



LESS

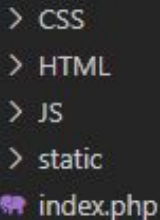


Significado

- Preprocesador de CSS que añade funcionalidades extras al CSS. Trabaja con archivos en formato .less para luego convertirlos en formato .css que pueda ser entendido por el navegador

Instalación LESS

1. Instalar node: <https://nodejs.org/es/>
2. Instalar less: `npm install -g less`
3. Ejecutamos comando `lessc` para comprobar que todo está correcto: `lessc CSS/P25.less > CSS/P25.css`



```
> CSS  
> HTML  
> JS  
> static  
🐼 index.php
```

4. (Windows) Abrimos consola powerShell como administrador y escribimos “Set-ExecutionPolicy Unrestricted”
5. (Recomendado) Instalar extensión Easy LESS



Variables

- Almacenamos valores en variables

```
@enlaces: green;
```

```
@selectorEnlaces:a;
```

```
@{selectorEnlaces}{  
    color: @enlaces  
}
```



```
a {  
    color: green;  
}
```

Anidación

- Nos permite anidar selectores

```
@enlaces-color: blue;
```

```
@enlaces-hover-color: red;
```

```
body{
```

```
  p{
```

```
    a{
```

```
      color:@enlaces-color;
```

```
      &:hover{
```

```
        color: @enlaces-hover-color;
```

```
    }
```

```
  }
```

```
}
```

```
}
```



```
body p a {
```

```
  color: blue;
```

```
}
```

```
body p a:hover {
```

```
  color: red;
```

```
}
```

Anidación

- Se puede mezclar con media queries para trabajar con componentes

```
.component {  
  width: 300px;  
}
```

```
@media (min-width: 768px) {  
  width: 600px;  
}
```

```
@media (min-resolution: 192dpi) {  
  background-image: url(/img/retina2x.png);  
}
```

```
@media (min-width: 1280px) {  
  width: 800px;  
}
```



```
.component {  
  width: 300px;  
}
```

```
@media (min-width: 768px) {  
  .component {  
    width: 600px;  
  }  
}
```

```
@media (min-width: 768px) and  
(min-resolution: 192dpi) {  
  .component {  
    background-image:  
    url(/img/retina2x.png);  
  }  
}
```

```
@media (min-width: 1280px) {  
  .component {  
    width: 800px;  
  }  
}
```

Mixins

- Nos permiten crear bloques de código CSS a los que asignaremos un nombre y podremos invocar llamando a dicho nombre

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

```
#menu a {  
  color: #111;  
  .bordered();  
}
```

```
.post a {  
  color: red;  
  .bordered();  
}
```



```
#menu a {  
  color: #111;  
  border-top: dotted 1px black;  
  border-bottom: solid 2px black;  
}
```

```
.post a {  
  color: red;  
  border-top: dotted 1px black;  
  border-bottom: solid 2px black;  
}
```

Mixins

- También admiten el paso de valores por parámetros

```
.radius(@radio:5px) {  
  -webkit-border-radius: @radio;  
  -moz-border-radius: @radio;  
  border-radius: @radio;  
}  
  
p {  
  .radius(30px);  
}
```



```
p {  
  -webkit-border-radius: 30px;  
  -moz-border-radius: 30px;  
  border-radius: 30px;  
}
```


- Existe un archivo llamado `elements`, que nos facilita la compatibilidad con navegadores

```
@import "elements"; // elements.less

.border-radius(5px, 0, 0, 5px);

.border-radius(@topright: 0, @bottomright: 0, @bottomleft: 0, @topleft: 0) {
    -webkit-border-top-right-radius: @topright;
    -webkit-border-bottom-right-radius: @bottomright;
    -webkit-border-bottom-left-radius: @bottomleft;
    -webkit-border-top-left-radius: @topleft;
    -moz-border-radius-topright: @topright;
    -moz-border-radius-bottomright: @bottomright;
    -moz-border-radius-bottomleft: @bottomleft;
    -moz-border-radius-topleft: @topleft;
    border-top-right-radius: @topright;
    border-bottom-right-radius: @bottomright;
    border-bottom-left-radius: @bottomleft;
    border-top-left-radius: @topleft;
    .background-clip(padding-box);
}
```



Mixins-Guards

- Mixins con estructuras de control

```
.textoOscuro (@value) when (@value=true) {  
    color: black;  
}
```

```
.textoOscuro (@value) when (@value=false) {  
    color: white;  
}
```

```
h1 {  
    .textoOscuro(true); // black  
    //.textoOscuro(false); // white  
}
```



Funciones

- Cuenta con una serie de funciones preestablecidas

```
@var: #004590;
```

```
div {  
  height: 50px;  
  width: 50px;  
  background-color: @var;
```

```
  &:hover {  
    background-color: fadeout(@var, 50%)  
  }  
}
```