

Documentación de Uso de SASS

1. Introducción

Este documento proporciona información sobre cómo configurar y utilizar SASS (Syntactically Awesome Stylesheets) en un entorno de desarrollo local. SASS es un preprocesador de hojas de estilo que facilita la escritura de código CSS más eficiente y mantenible.

2. Métodos de Configuración

1. Sass y Node.js:

Instalación:

Tener Node.js instalado en el PC. Luego, ejecutar el siguiente comando para instalar SASS globalmente:

- **npm install -g sass**

Uso:

1. Crear un archivo **.scss** o **.sass**.
2. Compilar el archivo a CSS con el siguiente comando:

- **sass input.scss output.css**

Utilizar la opción **--watch** para compilar automáticamente cuando se realicen cambios:

- **sass --watch input.scss:output.css**

2. Ruby y Sass Gem:

Instalación:

Instalar Ruby en el PC y luego ejecutar el siguiente comando para instalar la gema Sass:

- **gem install sass**

Uso:

1. Crear un archivo **.scss** o **.sass**.
2. Compilar el archivo a CSS con el siguiente comando:

- **sass input.scss output.css**

Utilizar la opción **--watch** para compilar automáticamente:

- **sass --watch input.scss:output.css**

3. Prepros:

Uso:

1. Descargar e instalar Prepros desde prepros.io.
2. Agregar el proyecto a Prepros y configurar las opciones según sea necesario.
3. Prepros compila automáticamente los archivos SASS al guardar.

4. Visual Studio Code (VSCode) y Extensiones:

Uso:

1. Instalar VSCode desde code.visualstudio.com.
2. Instalar una extensión de SASS, como "Live Sass Compiler" o "Sass".
3. Configurar el archivo **settings.json** según sea necesario.
4. Guardar el archivo **.scss** y la extensión lo compila automáticamente a CSS.

5. Gulp o Grunt:

Uso:

1. Configurar el proyecto para usar Gulp o Grunt.
2. Agregar tareas de compilación para SASS.
3. Ejecutar las tareas para compilar automáticamente los archivos SASS.

3. Recomendaciones Generales

- **Variables:** Utilizar variables para almacenar valores reutilizables a lo largo del código.
- **Nesting:** Anidar selectores para mejorar la legibilidad del código.
- **Mixins:** Crear mixins para reutilizar bloques de código.
- **Estructura del Proyecto:** Dividir el código en archivos SASS separados y utilizar la importación según sea necesario.