

Programador Web Inicial Front- End Developer

CSS y maquetado avanzado

CSS (parte 2)

Presentación

En esta unidad vemos las propiedades de CSS más avanzadas y que nos ayudarán en la maquetación del sitio.

Objetivos

Que los participantes logren...

- Conocer más propiedades de CSS.
- Entender y utilizar los distintos posicionamientos de los elementos.
- Conocer las propiedades para distribuir el espacio los elementos .

Bloques temáticos

1. Pseudo-clases /Pseudo-elementos.
2. Posicionamiento.
3. Flexbox.

1. Pseudo-clases /Pseudo-elementos

Pseudo-clases

Una **pseudoclase** CSS es una palabra clave que se añade a los selectores y que especifica un estado especial del elemento seleccionado. Por ejemplo, `:hover` aplicará un estilo cuando el usuario ponga el mouse sobre el elemento especificado por el selector.

Estructura

`selector:pseudoclase { propiedad: valor; }`

`:first-child` selecciona el primer elemento entre un grupo de elementos hermanos.

`:last-child` selecciona el último elemento entre un grupo de elementos hermanos.

`:nth-child()` selecciona elementos basándose en la posición de los hijos. Puede utilizar números expresiones y las palabras even y odd (par e impar respectivamente).

```
1 .persona:first-child{
2     margin-left: 0;
3 }
4
5 .servicio:nth-child(even) .info{
6     float: left;
7 }
8
9 .servicio:nth-child(even) figure{
10     float: right;
11     margin: 0 0 0 1em;
12 }
13
14 .servicio:nth-child(odd) figure{
15     float: left;
16     margin: 0 1em 0 0;
17 }
18
19 .servicio:nth-child(odd) .info{
20     float: right;
21 }
22
23 .galeria .foto:nth-child(5n+1){
24     border-color: red;
25 }
```

Pseudo-elementos

Al igual que las pseudo-classes, los **pseudo-elementos** se añaden a los selectores, pero en cambio, no describen un estado especial sino que, permiten añadir estilos a una parte concreta del documento. Por ejemplo, el pseudoelemento `::first-line` selecciona solo la primera línea del elemento especificado por el selector.

Estructura

```
selector::pseudo-elemento { propiedad: valor; }
```

::after crea un pseudo-elemento que es el **último** hijo del elemento seleccionado. Es comunmente usado para **añadir** contenido cosmético a un elemento con la propiedad content. Es en línea (inline) de forma predeterminada.

::before crea un pseudo-elemento que es el **primer** hijo del elemento seleccionado. Es usado normalmente para añadir contenido estético a un elemento, usando la propiedad content. Este elemento se muestra en línea con el texto de forma predeterminada.

::first-line aplica estilos a la **primera línea** de un elemento de bloque. Nótese que la longitud de la primera línea depende de muchos factores, incluyendo el ancho del elemento, el ancho del documento y el tamaño de fuente del texto.

::placeholder representa el **texto provisional** en un elemento <input> o un elemento <textarea>.



```
/* Añade una flecha después de los enlaces */  
a::after {  
    content: "→";  
}  
  
/* Añade un corazón antes de los enlaces */  
a::before {  
    content: "♥";  
}  
  
/* Selecciona la primera línea de un <p> */  
p::first-line {  
    color: red;  
}  
  
::placeholder {  
    color: blue;  
    font-size: 1.5em;  
}
```

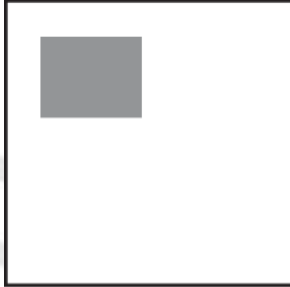
2. Posicionamiento

Los navegadores crean y posicionan de forma automática todas las cajas que forman cada página HTML. No obstante, CSS permite al diseñador modificar la posición en la que se muestra cada caja.

Utilizando las propiedades que proporciona CSS para alterar la posición de las cajas es posible realizar efectos muy avanzados y diseñar estructuras de páginas que de otra forma no serían posibles.

El estándar de CSS define cinco modelos diferentes para posicionar una caja:

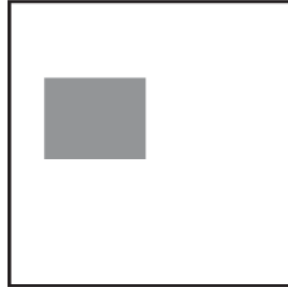
- **Posicionamiento normal o estático (static):** se trata del posicionamiento que utilizan los navegadores si no se indica lo contrario.
- **Posicionamiento relativo (relative):** variante del posicionamiento normal que consiste en posicionar una caja según el posicionamiento normal y después desplazarla respecto de su posición original.
- **Posicionamiento absoluto (absolute):** la posición de una caja se establece de forma absoluta respecto de su elemento contenedor y el resto de elementos de la página ignoran la nueva posición del elemento.
- **Posicionamiento fijo (fixed):** variante del posicionamiento absoluto que convierte una caja en un elemento inamovible, de forma que su posición en la pantalla siempre es la misma independientemente del resto de elementos e independientemente de si el usuario sube o baja la página en la ventana del navegador.



position: static;



position: absolute;
left: 200px



position: relative;
top: 20px



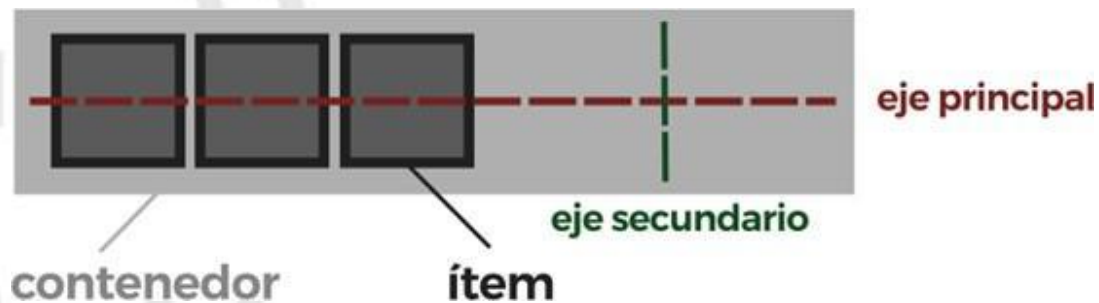
position: fixed;
top: 0;
left: 0;

3. Flexbox

Las cajas flexibles, comúnmente llamadas flexbox, fueron diseñadas como un modelo unidimensional de layout, y como un método que pueda ayudar a distribuir el espacio entre los ítems de una interfaz y mejorar las capacidades de alineación.

Conceptos

Para empezar a utilizar flexbox lo primero que debemos hacer es conocer algunos de los elementos básicos de este nuevo esquema, que son los siguientes:



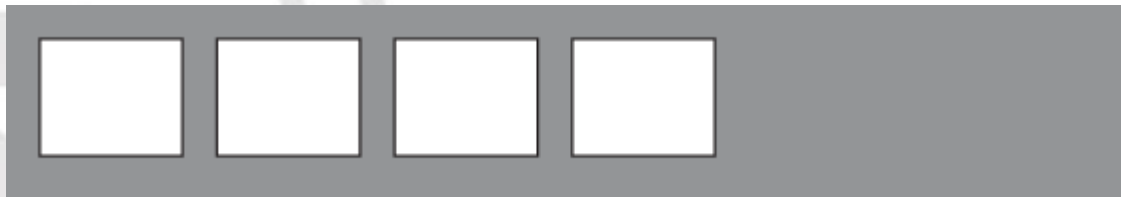
- **contenedor:** Existe un elemento padre que es el contenedor que tendrá en su interior cada uno de los ítems flexibles y adaptables.
- **ítem:** Cada uno de los hijos flexibles tendrá el contenedor en su interior.
- **eje principal:** Los contenedores flexibles tendrán una orientación principal específica. Por defecto, es en horizontal (fila).
- **eje secundario:** De la misma forma, los contenedores flexibles tendrán una orientación secundaria, perpendicular a la principal. Si la principal es en horizontal, la secundaria será en vertical, y viceversa.

Propiedad display

Controla la forma en la que se muestra el contenedor flexbox.

Valor	Descripción
flex	Establece un contenedor de ítems flexible en bloque.
inline-flex	Establece un contenedor de ítems flexible en línea (inline-block).

flex



inline-flex



Propiedad flex-direction

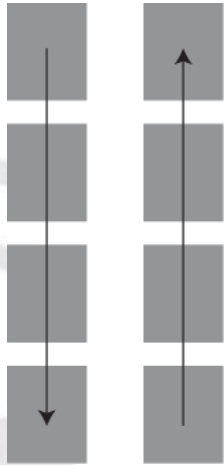
Define el eje principal y su dirección

Valor	Descripción
row	Establece la dirección del eje principal en horizontal.
row-reverse	Establece la dirección del eje principal en horizontal (invertido).
column	Establece la dirección del eje principal en vertical.
column-reverse	Establece la dirección del eje principal en vertical (invertido).

row/row-reverse



column / column-reverse

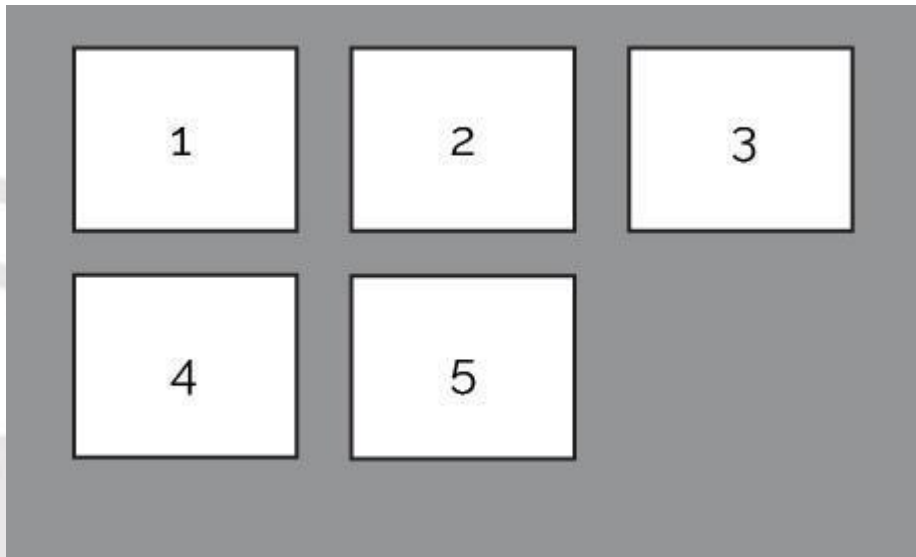


Propiedad flex-wrap

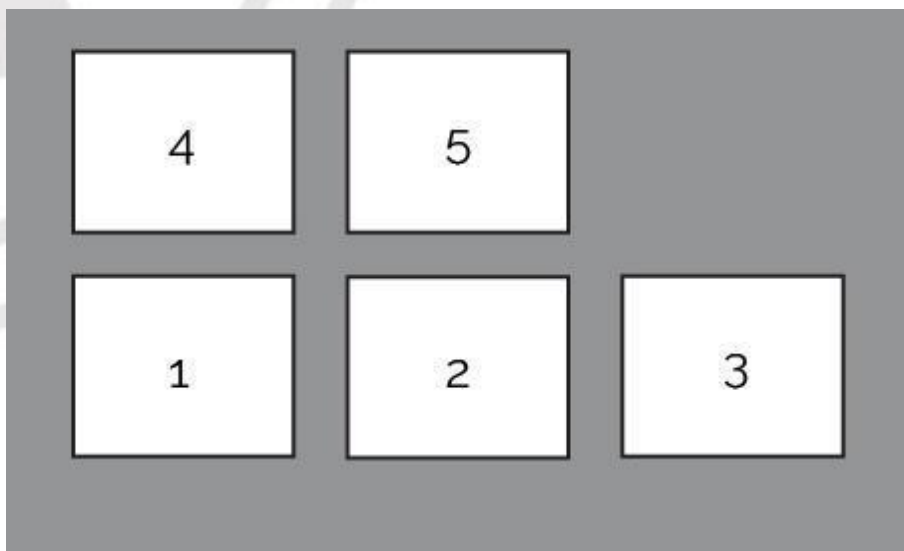
Controla el desbordamiento del contenedor cuando los ítems superan el ancho del mismo.

Valor	Descripción
nowrap	Establece los ítems en una sola línea (no permite desborde del contenedor).
wrap	Establece los ítems en modo multilínea (permite que se desborde el contenedor).
wrap-reverse	Establece los ítems en modo multilínea, pero en dirección inversa.

wrap



wrap-reverse



Propiedades de alineación de ítems

La propiedad **justify-content**, alinea los elementos en el eje principal

La propiedad **align-content**, alinea los elementos en el eje secundario cuando el contenedor es multilínea.

La propiedad **align-items**, alinea los elementos en el eje secundario cuando el contenedor tiene una línea única.

Estas son algunas de la características más usadas:

Valor	Descripción
flex-start	Los elementos se distribuyen desde el inicio de la línea.
flex-end	Los elementos se distribuyen al final de la línea.
center	Los elementos se centran a lo largo de la línea.
space-between	Los elementos se distribuyen uniformemente en la línea; el primer elemento está en la línea de inicio, el último elemento en la línea final.
space-around	Los elementos se distribuyen uniformemente en la línea con igual espacio a su alrededor. Tenga en cuenta que visualmente los espacios no son iguales, ya que todos los elementos tienen el mismo espacio en ambos lados.
space-evenly	Los elementos se distribuyen de manera que el espacio entre dos elementos (y el espacio hasta los bordes) sea igual.

flex-start



flex-end



center



space-between



space-around



space-evenly



Bibliografía utilizada y sugerida

Libros y otros manuscritos:

Castillo, Lucía. Iniciación al diseño Web. Editorial Dinámica. 2013.

Collell Puig, Jordi. CSS3-y-Javascript-Avanzado. Universidad Oberta de Catalunya.

Gauchat, Juan Diego. El gran libro de HTML5, CSS3 y Javascript. 1era Edición. 2012

Artículos de revista en formato electrónico:

Guia para utilizar Flexbox. Disponible desde la URL: ·
<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

Mozilla MDN Web Docs. Disponible desde la URL:
<https://developer.mozilla.org/es/>