***Instalación nodejs - npm - sass - nodemon***

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

1ro: Verificar en la consola si tenemos instalado nodejs: npm - - version

2do: si no lo tenes instalado, descargarlo del link de arriba

3ro: crear una carpeta que no sea la del repositorio para probar

4to: entrar en la consola como administrador a la carpeta creada recientemente

5to: en la consola, npm init

6to: package name: nombre del proyecto

Version

Description

Entry point

Test command

Git repository

Keywords

Autor

Licence

Y crea un json (un archivo de intercambio de javascript) – que es ¿? <https://es.wikipedia.org/wiki/JSON>

Nos va a pedir que confirmemos si este todo ok, ponemos todo ok.con yes

7° en la carpeta vamos a encontrar el package.json con las configuraciones que pusimos en la consola.

8° npm install -D node-sass nodemon

Que es nodemodules ¿? <https://lenguajejs.com/npm/administracion/carpeta-node_modules/#:~:text=La%20carpeta%20node_modules%20es%20un,paquetes%20o%20dependencias%20mediante%20npm%20>.

9° En el package.json dentro de la clave scripts, configuro lo siguiente:

 "build-css": "node-sass --include-path scss scss/prueba.scss css/pruebacss.css",

    "watch-css": "nodemon -e scss -x \"npm run build-css\""

"build-css": "node-sass --include-path scss scss/estilos.scss css/estilos.css", "watch-css": "nodemon -e scss -x \"npm run build-css\""

Que es lo que configuramos recién ¿?

Lo que indicamos en “build-css” es dónde encontrará el documento SCSS de entrada y dónde se depositará el documento CSS de salida. Lo que indica “watch-css” es donde vamos a permanecer observando los cambios sobre la última acción de compilación.

Entonces cuando lo **ejecutamos,** este script **vigilará cada archivo scss** en el directorio indicado, luego guarda el css compilado en el directorio css/ cada vez que cambiamos un archivo scss.

Para ejecutar nuestro script de una línea, necesitamos ejecutar el siguiente comando en la terminal:

10°  npm run watch-css

¡Y *voila*! Estamos vigilando y compilando SASS.

Más Info:

<https://nodejs.org/en/about/>

<https://devcode.la/blog/que-es-npm/>

<https://www.w3schools.com/nodejs/nodejs_npm.asp>

<https://www.sitepoint.com/beginners-guide-node-package-manager/>

live sass compiler :

es un pluggins de vscode que nos sirve para poder compilar sass en css.

Debemos primero instalar el plugin:

Live sass compiler

Luego en nuestro proyecto creamos la carpeta que se llame

scss con el archivo estilos.scss

creamos una carpeta que se llame .vscode y dentro tendrá un archivo que se llame settings.json

con lo que esta abajo dentro :

{

    "liveSassCompile.settings.formats":[

      {

         "format":"expanded",

         "extensionName":".css",

         "savePath":"/css"

      }

     ],

    "liveSassCompile.settings.excludeList":[

       "\*\*/node\_modules/\*\*",

       ".vscode/\*\*"

      ],

    "liveSassCompile.settings.generateMap":true,

    "liveSassCompile.settings.autoprefix":[

       ">1%",

       "last 2 versions"

   ]

 }

Para que el live sass compiler compile nuestro código debemos tener encedido en el visual , el watch, debe estar watching .

Para esta ultima opción no hay que instalar nada nodejs, no se crea la carpeta node\_modules. El archivo finalmente queda guardado en scss/estilos.css.

**Paso a paso para incorporar sass en nuestro proyecto ya iniciado:**

Versión con NodeJs:

-crear una rama desde master para incorporar sass.

-Nos parados en la rama

-Renombrar la hoja de estilos en cascada y le ponemos estilos2.css

-Poner npm init

-dentro del package json scss/estilos.scss css/estilos.css

-instalar node-sass y nodemon(ver comandos en el manual)

-npm run watch-css(para que compile sass a css)

-copiar todo el contenido de la hoja de estilos2.css y lo van a pegar en estilos.scss y grabar

como el watch está compilando si no hay errores se va a crear una nueva hoja de estilos en la carpeta css que se va a llamar estilos.css, esta nueva hojas va a tener todos los estilos que tenían en la rama hasta el momento.

-solo queda linkear la nueva hoja que se llama estilos.css en todos los archivos de HTML.

Desafío aplicar sass:

* Tienen que elegir una de las 5 hojas de html y empezar a identificar cuales son componentes

Ejemplo : header , sección principal, footer, article.

* Si esos componentes tienen clases propias con estilos en css y no tiene clases de Bootstrap empezaran aplicando el concepto de nesting(anidación).

Empiezan identificando el padre y dentro del padre ponen sus propiedades, dentro ponen a sus hijos.

Si en estos componentes usan clases genéricas que usan en varias partes del proyecto, esas clases no las anidan.

* Luego identifican aquellos valores de propiedades que se repite en todo el css y crean variables, en el lugar donde estaba el valor lo reemplazan por las variables. Recuerden que las variables van arriba de todo.
* Por último, empiezan a particionar su proyecto para crear hojas de sass nuevas:

\_fuentesiconos.css

\_variables.scss

\_reset.scss

\_header.scss

\_seccionprincipal.scss

\_footer.scss

\_mediaqueries.scss

estilos.scss

La única hoja que se compila es estilos.scss y en ella van a poner los import

@import “fuentesiconos.scss”;

@import “variables.scss”;

@import “reset.scss”;

@import “header.scss”;

@import “seccionprincipal.scss”;

@import “footer.scss”;

@import “mediaqueries.scss”;

La clase de sass2 vamos a ver que vamos a poner poner más código de css en hojas de estilo de css separado como los mixins.

Como organizar sass dentro de tu proyecto:

https://latteandcode.medium.com/c%C3%B3mo-organizar-los-archivos-sass-de-tu-proyecto-c8b02242d95

ejemplos de variables en sass:

https://desarrolloweb.com/articulos/variables-sass.html