

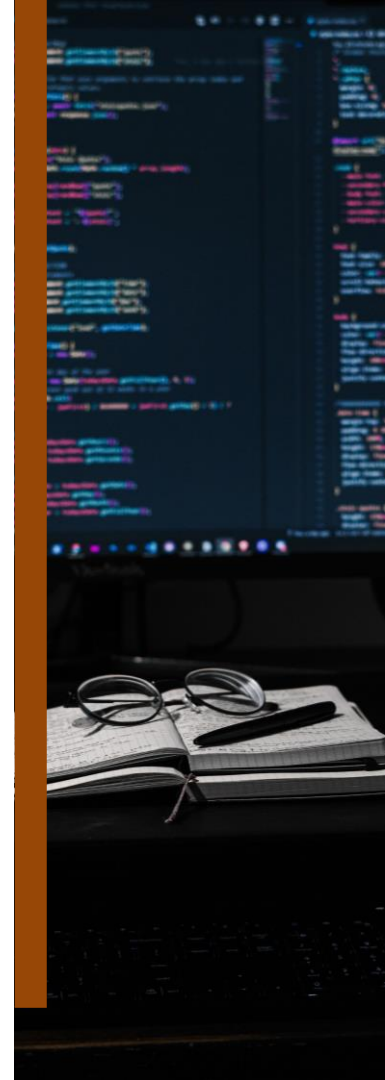
Tema 2

Diseño web con CSS

Programación web

Mada Cantabella Sabater

Ingeniería Informática



ÍNDICE

CONTENIDOS

1. Cajas
2. Maquetación de las cajas
3. Responsive Web Design
4. Medias Queries
5. Maquetación flexible

1. Cajas:

- ✓ Importante el concepto CAJA o CONTENEDOR.
- ✓ Espacio rectangular formado por:
 - Contenido
 - Margen interior (padding)
 - Borde (border)
 - Margen exterior (margin)
- ✓ La dimensión total del elemento caja se calcula del siguiente modo:

contenido + márgenes interiores + bordes + márgenes exteriores

1. Cajas

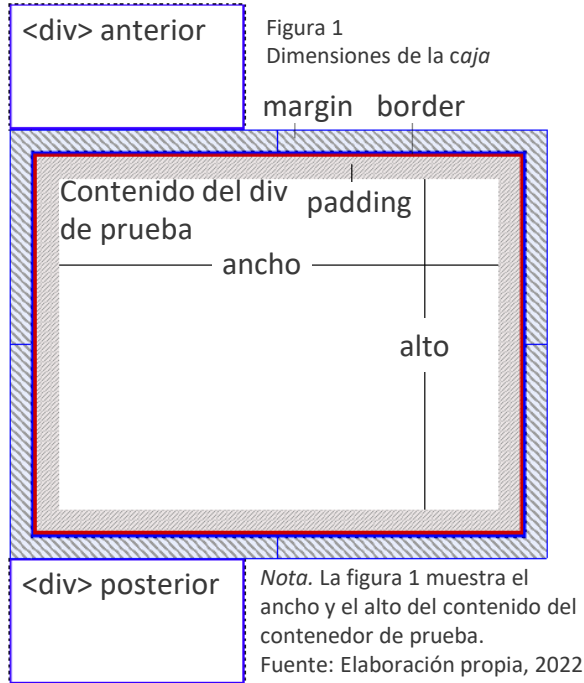
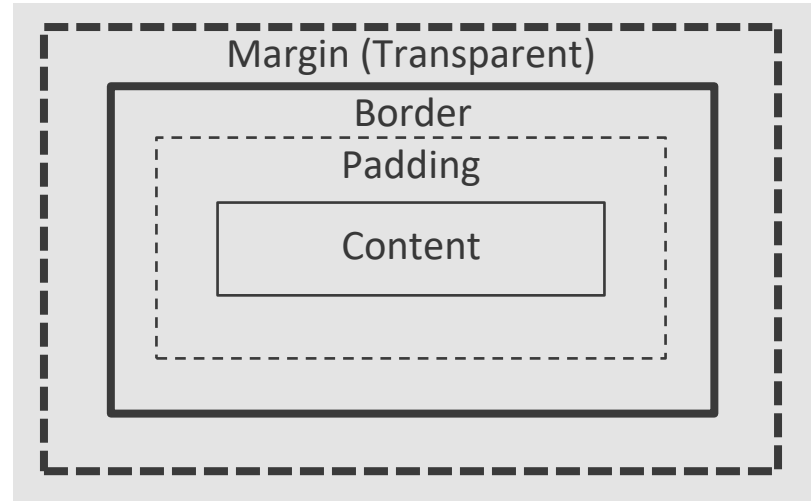


Figura 2
Contenido del contenedor

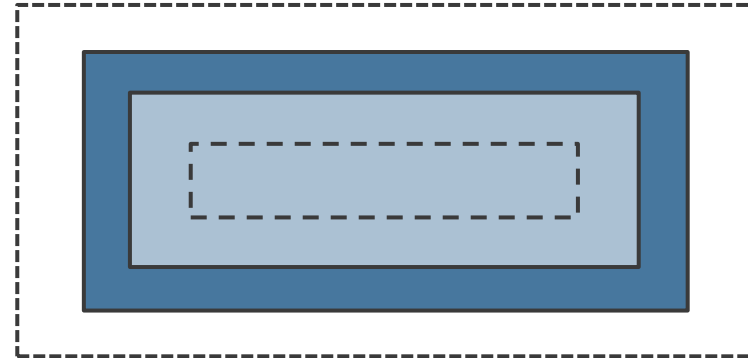


Nota. La figura 2 muestra que la caja cuenta con margen, borde, margen interior y margen exterior.
Fuente: Elaboración propia, 2022.

1. Cajas

Ejemplo

- ✓ Definimos una caja con las siguientes medidas:
- ✓ Espacio rectangular formado por:
 - Ancho 200px
 - Margen interior (padding) 20px
 - Borde (border) con anchura 10px
 - Margen exterior (margin) 20px



1. Cajas

Ejemplo

El total es son 300 píxeles:

La anchura del elemento caja es, de 300 píxeles, que se descomponen (de izquierda a derecha): 20 píxeles de margen exterior + 10 píxeles para el borde + 20 píxeles de margen interior + 200 píxeles de contenido + 20 píxeles de margen interior + 10 píxeles de borde + 20 píxeles de margen exterior.

```
<style type="text/css">
p { width: 200px;
    border: 10px solid rgb(125,165,205);
    padding: 20px;
    margin: 20px;
    background-color: rgb(215,230,245);
}</style>
```

1. Cajas: display

La propiedad de estilo **DISPLAY** permite redefinir un elemento en línea como un elemento en bloque, y a la inversa.

- En bloque (block)
- En línea (inline)

```

<!DOCTYPE html>
<html>
<head>
<style>
  p {display: inline;}
</style>
</head>
<body>

  <p>This is a paragraph.</p>
  <p>This is a paragraph.</p>
  <p>This is a paragraph.</p>
  <p>This is a paragraph.</p>
  <p>This is a paragraph.</p>

</body>
</html>

```

This is a paragraph. This is a paragraph. This is a paragraph. This is a paragraph. This is a paragraph.

<style>

p {display: block;} This is a paragraph.

</style> This is a paragraph.

This is a paragraph.

This is a paragraph.

This is a paragraph.

1. Cajas: width y height

- ✓ Fijan respectivamente la anchura y la altura del elemento.
- ✓ Valores width, height (auto, valor de longitud, %).
- ✓ Los valores especifican la anchura y la altura del contenido (sin márgenes interiores, sin bordes y sin márgenes exteriores).
 - El valor auto ajusta automáticamente la dimensión del contenido.
 - El valor de longitud especifica un valor fijo.
 - El porcentaje especifica un valor relativo.

1. Cajas: width y height


En CCS2

Las propiedades max-height, min-height, max-width y min-width no incluyen los márgenes interiores, los bordes ni los márgenes exteriores.

Max-height:	Fija la altura máxima
Min-height:	Fija la altura mínima
Max-width:	Fija la anchura máxima
Min-width:	Fija la anchura mínima

1. Cajas: width y height

En CCS3

- Para calcular las dimensiones de las cajas se usa la propiedad [box-sizing](#). 
- Se puede indicar si los valores de border y de padding deben incluirse en el cálculo de la anchura y la altura de las cajas.
 - › box-sizing: **content-box** indica que el cálculo de la anchura y de la altura de la visualización de la caja se realiza con los valores de las propiedades width, height, padding y border.
 - › box-sizing: **border-box** indica que el cálculo de la anchura y la altura de la visualización de la caja se realiza únicamente con los valores de las propiedades width y height.
 - Las dimensiones del elemento no tienen en cuenta la anchura del borde (border) ni el espaciado (padding).

1. Cajas: márgenes externos y espaciado interno

<code>margin:</code> <code>margin-top:</code> <code>margin-right:</code> <code>margin-bottom:</code> <code>margin-left:</code>	<code>padding:</code> <code>padding-top:</code> <code>padding-right:</code> <code>padding-bottom:</code> <code>padding-left:</code>	<code>auto</code> O bien valor de longitud O bien porcentaje
--	---	--

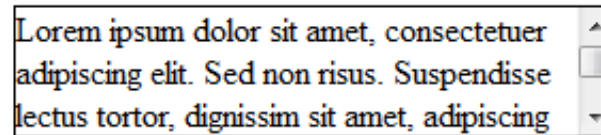
- ✓ El **valor auto** deja al navegador la gestión de los márgenes exteriores.
- ✓ Un **valor de longitud** define con precisión los márgenes exteriores.
- ✓ Un **porcentaje** define la longitud respecto a un elemento padre.

Las direcciones top, right, bottom, left diseñan, respectivamente, los bordes superior, derecho, inferior e izquierdo.

1. Cajas: overflow

Determina lo que debe hacer el navegador cuando un elemento es más grande que el elemento padre que lo contiene.

- **Hidden**, la parte que desborda se oculta, sin posibilidad de verla ni acceder a ella.
- **Scroll**, la parte que desborda se oculta, aunque queda accesible mediante barras de desplazamiento.
- **Visible**, la parte que desborda se muestra, ignorando las especificaciones del elemento padre que lo contiene.
- **Auto**, se deja la gestión al navegador.



2. Maquetación de las cajas

Posicionamiento de las cajas

Estático:

- Es el posicionamiento normal del elemento según la forma habitual de proceder del navegador.
- El diseñador no tiene el control.

Relativo:

- Es el posicionamiento de un elemento respecto a su posición normal o estática.
- El elemento permanece en el flujo de los datos aunque está descentrado respecto a su posición normal.

2. Maquetación de las cajas

Posicionamiento de las cajas

Absoluto:

- El posicionamiento absoluto crea un elemento independiente del resto del documento. Los elementos definidos con posición absoluta se retiran del flujo normal y se posicionan en el lugar exacto definido por el diseñador.
- NO RESPONSIVE

Fixed:

- Fijar un elemento en una posición respecto al origen de coordenadas del primer elemento contenedor posicionado o respecto a la esquina superior de la ventana de visualización.

› position:fixed;

position: static;	position: relative;	left: valor o %; top: valor o %; right: valor o %; bottom: valor o %;	position: absolute;	left: valor o %; top: valor o %; right: valor o %; bottom: valor o %;
----------------------	---------------------	--	---------------------	--

2. Maquetación de las cajas

Flotación de las cajas

Float (right, left, none):

- **Right** alinea a la derecha el elemento indicado, empujando a los demás elementos hacia la izquierda.
- **Left** alinea a la izquierda el elemento indicado, empujando a los demás elementos hacia la derecha.
- **None** no especifica nada y deja la gestión al navegador.

La posición float no puede aplicarse en caso de posicionamiento absoluto.

2. Maquetación de las cajas

Flotación de las cajas

Prohibir la flotación:

Clear

- El valor **right** anula los elementos flotantes a la derecha.
- El valor **left** anula los elementos flotantes a la izquierda.
- El valor **both** anula los elementos flotantes de ambos lados.
- El valor **none** anula los elementos flotantes.

3. Responsive Web Desing

Tamaño de las pantallas:

Los diseñadores de los sitios deben producir sitios web que sean perfectamente visibles y legibles, sea cual sea el soporte de consulta: pantalla de ordenador, tableta o smartphone.

En <head>, podemos utilizar el elemento **<meta>** con el atributo **viewport**. 

Para el atributo **content**, podremos utilizar el valor **device-width**, que se corresponde con la anchura física de la pantalla.

```
<meta name="viewport" content="width=device-width" />
```

4. Medias Queries:

Media query consiste en un tipo de medio que limita las hojas de estilo utilizando características del medio como ancho, alto y color.

```
<!-- CSS media query on a link element -->
<link rel="stylesheet" media="(max-width: 800px)" href="example.css" />

<!-- CSS media query within a style sheet -->
<style>
@media (max-width: 600px) {
  .facet_sidebar {
    display: none;
  }
}
</style>
```

W3C



MDN Web Docs



Rolandocaldas



Smartphone



320 x 480

Tablet



768 x 1024



1024 x 768

Laptop



1280 x 800

Desktop



1680 x 1050

4. Medias Queries:

Media query consiste en un tipo de medio que limita las hojas de estilo utilizando características del medio como ancho, alto y color.

Añadido en CSS3, dejan que la presentación del contenido se adapte a un rango específico de dispositivos de salida sin tener que cambiar el contenido en sí.

```
<!-- CSS media query on a link element -->
<link rel="stylesheet" media="(max-width: 800px)" href="example.css" />

<!-- CSS media query within a style sheet -->
<style>
@media (max-width: 600px) {
  .facet_sidebar {
    display: none;
  }
}
</style>
```

4. Medias Queries:

W3C

MDN Web Docs

Rolandocaldas

Medio	Descripción
all	Todos los medios definidos
braille	Dispositivos táctiles que emplean el sistema braille
embosed	Impresoras braille
handheld	Dispositivos de mano: móviles, PDA, etc.
print	Impresoras y navegadores en el modo <i>"Vista Previa para Imprimir"</i>
projection	Proyectores y dispositivos para presentaciones
screen	Pantallas de ordenador
speech	Sintetizadores para navegadores de voz utilizados por personas discapacitadas
tty	Dispositivos textuales limitados como teletipos y terminales de texto
tv	Televisores y dispositivos con resolución baja

5. Maquetación Flexible

Propiedad flex

- ✓ Permitir crear maquetaciones con columnas de manera sencilla.
- ✓ Se define un contenedor padre como caja flexible para que todos sus elementos hijos adopten su comportamiento.

Al definir las propiedades de la caja damos los valores en el archivo .css:

- ✓ Maquetación de cajas:
 - `display:flex`, activa la propiedad flex
 - `Flex-wrap: wrap`, ajusta el contenido al nuestra página
- ✓ Maquetación de menús:
 - `display:flex`
 - `flex-direction: column / row`

5. Maquetación Flexible

Diseño de caja flexible CSS



CSS TRICKS - A Complete Guide to Flexbox



CSS flex Property



Learn CSS



5. Maquetación Flexible

Propiedad Grid

- ✓ Cuadrículas bidimensionales: crear maquetaciones con columnas y filas.
 - ✓ Una cuadrícula es un conjunto de líneas horizontales y verticales que se cruzan y definen columnas y filas.
- ✓ **Al definir las propiedades de la caja damos los valores en el archivo .css:**
- ✓ Contenedor rejilla:
 - `display: grid`
 - `display: inline-grid`

Una vez activado todos los elementos secundarios directos de ese elemento se convertirán en elementos de cuadrícula.

5. Maquetación Flexible Grid

CSS Grid Property



CSS Grid Property2



3/4

GRID CSS

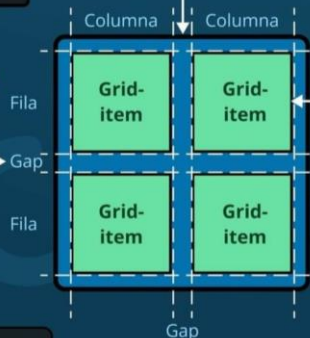
GRID CONTAINER

Es el contenedor o elemento "padre", **contiene a los Grid Items**.

Usa **gap** para crear separaciones entre elementos **(se hace desde el padre)**.

Grid no reemplaza a **FlexBox** y **Flexbox** no reemplaza a **Grid**.

Alexis Mora Angulo
UX Designer



GRID ITEMS

Son los "hijos" directos de un Grid Container y **quienes componen la cuadrícula**.

Para diseños bidimensionales, **controla filas y columnas a la vez**.

- Para diseños de **página completa**.
- Sistemas de columnas**.
- Galería de imágenes**.



FLEXBOX

FLEX CONTAINER

Es el elemento "padre", **contiene los Flex Items**.



PARA COMPONENTES PEQUEÑOS

Encabezados



Cards



Menús de navegación

FLEX ITEMS

Son todos los **elementos directos del Flex Container**, llamados elementos "hijos". **Los flex-items son flexibles**.

Usa **márgenes** para crear separaciones entre elementos **(se hace desde los hijos)**.

Para diseños unidimensionales, **controla solo filas o columnas**.



Mada Cantabella Sabater

mmcantabella@ucam.edu

UCAM Universidad Católica de Murcia

