

SASS



Syntactically Awesome Style Sheets

¿Qué es SASS?

- Se puede definir como CSS con superpoderes
- Añade potencia y elegancia a CSS
- Lenguaje de script que es traducido a CSS

Características

- Totalmente compatible con CSS3
- Permite el uso de variables, anidación y mixins
- Funciones avanzadas, como las directivas de control
- Personalización de la salida

Instalación

Dos formas:

- `$ gem install sass`
- Especificando la gema en el fichero Gemfile

```
source 'https://rubygems.org'

gem 'sinatra-contrib'
gem 'thin'
gem 'haml'
gem 'sass'
```

Sintaxis

Dos tipos de sintaxis:

- Por bloques → SCSS

```
h1 {  
  color: red;  
  font: sans-serif;  
}
```

- Por indentación → SASS

```
h1  
  color: red  
  font: sans-serif
```

Formas de uso

- Línea de comandos
 - `$ sass estilo.sass`
 - `$ sass estilo.sass estilo.css`
- Aplicación Web

```
template = File.read('public/sass/style.sass')
sass_engine = Sass::Engine.new(template)
output = sass_engine.render
puts output

fichero = File.new('public/css/style.css', 'w')

fichero.write(output)
fichero.close
```

Comentarios

- Comentarios de una línea

```
// Esto es un comentario de una línea.
```

- Comentarios de varias líneas

```
/* Esto es  
un comentario de  
varias líneas. */
```

Variables

- Facilitan los cambios a la hora de cambiar un mismo valor en varios sitios del código.

```
$color1: blue
$color2: red

h1
  color: $color1
h3
  color: $color1
p
  color: $color2
```


Anidación

- A la hora de modificar los valores de un elemento, cuando está dentro de otro, nos simplifica la tarea.

```
#cabecera  
font: sans-serif  
h1  
color: red  
h2  
color: blue
```

Importación

- Podemos incluir un fichero SASS dentro de otro.

```
@import 'herencia'  
@import 'anidacion'  
  
.centrar  
  text-align: center
```

Parciales

- Los ficheros que empiezan por un guión bajo (_cuerpo.sass), no son traducidos directamente, ya que forman parte de otro.

```
// Fichero _parte1.sass  
.izquierda  
  float: left
```

```
//Fichero _parte2.sass  
.derecha  
  float: right
```

```
@import '_parte1'  
@import '_parte2'
```

Mixins

- Funciones que aceptan el paso de parámetros.

```
// Creación del Mixin
=titulos($color)
  h1
    color: $color
  h2
    color: $color
  h3
    color: $color
  h4
    color: $color
  h5
    color: $color
  h6
    color: $color

// Llamada al Mixin
+titulos(red)
```

Operadores

- Podemos hacer cálculos gracias a los operadores como +, -, *, / o %.

```
.tres-elementos  
width: (900 / 3)px
```

- Útil dentro de un mixin

```
// Creación del Mixin  
=titulos($tamano)  
  .tres-elementos  
    width: ($tamano / 3)px  
  
// Llamada al Mixin  
+titulos(900)
```

Herencia

- Muy útil cuando queremos tener un elemento general, y otros elementos, más específicos, que tienen las mismas características salvo alguna que los distingue.

```
.cuadro
border: 2px solid black
font: sans-serif

.cuadro1
@extend .cuadro
color: green

.cuadro2
@extend .cuadro
color: red
```

Enlaces

Ejemplos publicados en:

<https://github.com/alu0100600674/sass.git>

Documentación de SASS:

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