

Stylus

Previously in "Clase 07"

En Sublime creamos un nuevo archivo suelto junto con los demás y lo nombramos:

stylus.styl

Dentro del archivo .styl ya guardado tipeamos respetando el indentado:

```
body  
  background red
```

Salvamos.

En la terminal, compilamos el archivo .styl

```
stylus stylus.styl -c -w
```

Veamos qué ocurre... jeje

Pre-Procesador CSS

¿Que es un Pre-Procesador CSS?

Herramientas para optimizar y administrar fácilmente nuestro código css.

- Trabajo más ordenado
- Trabajo más rápido

¿Cómo funciona?

Optimizar el código CSS para generar un archivo css más liviano. Optimizando el tiempo de carga de un sitio web.

Además de Stylus, los más conocidos son:

- Sass
- Less

Pre-Procesador CSS

Stylus. ¿Por qué Stylus?

Less no es profesional, es más para principiantes.

Sass es muy bueno y potente al igual que Stylus, pero su modo de codificar es más complicado, menos sencillo e intuitivo que Stylus.

The Stylus logo features the word "stylus" in a lowercase, elegant script font. A vibrant green swoosh underline starts under the 's' and extends under the 'us', adding a dynamic and modern feel to the branding.

Stylus es un nuevo lenguaje revolucionario, proporcionando una forma eficiente, dinámica y expresiva para generar CSS. Su sintaxis con sangría.

Stylus

1 Creamos nuestro archivo.styl y lo guardamos en la carpeta de proyecto.

2 Abrimos una terminal y nos posicionamos en la consola en donde guardamos nuestro archivo.styl.

3 Le decimos a Stylus que compile nuestro archivo.styl en un archivo.css.

Muchos de ustedes (o todos) se preguntarán para qué hacer todo este kilombo:

- Como veníamos escribiendo css, lo ordenábamos para "humanos", con espacios y tabulando para tener una buena estructura visual, esto, tiene un peso extra por todos los espacios en blanco y tabulaciones, Stylus quitará todos los espacios y hará una sola línea de código para optimizar al máximo el css.

- Podremos codear de una manera mucho más sencilla y sin complicaciones, primero no necesitamos poner las llaves y puntos, basta con colocar así:

```
body  
  background red
```


Stylus Variables

¿Qué es una variable?

Es un símbolo o texto, que toma un valor ya sea numérico o de texto.

Hagamos una variable.
En nuestro archivo .styl :

```
celeste=#18afbf
```

Escribir:

```
body
  background celeste
```

Hagamos otra variable.
`sans=Arial`

Escribir:

```
body
  background celeste
  font sans
```

Último ejemplo

```
tamano=14px
sans=tamano Arial
```

Escribir:

```
body
  background celeste
  font sans
```

Stylus Sintaxis

Sangría o Indentación

Sintaxis más corta y eficiente que nos hará ahorrar mucho tiempo.

El espacio en blanco puede ser usado en lugar de las llaves de apertura y de cierre.

Debemos de colocar el nombre de la clase, id o etiqueta, luego saltarnos a la siguiente línea y allí poner un espacio y colocar nuestras propiedades. Con eso indicamos a Stylus que lo que está dentro de la clase (hablando de forma indentada) será una propiedad y valor respectivamente.

```
C:\xampp\htdocs\estilo.styl - Sublime Text 2 (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

estilo.styl
1 body
2   background: red;
3   color: white;
4
5
6

C:\xampp\htdocs\estilo.css - Sublime Text 2 (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

estilo.styl  estilo.css
1 body {
2   background: #f00;
3 }
4
```

Stylus

```
gris_oscuro = #484848
body
  background #F1F1F2
  color gris_oscuro
  font-family Arial
  font-size 16px
```



Stylus

```
gris_oscuro = #484848
body
  background #F1F1F2
  color gris_oscuro
  font-family Arial
  font-size 16px

header
  background gris_oscuro
  color white
  padding .5em
  |
```



Stylus

```
amarillo = #fcbf16
gris_oscuro = #484848
body
  background #F1F1F2
  color gris_oscuro
  font-family Arial
  font-size 16px
header
  background gris_oscuro
  color white
  padding .5em
nav
  background amarillo
```



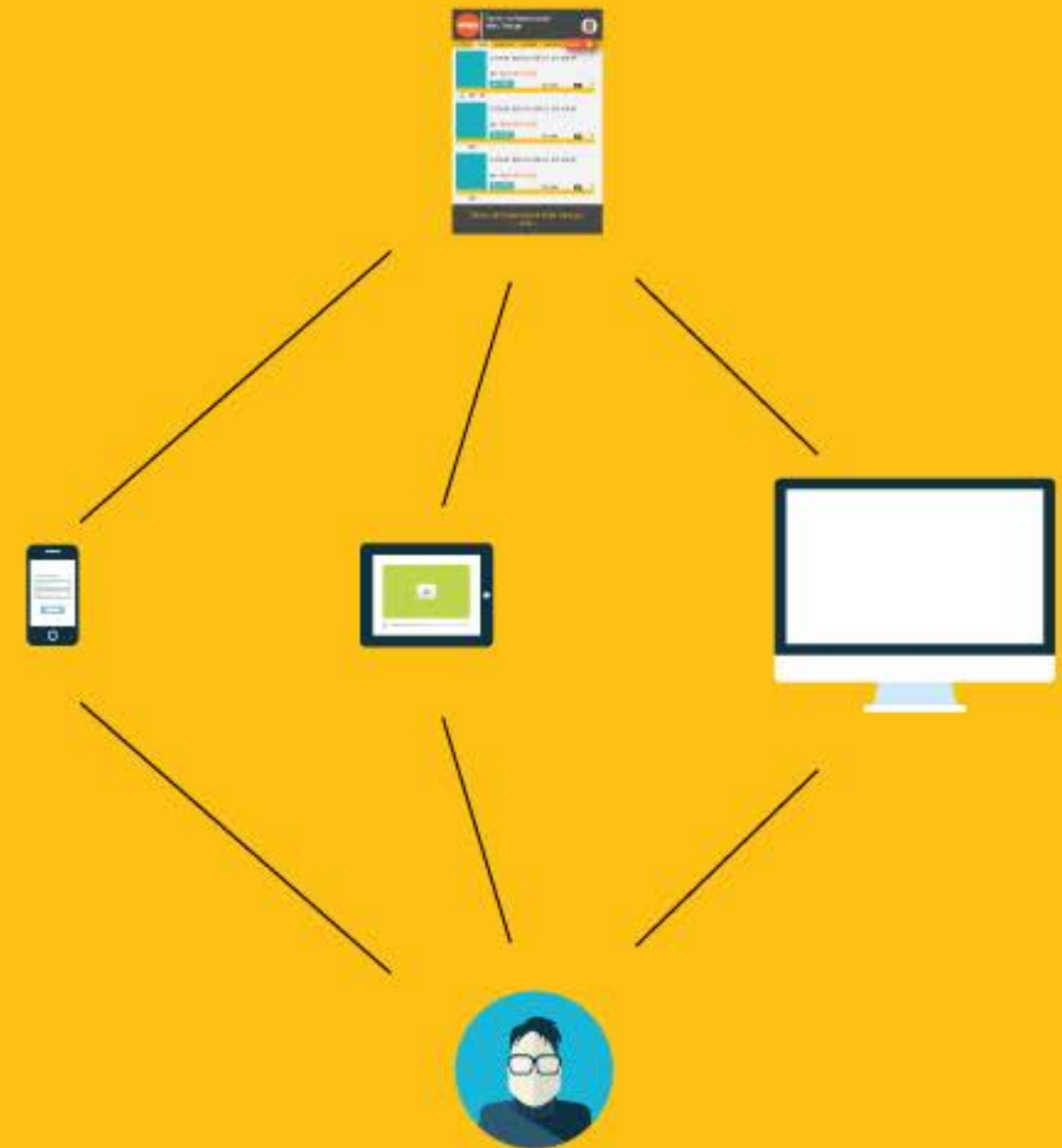
RWD

Fundamentos

Los usuarios han cambiado la forma en consumir contenidos web.

La llegada de celulares, tablets y notebooks han replanteado la forma de diseñar para web.

El RWD está todavía en etapa de desarrollo, no es estandar todavía.



CSS

Fundamentos

HTML

Describe la estructura de una página web.

HTML



CSS

Describe la presentación de una página web.



CSS

Fundamentos



Modelo de caja

Tiene 5 principales propiedades:

Width
Height
Padding
Border
Margin



CSS

Fundamentos



Modelo de caja

Tiene 5 principales propiedades:

Width
Height

Padding
Border
Margin

Width y **Height** se aplican al área de contenido.

El **padding** es lo que se "come" entre el border y el área de contenido.

El **border**, es eso, el borde que se le aplica al área de contenido por fuera.



Border + Padding afectan el ancho y alto del área de contenido.

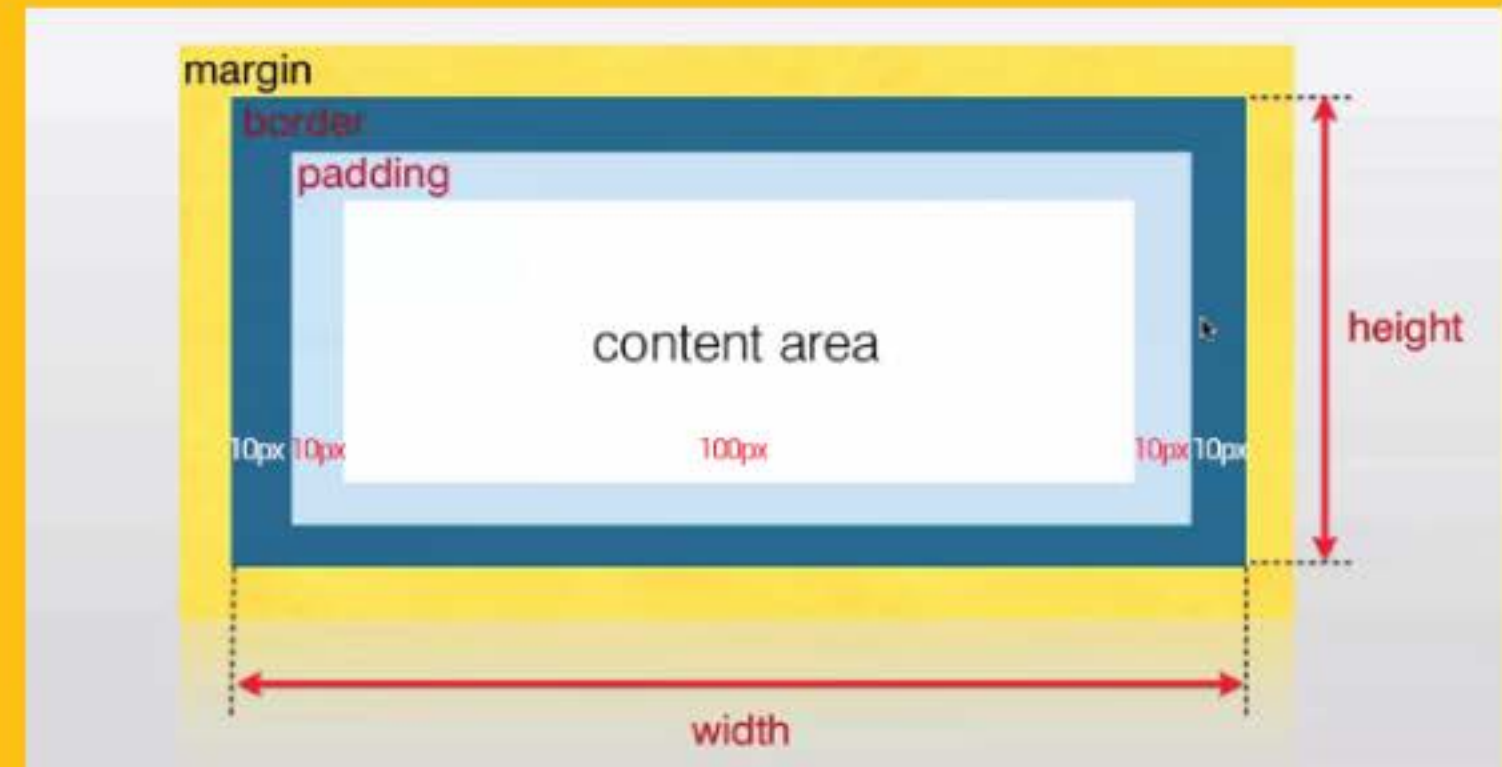
Margin lo que hace es agregar espacio entre el área de contenido y otras áreas de contenidos.

CSS

Modelos de caja



Supongamos que tenemos un área de contenido de 100px Width.
Le agregamos 10px Padding
También 10px Border
Width?



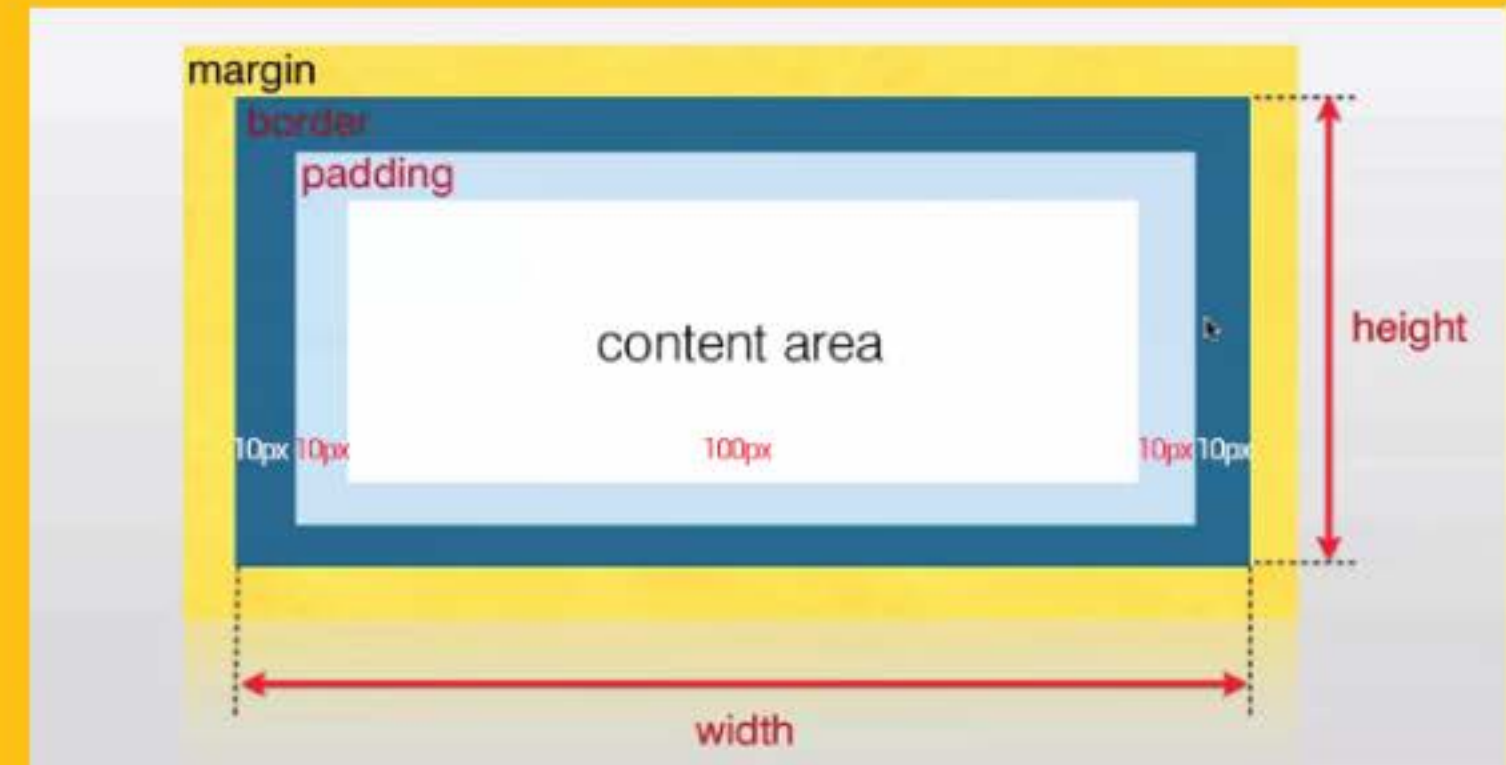
CSS

Modelos de caja



Supongamos que tenemos un área de contenido de 100px Width.
Le agregamos 10px Padding
También 10px Border

Width=140px



CSS

Modelos de caja



```
<!DOCTYPE html>
<html lang="es-ar">
<head>
  <title>Box</title>
  <link rel="stylesheet" href="normalize.css" />
  <link rel="stylesheet" href="box.css" />
</head>
<body>
  <div>
    <p>
      Primer párrafo
    </p>
  </div>
  <div>
    <p>
      Segundo párrafo
    </p>
  </div>
</body>
</html>
```

```
div,p{
  margin: 0;
  padding: 0;
}

html{
  background: white;
}

body{
  background: gray;
  font-family: Arial;
  font-size: 100%;
  margin: 0 auto;
  width: 500px;
}
```

CSS

Modelos de caja



Aplicamos estilo al div
Fijamos un width y height

```
div,p{  
  margin: 0;  
  padding: 0;  
}  
  
html{  
  background: white;  
}  
  
body{  
  background: gray;  
  font-family: Arial;  
  font-size: 100%;  
  margin: 0 auto;  
  width: 500px;  
}  
  
div{  
  background-color: yellow;  
  width: 450px;  
  height: 125px;  
}  
|
```


CSS

Modelos de caja



Agregamos padding.
Cambiamos p para ver el resultado.

```
html{
  background: white;
}

body{
  background: gray;
  font-family: Arial;
  font-size: 100%;
  margin: 0 auto;
  width: 500px;
}

div{
  background-color: yellow;
  width: 450px;
  height: 125px;
  padding: 10px;
}

p{
  color: white;
  background-color: black;
}
```

CSS

Modelos de caja



A esta altura nuestro div tiene 470px

450px width + 10px padding

```
html{  
  background: white;  
}  
  
body{  
  background: gray;  
  font-family: Arial;  
  font-size: 100%;  
  margin: 0 auto;  
  width: 500px;  
}  
  
div{  
  background-color: yellow;  
  width: 450px;  
  height: 125px;  
  padding: 10px;  
}  
  
p{  
  color: white;  
  background-color: black;  
}
```

CSS

Modelos de caja



Probemos de eliminar el height y width para ver qué pasa.

```
6  html{
7    background: white;
8  }
9
10 body{
11   background: gray;
12   font-family: Arial;
13   font-size: 100%;
14   margin: 0 auto;
15   width: 500px;
16 }
17
18 div{
19   background-color: yellow;
20   padding: 10px;
21 }
22
23 p{
24   color: white;
25   background-color: black;
26 }
```


CSS

Modelos de caja



Las 2 cajas ocupan todo el ancho y alto del body. Como "bloques" apilados uno por sobre el otro.

Pero el div no mide más 470px sino que al tomar todo el ancho, pero restando el padding, nos queda de 480px.

CSS

Modelos de caja



Agregamos un border de 10px blanco y sólido.

Nuestro div se "achica" porque además del padding le sumamos el border, en total ahora el ancho del div es de 460px.

```
3 padding: 0;  
4 }  
5  
6 html{  
7     background: white;  
8 }  
9  
10 body{  
11     background: gray;  
12     font-family: Arial;  
13     font-size: 100%;  
14     margin: 0 auto;  
15     width: 500px;  
16 }  
17  
18 div{  
19     border: 10px solid white;  
20     background-color: yellow;  
21     padding: 10px;  
22 }  
23  
24 p{  
25     color: white;  
26     background-color: black;  
27 }
```

CSS

Modelos de caja



A nuestro p ahora le agregamos un border de 5px solido y blanco.
Lo cual, nos reduce aún más nuestro div.

```
5
6  html{
7    background: white;
8  }
9
10 body{
11   background: gray;
12   font-family: Arial;
13   font-size: 100%;
14   margin: 0 auto;
15   width: 500px;
16 }
17
18 div{
19   border: 10px solid white;
20   background-color: yellow;
21   padding: 10px;
22 }
23
24 p{
25   border: 5px solid white;
26   color: white;
27   background-color: black;
28 }
```


CSS

Modelos de caja



Aplicamos margin de 10px a nuestro div. Y en nuestro ejemplo vemos que se agregó margin a los costados, pero en la parte superior no lo sigue al body y entre los p sólo hay 10px de separación.

(El problema de visualización de la parte superior, inferior y media, lo veremos más adelante, se llama vertical collapse)

```
1  div,p{
2      margin: 0;
3      padding: 0;
4  }
5
6  html{
7      background: white;
8  }
9
10 body{
11     background: gray;
12     font-family: Arial;
13     font-size: 100%;
14     margin: 0 auto;
15     width: 500px;
16 }
17
18 div{
19     border: 10px solid white;
20     background-color: yellow;
21     margin: 10px;
22     padding: 10px;
23 }
24
25 p{
26     border: 5px solid white;
```

CSS

Modelos de caja



Aplicamos margin de 10px a nuestro div. Y en nuestro ejemplo vemos que se agregó margin a los costados, pero en la parte superior no lo sigue al body y entre los p sólo hay 10px de separación.

(El problema de visualización de la parte superior, inferior y media, lo veremos más adelante, se llama vertical collapse)

Fljemos un ancho para ver qué pasa.

```
3     padding: 0;
4   }
5
6   html{
7     background: white;
8   }
9
10  body{
11    background: gray;
12    font-family: Arial;
13    font-size: 100%;
14    margin: 0 auto;
15    width: 500px;
16  }
17
18  div{
19    border: 10px solid white;
20    background-color: yellow;
21    margin: 10px;
22    padding: 10px;
23    width: 450px;
24  }
25
26  p{
27    border: 5px solid white;
28    color: white;
```


CSS

Modelos de caja



Necesitamos la propiedad que “esconda” lo que se excede y lo incluya en el div.

overflow hidden

(Necesitamos ponerlo en el html también además del body)

```
1  div,p{
2      margin: 0;
3      padding: 0;
4  }
5
6  html{
7      background: white;
8      overflow: hidden;
9  }
10
11 body{
12     background: gray;
13     font-family: Arial;
14     font-size: 100%;
15     margin: 0 auto;
16     overflow: hidden;
17     width: 500px;
18 }
19
20 div{
21     border: 10px solid white;
22     background-color: yellow;
23     margin: 10px;
24     padding: 10px;
25     width: 450px;
```