #9b9592, #3c3733);  
  background: -moz-linear-gradient(top, #9b9592, #3c3733);  
  background: -o-linear-gradient(top, #9b9592, #3c3733);  
  background: -ms-linear-gradient(top, #9b9592, #3c3733);  
  background: linear-gradient(top, #9b9592, #3c3733);  
}
```



2.6 Gradients





`linear-gradient(top, #9b9592, #3c3733)`

Linear Gradients

Using the non-prefixed property with the background mixin:

application.sass

```
@import "compass"

.content
  +background(linear-gradient(top, #9b9592, #3c3733))
```



outputs everything shown
in the initial example

2.6 Gradients



Radial Gradients

The radial gradient syntax also works within the mixin:

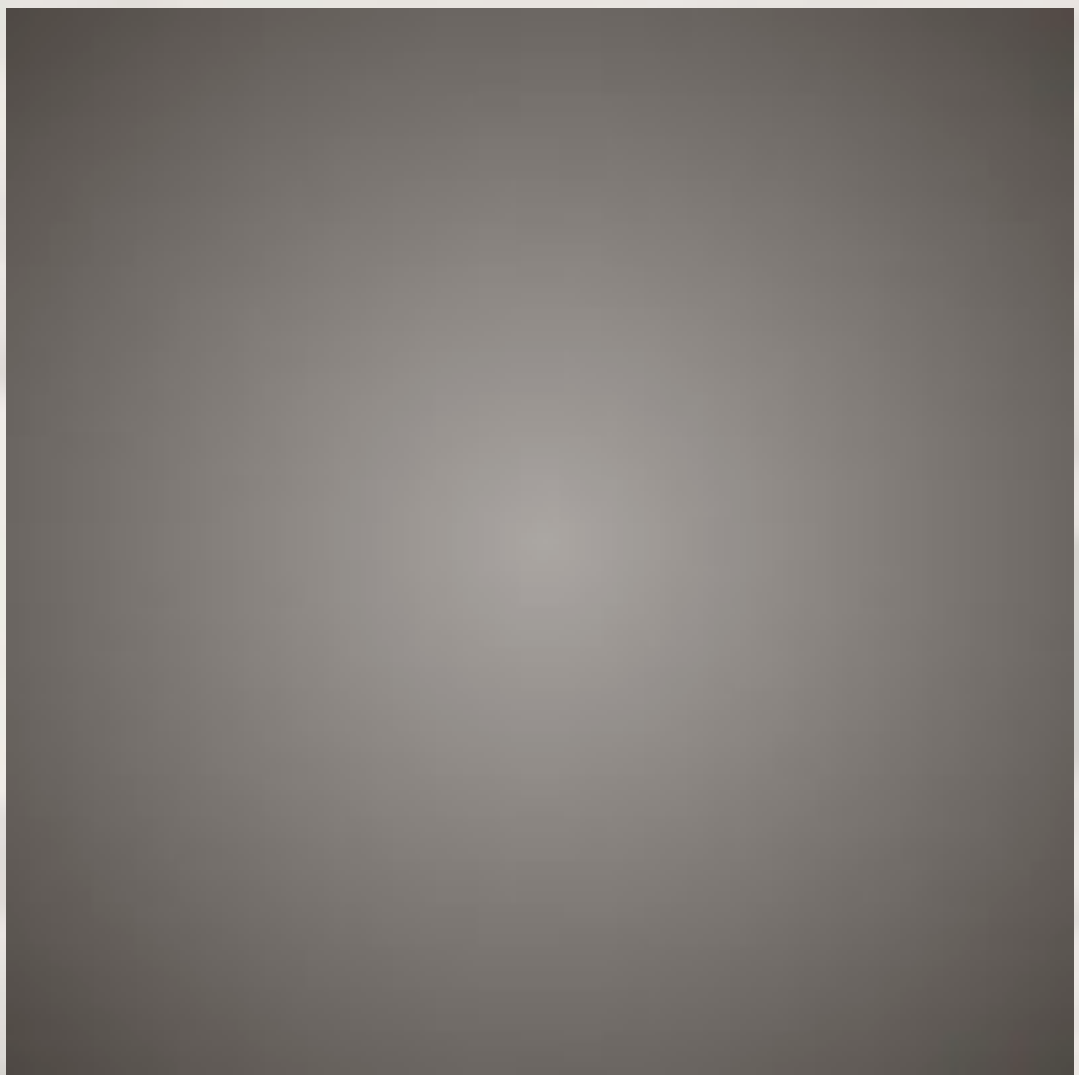
application.sass

```
@import "compass"

.content
  +background(radial-gradient(center, #9b9592, #3c3733))
```

2.6 Gradients





`radial-gradient(center, #9b9592, #3c3733)`

More robust color stops are also supported:

application.sass

```
@import "compass"

.content
  +background(linear-gradient(top, #9b9592, #e79e23 75%, #3c3733))
```



2.6 Gradients





```
linear-gradient(top, #9b9592, #e79e23 75%, #3c3733)
```

3 UTILITY



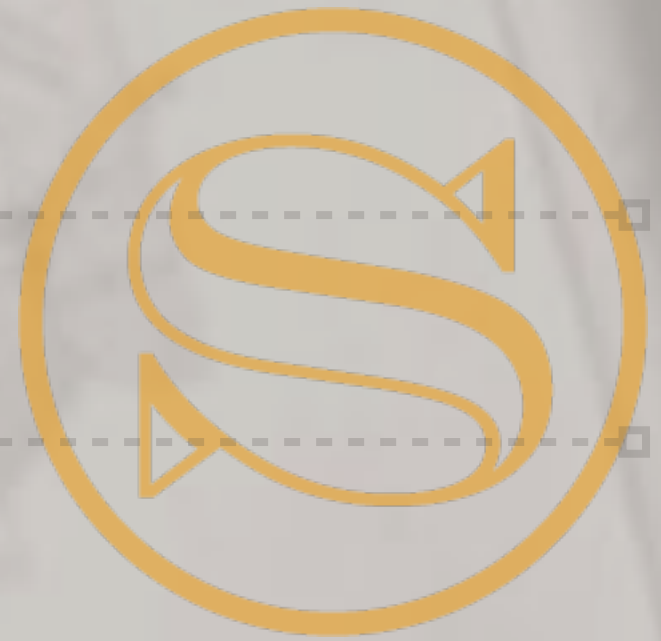
3.1 Use With Care

3.2 Helpers

3.3 Utilities & Stretch

3.4 Image Dimensions

3.5 Inline Data



- There's a **ton** of helpers and utilities in Compass:
<http://compass-style.org/reference/compass>
- We'll be going through a few of our favorites
- Not everything is optimal or current - **always** be aware of the output

3.1 Use With Care



application.sass

```
@import "compass"

%group,
.group
+clearfix
```



application.css

```
.group {
  overflow: hidden;
  *zoom: 1;
}
```

3.1 Use With Care



application.sass

```
@import "compass"
```



```
%group,  
.group  
+clearfix
```

application.sass

```
%group,  
.group  
  &:after  
    content: ''  
    display: table  
    clear: both
```



3.1 Use With Care



3.1 Use With Care

3.2 Helpers

3.3 Utilities & Stretch

3.4 Image Dimensions

3.5 Inline Data

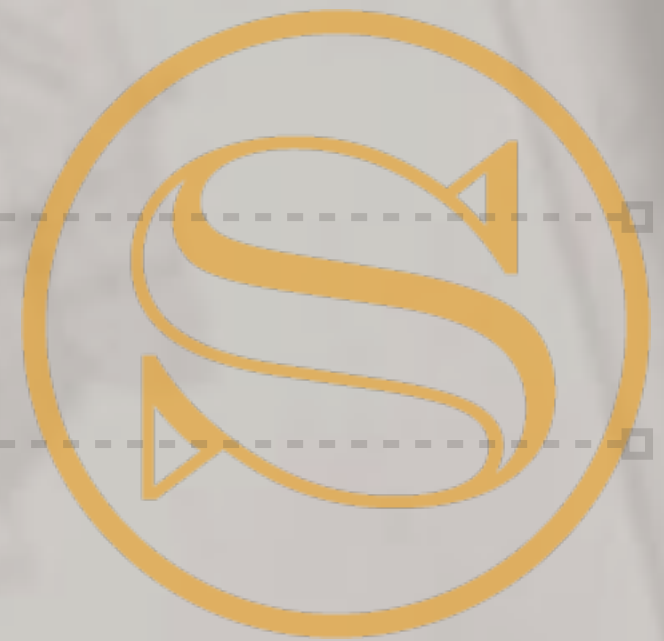


image-url and font-url are available post-configuration:

application.sass

```
@import "compass"

body
  background: image-url( 'bg-body.png' ) repeat
```



application.css

```
body {
  background: url( '/images/bg-body.png' ) repeat;
}
```

3.2 Helpers



scale-lightness shortcuts our scale_color use:

application.sass

```
.content  
  color: scale_color(#eee, $lightness: 7%)
```



application.sass

```
@import "compass"
```

```
.content  
  color: scale-lightness(#eee, 7%)
```



use a negative
value to darken



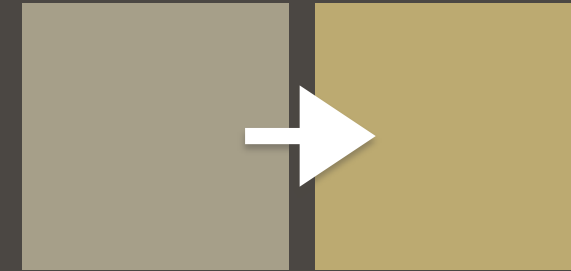
3.2 Helpers



scale-saturation also shortcuts scale_color:

application.sass

```
.content  
  color: scale_color(#a99f88, $saturation: 30%)
```



application.sass



```
@import "compass"
```

```
.content  
  color: scale-saturation(#a99f88, 30%)
```



3.2 Helpers



Assembly Tip

Color obsessed? Additional
Compass color functions
include `shade` and `tint`



Opposite Position

Returns the opposite side (or pair):

`application.sass`

```
@import "compass"

opposite-position(top) // bottom
opposite-position(left) // right
opposite-position(right bottom) // left top
```

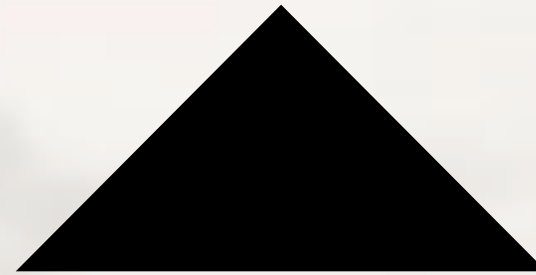
3.2 Helpers



Using opposite-position - CSS shapes:

application.sass

```
.caret  
  border: 100px solid transparent  
  border-bottom: 100px solid #000  
  border-top: 0  
  height: 0  
  width: 0
```



3.2 Helpers

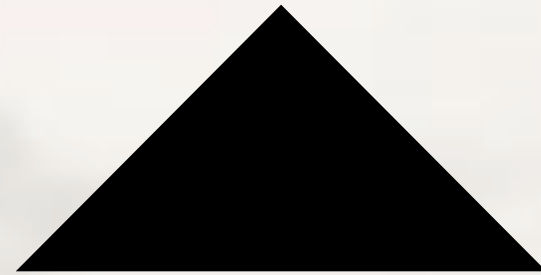


Using opposite-position - CSS shapes:

application.sass

```
=caret($point)
  border: 100px solid transparent
  border-bottom: 100px solid #000
  border-top: 0
  height: 0
  width: 0

.caret
  +caret(top)
```



3.2 Helpers

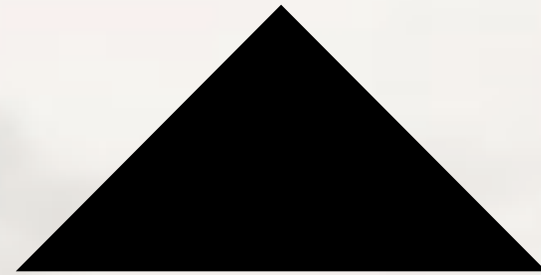


Using opposite-position - CSS shapes:

application.sass

```
=caret($point)
  $opposite: opposite-position($point)
  border: 100px solid transparent
  border-bottom: 100px solid #000
  border-top: 0
  height: 0
  width: 0

.caret
  +caret(top)
```



3.2 Helpers



Using opposite-position - CSS shapes:

application.sass

```
=caret($point)
  $opposite: opposite-position($point)
  border: 100px solid transparent
  border-#{$opposite}: 100px solid #000
  border-#{$point}: 0
  height: 0
  width: 0

.caret
  +caret(top)
```

Diagram illustrating the CSS shape creation for a caret. The code defines a function `=caret($point)` that uses the `opposite-position` helper to determine the opposite position of the specified `$point`. The shape is created using a border with a width of 100px and a height of 0. The border is transparent, except for the side specified by `$point`, which is solid black. The opposite side is also solid black. The diagram shows a black triangle pointing upwards, representing the caret shape. Arrows point from the labels `top` and `bottom` to the corresponding border properties in the code.



3.2 Helpers



Using opposite-position - CSS shapes:

application.sass

```
=caret($point)
  $opposite: opposite-position($point)
  border: 100px solid transparent
  border-#{$opposite}: 100px solid #000
  border-#{$point}: 0
  height: 0
  width: 0

.caret
  +caret(left)
```



3.2 Helpers



Assembly Tip

Responsive graphs?
Compass adds math
functions like `pi`, `sin`, `cos`,
`tan`, `e`, `log`, `sqrt`, and `pow`



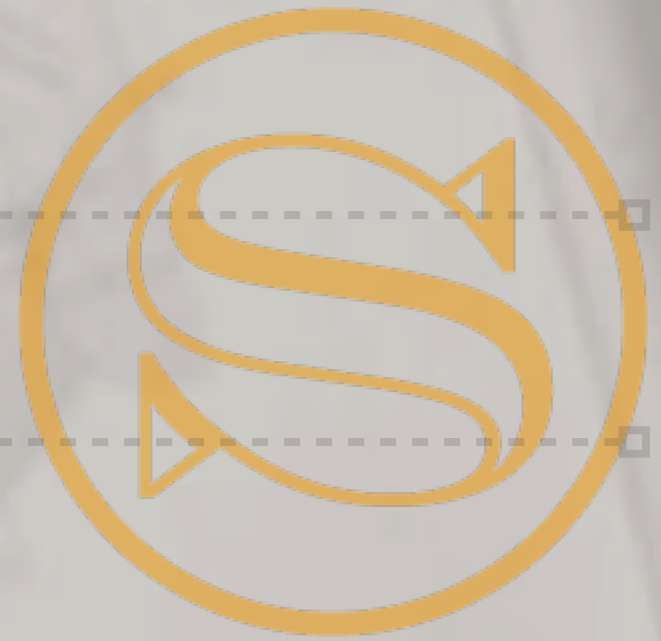
3.1 Use With Care

3.2 Helpers

3.3 Utilities & Stretch

3.4 Image Dimensions

3.5 Inline Data



Contrast Color

Returns a light or dark color based on an input color's lightness:

```
contrast-color(input-color, dark-color, light-color, threshold)
```



application.sass

```
@function button-text($bg)
  @return // which color?

.btn-a
  background: #222
  color: button-text(#222)

.btn-b
  background: #aaa
  color: button-text(#aaa)
```

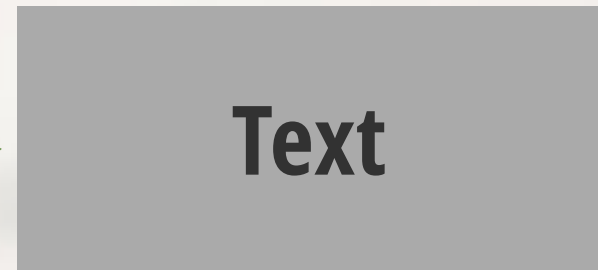
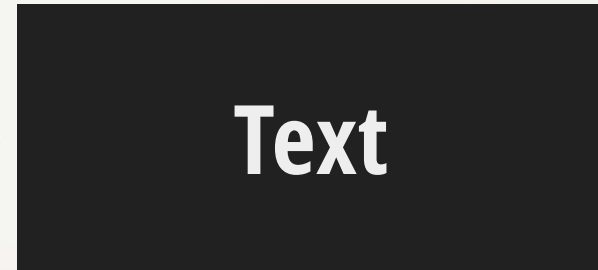
3.3 Utilities & Stretch



```
@function button-text($bg)
  @return contrast-color($bg, #333, #eee, 50%)

.btn-a
  background: #222
  color: button-text(#222) // #eee

.btn-b
  background: #aaa
  color: button-text(#aaa) // #333
```



3.3 Utilities & Stretch



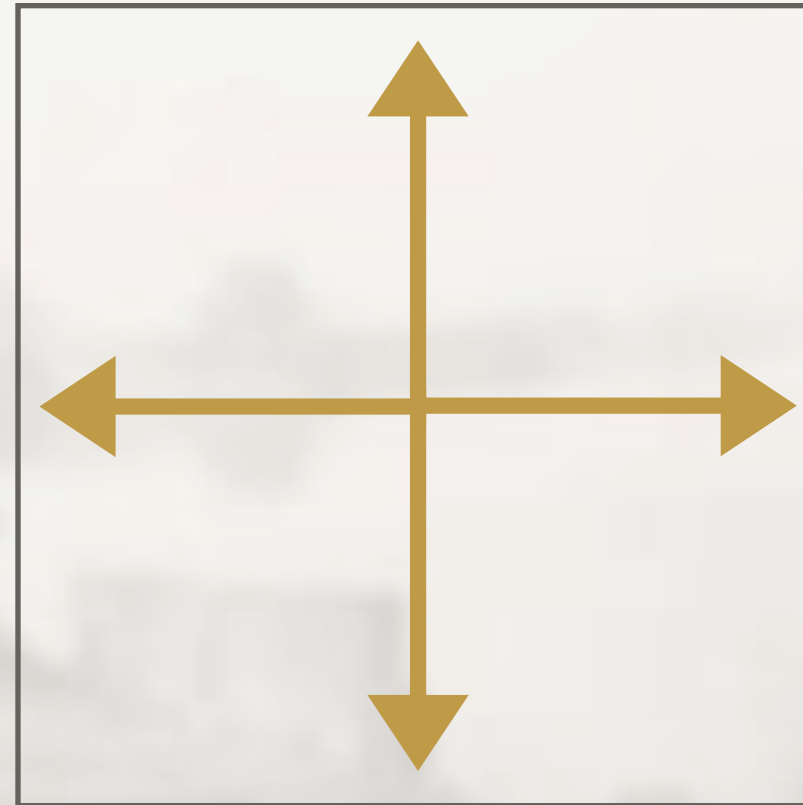
application.sass

```
.content
  height: 400px
  position: relative
  width: 400px

.sidebar
  background: #be9947
```

index.html

```
<section class="content">
  <aside class="sidebar"></aside>
</section>
```



we'd like .sidebar
to stick to
.content's edges

3.3 Utilities & Stretch



application.sass

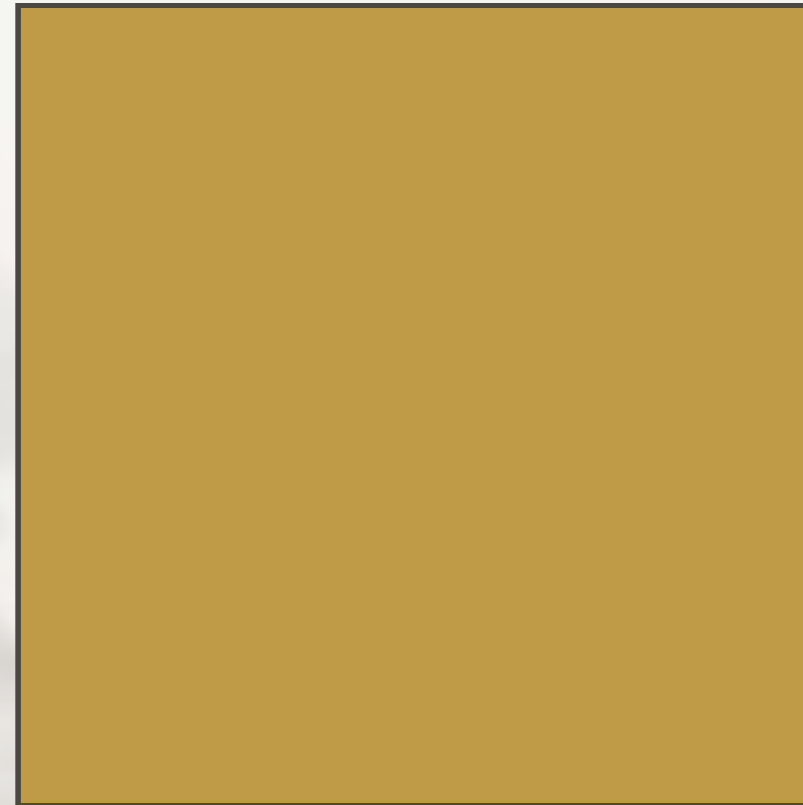
```
.content
  height: 400px
  position: relative
  width: 400px

.sidebar
  background: #be9947
  position: absolute
  top: 0
  right: 0
  bottom: 0
  left: 0
```



index.html

```
<section class="content">
  <aside class="sidebar"></aside>
</section>
```



3.3 Utilities & Stretch



application.sass

```
@import "compass/layout"
```

```
.content
```

```
  height: 400px
```

```
  position: relative
```

```
  width: 400px
```

```
.sidebar
```

```
  background: #be9947
```

```
  +stretch
```

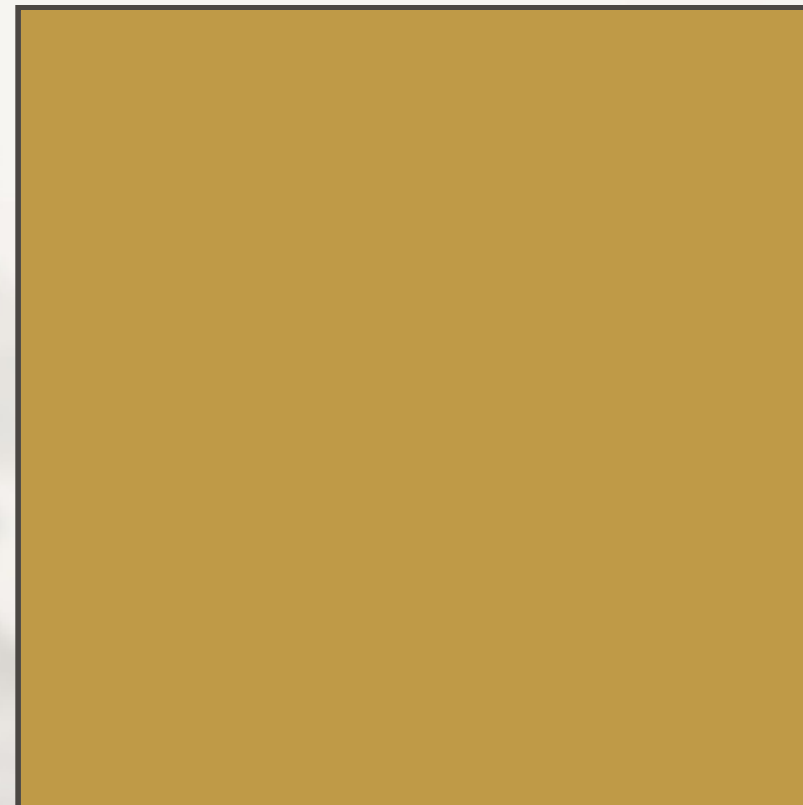


index.html

```
<section class="content">
```

```
  <aside class="sidebar"></aside>
```

```
</section>
```



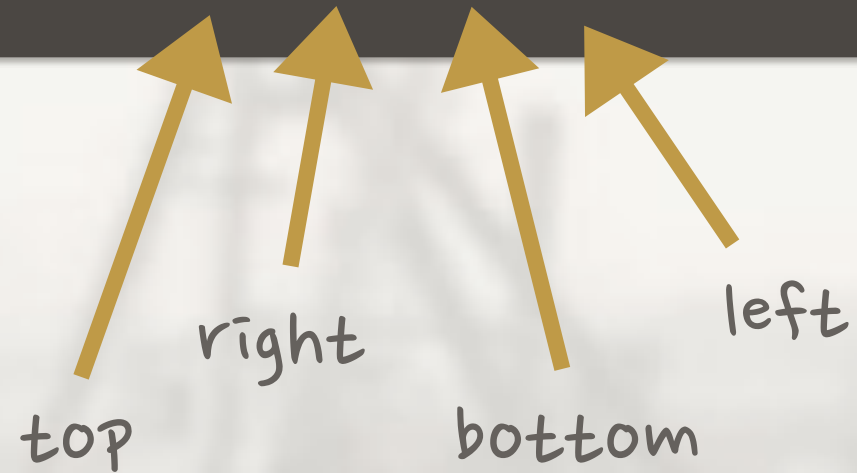
3.3 Utilities & Stretch



Stretch

Outputs positioning for each side of a container:

```
+stretch(0, 0, 0, 0)
```



3.3 Utilities & Stretch



application.sass

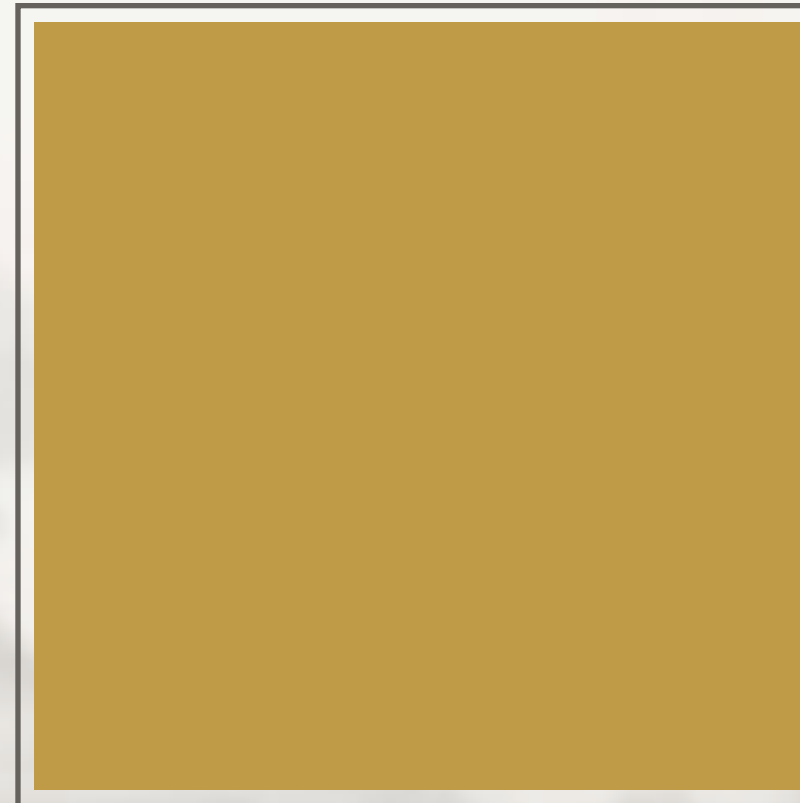
```
@include "compass/layout"

.content
  height: 400px
  position: relative
  width: 400px

.sidebar
  background: #be9947
  +stretch(5px, 5px, 5px, 5px)
```

index.html

```
<section class="content">
  <aside class="sidebar"></aside>
</section>
```



3.3 Utilities & Stretch



application.sass

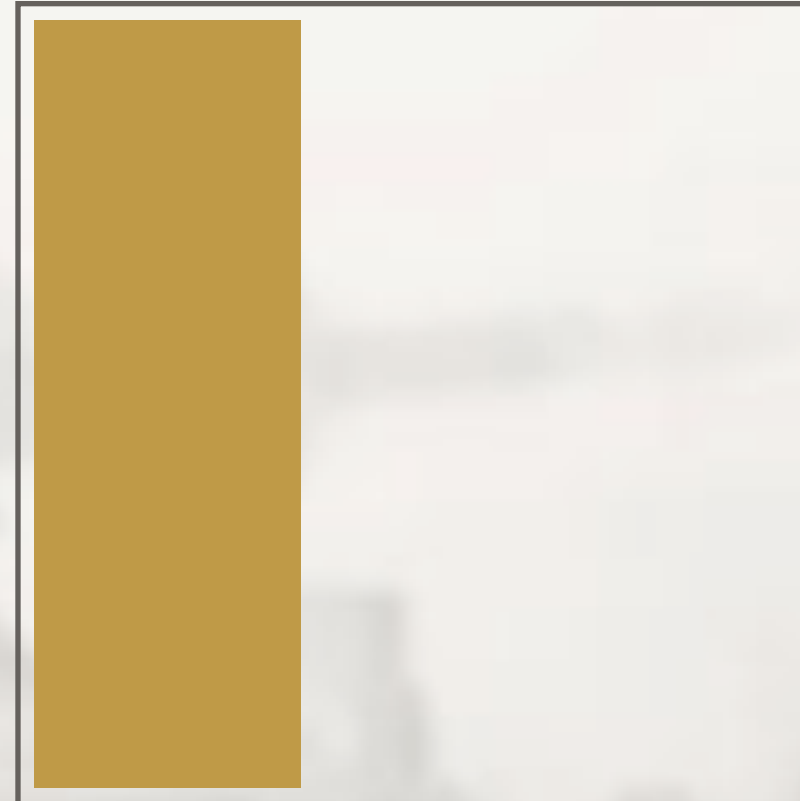
```
@include "compass/layout"

.content
  height: 400px
  position: relative
  width: 400px

.sidebar
  background: #be9947
  +stretch(5px, auto, 5px, 5px)
  width: 100px
```

index.html

```
<section class="content">
  <aside class="sidebar"></aside>
</section>
```



3.3 Utilities & Stretch



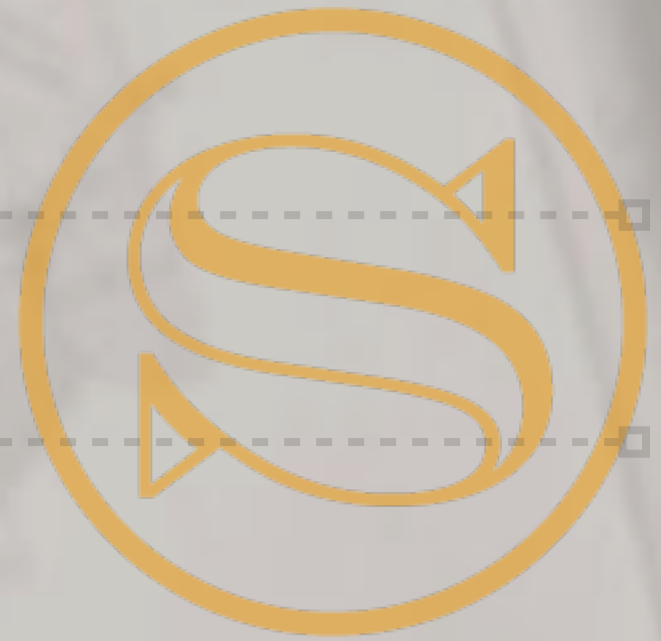
3.1 Use With Care

3.2 Helpers

3.3 Utilities & Stretch

3.4 Image Dimensions

3.5 Inline Data





s.png 236 x 236

application.sass

```
@import "compass"

.logo
  background: image-url('s.png')
  height: 236px
  width: 236px
```

application.css

```
.logo {
  background: url('/images/s.png');
  height: 236px;
  width: 236px;
}
```



3.4 Image Dimensions



application.sass

```
@import "compass"

.logo
  background: image-url('s.png')
  height: image-height('s.png')
  width: 236px
```

application.css

```
.logo {
  background: url('/images/s.png');
  height: 236px;
  width: 236px;
}
```



3.4 Image Dimensions



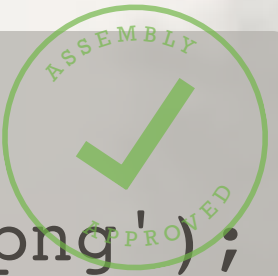
application.sass

```
@import "compass"

.logo
  background: image-url('s.png')
  height: image-height('s.png')
  width: image-width('s.png')
```

application.css

```
.logo {
  background: url('/images/s.png');
  height: 236px;
  width: 236px;
}
```



3.4 Image Dimensions



Use `inline-image` to base64 embed images into CSS:

application.sass

```
@import "compass"

.logo
  background: inline-image('s.png')
  height: image-height('s.png')
  width: image-width('s.png')
```

application.css

```
.logo {
  background: url('data:image/
png;base64,iVBORw0KGgoAAAANSUh
EUgAAAGQAAABkCAMAABHPGVmAAAAG
XRFWHRTb2Z0d2FyZQBBZO30bfS0uzz
2vD2rd3qkdLj4fP3ULjS2O/1...');
  height: 236px;
  width: 236px;
}
```

3.5 Inline Data



Inline Image

- Embeds the image into CSS (no more loading flash)
- ~10% image size increase, but reduces HTTP requests
- IE8+ (max 32KB), good mobile support



4

RHYTHM

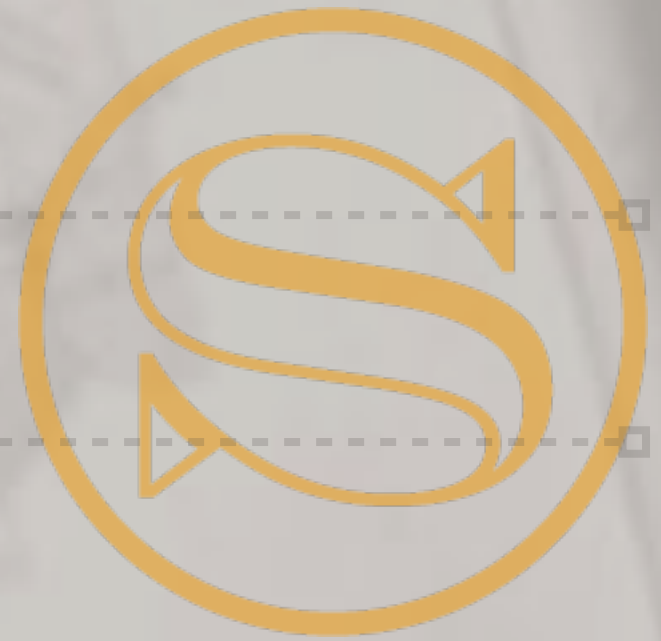


4.1 Vertical Rhythm

4.2 Baseline Setup

4.3 Size + Leading

4.4 Margin, Padding, + Borders

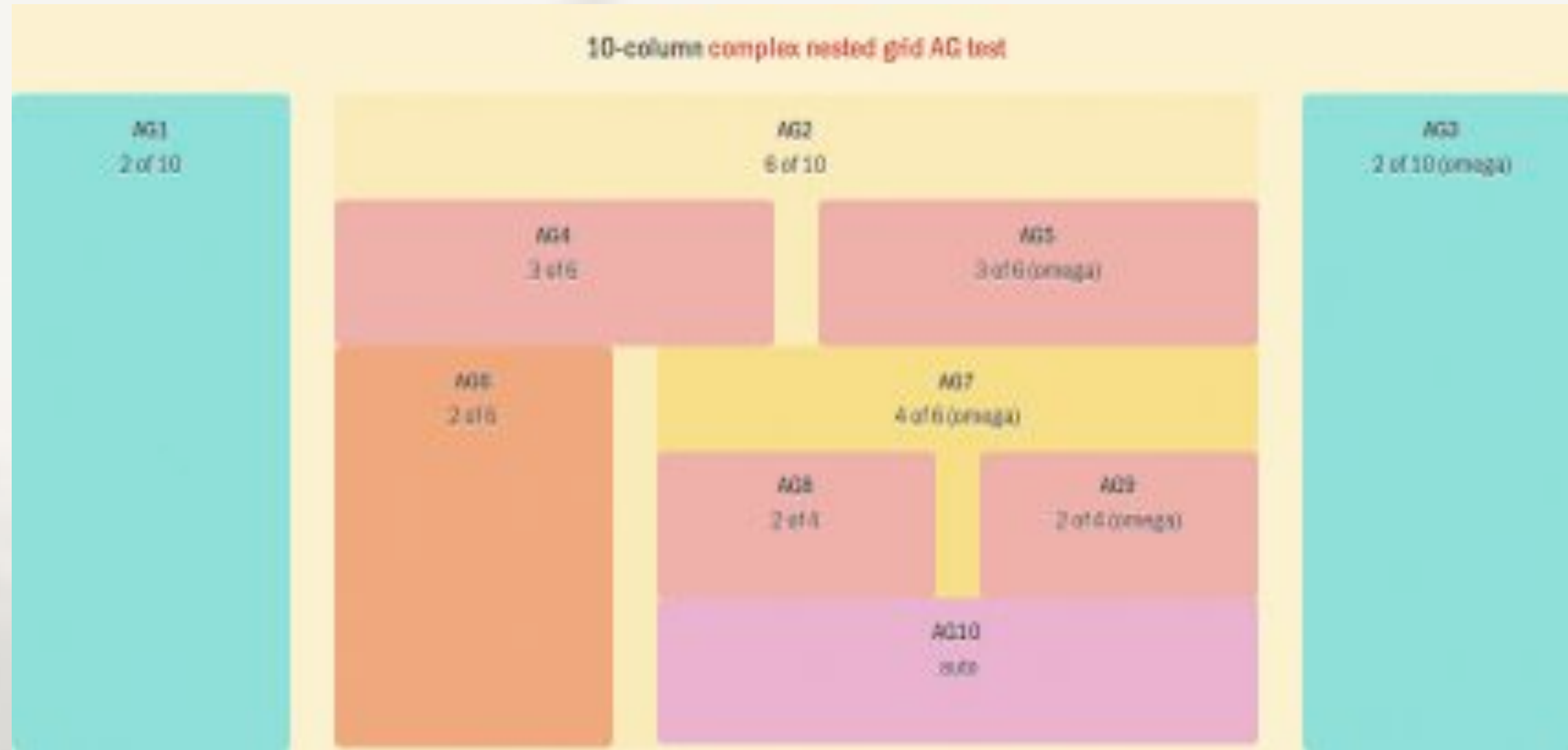


Grids

1. Horizontal



Susy - <http://susy.oddbird.net>



4.1 Vertical Rhythm



Grids

~~1. Horizontal~~

2. Vertical (Baseline)



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

We'll be adding rhythm to this snippet of HTML:

index.html

```
<h1>Rivets, Welds, and You</h1>
```

```
<p>High-quality welding work is a necessity where promotion  
consideration is concerned. One who neglects the smallest  
detail cannot be put in charge of the whole.</p>
```

```
<blockquote>Assembly is, and will always be, in the  
details.</blockquote>
```

```
<p>Rivets don't merely give metalwork character - they serve  
a vital function in joining pieces with strong cohesion. Use  
liberally where necessary, and always with care.</p>
```

4.1 Vertical Rhythm



Eyeballing tends to result in magic numbers and guesses:

application.sass

```
body
  font-size: 18px
  line-height: 1.66667
h1
  border-bottom: 2px solid #333
  font-size: 30px
  margin-bottom: 30px
blockquote
  background: #ccc
  line-height: 3.33333
  text-align: center
p
  margin: 18px 0
```



4.1 Vertical Rhythm



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

Feel the Rhythm, Feel the Rhyme

- We'll add a baseline grid and allow **it** to manage spacing, line heights, and margins
- Starting with just a reset:

4.2 Baseline Setup



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

Begin with Compass global variables and establish-baseline:

application.sass

```
@import "compass"

$base-font-size: 18px
$base-line-height: 30px
+establish-baseline
```

application.css

```
html {
  font-size: 112.5%;
  line-height: 1.667em;
}
```



4.2 Baseline Setup



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
$base-font-size: 18px
```

```
$base-line-height: 30px
```

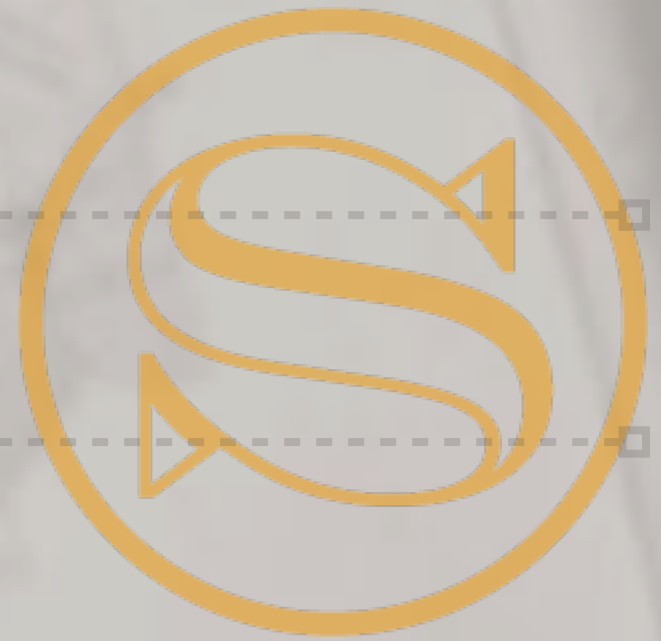
```
+establish-baseline
```

4.1 Vertical Rhythm

4.2 Baseline Setup

4.3 Size + Leading

4.4 Margin, Padding, + Borders



adjust-font-size-to keeps size alterations from disrupting the grid:

application.sass

```
@import "compass"

...

h1
  +adjust-font-size-to(50px)
```

application.css

```
h1 {
  font-size: 2.778em;
  line-height: 1.2em;
}
```



4.3 Size + Leading





Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
h1
```

```
+adjust-font-size-to(50px)
```

Typography Corner

Leading (distance between baselines) originates from typesetting by hand - strips of lead separated lines of text



adjust-leading-to tweaks the amount of vertical space occupied:

application.sass

```
@import "compass"
...

blockquote
  +adjust-leading-to(2)
  background: #ccc
```

application.css

```
blockquote {
  line-height: 3.333em;
  background: #cccccc;
}
```



4.3 Size + Leading



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.



Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
blockquote
```

```
  +adjust-leading-to(2)
```

```
  background: #ccc
```

Rhythm mixins need to be notified of font size adjustments:

application.sass

```
@import "compass"

...

blockquote
  +adjust-font-size-to(20px)
  +adjust-leading-to(2)
  background: #ccc
```



application.css

```
blockquote {
  font-size: 1.111em;
  line-height: 1.5em;
  line-height: 3.333em;
  background: #cccccc;
}
```



4.3 Size + Leading



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.



```
@import "compass"
```

```
...
```

```
blockquote
```

```
+adjust-font-size-to(20px)
```

```
+adjust-leading-to(2)
```

```
background: #ccc
```

Rhythm mixins need to be notified of font size adjustments:

application.sass

```
@import "compass"

...

blockquote
  +adjust-font-size-to(20px)
  +adjust-leading-to(2, 20px)
  background: #ccc
```



application.css

```
blockquote {
  font-size: 1.111em;
  line-height: 1.5em;
  line-height: 3em;
  background: #cccccc;
}
```



4.3 Size + Leading



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
blockquote
```

```
+adjust-font-size-to(20px)
```

```
+adjust-leading-to(2, 20px)
```

```
background: #ccc
```



Always be wary of the CSS generated by your Sass file:

application.sass

```
@import "compass"

...

blockquote
  +adjust-font-size-to(20px)
  +adjust-leading-to(2, 20px)
  background: #ccc
```

application.css

```
blockquote {
  font-size: 1.111em;
  line-height: 1.5em;
  line-height: 3em;
  background: #cccccc;
}
```



4.3 Size + Leading



Assembly Tip

Use Vertical Rhythm
carefully - misuse or
overuse can cause bloat

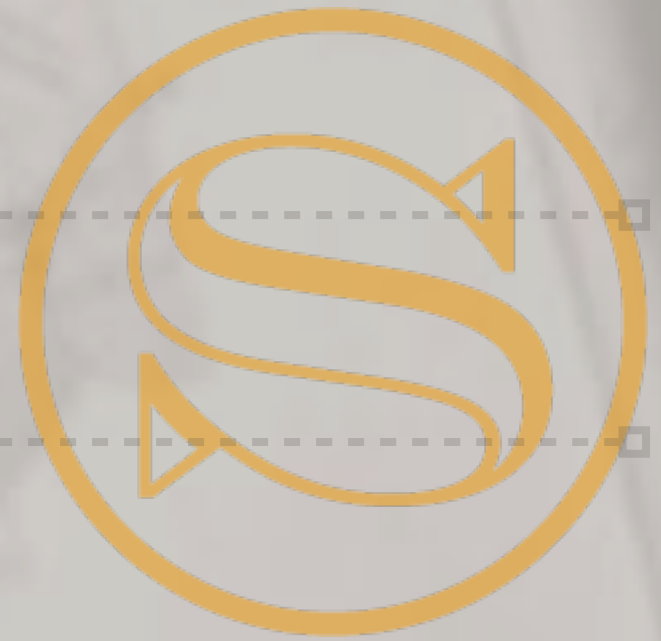


4.1 Vertical Rhythm

4.2 Baseline Setup

4.3 Size + Leading

4.4 Margin, Padding, + Borders



leader and trailer are used to add margin:

application.sass

```
@import "compass"

...

p
  +leader(1)
  +trailer(1)
```

application.css

```
p {
  margin-top: 1.667em;
  margin-bottom: 1.667em;
}
```



4.4 Margin, Padding, + Borders



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
p
```

```
+leader(1)  
+trailer(1)
```

Reminder - notify rhythm mixins of font size adjustments:

application.sass

```
@import "compass"

...

h1
  +adjust-font-size-to(50px)
  +trailer(2, 50px)
```



application.css

```
h1 {
  font-size: 2.778em;
  line-height: 1.2em;
  margin-bottom: 1.2em;
}
```



4.4 Margin, Padding, + Borders



Rivets, Welds, and You



High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
h1
```

```
+adjust-font-size-to(50px)
```

```
+trailer(2, 50px)
```

Other Spacing Mixins

- padding-leader
- padding-trailer
- rhythm

h2

```
+rhythm(1, 1, 1, 2, 36px)
```

leader

padding-leader

padding-trailer

trailer

adjusted font size

4.4 Margin, Padding, + Borders



Standard borders throw off our grid:

application.sass

```
@import "compass"

...

h1
  +adjust-font-size-to(50px
  +trailer(2, 50px)
  border-bottom: 10px solid #000
```



application.css

```
h1 {
  font-size: 2.778em;
  line-height: 1.2em;
  margin-bottom: 1.2em;
  border-bottom: 10px solid #000;
}
```



4.4 Margin, Padding, + Borders





Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
h1
```

```
+adjust-font-size-to(50px
```

```
+trailer(2, 50px)
```

```
border-bottom: 10px solid #000
```



leading-border and trailing-border are grid-friendly alternatives:

application.sass

```
@import "compass"
...
h1
  +adjust-font-size-to(50px
  +trailer(2, 50px)
  +trailing-border(10px, 1, 50px)
```



border width

units of padding

font size,
if necessary

application.css

```
h1 {
  font-size: 2.778em;
  line-height: 1.2em;
  margin-bottom: 1.2em;
  border-bottom-style: solid;
  border-bottom-width: 0.2em;
  padding-bottom: 0.4em;
}
```

4.4 Margin, Padding, + Borders



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
h1
```

```
+adjust-font-size-to(50px
```

```
+trailer(2, 50px)
```

```
+trailing-border(10px, 1, 50px)
```



rhythm-borders adds border and uniform whitespace:

application.sass

```
@import "compass"

...

blockquote
  +adjust-font-size-to(20px)
  +rhythm-borders(2px, 1, 20px)
```



application.css

```
blockquote {
  font-size: 1.111em;
  line-height: 1.5em;
  border-style: solid;
  border-width: 0.1em;
  padding: 1.4em;
}
```

4.4 Margin, Padding, + Borders



Rivets, Welds, and You

High-quality welding work is a necessity where promotion consideration is concerned. One who neglects the smallest detail cannot be put in charge of the whole.

Assembly is, and will always be, in the details.

Rivets don't merely give metalwork character - they serve a vital function in joining pieces with strong cohesion. Use liberally where necessary, and always with care.

```
@import "compass"
```

```
...
```

```
blockquote
```

```
  +adjust-font-size-to(20px)
```

```
  +rhythm-borders(2px, 1, 20px)
```

```
@import "compass"
$base-font-size: 18px
$base-line-height: 30px
+establish-baseline

h1
  +adjust-font-size-to(50px)
  +trailer(2, 50px)
  +trailing-border(10px, 1, 50px)
blockquote
  +adjust-font-size-to(20px)
  +rhythm-borders(2px, 1, 20px)
p
  +leader(1)
  +trailer(1)
```



4.4 Margin, Padding, + Borders



Assembly Tip

Baseline grids are
difficult on the web:
use when appropriate





SPRITE

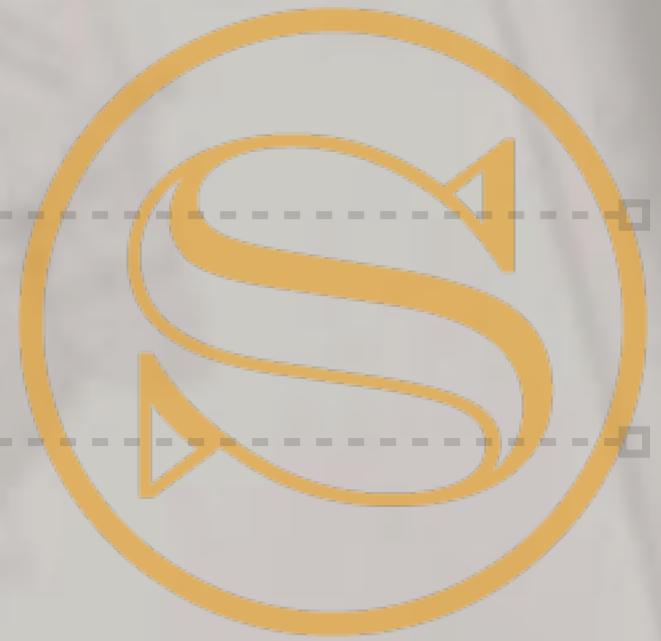


5.1 Sprites Revisited

5.2 Automatic

5.3 Manual

5.4 Further Configuration



- Images combined into one file to reduce HTTP requests
- Utilize `background-position` to shift the image and only show the needed portion
- Not universally loved

5.1 Sprites Revisited





youtube.com

Assembly Tip

Make sure you're using the appropriate solution: sprites, icon fonts, and embedded images all have their place





**images/icons/
cancel.png**
166 x 166



**images/icons/
next.png**
113 x 185



**images/icons/
warning.png**
185 x 169

Using `@import` to generate a sprite:

`application.sass`

```
@import "compass"  
@import "icons/*.png"
```



grabs all .png files
in the icons folder

5.2 Automatic





generated by compass



images/icons-s70d570c2ba.png

application.sass

```
@import "compass"  
@import "icons/*.png"  
+all-icons-sprites
```



the name of the folder
your images are in

application.css

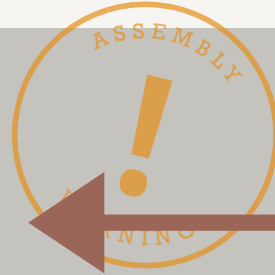
```
.icons-sprite, .icons-cancel,  
.icons-next, .icons-warning {  
  background: url('/images/icons-  
s451834c886.png') no-repeat;  
}  
.icons-cancel {  
  background-position: 0 -169px;  
}  
.icons-next {  
  background-position: 0 -335px;  
}  
.icons-warning {  
  background-position: 0 0;  
}
```



5.2 Automatic



```
.icons-sprite, .icons-cancel,  
.icons-next, .icons-warning {  
  background: url('/images/icons-  
s451834c886.png') no-repeat;  
}  
.icons-cancel {  
  background-position: 0 -169px;  
}  
.icons-next {  
  background-position: 0 -335px;  
}  
.icons-warning {  
  background-position: 0 0;  
}
```



lots of files =
lots of selectors

manual width/height
setting needed

5.2 Automatic

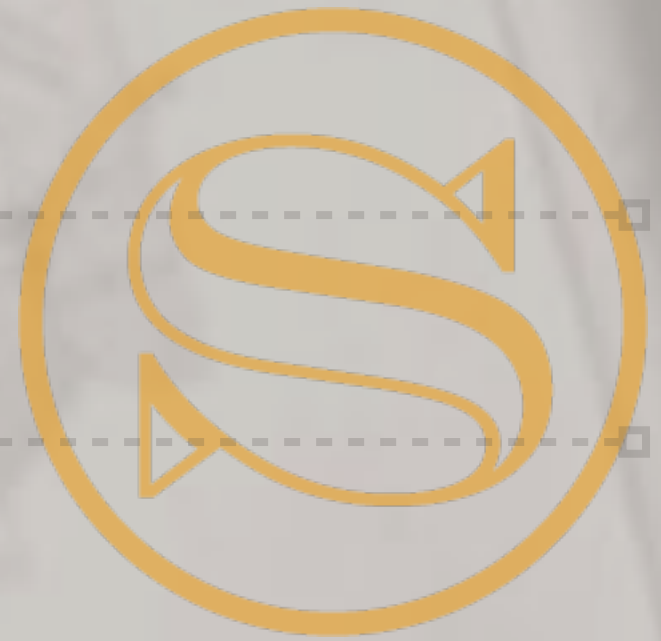


5.1 Sprites Revisited

5.2 Automatic

5.3 Manual

5.4 Further Configuration



Use `sprite-map` instead of `@import` to generate the sprite image:

`application.sass`

```
@import "compass"  
$icons: sprite-map("icons/*.png")
```

doesn't create the
sprite image on its own



We'll use the `i` element (rather than `chain`), similar to Bootstrap:

application.sass

```
@import "compass"
$icons: sprite-map("icons/*.png")

i
  background: $icons
  display: inline-block
```

application.css

```
i {
  background: url('/images/icons-
s23621d5361.png');
  display: inline-block;
}
```



5.3 Manual





images/icons-s23621d5361.png

`sprite-position` automatically finds the coordinates for us:

application.sass

```
@import "compass"
$icons: sprite-map("icons/*.png")

i
  background: $icons
  display: inline-block
.icn-cancel
  background-position: sprite-
    position($icons, cancel)
```

sprite map

file name, sans extension

application.css

```
i {
  background: url('/images/icons-
s23621d5361.png');
  display: inline-block;
}
.icn-cancel {
  background-position: 0 -169px;
}
```

5.3 Manual



`sprite-position` automatically finds the coordinates for us:

`application.sass`

```
@import "compass"
$icons: sprite-map("icons/*.png")

i
  background: $icons
  display: inline-block
.icn-cancel
  background-position: sprite-
    position($icons, cancel)
    +sprite-dimensions($icons, cancel)
```

`application.css`

```
i {
  background: url('/images/icons-
s23621d5361.png');
  display: inline-block;
}
.icn-cancel {
  background-position: 0 -169px;
  height: 166px;
  width: 166px;
}
```

5.3 Manual





```
<i class="icn-cancel"></i>
```



application.sass

```
...  
.icn-cancel  
  background-position: sprite-  
position($icons, cancel)  
  +sprite-dimensions($icons, cancel)  
  
.icn-next  
  background-position: sprite-  
position($icons, next)  
  +sprite-dimensions($icons, next)  
  
.icn-warning  
  background-position: sprite-  
position($icons, warning)  
  +sprite-dimensions($icons, warning)
```



application.css

```
.icn-cancel {  
  background-position: 0 -169px;  
  height: 166px;  
  width: 166px;  
}  
.icn-next {  
  background-position: 0 -335px;  
  height: 185px;  
  width: 113px;  
}  
.icn-warning {  
  background-position: 0 0;  
  height: 169px;  
  width: 185px;  
}
```



5.3 Manual



application.sass

```
...
.icn-cancel
  background-position: sprite-
position($icons, cancel)
  +sprite-dimensions($icons, cancel)

.icn-next
  background-position: sprite-
position($icons, next)
  +sprite-dimensions($icons, next)

.icn-warning
  background-position: sprite-
position($icons, warning)
  +sprite-dimensions($icons, warning)
```



application.css

```
.icn-cancel {
  background-position: 0 -169px;
  height: 166px;
  width: 166px;
}
.icn-next {
  background-position: 0 -335px;
  height: 185px;
  width: 113px;
}
.icn-warning {
  background-position: 0 0;
  height: 169px;
  width: 185px;
}
```



5.3 Manual



application.sass

```
...  
$icon-names: cancel next warning  
  
@each $i in $icon-names  
  .icn-#{$i}  
    background-position: sprite-  
position($icons, $i)  
    +sprite-dimensions($icons, $i)
```



forced to manually
update the list for new files

application.css

```
.icn-cancel {  
  background-position: 0 -169px;  
  height: 166px;  
  width: 166px;  
}  
.icn-next {  
  background-position: 0 -335px;  
  height: 185px;  
  width: 113px;  
}  
.icn-warning {  
  background-position: 0 0;  
  height: 169px;  
  width: 185px;  
}
```



5.3 Manual



application.sass

```
...  
  
@each $i in sprite_names($icons)  
  .icn-#{$i}  
    background-position: sprite-  
position($icons, $i)  
    +sprite-dimensions($icons, $i)
```



application.css

```
.icn-cancel {  
  background-position: 0 -169px;  
  height: 166px;  
  width: 166px;  
}  
.icn-next {  
  background-position: 0 -335px;  
  height: 185px;  
  width: 113px;  
}  
.icn-warning {  
  background-position: 0 0;  
  height: 169px;  
  width: 185px;  
}
```

5.3 Manual



```
@import "compass"
$icons: sprite-map("icons/*.png")

i
  background: $icons
  display: inline-block

@each $i in sprite_names($icons)
  .icn-#{$i}
    background-position: sprite-position($icons, $i)
    +sprite-dimensions($icons, $i)
```



Additional Sprite Options

- The generated sprite can be altered with a number of configuration variables, like spacing:
<http://compass-style.org/help/tutorials/spriting/customization-options/>
- Non-vertical layouts can be set:
<http://compass-style.org/help/tutorials/spriting/sprite-layouts/>

5.4 Further Configuration

