

Sass

# SASS

Instalación

# Sass

profesor: Vladimir Bataller

## Instalación de Ruby para SASS

SASS es un lenguaje de hojas de estilo que se transpila a css. El transpilador de SASS está desarrollado en Ruby. A continuación se indica como instalar Ruby en Windows.

Descargar el instalador (rubyinstaller-2.3.3-x64.exe) de la url:

<http://rubyinstaller.org/downloads/>

Ejecutar el instalador, aceptando todas las opciones por defecto, excepto que hay que indicarle que incluya la ruta de los ejecutables de Ruby en la variable PATH del sistema.

# Sass

profesor: Vladimir Bataller

## Instalación de Sass

SASS es un lenguaje de hojas de estilo que se transpila a css.

Para instalar SASS se debe ejecutar la siguiente línea del gestor de paquetes de Ruby (llamado Gems):

**gem install sass**

Nota: si el paso anterior falla, ejecutar:

**gem sources -a <http://rubygems.org/>**

Para verificar que está correctamente instalado ejecutar:

**sass -v**

Iniciar el servicio  
Sass

SASS es un lenguaje de hojas de estilo que se transpila a css.

Una vez instalado, la forma más cómoda de trabajar es indicarle que monitorice el archivo de fuentes de Sass (\*.scss) y que si cambia lo compile al archivo de destino (\*.css), para ello se debe ejecutar:

```
sass --watch archivoFuente.scss:archivoDestino.css
```

También se puede indicar lo mismo con todos los archivos de una carpeta de origen y otra de destino:

```
sass --watch carpetaFuente:carpetaDestino
```

Además se puede indicar el formato del archivo de salida con la opción **-style**.

```
sass --watch ./src_estilos:./www/estilos -style:expanded
```

Los valores posibles que se le pueden asignar a *style* son:

**:nested** -> agrega tabulaciones al selector para indicar el grado de anidamiento en el archivo de origen.

**:expanded** -> muestra cada selector sin tabulación, pero con cada propiedad en una línea.

**:compact** -> muestra cada selector sin tabulación y todas las propiedades en la misma línea.

**:compressed** -> toda la hoja de estilo se escribe en una sola línea sin espacios ni tabulaciones.

# Sass

profesor: Vladimir Bataller

## Anidamiento

Sass permite definir estilos anidados lo que facilita el mantenimiento.

estilos.scss		estilos.css
<pre>div#mensaje{   background-color:yellow;   p.contenido{     margin-left: 20px;     a{color:gray;}   } }</pre>		<pre>div#mensaje {   background-color: yellow; } div#mensaje p.contenido {   margin-left: 20px; } div#mensaje p.contenido a {   color: gray; }</pre>

# Sass

profesor: Vladimir Bataller

## Operador &

Por defecto, cada estilo anidado se aplica sobre un elemento anidado. El operador & representa al estilo superior pudiendo especificar si lleva o no espacio y por lo tanto si se aplica al elemento anidado o al mismo elemento.

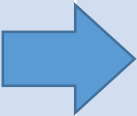
estilos.scss		estilos.css
<pre>div#mensaje{   background-color:yellow;   .contenido{     margin-left: 20px;     a{color:gray;}   } }</pre>	→	<pre>div#mensaje {   background-color: yellow; } div#mensaje .contenido {   margin-left: 20px; } div#mensaje .contenido a {   color: gray; }</pre> <p>Espacio-&gt; elemento anidado</p>
<pre>div#mensaje{   background-color:yellow;   &amp;.contenido{     margin-left: 20px;     a{color:gray;}   } }</pre>	→	<pre>div#mensaje {   background-color: yellow; } div#mensaje.contenido {   margin-left: 20px; } div#mensaje.contenido a {   color: gray; }</pre> <p>Sin espacio-&gt; mismo elemento</p>

# Sass

profesor: Vladimir Bataller

## Variables

Sass permite declarar variables (precedidas por \$) que se pueden emplear en diferentes puntos del documento. En la hoja de destino se reemplazará, en cada posición en donde aparece la variable, el valor que se la ha suministrado.

estilos.scss		estilos.css
<pre>\$color-bloques: green; div#mensaje{   background-color:\$color-bloques;   p.contenido{     margin-left: 20px;     a{color:gray;}   } } div#resultado{   background-color:\$color-bloques; }</pre>		<pre>div#mensaje {   background-color: green; } div#mensaje p.contenido {   margin-left: 20px; } div#mensaje p.contenido a {   color: gray; }  div#resultado {   background-color: green; }</pre>



# Sass

profesor: Vladimir Bataller

Importar archivos  
parciales

Sass trocear un una archivo fuente en varios archivos parciales (su nombre debe ir precedido de underscore, es decir guión bajo). Posteriormente esos archivos parciales se pueden importar con @import.

**\_parte1.scss**

```
div#resultado {  
  background-color: yellow; }
```

**estilos.scss**

```
div#mensaje{  
  background-color:yellow;  
  .contenido{  
    margin-left: 20px;  
    a{color:gray;}  
  }  
}  
  
@import "parte1";
```

**estilos.css**

```
div#mensaje {  
  background-color: yellow; }  
div#mensaje .contenido {  
  margin-left: 20px; }  
div#mensaje .contenido a {  
  color: gray; }  
  
div#resultado {  
  background-color: yellow; }
```

# Sass

profesor: Vladimir Bataller

## Mixins

Mediante la directiva `@mixin` se puede definir con un nombre, un bloque parametrizado que se puede emplear posteriormente en uno o varios puntos asignando diferentes valores a los parámetros.

estilos.scss	estilos.css
<pre>@mixin border-radius(\$radio) {   -webkit-border-radius: \$radio;   -moz-border-radius: \$radio;   -ms-border-radius: \$radio;   border-radius: \$radio; }  div#principal { @include border-radius (20px); } p { @include border-radius (10px); }</pre>	<pre>div#principal {   -webkit-border-radius: 20px;   -moz-border-radius: 20px;   -ms-border-radius: 20px;   border-radius: 20px; }  p {   -webkit-border-radius: 10px;   -moz-border-radius: 10px;   -ms-border-radius: 10px;   border-radius: 10px; }</pre>

# Sass

profesor: Vladimir Bataller

## Herencia

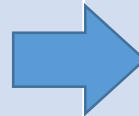
Mediante la etiqueta `@extend` se puede definir un estilo que hereda las propiedades de otro estilo.

### estilos.scss

```
.cuadro{  
  border: 1px solid black; padding:10px;  
}
```

```
.cuadro_centrado{  
  @extend .cuadro;  
  margin: 10px auto;  
}
```

```
.cuadro_importante{  
  @extend .cuadro;  
  border: 3px solid red;  
  background-color: pink;  
}
```



### estilos.css

```
.cuadro, .cuadro_centrado, .cuadro_importante{  
  border: 1px solid black;  
  padding: 10px;  
}
```

```
.cuadro_centrado {  
  margin: 10px auto;  
}
```

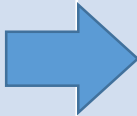
```
.cuadro_importante{  
  border: 3px solid red;  
  background-color: pink;  
}
```

# Sass

profesor: Vladimir Bataller

## Operadores matemáticos

En Sass a las propiedades se les puede asignar una expresión matemática para obtener el valor numérico que se generará para esa propiedad en la hoja de destino. Los operadores admitidos son: **+**, **-**, **\***, **/** y **%**.

estilos.scss		estilos.css
<pre>\$ancho:100%; .contenedor {   width: 100%; }  .izquierda {   float: left;   width: 600px / 960px * \$ancho; }  derecha {   float: right;   width: 300px / 960px * \$ancho; }</pre>		<pre>.contenedor {   width: 100%; }  .izquierda {   float: left;   width: 62.5%; }  derecha {   float: right;   width: 31.25%; }</pre>

# Sass

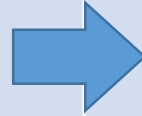
profesor: Vladimir Bataller

ampersand detrás

El operador ampersand (&) también se puede poner detrás de un selector, en ese caso hará referencia al padre, pero incluyéndolo como hijo del selector que hay delante del ampersand.

estilos.scss

```
input{  
  background-color: yellow;  
  p#alta & {  
    background-color: azure;  
  }  
}
```



estilos.css

```
input{  
  background-color: yellow;  
}  
p#alta input {  
  background-color: azure;  
}
```