

# LESS, UN PREPROCESADOR CSS

ADOLFO SANZ DE DIEGO

SEPTIEMBRE 2017

**1 EL AUTOR**

## 1.1 ADOLFO SANZ DE DIEGO

- Empecé desarrollando aplicaciones web, hasta que di el salto a la docencia.
- Actualmente soy Asesor Técnico Docente en el servicio TIC de la D.G de Infraestructuras y Servicios de la Consejería de Educación, Juventud y Deporte de la Comunidad de Madrid.
- Además colaboro como formador especializado en tecnologías de desarrollo.

## 1.2 ALGUNOS PROYECTOS

- Hackathon Lovers <http://hackathonlovers.com>: un grupo creado para emprendedores y desarrolladores amantes de los hackathones.
- Password Manager Generator <http://pasmangen.github.io>: un gestor de contraseñas online.
- MarkdownSlides  
<https://github.com/asanzdiego/markdownslides>: un script para crear slides a partir de ficheros MD.



## 1.3 ¿DONDE ENCONTRARME?

- Mi nick: asanzdiego
  - AboutMe: <http://about.me/asanzdiego>
  - GitHub: <http://github.com/asanzdiego>
  - Twitter: <http://twitter.com/asanzdiego>
  - Blog: <http://asanzdiego.blogspot.com.es>
  - LinkedIn: <http://www.linkedin.com/in/asanzdiego>
  - Google+: <http://plus.google.com/+AdolfoSanzDeDiego>

# 2 INTRODUCCIÓN

## 2.1 ¿QUÉ ES?

- Less es un pre-procesador de CSS.
- Añade características como variables, mixins, funciones, etc.

## 2.2 VENTAJAS

- El CSS es así más fácil de mantener, personalizable y extensible.
- Less tiene una sintaxis parecida a CSS.



## 2.3 CARACTERÍSTICAS

- Less se puede ejecutar desde NodeJS, desde un navegador, o desde Rhino.
- Además existen muchas herramientas que permiten compilar los archivos y ver los cambios en caliente.
- Es el preprocesador que usa Bootstrap 3.0.

**3 USANDO LESS**

## 3.1 INSTALACIÓN

- La forma más sencilla de instalar Less, es a través de npm, el gestor de paquetes de NodeJS:

```
$ npm install -g less
```

## 3.2 LÍNEA DE COMANDOS

- Una vez instalado, se puede compilar desde la línea de comandos:

```
$ lessc styles.less > styles.css
```

## 3.3 DESDE EL NAVEGADOR (I)

- Es la forma más fácil para empezar, pero no es recomendable usarlo así en producción.
- Descargar: <https://github.com/less/less.js/archive/master.zip>



## 3.4 DESDE EL NAVEGADOR (II)

- Enlazar tu archivo less que quieras precompilar, y luego el js de less:

```
<link rel="stylesheet/less" type="text/css" href="styles.less" />  
<script src="less.js" type="text/javascript"></script>
```

# 4 CARACTERÍSTICAS

## 4.1 VARIABLES (I)

- El siguiente código:

```
@nice-blue: #5B83AD;  
@light-blue: @nice-blue + #111;  
  
#header {  
    color: @light-blue;  
}
```

## 4.2 VARIABLES (II)

- Les compila a:

```
#header {  
  color: #6c94be;  
}
```

- Nota: las variables son en realidad "constantes" ya que sólo pueden ser definidas una vez.

## 4.3 EXTEND

- Son una forma de herencia:

```
.animal {  
  background-color: black;  
  color: white;  
}  
.bear {  
  &:extend(.animal);  
  background-color: brown;  
}
```



## 4.4 MIXINS (I)

- Los Mixins son una forma de **reutilizar propiedades ya definidas**:
- Imaginemos la clase `.bordered`:

```
.bordered {  
  border-top: dotted 1px black;  
  border-bottom: solid 2px black;  
}
```

## 4.5 MIXINS (II)

- Lo podemos usar así:

```
#menu a {  
  color: #111;  
  .bordered;  
}  
  
.post a {  
  color: red;  
  .bordered;  
}
```

- **Nota:** Además de clases, también se pueden utilizar **#ids** como **mixins**.

## 4.6 REGLAS ANIDADAS (I)

- Supongamos que tenemos el siguiente CSS:

```
#header {  
  color: black;  
}  
#header a {  
  color: blue;  
}  
#header a:hover {  
  color: red;  
}
```

## 4.7 REGLAS ANIDADAS (II)

- Pues con Less se puede escribir así:

```
#header {  
  color: black;  
  a {  
    color: blue;  
    &:hover {  
      color: red;  
    }  
  }  
}
```

- Se pueden usar pseudo-elementos, y llamar al selector padre actual, con &:

## 4.8 MEDIAS QUERIES ANIDADAS (I)

- El siguiente código:

```
@media screen {  
  .screencolor { color: green; }  
}  
@media screen and (min-width: 768px) {  
  .screencolor { color: red; }  
}  
@media tv {  
  .screencolor { color: black; }  
}
```



## 4.9 MEDIAS QUERIES ANIDADAS (II)

- Se podría escribir:

```
.screencolor{  
  @media screen {  
    color: green;  
    @media (min-width:768px) {  
      color: red;  
    }  
  }  
  @media tv {  
    color: black;  
  }  
}
```

## 4.10 OPERACIONES (I)

- Less puede hacer operaciones con números, colores o variables.
- Además sabe diferenciar cuando es un número o un color.

## 4.11 OPERACIONES (II)

```
@base: 5%;  
@filler: @base * 2;  
@other: @base + @filler;  
@base-color: #888 / 4;  
  
background-color: @base-color + #111;  
height: 100% / 2 + @other;
```

## 4.12 FUNCIONES (I)

- Less dispone de una variedad de funciones matemáticas, que manipulan cadenas, y que transforman los colores:

```
@base: #f04615;  
@list: 200, 500, 1200;  
  
.class {  
  width: extract(@list, 3);  
  color: saturate(@base, 5%);  
  background-color:  
    lighten(@base, 25%);  
}
```

## 4.13 FUNCIONES (II)

- El código anterior compila a:

```
.class {  
  width: 1200;  
  color: #f6430f;  
  background-color: #f8a58d;  
}
```



## 4.14 NAMESPACES (I)

- A veces, podemos querer **agrupar mixins**, por motivos de organización, o simplemente para encapsularlos.

## 4.15 NAMESPACES (II)

- Veamos como podemos agrupar varios mixins:

```
#bundle {  
  .button {  
    border: 1px solid black;  
    background-color: grey;  
  }  
  .tab { ... }  
  .citation { ... }  
}
```

## 4.16 NAMESPACES (III)

- Ahora podemos utilizar el mixin `.button` que está en el namespace `#bundle` de esta forma:

```
#header a {  
  color: orange;  
  #bundle > .button;  
}
```

## 4.17 SCOPE

- Los **ámbitos de las variables** en Less es muy similar a otros lenguajes:

```
@var: red;

#page {
  @var: white;
  #header {
    color: @var; // white
  }
}
```

## 4.18 IMPORTS

```
// foo.less is imported
@import "foo";

// foo.less is imported
@import "foo.less";

// foo.php imported as a less file
@import "foo.php";

// foo.css imported as a css file
@import "foo.css";
```



**5 ACERCA DE**

## 5.1 LICENCIA

- Estas transparencias están hechas con:
  - MarkdownSlides:  
<https://github.com/asanzdiego/markdownslides>
- Estas transparencias están bajo una licencia Creative Commons Reconocimiento-CompartirIgual 3.0:
  - <http://creativecommons.org/licenses/by-sa/3.0/es>

## 5.2 FUENTES

- Transparencias:
  - <https://github.com/asanzdiego/curso-interfaces-web-2017/tree/master/04-preprocesadores-css/04.01-less/slides>
- Ejercicios:
  - <https://github.com/asanzdiego/curso-interfaces-web-2017/tree/master/04-preprocesadores-css/04.01-less/src>

## 5.3 BIBLIOGRAFÍA

- Documentación oficial de Less
  - <http://lesscss.org>