



Mismo resultado de estilo con un código más eficaz, ¿es posible?

“La verdad solo se puede encontrar en un lugar: el código.” –Robert C. Martin, Ingeniero de software.

Una cuestión de (ahorro de) tiempo

Los archivos CSS, si bien nos permiten crear sitios web atractivos, por lo general suelen tener muchas líneas de código. Eso implica mayor tiempo invertido en escribir esas líneas (además del foco que tenemos que lograr para cometer la menor cantidad de errores posible).

Como con casi cualquier otra disciplina, los profesionales que se dedican a escribir código tratan de hacerlo cada vez mejor y/o más rápido. Los Full Stack Developers más eficientes buscan reducir la cantidad de código que deben escribir o, dicho de otra manera, hacer que su código sea muy efectivo en el menor tiempo posible. Para esto **existen dentro de CSS algunas herramientas que permiten hacer el maquetado más ágil y simplificado.**

Existen lo que se llaman **preprocesadores de CSS**, que permiten escribir un código muy similar a CSS con algunas pequeñas modificaciones y beneficios, como por ejemplo:

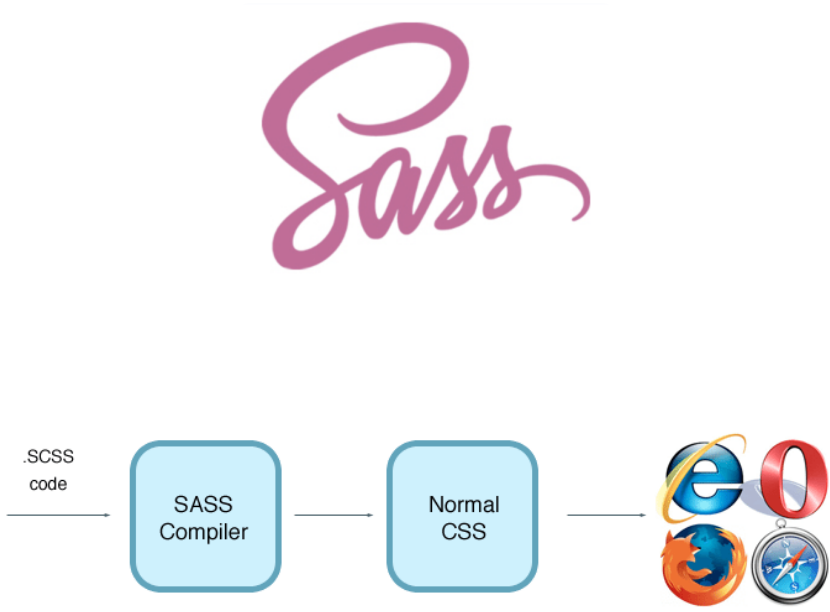
- **Extienden las funcionalidades de un CSS común**, permitiéndonos tener variables, funciones, mixins, reutilizar código, tener más flexibilidad al momento del desarrollo, etc.
- **El anidado de selectores hace que el código sea más fácil de leer**: en vez de escribirlos uno al lado del otro, nos permite ir anidando cada selector, lo cual ahorra muchísimo código y evita entrar en la repetición de elementos.
- **Podemos escribir nuestro código en módulos pero entregar al navegador un solo CSS**: nos evita peticiones HTTP de más al hacer imports
- **Nos ayudan a no repetir código**: los extends nos permiten compartir propiedades de un selector con otro
- **Podemos reutilizar código y modificarlo** mediante Mixins que nos permiten incrustar propiedades de una clase dentro de otra

Ya iremos viendo cómo hacer cada una de estas cosas a medida que trabajemos con los preprocesadores. Los más usados en la actualidad son: [Sass](#), [LESS](#), [Stylus](#) y [PostCSS](#).



Ahora bien, es importante saber que el navegador no puede interpretar el código escrito en el preprocesador de CSS, por lo que, una vez escrito, debemos **compilar el código y transformarlo en CSS convencional** para que el navegador lo pueda **interpretar sin problemas**.

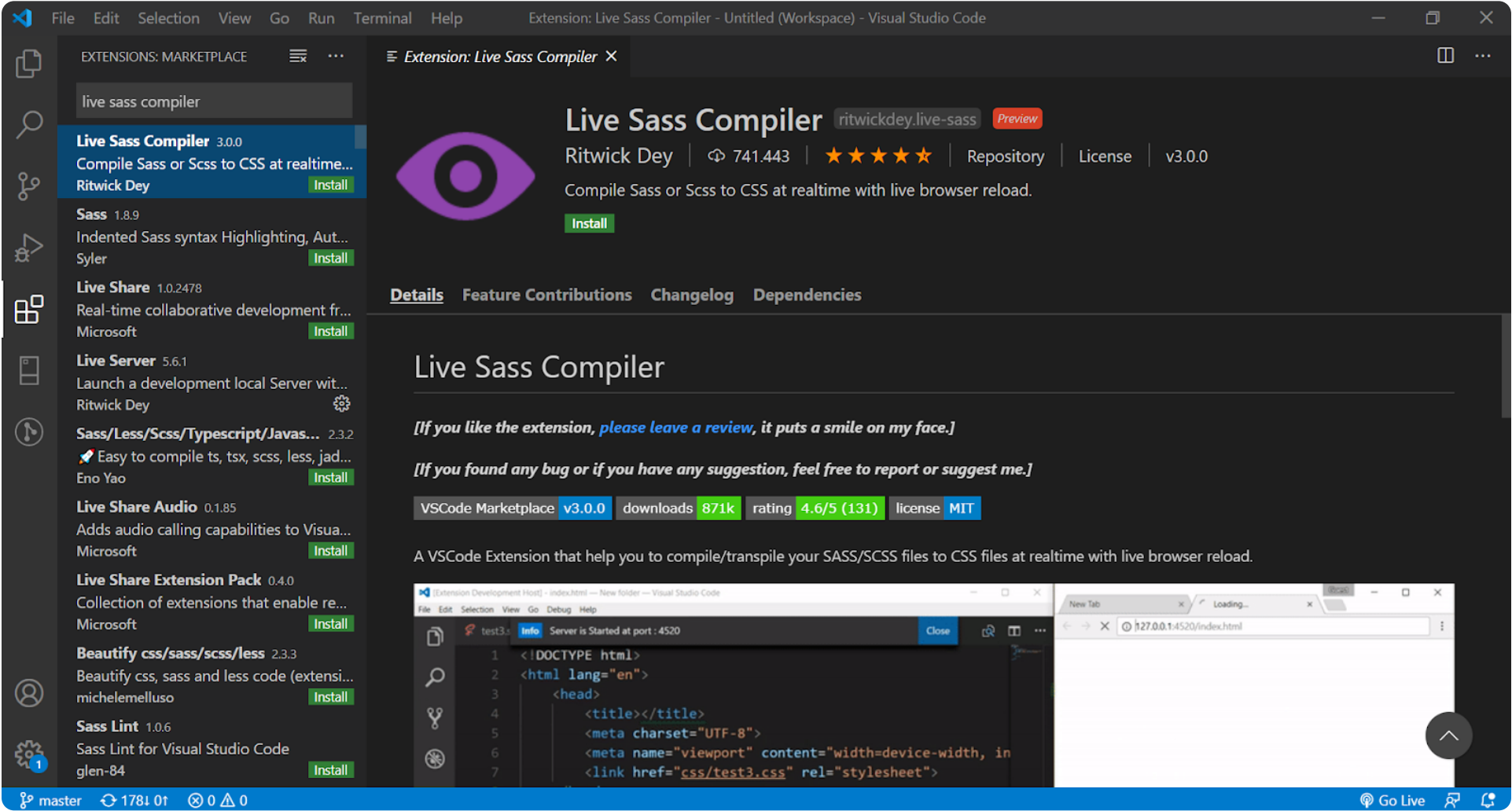
Básicamente, lo que necesitamos hacer es lo que ilustra el siguiente diagrama:



Koala es la herramienta que nos va a permitir convertir el código SASS en CSS. Se encargará de chequear los cambios que se pueden producir en el código para poder tener una hoja de estilos lista.

Queremos aclararte que esta es una primera aproximación a la simplificación del código CSS para ahorrarnos tiempo. Más adelante abordaremos formas más sofisticadas como la librería **node-sass** o la inclusión dentro del *building* de un proyecto. Aún no es momento de llegar a ese punto.

Primeros pasos con Live Sass Compiler



1. Abre el buscador de extensiones de tu Vistual Studio Code.
2. Busca Live Sass Compiler e instala la extensión.
3. Crea un proyecto con la siguiente estructura (presta atención a la extensión):

- `css/style.scss`

4. En el `index` crea un “Hola Mundo” en tag `h1` y dentro del incluye el archivo “`css/style.css`” (presta atención a la extensión)
5. Escribe tu archivo `style.scss` para que establezca un color de fondo al `body` a través de una variable.

```
$primary-color: #e96f6f;  
body{ background-color: $primary-color; }
```

6. Sobre la barra inferior azul de tu VSC presiona “Watch Sass”.
7. Abre tu archivo `index.html` a través del Live Server.

Por qué utilizar los preprocesadores

1. Atomización de estilos

- Mejora en mantenimiento.
- Escalabilidad de las hojas de estilo.
- Centralización de configuraciones generales (a través de variables)

2. Operaciones avanzadas

- Funciones predefinidas para operar con valores.
- Operaciones matemáticas.
- Desarrollo con unidades avanzado.

3. Permite implementar tecnologías

- Para organización de hojas de estilo (Atomic CSS, ITCSS)
- Para organización de estilos (BEM)

Esta es una breve introducción para que sepas de la existencia de los **preprocesadores** y puedas indagar un poco más sobre ellos. De todas formas, lo más importante es que aprendas a usar CSS de la mejor manera posible, para luego optimizar la escritura y usar otras herramientas. En nuestro próximo encuentro vamos compartir una primera aproximación práctica.

¡Prepárate para el próximo encuentro!



Profundiza

Te invitamos a conocer más sobre el tema de esta bitácora.



Herramientas

Programas necesarios para facilitar tu experiencia.



Challenge

Te proponemos el siguiente desafío, ¿te animas?