CSS, SASS

GOBIERNO DE ESPAÑA
MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO
EOI Escuela de organización industrial

Unión Europea
Fondo Social Europeo
Iniciativa de Empleo Juvenil
El FSE invierte en tu futuro

SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

¿Para qué nos sirve?

- Con **Html5** hemos visto como construir un proyecto **Html5** desde cero, como organizar su **estructura** y que elementos son más apropiados para representar un contenido u otro

- Ahora toca definir como se va a mostrar **(presentación)** en la pantalla ese documento **Html5** y para ellos nos vamos a apoyar en **CSS** y **SASS,** que nos van a permitir aplicar estilos (color, fuentes, tamaños , espacios , márgenes , ...) a nuestros elementos **Html5**

- **CSS (Cascading Style Sheets – Hojas de estilo en cascada) :** ficheros con extensión **.css**

- **SASS :** preprocesador css, ficheros con extensión **.scss**

Css y Sass

¿Para qué nos sirve?

Como ya indicamos anteriormente , al cargar el documento se genera un árbol DOM , un árbol CSSOM y luego se combinan los dos. En el caso de **SASS,** al ser un preprocesador de **CSS,** el código **SASS** se tiene que compilar para poder generar los archivos **.css** correspondientes.

Al usar **SASS** podemos usar variables, anidación de selectores, funciones (denominadas mixins) que con **CSS** no podemos usar y nos van a facilitar mucho el trabajo de construcción de nuestros **CSS**

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Módulos, Navegadores, ...

- **CSS** está **dividido en** varios documentos llamados **módulos** (Selectores, espaciones de nombres, Color , ...)

- La implementación de estos módulos en los **diferentes navegadores** se puede gestionar/señalar usando los prefijos del motor del mismo

-moz : Mozilla Firefox
-webkit : Safari y Chrome
-o : Opera
-ms : Microsoft

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Módulos, Navegadores, ...

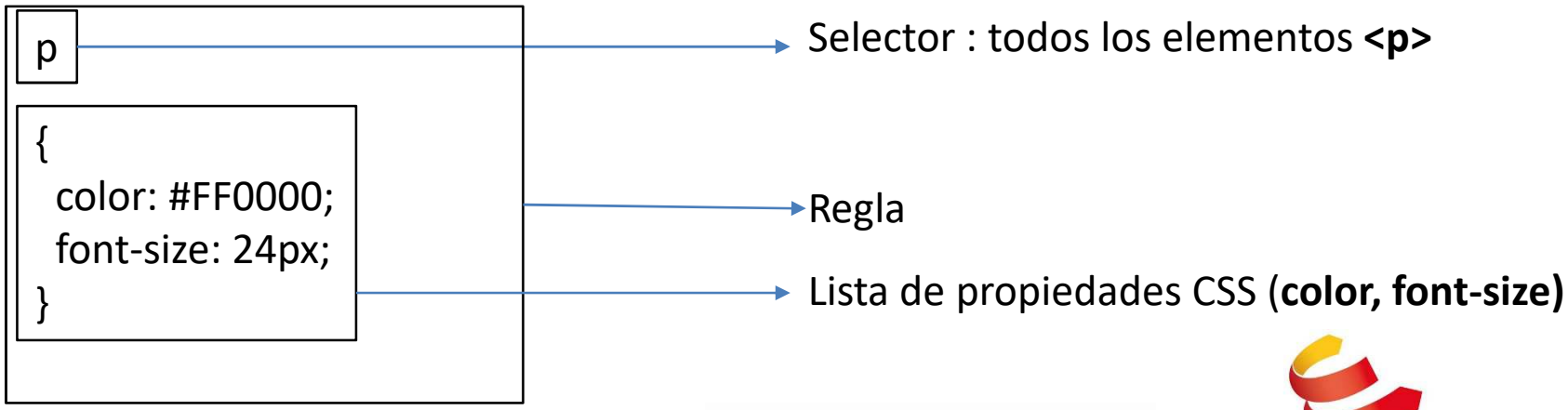
- **Unidades de medida** *:* la unidad de medida en pixeles **(px)** es la más utilizada , aunque cuando estemos creando sitios Web con diseño Responsive seguramente usemos otras como las siguientes

px : pixeles
pt : puntos
in : pulgadas
cm : centímetros
mm : milímetros
em : relativo al tamaño de la fuente del elemento
rem : relativo al tamaño de la fuente del documento
% : relativo con el tamaño del contenedor del elemento

Css y Sass

Reglas y selectores

- **Reglas :** para aplicar/cambiar varios estilos a la vez (esto es , varias propiedades CSS), se usan las reglas. Una regla es una lista de propiedades CSS declaradas entre llaves e identificadas por un selector. El selector indicará que elementos de nuestro documento **Html** se verás afectados por las propiedades. Para comentar una regla y que no se aplique usamos los carácteres /* ...reglas... */ como en Java, englobando la regla o reglas que no queremos que se apliquen



Css y Sass

Reglas y selectores

- Los **selectores** pueden sen de varios **tipos** :

- *Selector por Elemento* : se aplica a los elementos dentro del documento que coincidan con el selector

*Ejemplo : a todos los elementos **<p>** del documento se les aplicará el color rojo*

```
p{
 color:red;
}
<p> Aquí se aplica el color</p>
```

Css y Sass

Reglas y selectores

- *Selector por Clase :* se aplica a los elementos dentro del documento que su atributo **class** contenga ese selector. Empiezan por el símbolo **.**

*Ejemplo : a todos los elementos que contengan en el atributo **class** , **colorRojo** se les aplicará el color rojo*

```
.colorRojo{
 color:red;
}
<div class="colorRojo">
```

Css y Sass

Reglas y selectores

- *Selector por Identificador :* se aplica a los elementos cuyo atributo **id** coincida con el selector. el selector empieza por el símbolo **#**

*Ejemplo : a todos los elementos del documento con el atributo **id = colorRojo** se les aplicará el estilo color rojo*

```
#colorRojo{
 color:red;
}
<div id="colorRojo">
```

Css y Sass

Reglas y selectores

- *Selector por nombre* : se aplica alos elementos cuyo atributo **name** coincida con el selector.

Ejemplo :

```
p[name="apellidos"]
{
 color:red;
}
```

```
<p name="apellidos">
```

** **Importante :** Los estilos se aplican en cascada , por lo que los elementos de los niveles inferiores heredan los estilos de un elemento en un nivel superior de la jerarquía (De su padre)

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Aplicar estilos

- Para asignar los estilos a los documentos , existen 3 formas :

- *Aplicar estilos en línea (Esta opción no es muy usada):* usando el atributo **style** dentro de cualquier elemento y separando por **;** todas las propiedades que quedamos aplicar

Ejemplo

```
<p style="font-size: 20em;color:red;">Prueba de estilo en línea</p>
```

Css y Sass

Aplicar estilos

- *Aplicar en el **<head>** usando el elemento **<style>** **:** Declarar los estilos dentro de un element **<style>** en el elemento **<head>** del documento*

Ejemplo

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Prueba</title>
  <meta charset="utf-8">
  <style>
    p {    font-size: 20px;    }
  </style>
</head>
<body><main><section><p>Texto</p></section></main></body>
</html>
```

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Aplicar estilos

- *Aplicar los estilos desde un archive externo usando **<link>** :* Declarar todos los estilos en un archivo externo con extensión **.css** e incluir una referencia al mismo usando **<link>** dentro de **<head>**

```
<!DOCTYPE html>
<html lang="es">
<head>
 <title>Prueba</title>
 <meta charset="utf-8">
 <link rel="stylesheet" href="estilos.css">
</head>
<body><main>
<section><p>Texto</p></section>
</main>
</body></html>
```

Contenido del fichero **estilos.css**

```
p{
  color:red;
}
```

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio1

Añadir hoja de estilo

Crear una regla CSS que se aplique a los elementos **<p>** y que establezcan el color del texto a rojo, aplicar este estilo al texto **CURSO CSS EOI**

Ejercicio1

Añadir hoja de estilo

CURSO CSS EOI

GOBIERNO DE ESPAÑA
MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO
EOI Escuela de organización industrial
Unión Europea
Fondo Social Europeo
Iniciativa de Empleo Juvenil
El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio2

Estilos heredados y tamaños

A partir del código fuente del *ejercicio1* introducir el elemento **<p>** dentro de un elemento **<section>** y crear un estilo para el elemento **<section>** que establezca el tamaño de la fuente a **5em** Modificar también el estilo del elemento **<p>** para que se establezca el tamaño de fuente de **2em** Introducir en texto **SECCIÓN** en el contenido de **<section>** fuera del elemento **<p>** Pensar porque se está visualizando de esta manera. Crear una nueva hoja de estilo llamada *ejercicio2.css*

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio2

Estilos heredados y tamaños

SECCION

CURSO HTML5

EOI

GOBIERNO DE ESPAÑA
MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO
EOI Escuela de organización industrial
Unión Europea
Fondo Social Europeo
Iniciativa de Empleo Juvenil
El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Aplicar reglas a varios selectores

- Hasta ahora hemos visto como aplicar una regla a un selector en concreto
- Podemos **aplicar la misma regla , a un conjunto de selectores separando los selectores por ,**

Ejemplo :

```
p, span , article{
        color:red;
}
```

Esta regla se aplicará a los elementos *<p>, * y *<article>*

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Aplicar reglas a elementos hijos

- Se aplican las reglas a los elementos que sea descendientes directos del primer elemento del selector.

- Se consigue usando el símbolo **>** , de la siguiente manera ***padre > hijo***

Ejemplo :

```
p > span {
        color:red;
}
```

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Aplicar reglas a cualquier descendiente

- Se aplican las reglas a elementos que están dentro de otros elementos, independientemente de su lugar en la jerarquía . Descendientes del primer elemento del selector (Cuando hablemos de jerarquía pensemos en la representación del árbol)

- Se consigue separando los selectores por espacios

Ejemplo :

```
p span {
        color:red;
}
```

Css y Sass

Aplicar reglas a hermanos adyacentes

- Selecciona elementos hermanos adyacentes. Tienen que compartir el mismo padre
- Se consigue separando los selectores por +

Ejemplo :

```
p+span {
        color:red;
}
```

Css y Sass

Aplicar reglas a hermanos no necesariamente adyacentes

- Selecciona hermanos. Deben compartir el mismo padre
- Se consigue separando los selectores por ~

Ejemplo :

```
p~span {
        color:red;
}
```

Css y Sass

Aplicar reglas por selectores con atributos

- Antes habíamos visto que podíamos aplicar reglas por selectores construidos con los atributos : *id, class, name*

- Vamos a ver como jugar con los atributos para construir reglas muy diversas

- *Selector que detecte la inclusión de una palabra concreta dentro del valor de un atributo (~)*

p[name~=“nom”]{ color:red}

- *Selector que detecte una palabra concreta al principio del valor de un atributo (^)*

p[name^=“nom”]{ color:red}

Css y Sass

Aplicar reglas por selectores con atributos

- *Selector que detecte una palabra concreta al final del valor de un atributo (**$**)*

p[name$=“nom”]{ color:red}

- *Selector que detecte una cadena de texto concreta dentro del valor de un atributo (*****)*

p[name*=“nom”]{ color:red}

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio3

Aplicar reglas con selectores de herencia

A partir del código fuente del *ejercicio2* eliminar todo el contenido del elemento **<body>** y realizar las siguientes tareas , ir comentando las reglas creadas hasta el momento con cada nueva tarea para poder probar las reglas de manera aislada:

a) Crear dos elementos **<section>** dentro del elemento **<body>** con los atributos ***name*** , *seccion1* y *seccion2* y con los textos *SECCION1* y *SECCION2*

b) Crear dentro de cada elemento **<section>** un elemento <**article>** con el atributo ***class*** *articulo1* para uno de ellos y *articulo2* para el otro , incluir con los textos *ARTICULO1 y ARTICULO2* respectivamente en cada uno de los artículos. El texto del *articulo2* meterlo dentro de un elemento **<p>**

Ejercicio3

Aplicar reglas con selectores de herencia

c) Crear una regla que establezca el color rojo del texto al elemento con atributo **name="seccion1"**

d) Añadir a la regla anterior un selector para que se seleccione los elementos con atributo **class="articulo2"**

e) Incluir un elemento **<p>** entre los dos elementos <section> con el texto **SEPARA ARTÍCULOS .** Crear una regla que establezca el color rojo del texto al elemento **<section>** que vaya detrás del elemento **<p>**, usar el operador **+**

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio3

Aplicar reglas con selectores de herencia

f) Crear una regla que establezca el color rojo al elemento **<article>** con el atributo **class** que contenga la palabra **articulo2**

g) Meter todos los elementos **<section>** dentro de un elemento **<main> .** Crear una regla de establezca el color rojo los elementos **<section>**

h) Crear una regla que establezca el color rojo a todos los elementos que contengan en el atributo **class** el texto **articulo**

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio3

Aplicar reglas con selectores de herencia

i) Crear una regla que establezca el color rojo a todos los elementos que contengan en el atributo **class** el texto **2** al final del valor

j) Crear una regla que establezca el color rojo a todos los elementos que contengan en el atributo **class** el texto **art** al inicio del valor

k) Establecer el atributo **id="sec1"** en el elemento **<section>** con atributo **name="seccion1"** y crear una regla que establezca el color rojo al elemento con atributo **id="sec1"**

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Trabajar con pseudoclases

- Las pseudo-clases son características especiales en CSS que nos permiten hacer referencia a elementos HTML por sus características, como la posición en el código o las condiciones actuales. Se ponen a continuación del selector con el que queremos trabajar. Permiten la selección de elementos comprobando la información de estado que no está contenida en el árbol de documentos

:nth-child(value**)**

Selecciona un elemento de una lista de hermanos adyacentes que está en la posición especificada por el valor entre paréntesis. En value se puede identificar si la posición es par con **even** o impar con **odd**

Ejemplo : selecciona el elemento en la posición 2 de entre todos los elementos **<p>**

```
p:nth-child(2) {
  font-size: 20px;
}
```

- **:first-child :** Selecciona el primer elemento de una lista de hermanos.

Css y Sass

Trabajar con pseudoclases

- **:last-child**

Elige el último elemento en una lista de hermanos.

- **:only-child**

Selecciona un elemento cuando es el único hijo de otro elemento.

- **:first-of-type**

Selecciona el primer elemento en una lista de elementos del mismo tipo.

- **:not(**selector**)**

Selecciona los elementos que no coinciden con el selector entre paréntesis.

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio4

Pseudoclases

Crear una nueva plantilla **Html5** y añadir el siguiente código dentro del elemento **body**

```
<main>
  <section id="sec1" name="seccion1">
    <p class="p1">Párrafo 1</p>
    <p class="p2">Parrafo 2</p>
    <p class="p3">Párrafo 3</p>
    <p class="p4">Párrafo 4</p>
  </section>
</main>
```

Ejercicio4

Pseudoclases

Realizar las siguientes selectores con Pseudo-clases :

a) Para el segundo elemento **<p> ,** establecer el color de texto rojo

b) Para el primer elemento **<p> ,** establecer el color de texto rojo

c) Para el último elemento **<p> ,** establecer el color de texto rojo

d) Para los elementos en posición **par** establecer color de fondo rojo y para los elementos en posición **impar** establecer color de fondo amarillo

Ejercicio4

Pseudoclases

Realizar las siguientes selectores con Pseudo-clases :

e) Para el elemento **<section>** hijo único de **<main>** establecer el color de texto rojo

f) Para el primer elemento de los elementos del mismo tipo de elemento **<p>** establecer el color de texto rojo

g) Usando la Pseudo-clase **:not(selector)** establecer para los elementos **<p>** un color de fondo rojo

Css y Sass

Formato y diseño

- Las propiedades de CSS definen los estilos que podemos aplicar a un elemento. Para simplificar su estudio, pueden clasificarse en dos tipos: propiedades de formato y propiedades de diseño.

➢Las propiedades de formato : asignan un formato a los elementos y su contenido. Las propiedades de formato pueden clasificarse según el tipo de modificación que realicen (Fuente, bordes, color de fondo , ...)

➢Las propiedades de diseño : determinan el tamaño y la posición de los elementos en la pantalla.

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Texto

- **font-family :** familias de Fuente separados por coma (Si no encuentra la primera , busca la siguiente y así sucesivamente)
- **font-size :** tamaño de la fuente (px, %, em, rem,pt , ...)
- **font-weight :** determina si es negrita o no *(normal, bold).* Si es negrita, podemos indicar la intensidad de la negrita (100..900)
- **font-style :** estilo de la fuente (*normal, italic , oblique)*
- **font :** propiedad shorthand que nos permite declaras multiples estilos en una línea, todos los estilos se separan por espacio y con un orden determinado (*style* y *weight* tiene que ser declarados antes del tamaño y la *family* tiene que ir al final)

Ejemplo: **font: bold 2em Arial,sans-serif**;

- **text-align :** alinea el texto que está dentro de un elemento (*left, right, center y justify)*
- **text-align-last :** alinea la última línea de un párrafo **<p>** (*left, right, center y justify)*

Css y Sass

Texto

- **text-indent :** define la cantidad de sangría de un párrafo
- **letter-spacing :** define el espacio entre letras
- **word-spacing :** define el espacio entre palabras
- **line-height :** define el espacio entre líneas
- **vertical-align :** alinea elementos verticalmente. Usado normalmente para alinear texto con imágenes, la propiedad debería ser aplicada en este caso a la imagen.
- **text-decoration :** decora el texto con una línea *(underline, overline, line-through, none).* Esta propiedad es muy útil al usar los enlaces con el element **<a>**

Safe Fonts : fuentes que están disponibles en casi todos los equipos y pueden ser usadas con la tranquilidad de que van a cargarse correctamente en cualquier equipo. ¿Pero qué pasa si queremos usar una fuente específica, cómo la incluimos? @font-face

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Colores

Hay dos formas de declaras los colores en CSS, podemos usar una combinación de los tres colores básicos, rojo, verde y azul, o definir el nivel de matiz, saturación y ligereza. Dependiendo del tipo de sistema que utilizamos para definir el color, tenemos que declarar los niveles usando números hexadecimales (de 00 a FF), números decimales (de 0 a 255), o porcentajes.

- **rgb(***red, green, blue***) :** define un color desde los valores pasados como parámetros
- **rgba(***red, green, blue, alpha***) :** define un color desde los valores pasados como parámetros, se añade opacidad (0 , transparente y 1, opaco)
- **hsl(***hue, saturation, lightness***)** define un color desde los valores pasados como parámetros
- **hsla(***hue, saturation, lightness, alpha***) :** define un color desde los valores pasados como parámetros
- **color :** define el color del contenido de un elemento (Puede ser el color en texto o en hexadecimal : red, #FF0000)

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

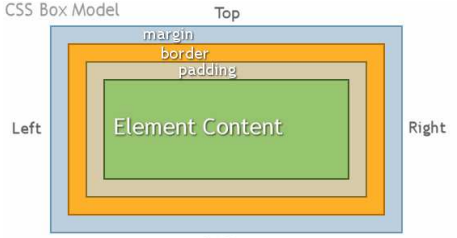
Tamaños

De forma predeterminada, el ancho de los elementos estructurales y otros elementos como <p> viene determinado por el tamaño del contenedor. Los elementos se expanden para ocupar el espacio horizontal disponible en el contenedor. Por otro lado, la altura está determinada por el tamaño de su contenido. Si queremos establecer un tamaño personalizado, tenemos que asignar las siguientes propiedades al elemento

- **width :** declara el ancho de un elemento. Cuando el valor se especifica en porcentaje, el ancho se calcula por el explorador a partir del ancho del contenedor y, cuando se declara como automático, el elemento se expande para ocupar todo el espacio horizontal disponible en el contenedor.

- **height :** declara el alto de un element. El cálculo es similar al de **width** pero con respecto a la propìedad **height** del element

Los navegadores generan una caja alrededor de los elementos que determinan el área que ocupan en la pantalla (modelo de cajas)



GOBIERNO DE ESPAÑA
MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO
EOI Escuela de organización industrial
Unión Europea
Fondo Social Europeo
Iniciativa de Empleo Juvenil
El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Tamaños

El problema con los elementos con ancho y alto personalizados es que a veces el contenido no encaja en el área. De forma predeterminada, los exploradores muestran el resto del contenido fuera del cuadro. En consecuencia, parte del contenido de nuestra caja se solapa con el contenido del elemento siguiente

- **overflow :** especifica cómo se va a mostrar el contenido que se desborda de su contenedor. Los posibles valores son (*visible , hidden , scroll, auto*)
- **overflow-x :** especifica cómo se va a mostrar el contenido que desborda su contenedor horizontalmente
- **overflow-y :** especifica cómo se va a mostrar el contenido que desborda su contenedor verticalmente
- **overflow-wrap :** indica si una palabra debe romperse o no en un punto arbitrario cuando no hay suficiente espacio para mostrarla en la línea actual *(normal, break-word)*

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio5

Texto y tamaños

A partir del código fuente del **ejercicio4** realizar las siguientes tareas

a) Crear una regla para que la clase **p1** tenga un tamaño de fuente 3 veces más grande relativo al tamaño de la fuente del elemento y tenga el texto alineado al centro

b) Crear una regla para que la clase **p3** tenga un tamaño de fuente 10 veces más grande relativo al tamaño de la fuente del elemento y tenga una decoración de texto **overline**

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Márgenes

- **margin :** declara el margen de un elemento

margin: upper_margin right_margin bottom_margin left_margin

También se puede usar de manera independiente :
margin-top, margin-right, margin-bottom, margin-left

Ejemplo : **margin: 2em 2em 2em 2em;**
margin-top:2em; margin-right:2em; margin-bottom:2em; margin-left:2em;

- **padding :** declara el padding del elemento (Mismo uso que **margin)**

- Para calcular el área de un elemento se utiliza la siguiente fórmula :

Área elemento = tamaño_elemento + márgenes + paddings + borders

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Márgenes

- Los elementos se pueden clasificar como elementos de bloque o elementos en línea. Los elementos de bloque pueden tener un tamaño personalizado pero los elementos en línea solo pueden ocupar el espacio determinado por su contenido

- Los elementos estructurales que hemos visto en el tema anterior de **Html5** están clasificados como elementos de bloque por lo que podemos jugar con su tamaño

  header, main, section, aside, article, ...

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio6

Márgenes

A partir del código fuente del **ejercicio5** realizar las siguientes tareas

a) Crear una regla para que la clase **p1** tenga margen de tamaño **1em**

b) Crear una regla para que la clase **p3** tenga un margen superior de **2em**

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Fondo

Los elementos pueden incluir un fondo que se muestra detrás del contenido del elemento y en todo el área ocupada por el contenido y el relleno. Debido a que el fondo puede estar compuesto de colores e imágenes

- **background-color :** color de fondo
- **background-image :** imagen de fondo (Se usa url('nombre de imagen') para cargar la imagen)
- **background-position :** declara la posición inicial de la imagen de fondo (*center, left, right, top,* y *bottom*)
- **background-size :** declara el tamaño de la imagen de fondo (Tamaño en unidades o usando : *cover -> escala la imagen hasta que al menos su ancho o su altura encaja en la caja del elemento, contain -> escala la imagen hasta que toda la imagen encaje en la caja del elemento)*
- **background-repeat :** decide como se va a distribuir la imagen (*repeat, repeat-x, repeat-y,no-repeat)*
- **background-origin :** determina si la imagen será posicionada relativamente al border, padding, o el contenido de la caja (*border-box, padding-box, content-box)*

Css y Sass

Fondo

- **background-clip :** delcara el área que va a ser pintada (*border-box,padding-box, content-box)*
- **background-attachment :** determina si la imagen es estática o se tiene scroll (*scroll, fixed)*
- **background :** propiedad shorthand
  - Ejemplo *:*
  - *background-image: url("brickslight.jpg");*
  - *background-color: #CCCCCC;*
  - *background-repeat: repeat-y;*
  - o de con la propiedad shorthand de esta manera
  - *background: #CCCCCC url("brickslight.jpg") repeat-y;*

Ejercicio7

Fondos

A partir del código fuente del **ejercicio6** realizar las siguientes tareas

a) Crear una regla para que el elemento **<section>** tenga como imagen de fondo **logoEOI.jpg** y esta se repita horizontalmente y cubra toda la sección.

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Bordes

- Los elementos pueden incluir un borde en los bordes de la caja. Por defecto los navegadores no muestran ningún borde para ningún elemento.

- **border-width :** define el ancho del borde (Se puede usar de manera independiente como en **margin** , *border-top-width, border-right-width, border-bottom-width, border-left-width)*

- **border-style :** define el estilo del borde *(none, hidden, dotted, dashed, solid, double, groove, ridge, inset,* y *outset)*

- **border-color :** color del borde. Se pueden definir los border de manera independiente como en **border-width**

- **border-radius :** define el radio del círculo virtual que el navegador va a utilizar para dibujar las esquinas redondeadas (*top-left, top-right, bottom-right* y *bottom-left*), también se pueden declarar de manera independiente (*border-top-left-radius, border-top-right-radius, border-bottom-right-radius* y *border-bottom-left-radius*)**.** Para definirlo con su propiedad shorthand seguir siempre la regla de las agujas del relog (*top-left-corner, top-right corner, bottom-right corner* y *bottom-left corner)*
  - Ejemplo *border-radius: 20px 10px 30px 50px;*

Css y Sass

Bordes

- Al igual que las propiedades margin y padding, border-radius también puede funcionar con solo dos valores. El primer valor se asigna a la primera y tercera esquina (arriba-izquierda, abajo-derecha) y la segunda a la segunda y cuarta esquina (arriba-derecha, abajo a la izquierda).

  Ejemplo
  - *border-radius: 40px / 20px;*

- También podemos dar forma a las esquinas proporcionando nuevos valores separados por una barra diagonal. Los valores de la izquierda representan el radio horizontal y los valores a la derecha del radio vertical. La combinación de estos valores genera una elipse.

  Ejemplo

  - *border-radius: 40px / 20px;*

Css y Sass

Bordes

También podemos delinear el elemento con un segundo borde (contorno) que se representa lejos de los bordes. El propósito es hacer que el elemento destaque. Algunos navegadores lo utilizan para delinear el texto, pero la mayoría de ellos dibujan un borde fuera de los límites de la caja del elemento.

- **outline-width :** define el ancho del contorno. Con unidades de medida o *(thin, medium* y *thick)*
- **outline-style :** estilo del contorno *(none, auto, dotted, dashed, solid, double, groove, ridge, inset* y *outset)*
- **outline-color :** color del contorno
- **outline-offset :** define el offset del contorno
- **outline :** propiedad shorthand (*width, style, y color*)

Css y Sass

Bordes

CSS nos permite definir bordes personalizados usando imágenes para que no sean solo líneas o curvas como hemos visto hasta ahora.

- **border-image-source :** define la imagen que se va a usar como borde (*url('fichero'))*
- **border-image-width :** ancho del borde
- **border-image-repeat :** define como se va a repetir la imagen *(repeat, round, stretch* y *space)*
- **border-image-slice :** define como se va a "rebanar" la imagen para mostrar las esquinas y el centro del borde (*4 valores)*
- **border-image-outset :** define el offset del borde
- **border-image :** propiedad shorthand

Css y Sass

Bordes

Ejemplo

- *border-image-source: url("diamonds.png");*
- *border-image-slice: 29;*
- *border-image-repeat: stretch;*

- *border-image: url("diamonds.png") 29 round;*

Ejercicio8

Bordes

A partir del código fuente del **ejercicio7** realizar las siguientes tareas

a) Crear una regla para que la clase **p2** defina una esquina reondeada de **2em** con borde de color Azul. El ancho del elemento tiene que ser de **10em**

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Sombras

CSS incluye las siguientes propiedades para generar una sombra para el cuadro del elemento, así como formas irregulares como texto.

- **box-shadow :** genera una sombra para el elemento. Acepta 6 valores (*Los desplazamientos horizontales y verticales de la sombra, el radio de desenfoque y el valor de dispersión, el color de la sombra y también incluyen la palabra clave* ***inset*** *para indicar que la sombra se proyectará dentro del cuadro*)

- **text-shadow :** genera una sombre a partir de un texto. Acepta 4 valores (*Podemos declarar los desplazamientos horizontales y verticales de la sombra, el radio de desenfoque y el color de la sombra*)

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Sombras

El desplazamiento (offset) puede ser positivo o negativo. Los valores indican la distancia horizontal y vertical desde la sombra hasta el elemento. Los valores negativos posicionan la sombra a la izquierda y encima del elemento, mientras que los valores positivos crean una sombra a la derecha y debajo del elemento. El valor 0 posiciona la sombra detrás del elemento, ofreciendo la posibilidad de generar un efecto de desenfoque a su alrededor.

Ejemplos

Crear una sombra a un elemento

- *box-shadow: rgb(150,150,150) 5px 5px;*

Añadir desenfoque

- *box-shadow: rgb(150,150,150) 5px 5px 20px;*

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Sombras

Extender la sombra añadimos el último valor

– *box-shadow: rgb(150,150,150) 10px 10px 20px 10px;*

Crear una sombra a un texto

– *text-shadow: rgb(150, 150, 150) 3px 3px 5px;*

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Degradados

Un degradado es un rango de colores que varía continuamente con una transición suave de un color a otro. Los degradados se crean como imágenes que tenemos que agregar al elemento como una imagen de fondo con la imagen de fondo o las propiedades de fondo.

- **linear-gradient(**position, angle, color-stop**) :** crea un degradado lineal (*position -> top, bottom, left* y *right, angle -> dirección del degradado, deg (degrees), grad (gradians), rad (radians) o turn (turns),* color-stop -> lista de colores separados por coma *)*
- **radial-gradient(**position, shape, color-stop, extent**) :** crea un degradado radial (*shape -> forma del degradado , circle* o *ellipse, extent -> forma que va a tener el degradado al final, closest-side, closest-corner, farthest-side* y *farthest-corner)*

Ejemplos

- *background: -webkit-linear-gradient(top, #FFFFFF, #666666);*
- *background: -webkit-linear-gradient(top right, #FFFFFF, #666666);*
- *background: -webkit-linear-gradient(30deg, #FFFFFF, #666666);*
- *background: -webkit-linear-gradient(top, transparent, #666666);*

Ejercicio9

Sombras y degradados

A partir del código fuente del **ejercicio8** realizar las siguientes tareas

a) Crear una regla para la clase **p2** que cree una sombra de texto para el texto del elemento (Elegir los valores)

b) Crear una regla para la clase **p4** que cree un degradado linear (Elegir los valores)

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Transformaciones 2D

Una vez creados, los elementos HTML permanecen estáticos, pero podemos modificar su posición, tamaño y apariencia con la propiedad **transform**. Esta propiedad realiza cuatro transformaciones básicas en un elemento. Podemos escalar, rotar, sesgar y traducir

- **scale(**x, y**) :** modifica la escala de un documento (Se puede usar **scaleX** y **scaleY**)
- **rotate(**angle**) :** rota un angulo el elemento (*deg (degrees), grad (gradians), rad (radians), or turn (turns))*
- **skew(**angle**) :** sesga un element
- **translate(**x, y**) :** mueve el elemento a la posición indicada

Si ***scale*** recibe un solo parámetro, ese mismo se usa como argumento **x** e **y**

Ejemplos

- *transform: scale(2);*
- *transform: rotate(30deg);*
- *transform: skew(20deg);*

Css y Sass

Transformaciones 2D

La pantalla de un dispositivo se divide en filas y columnas de píxeles (la unidad mínima de la pantalla). Con el propósito de identificar la posición de cada píxel, los equipos utilizan un sistema de coordenadas, donde las filas y columnas de píxeles se cuentan desde la esquina superior izquierda hasta la esquina inferior derecha de la cuadrícula, comenzando desde 0 (los valores aumentan de izquierda a derecha y arriba a abajo).

La función **translate()** utiliza el sistema de coordenadas para establecer la posición del elemento, con su posición actual utilizada como referencia. La esquina superior izquierda del elemento se considera que está en la posición 0,0 para que los valores negativos muevan el elemento a la izquierda o por encima de la posición original, y los valores positivos lo muevan hacia la derecha o por debajo.

Ejemplos

- *transform: translateY(100px) rotate(45deg) scaleX(0.3);*

Css y Sass

Transformaciones 3D

Además de las transformaciones 2D, también podemos realizar transformaciones 3D en elementos HTML. Estos tipos de transformaciones se realizan teniendo en cuenta un tercer eje que representa la profundidad y se identifica con la letra z

- **scale3d(**x, y, z**) :** asigna una escala 3d al element
- **rotate3d(**x, y, z, angle**) :** rota el elemento *(rotate3d(5, 2, 6, 30deg))*
- **translate3d(**x, y, z**) :** mueve el elemento a otra posición en el espacio 3D
- **perspective(**value**) :** añade profuncidad a la escena aumentando el tamaño del elemento que está más cerca del usuario

Ejemplos

- *transform: perspective(500px) rotate3d(0, 1, 0, 45deg);*

Ejercicio10

Transformaciones

A partir del código fuente del **ejercicio9** realizar las siguientes tareas

a) Crear una regla para la clase **p2** que le haga rotar 180 grados

b) Crear una regla para la clase **p4** que use **rotate3d**

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Transiciones

Una animación real requiere una transición entre los dos pasos del proceso.

- **transition-property :** esta propiedad especifica las propiedades que participan en la transición. La propiedad incluye el valor **all** para especificar que todas las propiedades se transición a los nuevos valores
- **transition-duration :** duración en segundos
- **transition-timing-function :** establece la función usada para calcular los valores de la transición *(ase, ease-in, ease-out, ease-in-out, linear, step-start* y *step-end)*
- **transition-delay :** indica al navegador el tiempo que tiene que esperar para comenzar la animación
- **transition :** propiedad shorthand

Ejemplos

- *transition: transform 1s ease-in-out 0s;*

Css y Sass

Animaciones

La propiedad de transición crea una animación básica, pero solo dos estados están implicados en el proceso, el estado inicial determinado por los valores actuales de las propiedades y el estado final, determinados por los nuevos valores. Para crear una animación real, necesitamos declarar más de dos estados, como los fotogramas de una película

- **animation-name :** nombre de la animación para poder identificarla
- **animation-duration :** duración de la animación en segundos
- **animation-timing-function :** determina cómo se llevará a cabo el proceso de animación en función de una curva de Bézier declarada por las palabras clave *(ease, linear, ease-in, ease-out* y *ease-in-out)*
- **animation-delay :** tiempo que se espera hasta comenzar la animación
- **animation-iteration-count :** número de veces que la animación será ejecutada (*infinite)*
- **animation-direction :** dirección de la animación *(normal , reverse, alternate* y *alternate-reverse)*
- **animation-fill-mode :** define como interactua la animación con los estilos del elemento (*none , forwards -> mantiene el elemento de estilo con las propiedades aplicadas en el último fotograma de la animación, backwards-> aplica los estilos del primer fotograma al elemento tan pronto como se define la animación, both)*
- **animation :** propiedad shorthand

Css y Sass

Animaciones

Las propiedades anteriores configuran la animación, pero los fotogramas se declaran con la regla **@keyframes**. La regla debe identificarse con el nombre utilizado para configurar la animación, y tiene que incluir la lista de las propiedades que queremos modificar en cada fotograma. La posición de cada fotograma en la animación viene determinada por un valor en porcentaje, donde 0% es el primer fotograma o el inicio de la animación y 100% es el final.

```
article {
  margin: 30px;
  padding: 15px;
  text-align: center;
  border: 1px solid;
  animation: mianimacion 1s ease-in-out 0s infinite normal none;
}
```

Css y Sass

Animaciones

```
@keyframes mianimacion{
 0% {
  background: #FFFFFF;
 }
 50% {
  background: #FF0000;
 }
 100% {
  background: #FFFFFF;
}
}
```

Ejercicio11

Animaciones

A partir del código fuente del **ejercicio10** realizar las siguientes tareas

a) Crear una animación para la clase **p2** que vaya cambiando de color (rojo,amarillo,azul, verde,rojo) y de posición (0,0),(200,0),(200,200),(0,200),(0,0)

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Modelo de cajas

Los navegadores crean una caja virtual alrededor de cada elemento para determinar el área que ocupan. Para organizar estos cuadros en la pantalla, los elementos se clasifican en dos tipos básicos: Block y inline. Las principales diferencias entre estos dos tipos de elementos es que los elementos Block tienen un tamaño personalizado e insertan saltos de línea, mientras que los elementos inline son del tamaño de su contenido y no insertan saltos de línea. Debido a sus características, los elementos Block se colocan uno por línea, y los elementos inline se colocan junto a otros elementos inline

Elementos que definen la estructura/diseño de nuestro documento, como <section>, <nav>, <header>, <footer> o , se declaran como elementos Block de forma predeterminada, y otros como , o , que representan el contenido de esos elementos son declarados por los navegadores como Elementos inline.

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Modelo de cajas

- **display :** define el tipo de caja usada para renderizar el element (*none, block, inline, inline-block)*
- **visibility :** determina si un element va a ser visible o no (*hidden -> el element y su contenido son renderizados, ocupan el espacio en la pantalla, pero no visibles)*
- **float :** permite a un elemento de tipo *bloque ,* flotar en un la o u otro de su contenedor (*none,left, right)*
- **clear :** recupera el flujo normal del documento, no permitiendo que el elemento se mantenga flotando a la izquierda, derecha o ambos lados *(none, left, right* y *both)*

Ejemplo

```
.caja1 {
  float:left;
}
.limpiafloat
{
  clear:both;}
```

Css y Sass

Modelo de cajas : posicionamiento

La posición de un elemento puede ser relativa o absoluta. Con el posicionamiento relativo, las cajas se colocan una tras otra en el espacio designado por su contenedor. Si no hay espacio disponible o los elementos no están flotando alrededor de otros elementos, se colocan en una nueva línea. Este es el modo de posicionamiento por defecto, pero hay otro modo para posicionar los elementos denominados posicionamiento absoluto. El posicionamiento absoluto nos permite establecer las coordenadas exactas donde queremos posicionar cada elemento.

- **position :** define el tipo de posicionamiento usado por el element (***static*** *(según el flujo del documento),* ***relative*** *(relativo a la posición original del documento),* ***absolute*** *(posición absoluta relativa al contenedor del elemento),* ***fixed*** *(posición relative a la ventana del navegador))*
- **top :** especifica la distancia entre el margen superior del elemento y el margen superior de su contenedor
- **bottom :** especifica la distancia entre el margen inferior del elemento y el margen inferior de su contenedor
- **left :** especifica la distancia entre el margen izquierdo del elemento y el margen izquierdo de su contenedor
- **right :** especifica la distancia entre el margen derecho del elemento y el margen derecho de su contenedor
- **z-index :** define un índice que determina la posición del elemento en el **eje z**. Un elemento con un índice más alto se dibujará sobre un elemento con un índice inferior

Css y Sass

Modelo de cajas : columnas

Además de las propiedades que hemos visto para organizar las casillas en la pantalla, CSS también incluye un grupo de propiedades para facilitar la creación de columnas.

- **column-count :** especifica el número de columnas que el navegador tiene que generar para distribuir el contenido del elemento
- **column-width :** especifica el ancho de las columnas
- **column-span :** se aplica a los elementos dentro de un cuadro para determinar si el elemento se colocará dentro de una columna o se extendió a través de varias columnas. Los valores posibles son **all**, lo que significa "todas las columnas ", y **none** (por defecto).
- **column-fill :** especifica como se dividirá el contenido en columnas (*auto, balance)*
- **column-gap :** define el espacio entre columnas
- **columns :** propiedad shorthand (*column-count* y *column-width*)

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio12

Modelo de cajas

A partir del código fuente del **ejercicio11** realizar las siguientes tareas

a) Disponer todos los elementos **<p>** en un modelo de cajas alineados a la izquierda si es elemento impar y a la derecha si es elemento par

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Modelo de cajas flexible

El modelo de caja flexible resuelve los problemas del modelo de caja tradicional de una manera elegante. Aprovecha las herramientas utilizadas por el modelo de caja tradicional, como el posicionamiento absoluto y las columnas, pero en lugar de flotar los elementos ,organiza los cuadros utilizando contenedores flexibles. Un contenedor flexible es un elemento que permite que su contenido sea flexible. En este nuevo modelo, cada conjunto de cuadros debe tener un cuadro primario que configure las características de todos los cuadros, como en el ejemplo siguiente.

Para este propósito, CSS ofrece los valores **flex** y **inline-flex** para la propiedad **display**. Así declaramos un contenedor flexible

```
section{
  display: flex;
}
```

Para que un elemento dentro de un contenedor flexible se convierta en flexible, debe declararse como tal.

Css y Sass

Modelo de cajas flexible

Las cajas flexibles se expandirá o reducirá para llenar el espacio adicional dentro de su caja principal. La distribución del espacio depende de las propiedades del resto de las cajas.

Si todas las casillas se configuran como flexibles, el tamaño de cada una de ellas dependerá del tamaño de su cuadro principal y del valor de la propiedad **flex.**

Ejemplo

```
.padre{
        display: flex;
}
```

- **flex-direction :** define el orden y la orientación de los elementos en un contenedor *flexible ,* es el eje principal *(row, row-reverse, column* y *column-reverse),*
- **order :** específica el orden de los elementos
- **justify-content :** alinea los elementos en el eje principal *(flex-start, flex-end, center, space-between* y *space-around)*
- **align-items :** alinea los elementos en el eje transversal (*flex-start, flex-end, center, baseline, stretch)*
- **align-self :** alinea un elemento *(auto, flex-start, flex-end, center, baseline, stretch)*
- **flex-wrap :** determina si se permiten usar varias líneas para disponer los elementos *(nowrap, wrap, wrap-reverse)*

Css y Sass

Modelo de cajas flexible

- **align-content :** distribuye el espacio de los elementos en el eje transversal *(flex-end, center, space-between, space-around, stretch)*

El tamaño de cada caja se calcula multiplicando el valor del tamaño de la caja padre por el valor de su propiedad **flex** dividido por la suma de los valores de **flex-grow** de todos los cuadros

- **flex-grow:** indica cuánto se va a expandir un elemento. El tamaño se determina teniendo en cuenta los valores asignados al resto de los elementos en el cuadro (los hermanos).
- **flex-shrink :** indica cuánto se va a reducir un elemento. El tamaño se determina a partir de los valores asignados al resto de los elementos en el cuadro (los hermanos).
- **flex-basis :** declara un tamaño inicial para el elemento
- **flex :** propiedad shorthand *(flex-grow, flex-shrink)*
- **max-width :** establece el máximo ancho permitido para un elemento
- **min-width :** establece el mínimo ancho permitido para un elemento
- **max-height :** establece el máximo alto permitido para un elemento
- **min-height :** establece el mínimo alto permitido para un elemento

https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Flexbox

Css y Sass

Modelo de cajas flexible

prueba.css U

becaeoi > HTML > css > # prueba.css > .article1

```css
.flex-col{
    display: flex;

    /* Eje principal : columna */
    /* Eje transversal : fila */
    flex-direction: column;
    justify-content:space-around;
}
section{
    border:1px solid blue;
    width: 200px;
    height: 200px;
}

.article1{
    border:2px solid red;
    margin:20px 0;
    height: 30vh;
}

.article2{
    border:2px solid red;
    margin:20px 0;
    height: 60vh;
}
```

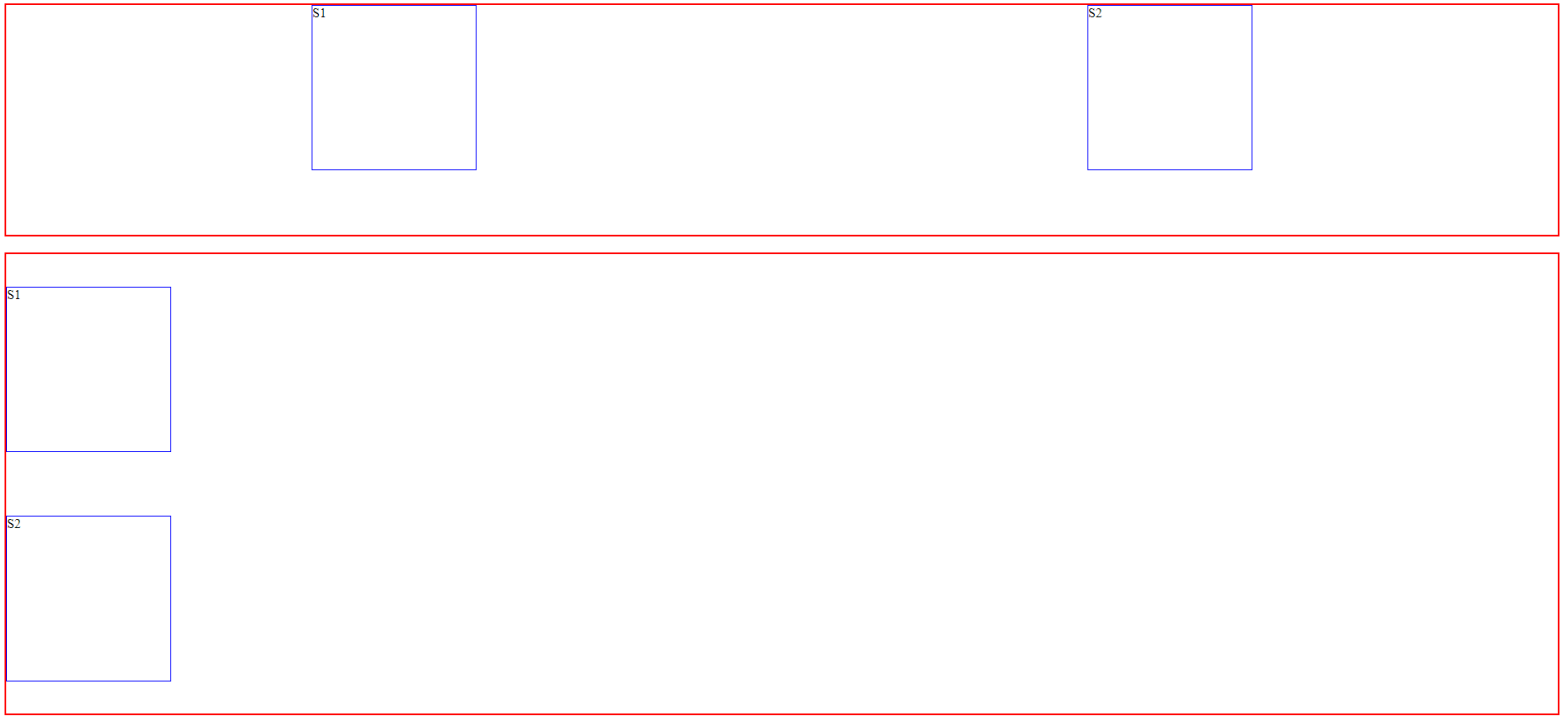
prueba.html M

becaeoi > HTML > prueba.html > html > body > article.flex-col.article2

```html
<title>Prueba</title>
</head>
<body>

    <article class="flex-row article1">
        <section>
            S1
        </section>

        <section>
            S2
        </section>

    </article>

    <article class="flex-col article2">
        <section>
            S1
        </section>

        <section>
            S2
        </section>

    </article>
```

Css y Sass

Modelo de cajas flexible



GOBIERNO DE ESPAÑA
MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO
EOI Escuela de organización industrial
Unión Europea
Fondo Social Europeo
Iniciativa de Empleo Juvenil
El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Ejercicio13

Modelo de cajas flexibles

A partir del código fuente del **ejercicio13** realizar las siguientes tareas

a) Disponer todos los elementos **<p>** en un modelo de cajas flexible. Disponerlos todos en la misma fila, uno a continuación del otro

b) Disponer todos los elementos **<p>** en un modelo de cajas flexible. Disponerlos todos en la misma columna, uno a continuación del otro. No dejar que el contenido de las cajas ocupe el eje x por completo.

c) Hacer lo mismo que en el punto anterior pero posiciones las cajas a la derecha de la pantalla

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Modelo en grid

Permite la alineación de los elementos en varias direcciones (filas o columnas)
Define una página como un **grid** con **grid rows** , **grid columns** y **grid áreas.**

- **display:** esta propiedad será **grid** en lugar de **flex** y **block** como hasta ahora

- **grid-gap :** valor para separar los elelemtos del grid
  - *grid-gap:1em;*

- **grid-template-columns :** definimos el número de columnas y el ancho de las mismas
  - *grid-template-columns: 5em 5em 5em; (Definimos 3 columnas de 5em de ancho)*

- **grid-template-rows :** definimos el número de filas y el tamaño de las mismas
  grid-template-rows: 5em 5em 5em; (definimos 3 filas)

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Modelo en grid

En la definición de las filas y columnas podemos poner el valor **auto** para que las calcule automáticamente.

Tiene muchas propiedades en comun con el modelo de caja flexible. Podemos consultar el resto de carácterísticas en el siguiente enlace.

https://developer.mozilla.org/en-US/docs/Web/CSS/grid

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Media queries

Una media Query es una regla reservada en CSS que se incorporó para permitir a los desarrolladores detectar el medio en el que se muestra el documento. Por ejemplo, mediante media queries, podemos determinar si el documento se está visualizando en un monitor o si se envía a una impresora y se asignan los estilos apropiados para cada caso. Para este propósito, media queries ofrece las siguientes palabras clave.

- **width :** determina el ancho en el que se aplicarán las propiedades
- **height :** determina el alto en el que se aplicarán las propiedades
- **min-width :** determina el mínimo ancho en el que se aplicarán las propiedades
- **max-width :** determina el máximo ancho en el que se aplicarán las propiedades

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

Media queries

- **aspect-ratio :** determina el aspect en las cuales las propiedades serán aplicadas

- **orientation :** determina la orientación en las cuales las propiedades serán aplicadas (portrait, landscape)

- **resolution :** determina la densidad de píxeles a la que se aplicarán las propiedades. Puede tomar valores en puntos por pulgada (DPI), puntos por centímetro (DPCM) o por ratio de píxel (dppx).
Por ejemplo, para detectar una pantalla con una visualización retina de una escala de 2, podemos utilizar el valor 2dppx

Css y Sass

Media queries

Ejemplos de media queries :

<link rel="stylesheet" media="(max-width: 480px)" href="responsivephones.css">

Además de incluirlo a la hora de decidir que hoja de estilos cargar según la media query,
Podemos definir los media queries dentro de las hojas de estilo con la regla **@media**

```
 @media (max-width: 1024px) {
  body {
    background-color: #3333FF;
  }
}
@media (max-width: 768px) {
  body {
    background-color: #FF33FF;
  }}
```

Css y Sass

SAS

Los preprocesadores de CSS son herramientas que facilitan el desarrollo de código CSS, ampliando los elementos a nuestra disposición para su escritura, como es el manejo de variables,

condicionales, funciones y otros elementos propios de lenguajes de programación más avanzados. SASS nos permite emplear esos elementos que, una vez terminados, procesará para crear el

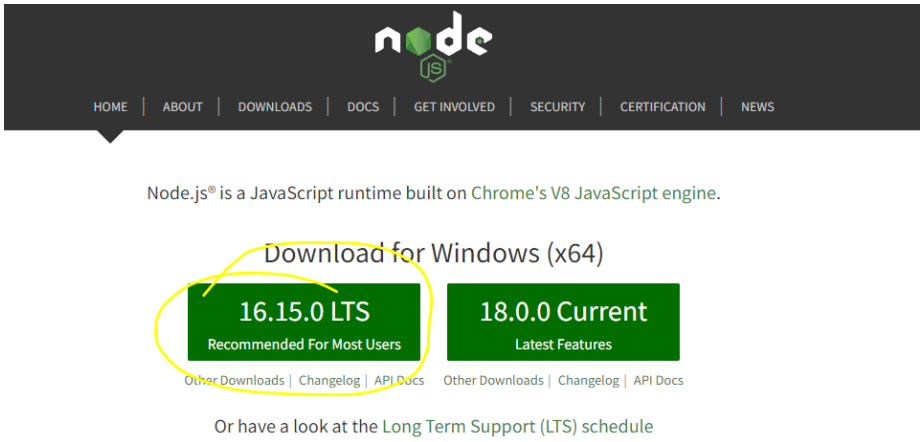
código CSS válido resultante. Es un meta-lenguaje en la parte superior de CSS que se utiliza para describir el estilo de un documento de forma limpia y estructurada

SASS es un preprocesador de css podemos usar nodejs mediante la línea de comandos para hacer la instalación y a la hora de compilar el código para obtener el archivo css resultante. A día de hoy existen dos variaciones de código en SASS, una para archivos .scss (más actual) y otra para archivos .sass (más antigua).

Css y Sass

SAS : instalación

- Instalaremos primero NodeJs https://nodejs.org/en/



Css y Sass

SAS : instalación

- Teniendo instalado **nodejs**, podemos instalarlo a través de **npm**

npm install –g sass
npm install –g node-sass

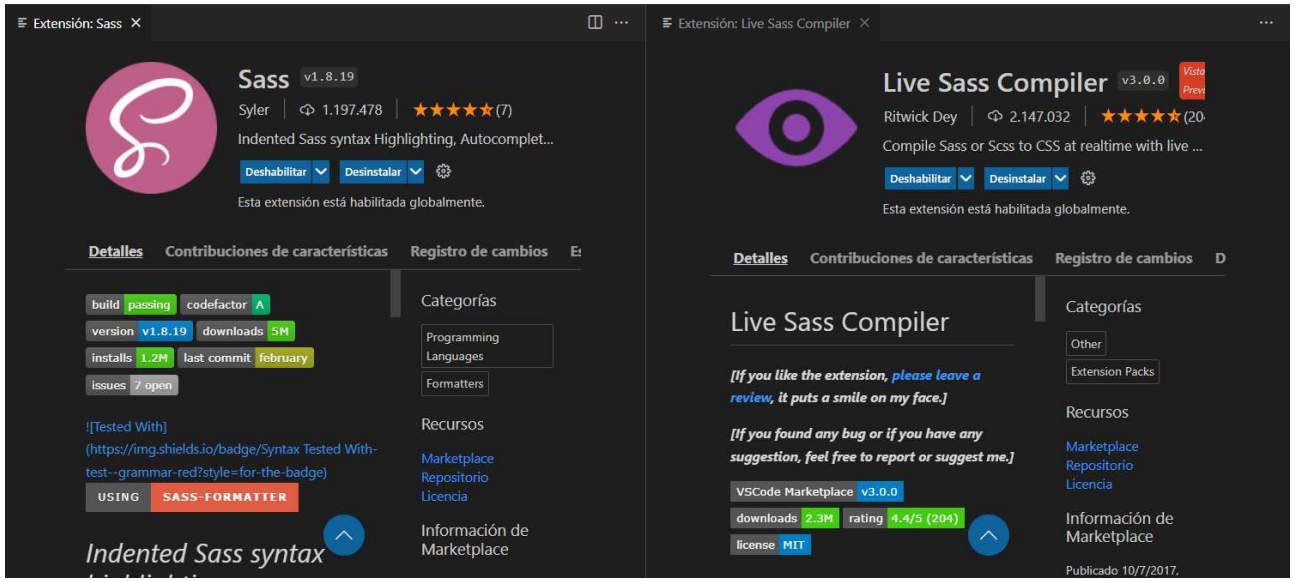
Una vez instalado lo tenemos disponible por línea de comando

Si ejecutamos en un terminal el comando **sass** nos saldrá la información de cómo usarlo.

Css y Sass

SAS : instalación

Extensiones de VScode



Css y Sass

SAS : reglas anidadas

Cada elemento HTML puede tener en su interior otros elementos (por ejemplo). Si en CSS es necesario identificar todos los elementos por separado, con SASS se pueden anidar.

```
nav {
        width: 80%;
        height: 23px;
        ul { list-style-type: none; }
        li {
                float: left;
                a { font-weight: bold; }
        }
}
```

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

SAS : referencia a selectores padres

Se puede utilizar el carácter & para hacer referencia al selector padre dentro del cual se encuentra la regla anidada.

```
nav {
        width: 80%;
        height: 23px;
        &:hover { text-decoration:underline;}
        ul { list-style-type: none; }
        li {
                float: left;
                a { font-weight: bold; }
        }
}
```

Vamos a implementar este ejemplo y veremos como funciona **Sass** rápidamente. Vamos a instalar un plugin nuevo para Atom **sass-autocompile**, el cual nos facilitará el proceso de la compilación de .scss a .css, solo guardando el fichero

Ejercicio14

Plantilla SASS

Crear una plantilla Html5 llamada **ejercicioSass.html** y un archivo SASS **scss/ejercicio14.scss** con el siguiente código. Una vez creada la plantilla compilar la plantilla para que genere el código **css** correspondiente y analizarlo.

```
nav {
        width: 80%;
        height: 23px;
        &:hover { text-decoration:underline;}
        ul { list-style-type: none; }
        li {
                float: left;
                a { font-weight: bold; }
        }
}
```

Css y Sass

SAS : variables

Declaración de variables mediante el símbolo **$**, en las que se almacenan las propiedades del elemento. Las variables también se pueden emplear para crear nombres de propiedades o selectores mediante la sintaxis ***#{$variable}***

Ejemplo

```
$altura: height;
$ancho: width;
$el1:5em;
nav {
        #{$ancho}: 80%;
        #{$altura}: 23px;
        ul { list-style-type: none; }
        li {
                float: left;
                a { font-weight: bold; font-size: $el1;}
        }
}
```

Css y Sass

SAS : variables de texto

Ejemplo

```
$altura: height;
$ancho: width;
nav {
        #{$ancho}: 80%;
        #{$alto}: 23px;
        ul { list-style-type: none; }
        li {
                float: left;
                a { font-weight: bold; }
        }
}
```

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

SAS : operaciones y funciones

Se pueden realizar operaciones básicas con los números como +, -, *, /, y %. Por otro lado, SASS permite realizar ciertas funciones sobre colores y textos.

A la hora de operar, hay que tener en cuenta que los operandos implicados en la operación tienen que pertenecer al mismo dominio de la unidad (px, pt, em, etc.).

Algunas funciones

if($condition, $if-true, $if-false) : devuelve uno de dos valores, según si se cumple la condición o no.

type-of($value) : devuelve el tipo de un valor.

to-upper-case($string) Convierte una cadena a mayúsculas

Css y Sass

SAS : operaciones y funciones

index($list, $value) Devuelve la posición de un elemento concreto de una lista.

length($list) Devuelve la cantidad de elementos de una lista.

nth($list, $n) Devuelve un elemento concreto de una lista.

join($list1, $list2, [$separator]) Une dos listas en una.

append($list1, $val, [$separator]) Añade un valor al final de una lista.

rgb($red, $green, $blue) Crea un color a partir de los valores RGB

grayscale($color) Convierte un color a escala de grises.

round($value) Redondea al número entero más cercano.

Css y Sass

SAS : operaciones y funciones

min($numbers…) Encuentra el mínimo de una serie de números.

str-index($string, $substring) Índice de la primera aparición de $substring en $string

str-length($string) Devuelve el número de caracteres de una cadena

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

SAS : listas y mapas

Las listas son el tipo de dato que utiliza Sass para representar los valores que normalmente se utilizan en las propiedades CSS como **margin: 10px 15px 0 0** o **font-face: Helvetica, Arial, sans-serif**. Las listas son simplemente una colección de valores separados por comas o espacios en blanco

```
$altura: height;
$ancho: width;
$el1:1em;
$listaMargen:(10px 15px 0 0);
nav {
        #{$ancho}: 80%;
        #{$altura}: 23px;
        ul { list-style-type: none; }
        li {
                float: left;
                a { font-weight: bold; font-size: $el1; margin: $listaMargen;}
        }}
```

Css y Sass

SAS : listas y mapas

Los mapas son asociaciones de claves y valores. Para recuperar un valor se usa la función *map-get($mapa,"clave")*

```
$altura: height;
$ancho: width;
$el1:1em;
$map: (clave1: 1em, clave2: 2em, clave3: 3em);
$listaMargen:(10px 15px 0 0);
nav {
        #{$ancho}: 80%;
        #{$altura}: 23px;
        ul { list-style-type: none; }
        li {
                float: left;
                a { font-weight: bold; font-size: map-get($map,"clave1"); margin: $listaMargen;}
        }
}
```

Css y Sass

SAS : directivas de control

Las directivas de control son recursos avanzados que no se recomiendan para el mantenimiento diario de los estilos. En esos casos es más indicado emplear los mixins que veremos después

@while

```
$i: 6;
@while $i > 0 {
        .item-#{$i} { width: 2em * $i; }
        $i: $i - 2;}
```

Css y Sass

SAS : directivas de control

@if y @else if

```
$type: monster;
p {
@if $type == ocean {
color: blue;
} @else if $type == matador {
color: red;
} @else if $type == monster {
color: green;
} @else {
color: black;}}
```

Css y Sass

SAS : directivas de control

@for

```
@for $i from 1 through 3 {
.item-#{$i} { width: 2em * $i; }
}
```

@each

```
@each $animal in puma, sea-slug, egret, salamander {
.#{$animal}-icon {
  background-image: url('/images/#{$animal}.png');
 }
}
```

Css y Sass

SAS : mixins

Mixins son elementos similares a las funciones en otros lenguajes, que permiten reutilizar estilos, propiedades y selectores. Se declaran mediante la directiva @mixin, y a continuación se crea el estilo. Los mixins pueden recibir argumentos. Para invocarlos, basta con hacerlo a través de la directiva @include seguida por el nombre del mixin.

```
@mixin rounded-top-left {
        $vert: top;
        $horz: left;
        $radius: 10px;
        border-#{$vert}-#{$horz}-radius: $radius;
        -moz-border-radius-#{$vert}#{$horz}: $radius;
        -webkit-border-#{$vert}-#{$horz}-radius: $radius;
}

#navbar li { @include rounded-top-left; }
#footer { @include rounded-top-left; }
```

Css y Sass

SAS : mixins

Los argumentos se indican entre paréntesis y separados por comas.También se les puede dar un valor por defecto.

```
@mixin rounded($vert, $horz, $radius: 10px) {
        border-#{$vert}-#{$horz}-radius: $radius;
        -moz-border-radius-#{$vert}#{$horz}: $radius;
        -webkit-border-#{$vert}-#{$horz}-radius: $radius;
}
#navbar li { @include rounded(top, left); }
#footer { @include rounded(top, left, 5px); }
#sidebar { @include rounded(top, left, 8px); }
```

Un mixin, además de contener sus propios estilos, podemos incluirle un bloque de estilos externo mediante la directiva @content indicada dentro del mixin. Al incorporar un mixin mediante @include, podemos incluir otros estilos que serán agregados dentro del mixin en el lugar donde se haya declarado la directiva @content

Css y Sass

SAS : mixins

```
@mixin apply-to-ie6-only {
        * html {
                @content;
        }
}

@include apply-to-ie6-only {
        #logo {
                background-image: url(/logo.gif);
        }
}
```

Css y Sass

SAS : extender clases

La directiva @extend permite que varias clases compartan estilos. En css el equivalente sería a crear una serie de estilos a dos clases separadas por comas en la declaración. No se duplica el código de una clase a la otra.

```
.error {
        border: 1px #f00;
        background-color: #fdd;
}
.seriousError {
        @extend .error;
        border-width: 3px;
}
```

Css y Sass

SAS : importar estilos

Mediante la directiva @import se pueden importar estilos de otros archivos. Se puede no indicar la extensión del archivo, de esta manera se hace compatible con los dos tiempos, .scss y .sass. Al importar archivos, debemos tener cuidado de no importar más de una vez el mismo archivo

@import "roundedfunc";

```
#navbar li { @include rounded(top, left); }
#footer { @include rounded(top, left, 5px); }
#sidebar { @include rounded(top, left, 8px); }
```

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea Fondo Social Europeo Iniciativa de Empleo Juvenil El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

Css y Sass

SAS : funciones propias

SASS nos permite crear nuestras propias funciones, las cuales pueden aceptar argumentos y devolver valores. Se realiza mediante la directiva @function seguida por el nombre de la función. Por último se indicarían los parámetros de entrada, y para devolver el resultado
se emplearía la directiva @return.

```
$grid-width: 40px;
$gutter-width: 10px;

@function grid-width($n) {
        @return $n * $grid-width + ($n - 1) * $gutter-width;
}
#sidebar { width: grid-width(5); }
```

Css y Sass

SAS : estilos parciales

Consisten en archivos .scss que serán importados con @import y que sirven de apoyo para otros archivos. El nombre del archivo debe comenzar por un guión bajo para diferenciarlos del resto, y entre las ventajas de su uso se encuentran:

- Estructuras modulares
- Mejor organización
- Posibilidad de reutilizar variables y/o mixins

Al mismo tiempo, un estilo parcial también puede importar otros archivos o mixins. Solo debemos tener cuidado de no repetir código en el archivo final .css.

/* Archivo _variables.scss */

$ancho: 960px;

$azul: #3B56FF;

Css y Sass

SAS : estilos parciales

```
/* Archivo _cabecera.scss */
.cabecera{
width: $ancho;
}
/* Archivo styles.scss */
// Importa cada estilo parcial
@import "variables";
@import "cabecera";
```

Ejercicio15

Prueba de SASS

A partir del código fuente del *ejercicio14* realizar más pruebas de lo visto con Sass. Hacer especial hincapié en los *Funciones propias*, *Mixins, Funciones* y *estilos parciales*

GOBIERNO DE ESPAÑA · MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO · EOI Escuela de organización industrial
Unión Europea · Fondo Social Europeo · Iniciativa de Empleo Juvenil · El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL

⇨Enlaces de interés

- https://developer.mozilla.org/es/docs/Web/CSS
- https://developer.mozilla.org/es/docs/Web/CSS/Referencia_CSS
- https://fonts.google.com/
- https://htmlcolorcodes.com/
- https://sass-lang.com

GOBIERNO DE ESPAÑA MINISTERIO DE INDUSTRIA, COMERCIO Y TURISMO EOI Escuela de organización industrial
Unión Europea
Fondo Social Europeo
Iniciativa de Empleo Juvenil
El FSE invierte en tu futuro
SISTEMA NACIONAL DE GARANTÍA JUVENIL