



CSS

Definición

CSS ...es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario escritas en HTML...



Añadir estilos

- In-line: No recomendable, si hay algún cambio deberías cambiar todos los h1 del fichero.

```
<h1 style="color:red">Título 1</h1>
```

- <style> en la cabecera: No recomendable, si hay algún cambio deberías cambiar todas las páginas html.

```
<style> h3 {color:orange;}</style>
```

- Directiva @import: hay un inconveniente. El rendimiento, ya que es bloqueante en algunos navegadores, y se para la descarga del resto de la página hasta que no se ha descargado la hoja de estilos.

```
<style>
```

```
@import 'css/estilo.css';
```

```
</style>
```

- Etiqueta link

```
<link rel="stylesheet" type="text/css" href="css/estilo.css"/>
```

Etiqueta Link

Un ejemplo de uso de esta etiqueta, que se pone **dentro de la cabecera <head>**, es:

```
<link href="css/estilos.css" rel="stylesheet" type="text/css" />
```

Tiene tres atributos en los que indicamos:

- href: ruta a la hoja de estilos.
- rel: tipo de contenido con el que estoy enlazando.
- type: tipo de fichero.



Selectores CSS

Los selectores CSS son **reglas o patrones** que nos van a permitir **seleccionar los distintos elementos** de mi página web para poder **modificar sus propiedades o estilos**.

Estos selectores es lo que vamos a incluir en nuestros archivos CSS y su sintaxis de manera general es la siguiente:

```
selector {  
    propiedad1:valor1;  
    propiedad2:valor2;  
    ...  
}
```

Selectores más importantes

***** es el selector universal. Selecciona todo los elementos.

#id selecciona el elemento que tenga ese valor (id en este ejemplo) en el atributo id.

.class selecciona los elementos que tengan ese valor (class en este ejemplo) en el atributo class.

etiqueta selecciona esas etiquetas concretas.

selector1,selector2 sirve para cambiar las propiedades de los elementos seleccionados por ambos selectores

selector1 selector2 sirve para cambiar las propiedades de los elementos seleccionados por selector2 que se encuentran dentro de aquellos seleccionados por selector1.

Algunos selectores más

selector1>selector2 sirve para cambiar las propiedades de los elementos seleccionados por selector2 que son hijos directos (en el árbol DOM) de aquellos que se seleccionen con selector1.

selector1+selector2 sirve para cambiar las propiedades de los elementos seleccionados por selector2 que están justo después de aquellos que se seleccionan mediante selector1.

selector1~selector2 (justo antes)

[atribut expre valor] siendo expre (=, ~, |, \$, *, ..) para seleccionar elementos atendiendo al valor de sus atributos.






Resumen de selectores

Hoja de estilos en “Cascada”

Ya desde la misma definición de CSS dijimos que la C hacía referencia a “en Cascada”. Pero exactamente, ¿qué significa eso?.

Básicamente dos cosas:

- a) Los estilos se van propagando hacia abajo o lo que es lo mismo **si especificamos una propiedades en un elemento padre los hijos tienen el mismo valor** para esas propiedades.
- b) Si hay más de una regla que se puede aplicar al mismo elemento y hay conflicto, entonces **se aplica la regla más específica**.

- 
- 
- 
- Color
 - Fondos
 - Dimensiones y Unidades
 - Márgenes, Bordes y Padding
 - Texto
- 
- 

Colores

Para establecer el color del texto de nuestra web lo podemos establecer usando la propiedad color. Por ejemplo:

```
selector { color: red (nombre en inglés)
```

```
o
```

```
color: #FFFFFF (eso es blanco)
```

```
o
```

```
color: rgb(255,255,255)(blanco)
```

```
}
```

FONDOS

```
selector {  
    background:  
    background-color:  
    background-image:  
    background-repeat:  
  
}
```



Dimensiones y unidades

Las dimensiones de los elementos de nuestra página se establecen usando las siguientes propiedades:

width para la anchura de nuestro elemento.

height para altura de nuestro elemento.

Y ambas podemos determinar usar varios tipos de unidades:

px: En píxeles

%: En relación a lo que ocupe el padre del elemento dentro del árbol DOM.

em: En relación al tamaño por defecto de la letra del navegador en ese instante (normalmente 16px..)

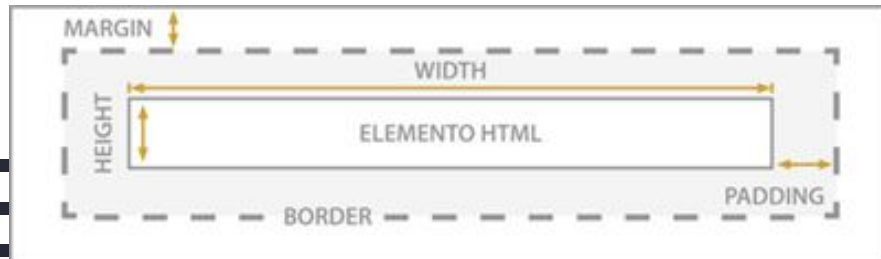
rem: En relación al tamaño por defecto de la letra que tiene la etiqueta HTML.

Márgenes, Bordes y Paddings

Son propiedades para establecer las dimensiones de los elementos de la caja.

Para márgenes y paddings tenemos varias formas de hacerlo. Si tenemos en cuenta que A(Arriba)- D(Derecha)- AB(Abajo)- IZQ(Izquierda):

Para el padding sería exactamente lo mismo. Sólo tenemos que sustituir margin por padding.



```
selector {  
  /* A -D -AB-IZQ */  
  margin: 20px 50px 20px 50px;  
  
  /* A y AB - DCHA e IZQDA */  
  margin: 20px 50px;  
  
  /* Todos */  
  margin: 50px;  
  
  /* O de manera individual */  
  margin-left: 10px;  
  margin-top: 10px;  
  margin-bottom: 10px;  
  margin-right: 10px;  
}
```

En relación al borde de un elemento tenemos también varias posibilidades:

```
selector {  
border: 1px solid black;  
border-color: black;  
border-width: 1px;  
border-style: solid;  
}
```

Estilo de borde CSS:

dotted - Define un borde punteado

dashed - Define un borde punteado

solid - Define un borde sólido

double - Define un borde doble

```
/* De manera general */  
border: 1px solid black;
```

```
/* Sólo el borde */  
border-color: black;
```

```
/* Sólo la anchura del borde */  
border-width: 1px;
```

```
/* Sólo el estilo de la línea del borde */  
/* posibles valores solid, dashed, dotted */  
border-style: solid;
```

Estilos para el texto

Hay multitud de propiedades para establecer los estilos del texto de mi página web. Algunas de las más destacables son:

```
/* Para establecer el tipo de fuente */  
font-family: "Times New Roman", Times, serif;
```

```
/* Para establecer el tamaño de la fuente */  
font-size: 2em;
```

```
/* Para establecer el grosor del tipo de letra */  
/* Posibles valores: bold, bolder, lighter */  
font-weight: bold;
```

```
/* Para establecer la alineación texto */  
/* Posibles valores: center, left, right, justify */  
text-align: center;
```

```
/* Para establecer la decoración de texto */  
/* Posibles valores: underline, overline, none, line-through */  
text-decoration: underline;
```

```
/* Para establecer tabulaciones */  
text-indent: 10px;
```

```
/* Para transformar un texto todo a mayúscula o minúsculas */  
/* Posibles valores: uppercase, lowercase, capitalize */  
text-transform: uppercase;
```



Problema 1

Consigue la siguiente Página Web.

Crea un documento HTML5 que represente un artículo con la información facilitada en la carpeta “problema5”. Debes incluir los siguientes elementos:

- a. Un elemento contenedor con el contenido del cuerpo de la web.
- c. Párrafos, títulos y subtítulos.
- d. Una imagen.
- e. Un enlace externo a <https://www.educantabria.es> que se abra en una pestaña nueva.



Crea un documento CSS3, que enlazado con el ejercicio anterior aplique el siguiente estilo:

- a. El color de fondo de la Web es #31B404
- b. El elemento contenedor con el contenido del cuerpo: color de fondo en blanco y fuente para todos los textos Calibri. Tiene una anchura del 80%, padding de 2em y está centrada
- c. Títulos y subtítulos: color #31B404, con alineación centrada. Párrafos con alineación justificada
- d. Cabecera color blanco y borde doble de color #31B404
- e. La lista de navegación con fondo de color #31B404, con el texto centrado y blanco.
- f. Imagen centrada con borde color #F5B041, ocupando un ancho del 60%.



III Semana Europea de la FP

[Eventos](#)[Galería](#)[Educantabria](#)

La III Semana Europea de la Formación Profesional quiere acercar la FP a todos



La cita en el Palacio de Exposiciones de Santander

Un año más, la Consejería se ha sumado a esta iniciativa de la Comisión Europea con la que se persigue acercar la Formación Profesional de nuestra región a la sociedad, y así concienciar sobre la amplia gama de oportunidades que brinda, animando a más personas a que participen y descubran lo que puede ofrecerles. Bajo el lema 'Descubre tu talento', miles de jóvenes europeos han tenido la posibilidad durante esa semana en los distintos países de la UE, de informarse sobre las oportunidades que ofrece.

Más de 5.000 personas visitaron la III Semana Europea de la Formación Profesional, que se celebró del 14 al 16 de noviembre en el Palacio de Exposiciones de Santander.

El objetivo de "salir de los centros" para llevar la FP a la ciudadanía y "prestigiar" esta formación con una tasa de ocupación superior a la general se cumplió con creces, superando cualquier expectativa.

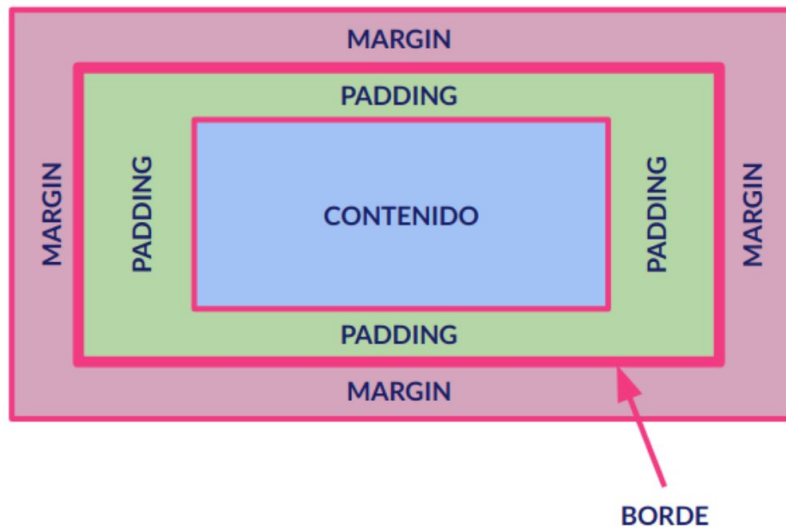


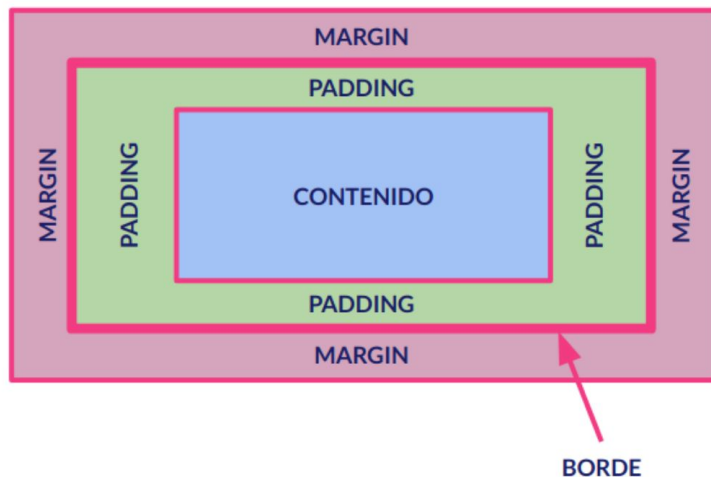
Maquetación web con CSS

Modelo de cajas

Cada uno de los elementos de mi página web son cajas:

Todos los elementos que representan algo en HTML tiene la siguiente estructura visual:





- El contenido que es lo “importante”, el texto en las etiquetas de texto, la imagen en las etiquetas de imagen etc..
- El padding que es la distancia que existe entre el contenido y el borde de la caja.
- El borde que es el elemento que marca la división entre el elemento y el resto de los elementos de la página.
- El margen que es la distancia entre el elemento y el resto de los elementos de la página.



La siguiente estructura html está compuesta:

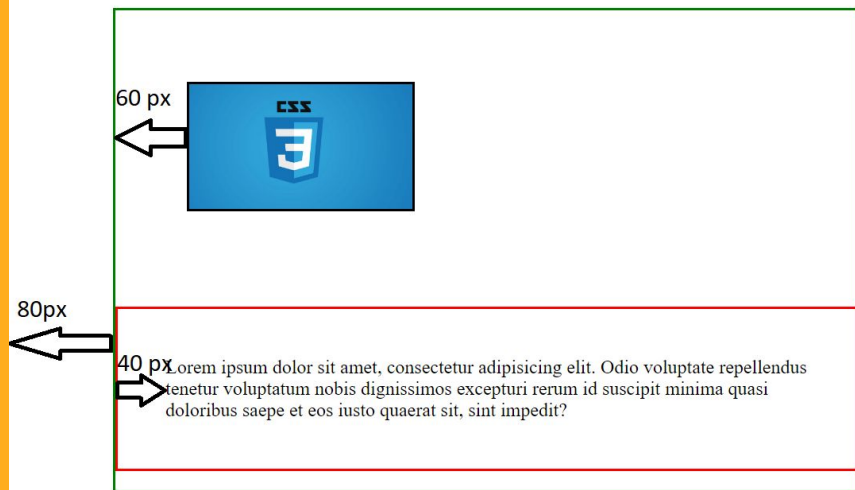
```
<div>
```

```
  <img/>
```

```
  <p></p>
```

```
</div>
```

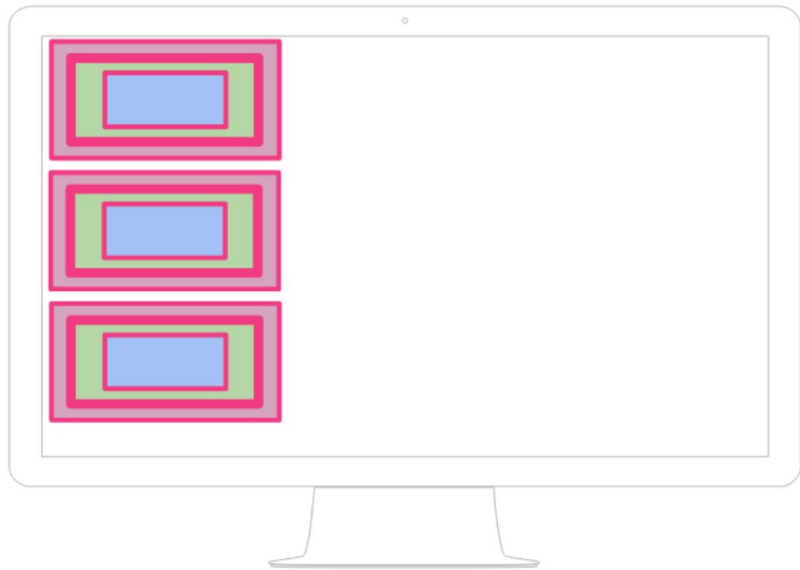
Según el modelo de cajas que margin, padding y border tienen definidos en CSS:



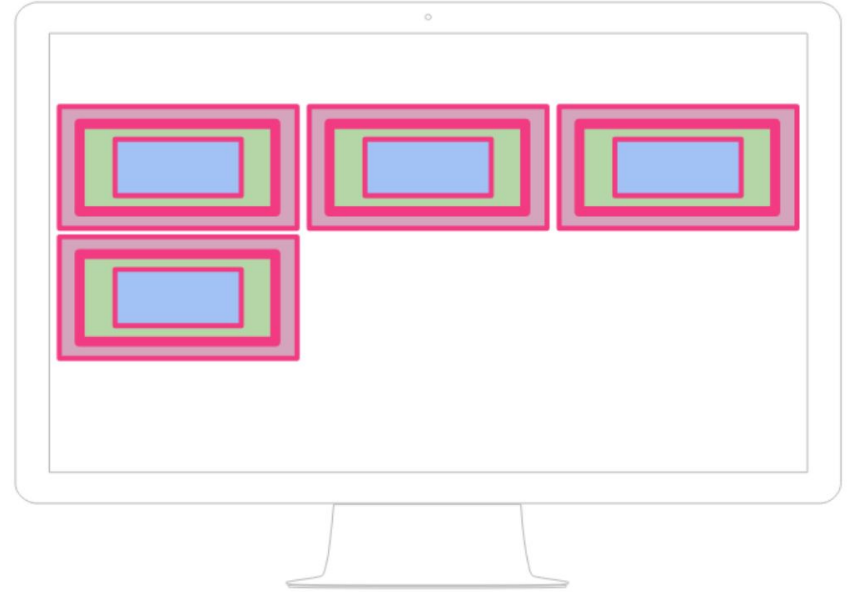
atum nobis dignissimos excepturi rerum id suscipit minima quasi doloribus

Comportamiento de las cajas

Los elementos en **bloque** son elementos que, independientemente de la anchura que tengan, se separan verticalmente de los elementos anteriores y posteriores. Es como si “provocaran” un salto de línea antes y uno después:



En cambio, los **elemento en línea** se van sucediendo a lo largo de la misma línea, mientras caben, uno detrás de otro y de izquierda a derecha (al menos en nuestro idioma). Cuando no caben pasan a la línea siguiente.



Elementos en bloque y en línea

Bloque:

`<address>` `<blockquote>` `<hr>`
`<article>` `<aside>` `<footer>` `<div>` `<main>` `<nav>` `<section>`
`<header>` `<h1>-<h6>` `<p>`
`<dd>` `<dl>` `<dt>` `` `` ``
`<table>`
`<form>` `<fieldset>`
`<figcaption>` `<figure>`

Linea:

`<a>` `` `
` `<cite>` `<i>` `strong` `<sub>` `<sup>`
`` `<cite>` ``
`<button>` `<input>` `<label>` `<textarea>`





MAQUETACIÓN WEB



DEFINICIÓN

La maquetación web es la parte que se encarga de **definir la disposición de los distintos elementos** (imágenes, texto etc...) que forman nuestra página web para definir la **estructura visual** de la misma de acuerdo a unos objetivos de comunicación definidos.



Que utilizamos:

ETIQUETAS HTML+CSS

div, header, footer, nav, section, article, aside,
dialog, main, summary...

Etiquetas semanticas: aportan significado a las partes del documento, pero estructuralmente se comportan igual que el div

Recomendaciones

Realiza siempre un esquema previo.

Diseña siempre “de lo grande a lo chico”.

Copia.

Prueba en distintos navegadores.



Flujo de los elementos en la ventana de nuestro navegador:

- Los navegadores no hacen NADA para controlar el diseño de nuestra página Web.
- Los navegadores, lo único que hacen es mostrar los elementos de nuestra página HTML en el mismo ORDEN en el que los hemos escrito.
- Siguen solo dos reglas básicas dependiendo de las propiedades de las cajas:
 - Elementos en línea: va poniendo uno detrás del otro
 - Elementos en bloque: introduce un "salto de línea".

Propiedad Display

Los elementos o fluyen o van a la siguiente línea, esto puede ser determinado por defecto o bien, establecido por la propiedad CSS Display.

VALORES DE DISPLAY MÁS USADOS:

- inline
- inline-block
- block
- None
- Relacionados con tablas

INLINE, INLINE-BLOCK

Inline: Los elementos Inline no rompen el flujo de la línea y se van colocando uno detrás de otro mientras caben. Aceptan margin y padding (horizontal). Ignoran width y height.

Etiquetas display inline por defecto: ``, `<a>`, ``, ``

Inline-Block: Como inline pero podemos asignarles width y height.

BLOCK

BLOCK: Los elementos en bloque rompen el flujo de la línea y provocan “una salto de línea” tanto anterior como posterior. Por defecto si no especificamos una anchura ocupan toda la del elemento que los contiene (la etiqueta padre)

Ejemplos de elementos en bloque por defecto:

`<h1>,<p>,<section>,<div>,,,<nav>...`



NONE

Una vez fijado el elemento desaparece. No deja espacio vacío aunque siga en el código HTML.



VALORES DE DISPLAY RELACIONADOS CON TABLAS

Si quiero emular el comportamiento de una tabla puedo dar los siguientes valores a la propiedad display:

- Table (se comporta como una tabla)
- Table-row (se comporta como un tr)
- Table-cell (se comporta como un td, th)
- Table-caption
- Table-colgroup



holaAdios



Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Lorem Ipsum
is simply
dummy text of
the printing
and typesetting
industry.
Lorem Ipsum
has been the
industry's
standard
dummy text
ever since the
1500s, when an
unknown
printer took a
galley of type
and scrambled
it to make a
type specimen
book.

Algo de texto

Lorem Ipsum
is simply
dummy text of
the printing
and typesetting
industry.
Lorem Ipsum
has been the
industry's
standard
dummy text
ever since the
1500s, when an
unknown
printer took a
galley of type
and scrambled
it to make a
type specimen
book.

Algo de texto



Texto

Texto Lorem Ipsum is simply dummy text of the printing and typesetting industry.

The image features a dark blue background with a large orange horizontal bar. On the left side of the orange bar, there are four dark blue horizontal bars. To the right of the orange bar, there is a grid of small orange squares. In the top right corner, there are four white horizontal bars. In the bottom right corner, there are four orange horizontal bars and a grid of small grey squares.

TIPOS DE LAYOUTS

TIPOS

- Fixed
- Elastic
- Fluid (%)
- Con Min/Max Sizing
- Responsive
- Mezclas de varios.



Fixed Layouts

Este tipo de layouts establecen un tamaño fijo en pixels para la anchura de los distintos elementos.

VENTAJA

- Siempre el mismo tamaño
- Total control

DESVENTAJA

- Pantallas pequeñas -> Scroll Horizontal
- Pantalla grandes -> Mucho espacio en blanco a los lados si el contenedor principal no es muy ancho.

Elastic Layouts

Este tipo de layouts establece la anchura de los elementos en *em* que es el tamaño de letra por defecto (suele ser 16px).

VENTAJA

- Tamaño en *em* escala correctamente

DESVENTAJA

- Elementos adyacentes pueden solaparse
- Habría que hacer pruebas en todo tipo de dispositivos y con todo tipo de tamaños de letra

Fluid Layouts

Este tipo de layouts establecen el ancho de los distintos contenedores en % (con respecto al contenedor/etiqueta padre).

VENTAJA

- La proporción de los elementos es siempre la misma

DESVENTAJA

- En pantallas pequeñas las columnas puede ser muy estrechas.
- En columnas estrechas texto largos -> celdas muy altas.
- Problemas si hay imágenes y vídeos

Layout con max/min width (Híbridos)

El tamaño (anchura) puede crecer/encoger con unos límites max-width / min-width expresados en pixels.

max-width, en el caso de crecer, mi contenedor no supere esa anchura.

min-width, en caso de encoger, mi contenedor no sea menor que esa anchura.

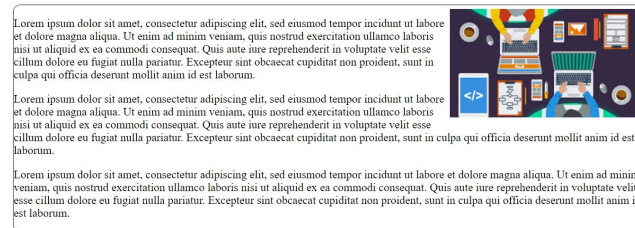




Elementos Flotantes

Elementos flotantes (float)

La propiedad float inicialmente estaba pensada para especificar cómo se situaba el texto alrededor de una imagen.



La propiedad float permite que los elementos se coloquen en una determinada posición con respecto a los otros elementos de la web:

- float: none; → la posición flotante por defecto del elemento en la página.
- float: right; → la posición flotante hacia el lugar más a la derecha posible.
- float: left; → la posición flotante hacia el lugar más a la izq posible.

Clear

Si quiere “forzar” que un elemento deje de flotar debo añadir la propiedad `clear:right` (o `left` o `both`) y ese elemento ya se añadirá, tras el fin del elemento flotante.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Cuidado con la altura del contenedor

Si la imagen es mayor que la altura del contenedor:



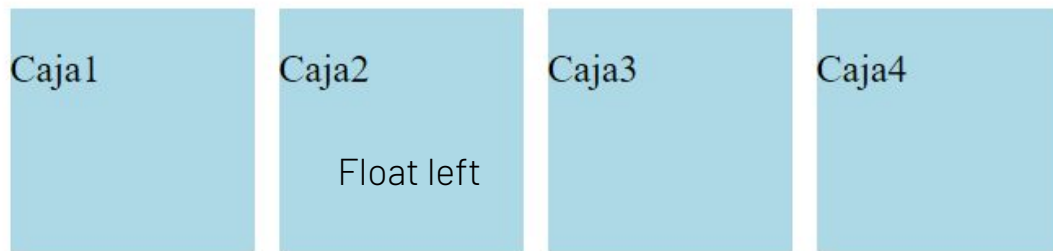
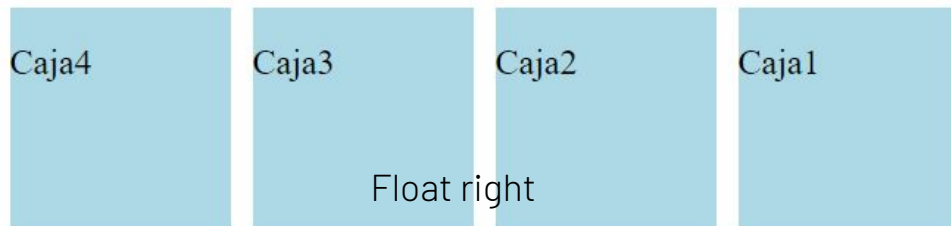
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Podemos adaptarla utilizando en el contenedor la propiedad: `overflow-y: auto;`



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

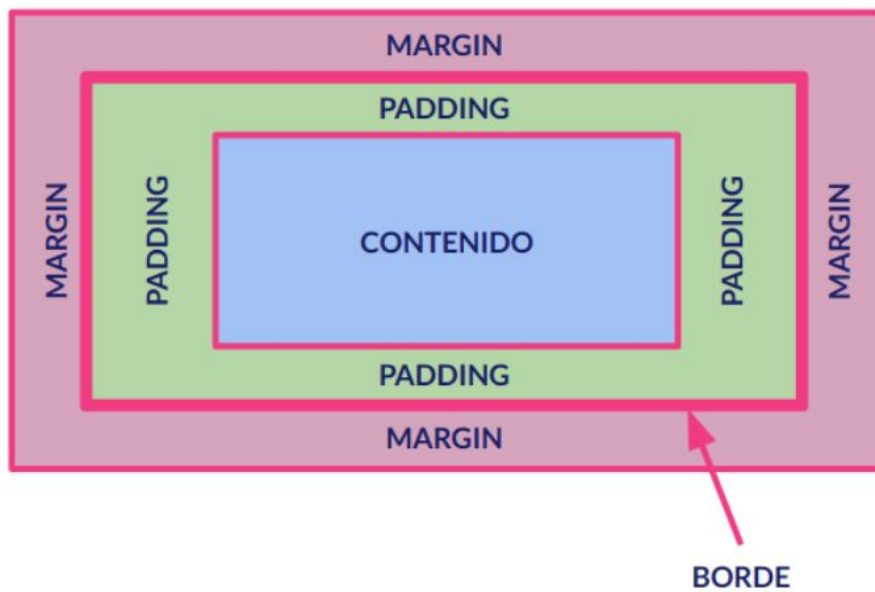
Ejemplo 1





LA PROPIEDAD BOX-SIZING

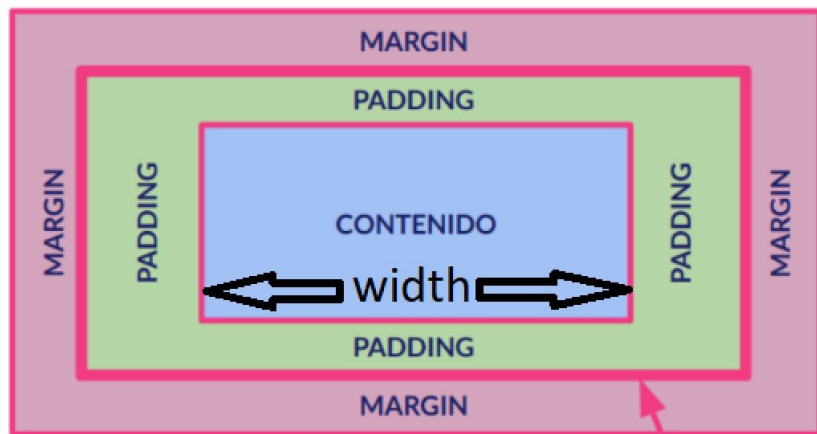




Altura real del elemento sin box-sizing= altura del contenido(height)+padding + borde

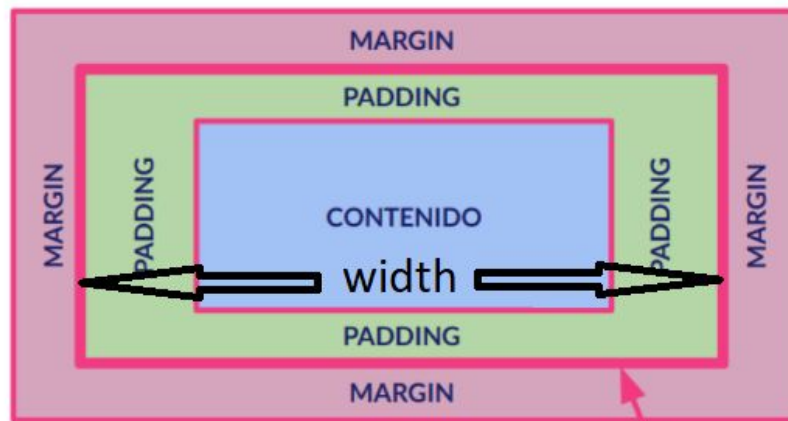
Anchura real del elemento sin box-sizing= anchura del contenido(width)+ padding + borde

La mejor solución para eso es establecer la propiedad CSS **box-sizing: border-box** y de esta manera no tendremos que echar cuentas con los bordes y los paddings. El tamaño que demos al elemento será la medida hasta el borde.



Por defecto (sin BOX-SIZING)

BORDE



Con BOX-SIZING

BORDE



Maquetar con elementos flotantes

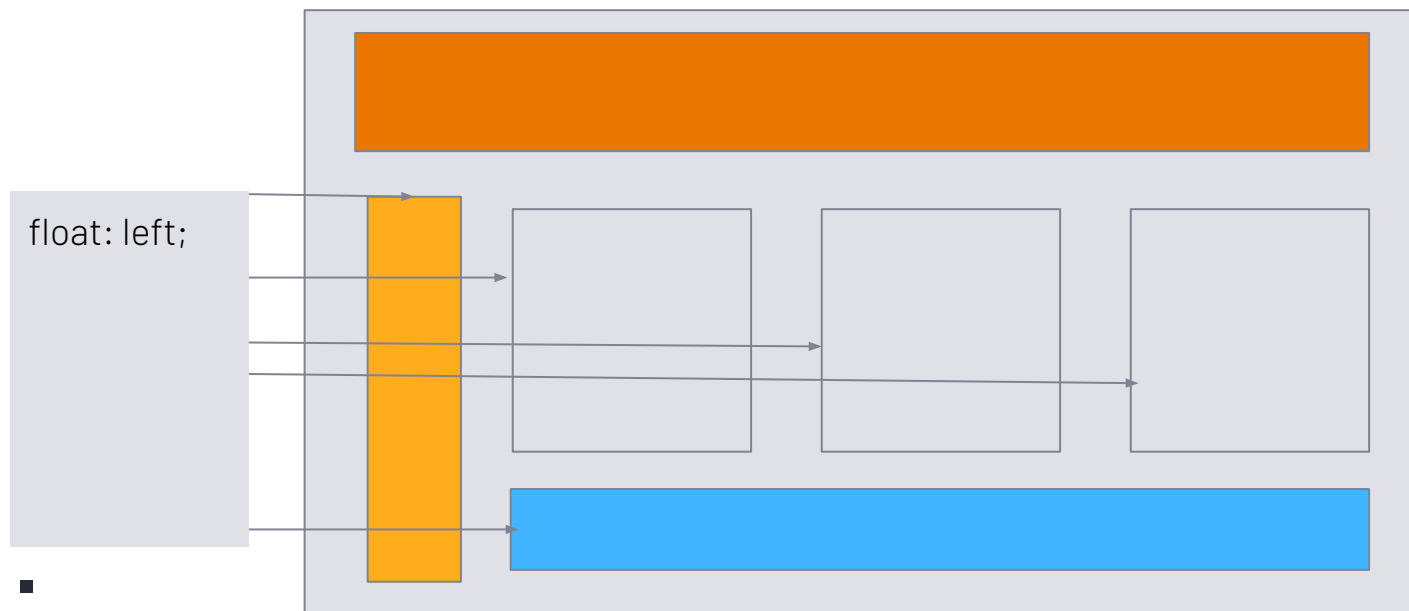


Maquetar con elementos flotantes

- Flotando los elementos según necesitemos (div, nav, header...)
- Dándoles las dimensiones adecuadas.

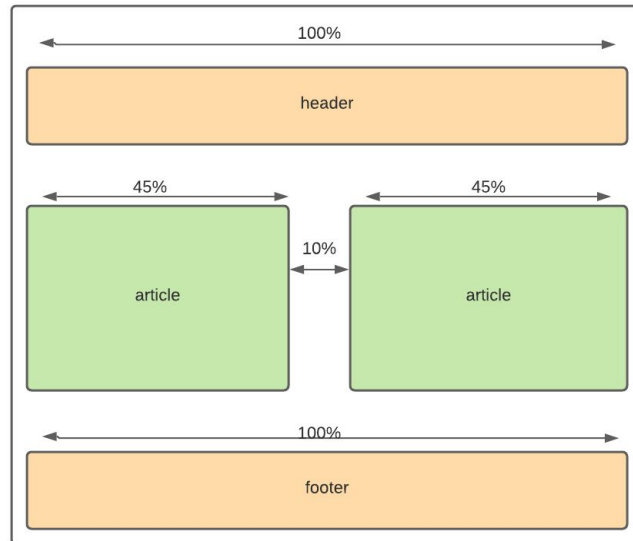


Ejemplo float:



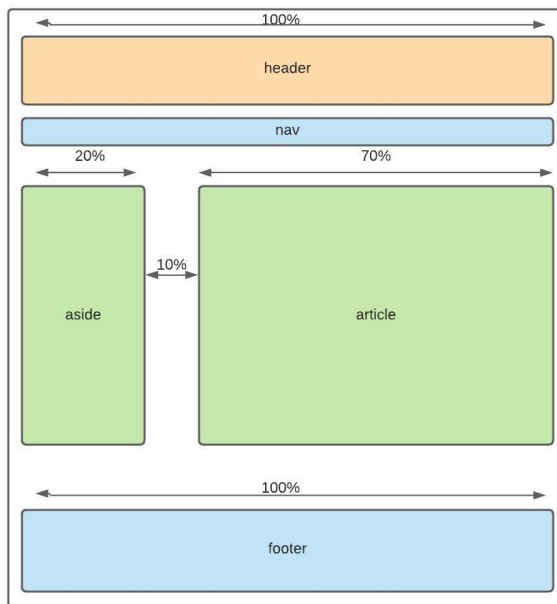
Ejemplo2:

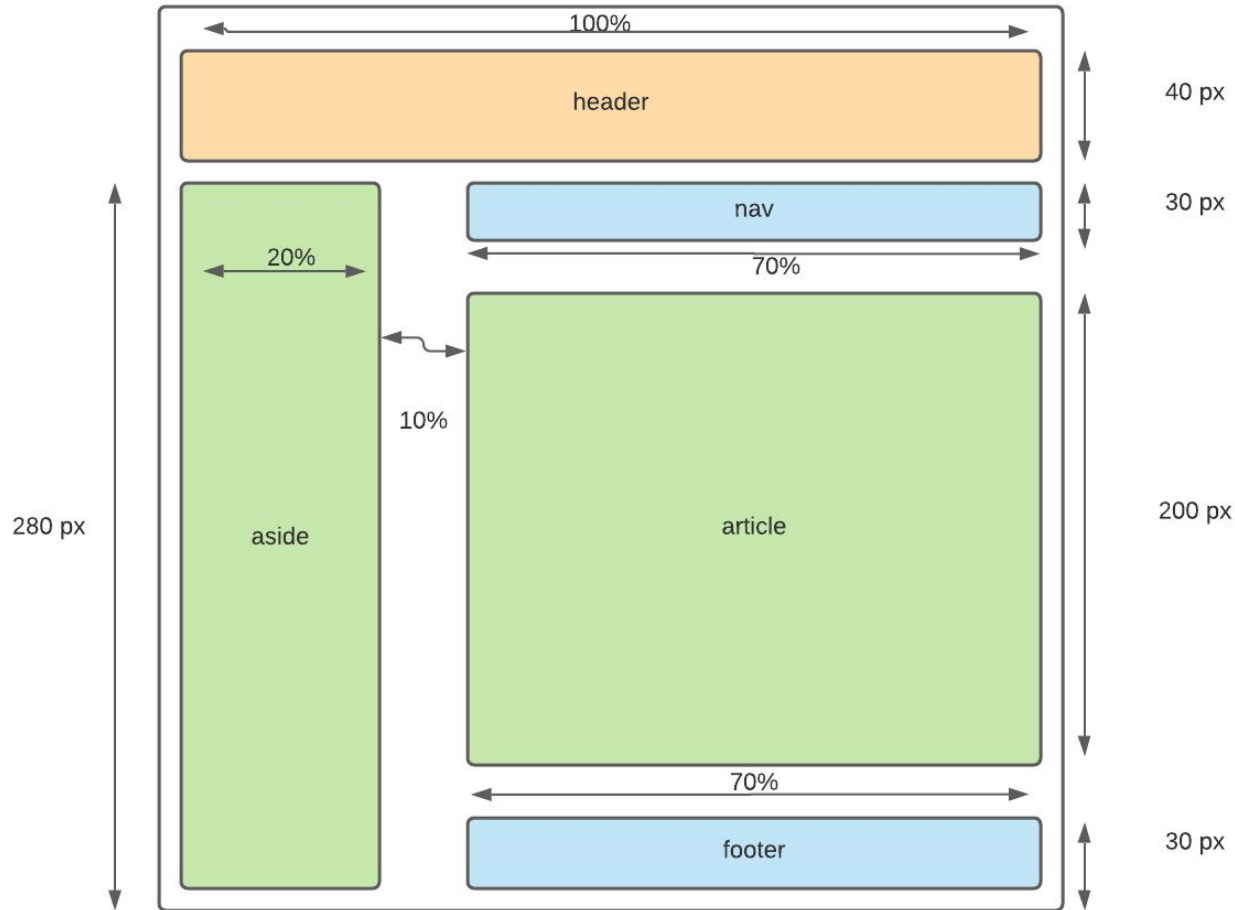
Realiza una página web con el siguiente diseño:



Ejercicio 1

Realiza una página web con el siguiente diseño:







DISEÑO 'RESPONSIVE'



¿QUÉ ES EL DISEÑO RESPONSIVO?

Permite que nuestras página web se vean correctamente en una variedad de dispositivos y de tamaño de pantalla, adaptando el layout al entorno de visualización.

Para ello usa CSS+ ETIQUETAS+ MEDIA QUERIES



Viewport

Área de la pantalla en la que el navegador puede renderizar contenido, es decir, el espacio disponible para mostrar mi página web (**pantalla de nuestro navegador**).

Para maquetar diseños responsivos de manera eficiente necesitamos añadir:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

- The width=device-width establece el ancho de la página según el ancho de pantalla del dispositivo.
- The initial-scale=1.0 establece el nivel de zoom inicial cuando el navegador carga la página por primera vez.

Breakpoint

Podemos agregar un punto de interrupción (Breakpoint) donde ciertas propiedades del diseño se comportarán de manera diferente en cada lado del punto de interrupción. Podemos agregar tantos puntos de interrupción como deseemos.



Media Query

Media es una técnica de CSS3. Usa la regla `@media` para incluir un bloque de propiedades CSS cuando una condición es cierta.



Breakpoint en función de la anchura:

Podemos agregar un punto de interrupción (Breakpoint) a partir del cual, cambiarán ciertas propiedades del diseño:

```
body {  
  background-color: lightgreen;  
}  
@media only screen and (max-width: 600px) {  
  body {  
    background-color: lightblue;  
  }  
}
```

Orientación:

Las consultas de medios también se pueden utilizar para cambiar el diseño de una página según la orientación del navegador. Puede tener un conjunto de propiedades CSS que solo se aplicarán cuando la ventana del navegador sea más ancha que su altura, la denominada orientación "Horizontal":

```
@media only screen and (orientation: landscape) {  
  body {  
    background-color: lightblue;  
  }  
}
```


Ocultar elementos

Otro uso común de las media query es ocultar elementos en diferentes tamaños de pantalla:

Por ejemplo, si el tamaño de la pantalla es de 600 px de ancho o menos, se oculta el elemento:

```
@media only screen and (max-width: 600px) {  
  div.example {  
    display: none;  
  }  
}
```

Cambiar el tamaño de fuente

También puede usar consultas de medios para cambiar el tamaño de fuente de un elemento en diferentes tamaños de pantalla.

Por ejemplo, si el tamaño de la pantalla es **601px o más**, establezca el tamaño de fuente de <div> en **30px** y si el tamaño de la pantalla es de **600 px o menos**, establece el tamaño de fuente de <div> en **20 px** :

```
/* si la pantalla mide 601px o más, font-size=30px */
@media only screen and (min-width: 601px) {
  body {
    font-size: 30px;
  }
}

/* Si la pantalla mide 600px o menos, font-size=20px */
@media only screen and (max-width: 600px) {
  body{
    font-size: 20px;
  }
}
```

Activar Windows