

TAREA DE INVESTIGACIÓN DEL PREPROCESADOR SASS

Realizado por James Solis Fernandez

Variables Sass

Las variables se definen igual que otros atributos CSS, solo que no están ligados a un selector en concreto. Además, en Sass las variables comienzan siempre con “\$”. Si usamos una variable, podríamos solucionar varias cosas al mismo tiempo, facilitando la codificación y el mantenimiento del código CSS.

Por ejemplo el color de algún texto. Probablemente vayamos a usar ese color en más de una ocasión, asignado a varios selectores. Lo más usual sería escribir ese color en todos los lugares donde lo vayamos a necesitar, repitiendo una y otra vez el mismo valor rgb aquí y allá. Sin embargo, esto da confusiones en el código si queremos realizar algún cambio. Pero con Sass podríamos hacer un mejor uso y mejor entendimiento.

Nesting

Sass te permite guardar tus selectores de CSS de una manera que siga la misma jerarquía de HTML.

Ejemplo:

```
nav {
  ul {
    margin: 0;
    padding: 0;
    list-style: none;
  }
  li {
    display: inline-block;
  }
  a {
    display: block;
    padding: 6px 12px;
    text-decoration: none;
  }
}
```

Mixins

Un mixin te deja hacer grupos de declaraciones de CSS que desea reutilizar en todo su sitio.

Para crear un mixin, usa la directiva `@mixin` y le da un nombre. Hemos nombrado nuestra transformación `mixin`. También `$` dentro del paréntesis para que podamos pasar una transformación de lo que queramos. Después de crear su mixin, puede usarlo como una declaración CSS comenzando con `@include` seguido del nombre del mixin.

Ejemplo:

```
@mixin transform($property) {  
  -webkit-transform: $property;  
  -ms-transform: $property;  
  transform: $property;  
}  
.box { @include transform(rotate(30deg)); }
```

Partials

Es un archivo Sass nombrado con un guión bajo se utilizan con la regla `@use`. Se podría nombrar como `_partial.scss`. El guión bajo permite a Sass saber que el archivo es solo un archivo Partial y que no te debe generar el mismo en un archivo CSS. Se pueden crear archivos Sass Partials que contengan pequeños fragmentos de CSS que puede incluir en otros archivos Sass. Esta es una excelente manera de modularizar su CSS y ayudar a que las cosas sean más fáciles de mantener.

Ejemplos:

```
//application.scss  
  
@import "reset";  
@import "variables";  
@import "functions";  
@import "mixins";  
@import "base";  
@import "buttons";
```

Modules

No tienes que escribir todo tu Sass en un solo archivo. Puedes dividirlo como desees con la regla `@use`.

Esta regla carga otro archivo Sass como módulo, lo que significa que puede hacer referencia a sus variables, mixins y funciones en su archivo Sass con un espacio de nombres basado en el nombre del archivo.

Ejemplos:

```
// _base.scss
$font-stack: Helvetica, sans-serif;
$primary-color: #333;

body {
  font: 100% $font-stack;
  color: $primary-color;
}
```

```
// styles.scss
@use 'base';

.inverse {
  background-color: base.$primary-color;
  color: white;
}
```

```
body {
  font: 100% Helvetica, sans-serif;
  color: #333;
}

.inverse {
  background-color: #333;
  color: white;
}
```

Información e imágenes extraídas de:

- https://www.w3schools.com/sass/sass_nesting.asp
- <https://sass-lang.com/guide>
- https://dev.to/sarah_chima/using-sass-partials-7mh

