Preprocesador

<https://developer.mozilla.org/en-US/docs/Glossary/CSS_preprocessor>

A **CSS preprocessor** is a program that lets you generate CSS from the preprocessor's own unique syntax. There are many CSS preprocessors to choose from, however most CSS preprocessors will add some features that don't exist in pure CSS, such as mixin, nesting selector, inheritance selector, and so on. These features make the CSS structure more readable and easier to maintain.

To use a CSS preprocessor, you must install a CSS compiler on your web server; Or use the CSS preprocessor to compile on the development environment, and then upload compiled CSS file to the web server.

Entorno

* Visual Studio Code
* Navegadores: Chrome, Mozilla
* Sass
* Npm

Extensiones VS Code

* AutoCloseTag
* AutoRenameTag
* LiveServer
* HTML CSS Support
* Intellisense for CSS Class Names
* Sass
* Sass Lint
* Scss Intellisense

Instalación

* Como instalación independiente
* Como programa de gestión de paquetes

<https://sass-lang.com/install>

<https://github.com/sass>

Como vemos hay diferentes formas de instalar SaSS. Una de las más populares es usar node.js (node-sass).

**Node:**

javascript runtime environment, which allows us to run javascript applications on the server

npm:

command line programs, which allows us to install and manage packages

**Preparando el entorno de trabajo:**

Instalar node (si no está instalado)

<https://nodejs.org/es/download/>

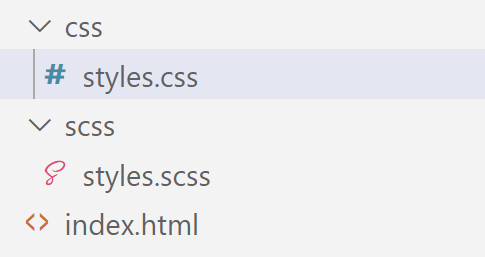
Comprobar node está instalado. Desde la línea de comandos

node --version

Comprobar npm está instalado.

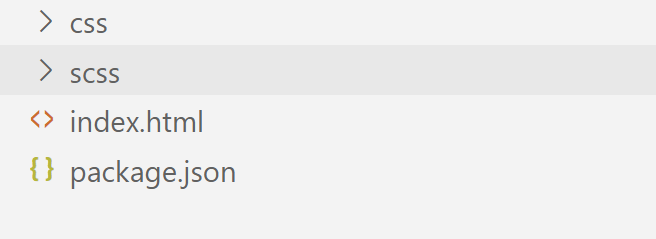
npm --version

¿Cómo vamos a trabajar?

css -> carpeta para estilos definitivos después de la compilación. NUNCA escribo aquí, solo es el resultado de la compilación

scss -> aquí escribimos estilos en SaSS.

Para realizar algunas tareas como la compilación vamos a crear el fichero package.json

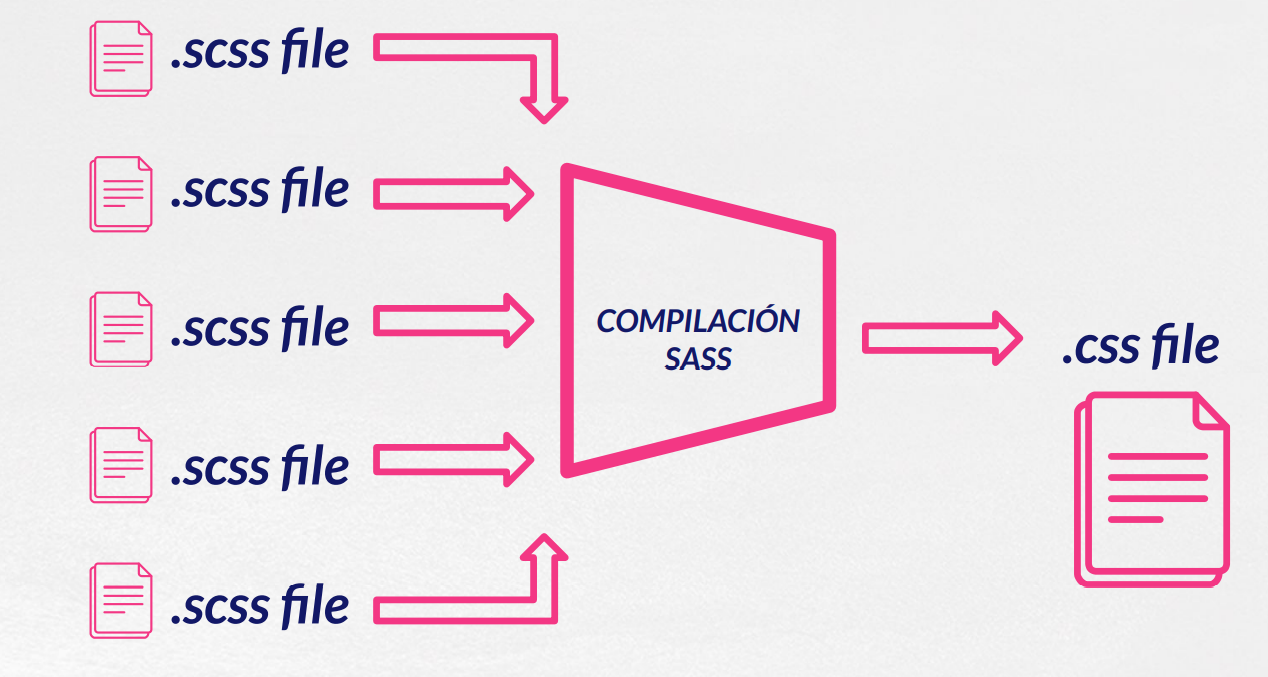
Crear package.json

npm init –yes

Contenido



Esquema general



Elementos básicos

* Variables
* Ámbito de las variables
* Comentarios
* Anidamiento
* Anidamiento referencia al padre
* Mixins
* Funciones
* Extend
* Placeholder

Compilación

**Tipos**

* Simple
* Múltiple
* Expandida
* Comprimida
* Vigilando los cambios

Simple

Sass file.scss output.css

Múltiple

Sass file1.scss:output1.css … filen.scss:outputn.scss

Expandida

Sass –style=expanded File.scss output.css

Comprimida

Sass –style=compressed File.scss output.css

Vigilando

Sass –watch file.scss output.css

Ficheros map

Se generan durante la compilación. Contienen el mapeado entre las reglas SCSS y el resultado CSS.

Comentarios en la compilación

Podemos decidir se queremos que nuestros comentarios aparezcan o no en el CSS

// comentario que no aparece en CSS

/\* Comentario que se incluye menos en compressed\*/

/\*! Comentario que se incluye tembién en compressed\*/

Compilar Sass con node

Instalar node-sass

//opción -g para indicar que es global y lo podremos usar en los demás proyectos

npm i -g node-sass

¿Cómo compilar? En la línea de comandos

Compilación manual

*node-sass -o css scss\styles.scss*

donde

-o css -> carpeta destino, resultado compilación

scss\styles.scss -> lo que quiero compilar

Compilación automática

*node-sass -o css scss\styles.scss* -*w*

En el fichero json

Lo modificamos y añadimos un script para la compilación



Para llamarlo

*npm run watch*

Proyecto restaurante

Preparar estructura de ficheros

Preparar package.json compilación vigilante comprobar que funciona

Hacer estructura de header