

Sass

Sass

Los preprocesadores CSS son herramientas para los desarrolladores de sitios web, que permiten traducir un código de hojas de estilo no estándar, específico del preprocesador en cuestión, a un código CSS estándar, entendible por los navegadores.

Los preprocesadores básicamente nos ofrecen diversas utilidades que a día de hoy no se encuentran en el lenguaje CSS, o bien no son compatibles con todos los navegadores. Ejemplos pueden ser variables, anidación de selectores, funciones (denominadas mixins), etc.

Al desarrollar con un preprocesador se consigue principalmente un ahorro de tiempo, ya que tenemos que escribir menos código para hacer las cosas. Pero también conseguimos una mayor facilidad de mantenimiento del código, gracias a una mejor organización del código y la posibilidad de editar una vez ciertos valores y que afecten a decenas, o cientos, de lugares del código CSS generado.

Sass

¿Qué es sass?

Lo primero que debes conocer de Sass es que existen dos tipos de sintaxis para escribir su código:

Sintaxis Sass: esta sintaxis es un poco diferente de la sintaxis de CSS estándar. No difiere mucho. Por ejemplo, te evita colocar puntos y coma al final de los valores de propiedades. Además, las llaves no se usan y en su lugar se realizan indentados.

Sintaxis SCSS: Es una sintaxis bastante similar a la sintaxis del propio CSS. De hecho, el código CSS es código SCSS válido. Podríamos decir que SCSS es código CSS con algunas cosas extra.

Sass

Instalar Live Server

`npm install -g node-sass`

`npm init -y`

genera un archivo json

Actualizar el archivo json para compilar el estilo:

```
"scripts": {  
  "test": "echo \"Error: no test specified\" && exit 1",  
  "start": "sass ./sass/index.scss ./css/style.css"  
}
```

Sass

```
npm init -y
```

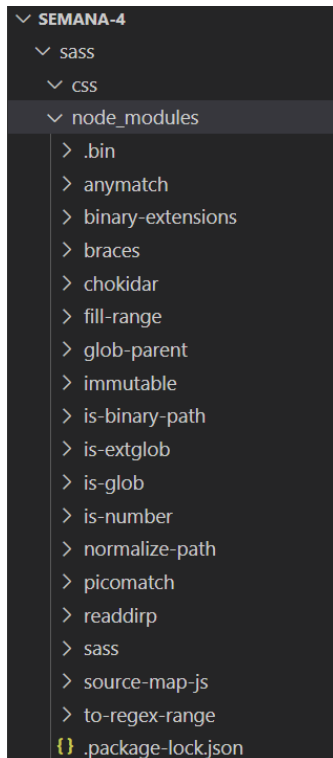
```
{  
  "name": "sass-app",  
  "version": "1.0.0",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
  },  
  "keywords": [],  
  "author": "",  
  "license": "ISC"  
}
```

<- Cambiar nombre

Sass

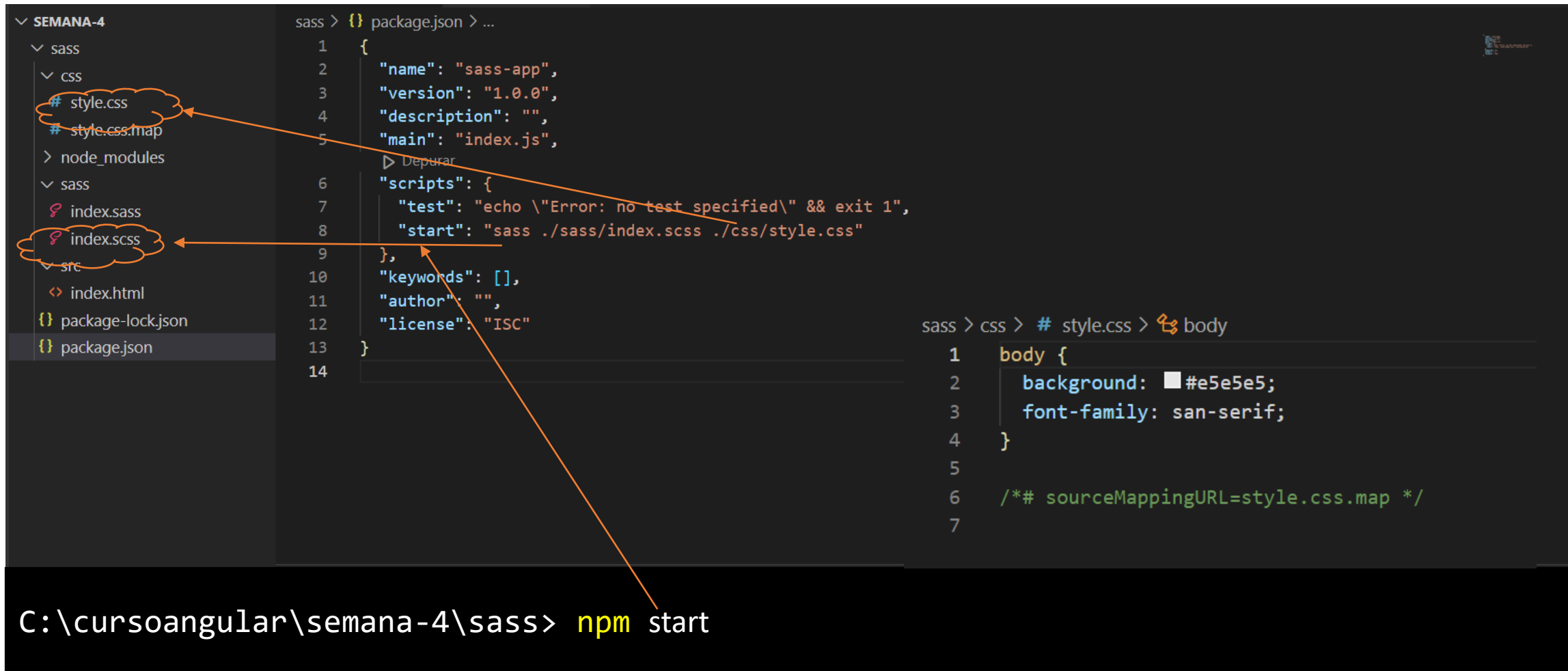
Instalar Sass y paquetes necesarios al proyecto

```
C:\cursoangular\semana-4\sass> npm i sass -D
```



```
sass > sass > index.scss > body
1 // Creacion de Variable
2 $primary-color: #e5e5e5;
3 $primary-font: san-serif;
4
5 body{
6     background: $primary-color;
7     font-family: $primary-font;
8 }
```

Sass



SEMANA-4

- sass
 - css
 - # style.css
 - # style.css.map
 - node_modules
 - sass
 - index.sass
 - index.scss
 - src
 - index.html
 - package-lock.json
 - package.json

```
sass > {} package.json > ...
1  {
2    "name": "sass-app",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1",
8      "start": "sass ./sass/index.scss ./css/style.css"
9    },
10   "keywords": [],
11   "author": "",
12   "license": "ISC"
13 }
14
```

sass > css > # style.css > body

```
1  body {
2    background: #e5e5e5;
3    font-family: san-serif;
4  }
5
6  /*# sourceMappingURL=style.css.map */
7
```

C:\cursoangular\semana-4\sass> npm start

Sass

```
"start": "sass ./sass/index.scss ./css/style.css --watch"
```

```
PS C:\cursoangular\semana-4\sass> npm start
```

```
> sass-app@1.0.0 start
```

```
> sass ./sass/index.scss ./css/style.css --watch
```

```
Sass is watching for changes. Press Ctrl-C to stop.
```


Sass

Compilar el css

Instalar:

Live Server

Live Sass Compiler