

Estilos CSS

Etiquetas “auxiliaries”: div y span

- Hasta ahora hemos visto diferentes etiquetas, y cada una de ellas nos permitía escribir texto en diferentes formatos. Ahora bien, existen algunos parámetros que **no se basan en nuevas etiquetas**, sino en **atributos** de etiquetas ya existentes.
- Pongamos por caso el hecho de la **alineación** del texto (a la izquierda, a la derecha, centrado, justificado). Se trata de una característica que se define en un **atributo**, ya sea de una etiqueta <p>, , <a>, etc...
- Ahora bien, qué pasa con aquellos elementos que **no** pertenecen a ninguna etiqueta todavía?

Etiquetas “auxiliaries”: div y span

- Para ello utilizaremos dos etiquetas auxiliares, **<div>** y ****.
 - **<div>...</div>**: se utiliza para definir **bloques definidos** que tendrán un formato determinado. Se suele utilizar para alinear párrafos, crear marcos independientes...
 - **...**: a diferencia de <div>, utilizaremos la etiqueta para **pequeños fragmentos** que tengan una diferencia de formato con su alrededor, como por ejemplo, si queremos resaltar una palabra en otro color.
- Como podemos ver, ninguna de ellas define ningún elemento concreto, sino que las utilizaremos para poder aplicar formato a fragmentos de código determinados.

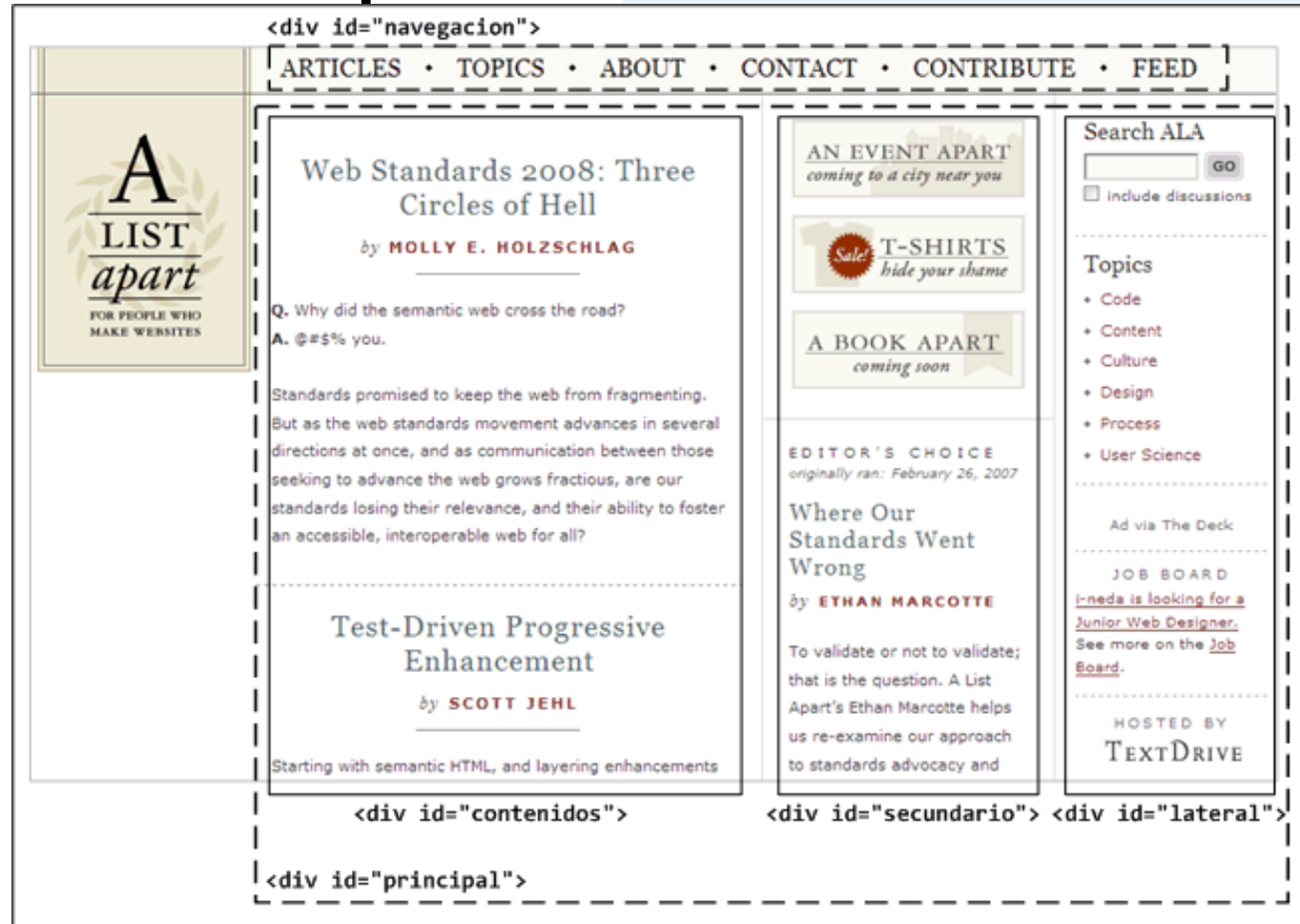
Etiquetas “auxiliaries”: div y span

HTML se encarga de agrupar los elementos de la página en diferentes divisiones en función de su finalidad:

- La zona de la cabecera de la página
- La zona de contenidos
- Una zona lateral para el menú
- Otras secciones menores
- La zona del pie de página
- etc.

Etiquetas “auxiliaries”: div y span

No es posible crear estas estructuras tan complejas utilizando solo el HTML , ya que es imprescindible emplear las hojas de estilos CSS



¿Qué es CSS?

El lenguaje CSS se utiliza para definir el aspecto de **TODOS** los contenidos: el color, tamaño y tipo de letra de los párrafos de texto, la separación entre titulares y párrafos, la tabulación con la que se muestran los elementos de una lista.

¿Cómo usar CSS?

1. Incluir CSS en el mismo documento HTML dentro de la etiqueta

<p style="color: black; font-family: Verdana;">

¿Cómo usar CSS?

2. Incluir CSS en el mismo documento HTML

```
<style>
```

```
p { color: black; font-family: Verdana; }
```

```
</style>
```

Este código lo incluimos dentro del HEAD entre las etiquetas `<head>` y `</head>`.

¿Cómo usar CSS?

3. Incluir CSS en un archivo externo mediante el siguiente código:

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

Este código lo incluimos dentro del HEAD entre las etiquetas <head> y </head>.

¿Cómo usar CSS?

Estilos sobre una etiqueta

```
<head>
<title>El lenguaje
HTML</title>
<style>
body {
    background-color: #CCCCCC;
    color: #000000;
    margin-left: 20%;
    margin-right: 20%;
    border: 1px dotted gray;
    padding: 10px 10px 10px 10px;
    font-family: sans-serif;
}
</style>
</head>
```

Estilos sobre “class” e “id”

```
<head>
<title>El lenguaje
HTML</title>
<style>
.normalAzul {
    font-size:18px;
    color: blue;
}
#normal {
    font-size:18px;
    color: orange;
    font-weight: bold;
}
</style>
</head>
```

Class

id

Glosario básico

Comentarios

/ Este es un comentario en CSS */*

/ Este es un
comentario CSS de varias
líneas */*

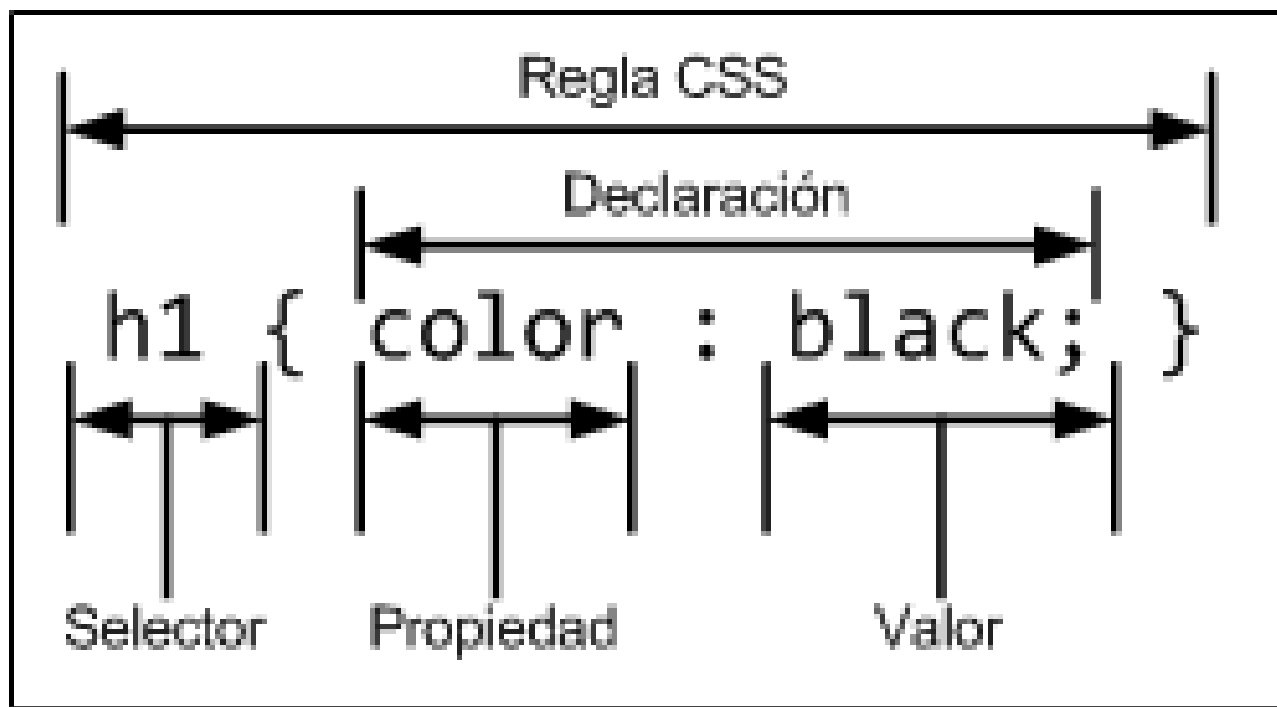
La sintaxis de los comentarios CSS es muy diferente a la de los comentarios HTML.

<!-- Este es un comentario en HTML -->

*<!-- Este es un
comentario HTML de varias
líneas -->*

Glosario básico

Componentes de un estilo CSS básico



Selectores

Selector universal:

```
* {  
    margin: 0;  
    padding: 0;  
}
```

Selector de tipo o etiqueta:

```
p {  
    ...  
}  
h1 {  
    ...  
}  
h2 {  
    ...  
}
```

Selector múltiple:

```
h1, h2, h3 {  
    color: #8A8E27;  
    font-weight: normal;  
    font-family: Arial, Helvetica, sans-serif;  
}
```

Selectores

Selector descendente: Selecciona los elementos que se encuentran dentro de otros elementos. Un elemento es descendiente de otro cuando se encuentra entre las etiquetas de apertura y de cierre del otro elemento.

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

La sintaxis formal del selector descendente se muestra a continuación:

elemento1 elemento2 elemento3 ... elementoN

```
p a span em {  
text-decoration: underline;  
}
```

¿A que elementos se les aplica los estilos de la regla anterior?

Selectores

Selector descendente

No debe confundirse el selector descendente con la combinación de selectores:

/ El estilo se aplica a todos los elementos "p", "a", "span" y "em" */*

```
p, a, span, em {  
  text-decoration: underline;  
}
```

/ El estilo se aplica solo a los elementos "em" que se encuentran dentro de "p a span" */*

```
p a span em {  
  text-decoration: underline;  
}
```

Selectores

Selector de classe: consisteix en utilitzar el atributo class de HTML sobre ese elemento para indicar directamente la regla CSS que se le debe aplicar.

```
.destacado {  
  color: red;  
}
```

```
<p class="destacado">Este párrafo se le aplica el estilo...</p>  
<p>Este párrafo NO se le aplica el estilo...</p>
```

En una misma página HTML varios elementos diferentes pueden utilizar el mismo valor en el atributo class:

```
<p class="destacado">Es un texto de prueba...</p>  
<p> Es un <a href="#" class="destacado"> texto de prueba </a>...</p>  
<p> Es un texto de <em class="destacado">prueba</em>...</p>
```


Selectores

Selectores de ID: permite seleccionar un elemento de la página a través del valor de su atributo id.

La sintaxis de los selectores de ID es muy parecida a la de los selectores de clase, salvo que se utiliza el símbolo de la almohadilla (#) en vez del (.)

```
#destacado {  
    color: red;  
}
```

```
<p>Primer párrafo</p>
```

```
<p id="destacado">Segundo párrafo</p>
```

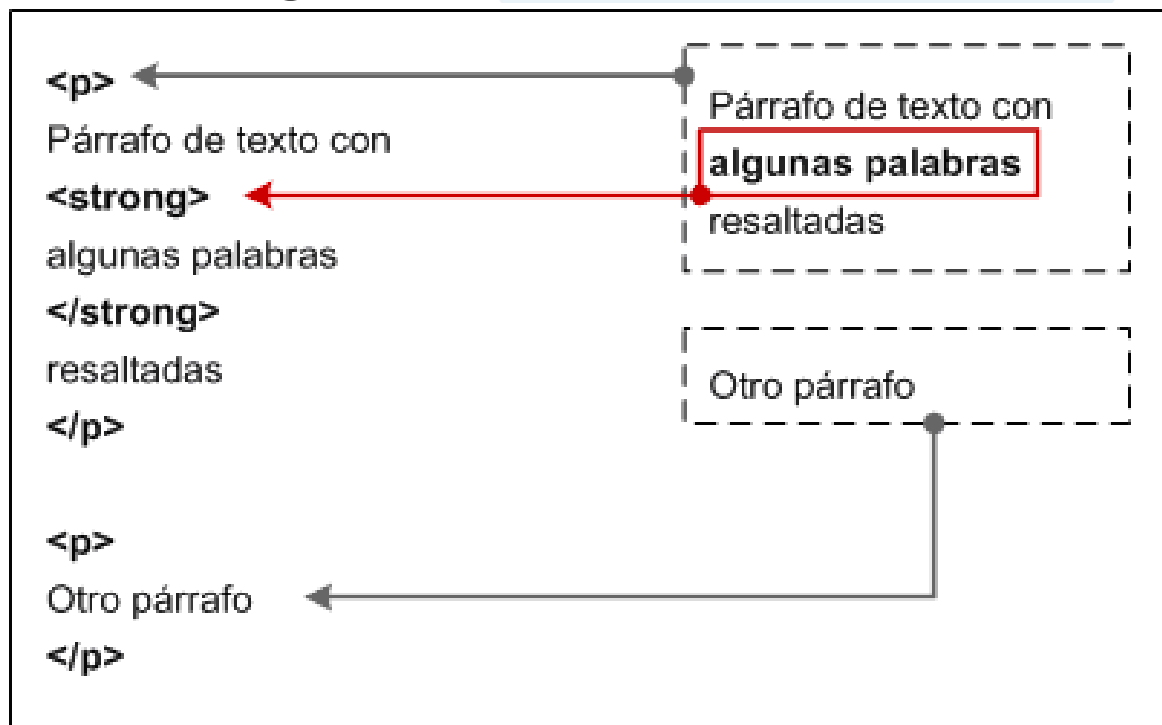
```
<p>Tercer párrafo</p> punto (.)
```

Ejercicio 1 y 2

Modelo de cajas

El diseño de cualquier página XHTML está compuesto por cajas rectangulares.

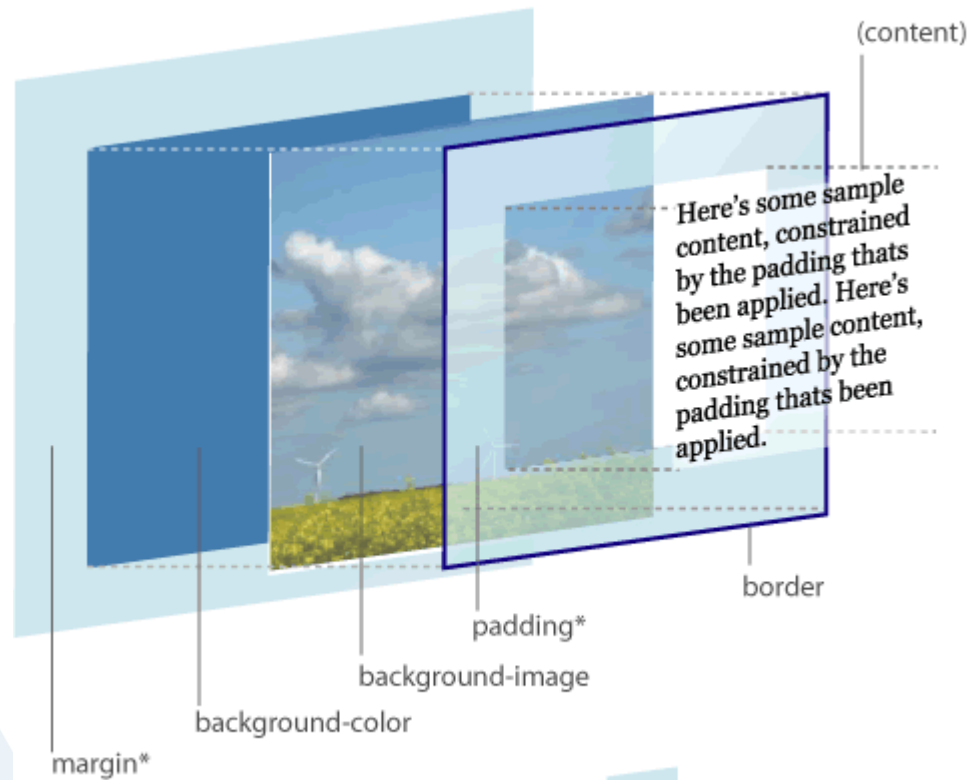
Las cajas de una página se crean automáticamente. Cada vez que se inserta una etiqueta o elemento en la página, se crea una nueva caja rectangular que encierra los contenidos del elemento.



Modelo de cajas

Cada una de las cajas está formada por seis partes.

THE CSS BOX MODEL HIERARCHY



Modelo de cajas

A continuación comentamos las propiedades más comunes:

Fondo y bloque

background-color: *color de fondo*

background-image: *imagen de fondo*

vertical-align: *alineamiento vertical*

text-align: *alineamiento del texto*

padding: *margen **interno***

margin: *margen **externo***

Estas dos últimas, puestas así, se refieren a los **cuatro márgenes**. Se pueden especificar añadiéndoles al final **-left**, **-right**, **-top** o **-bottom**, como por ejemplo:

margin-right: 5px;

Modelo de cajas

Bordes

border-left-color:	color del borde izquierdo
border-left-style:	estilo del borde (solido, punteado...)
border-left-width:	ancho del borde
border-right-color...	las mismas propiedades para los tres bordes
border-top-color...	
border-bottom-color...	

Si se quiere establecer el mismo estilo, color o tamaño para todos los bordes:

border-color, border-style, border-width

Estilo completo de todos los bordes:

border: width style color;

Modelo de cajas

Otros

width: Anchura

height: Altura

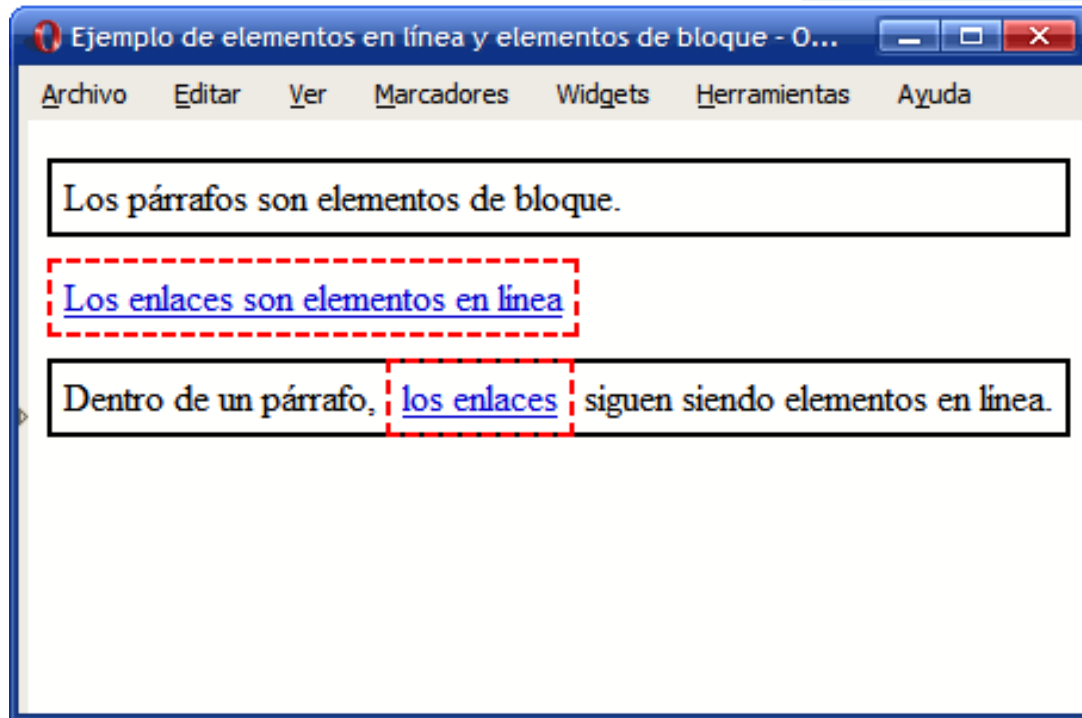
Valores: medida | porcentaje | auto | inherit

...

Ejercicio 3 y 4

Posicionamiento

Cajas creadas por los elementos de línea y los elementos de bloque



Posicionamiento

Diferentes modelos para posicionar una caja:

- **Posicionamiento normal o estático:** se trata del posicionamiento que utilizan los navegadores si no se indica lo contrario.
- **Posicionamiento relativo:** consiste en posicionar una caja según el posicionamiento normal y después desplazarla respecto de su posición original.
- **Posicionamiento absoluto:** 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:** 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.
- **Posicionamiento flotante:** desplaza las cajas todo lo posible hacia la izquierda o hacia la derecha de la línea en la que se encuentran.

Posicionamiento

El posicionamiento de una caja se establece mediante la propiedad **position**:

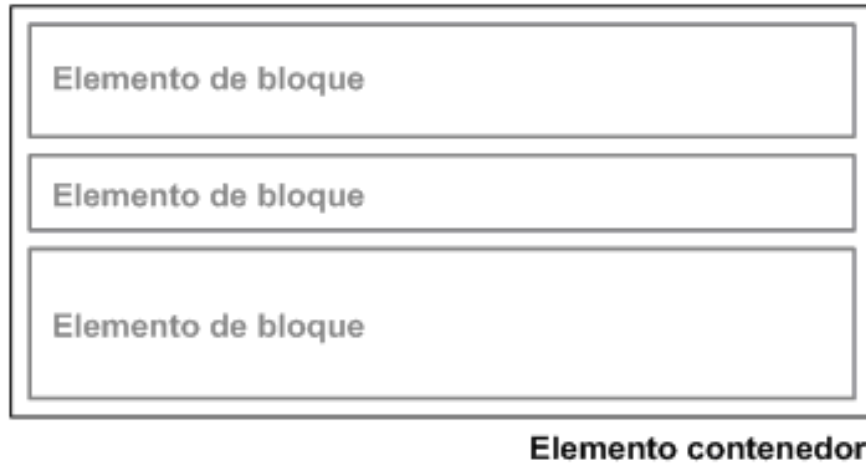
position	Posicionamiento
Valores	<code>static</code> <code>relative</code> <code>absolute</code> <code>fixed</code> <code>inherit</code>

El desplazamiento de la caja se controla con las propiedades **top**, **right**, **bottom** y **left**.

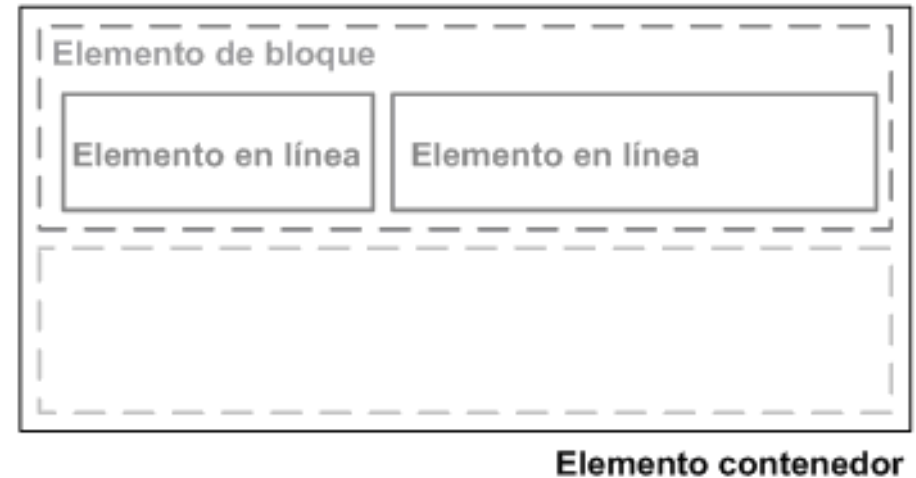
top right bottom left	Desplazamiento superior Desplazamiento lateral derecho Desplazamiento inferior Desplazamiento lateral izquierdo
Valores	<code><medida></code> <code><porcentaje></code> <code>auto</code> <code>inherit</code>

Posicionamiento

Posicionamiento normal



Posicionamiento normal de los
elementos de bloque

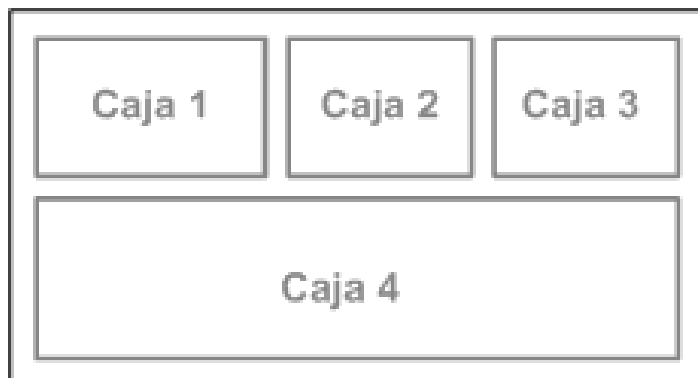


Posicionamiento normal de los
elementos de en línea

Posicionamiento

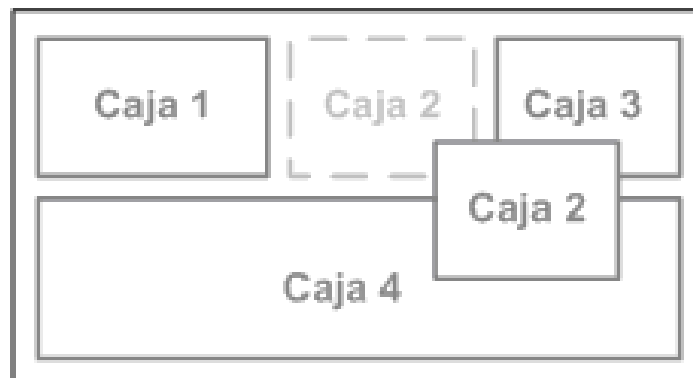
Posicionamiento relativo

Elemento contenedor



Posicionamiento normal

Elemento contenedor



Posicionamiento relativo de la caja 2

El posicionamiento relativo permite desplazar una caja respecto de su posición original establecida mediante el posicionamiento normal. El desplazamiento de la caja se controla con las propiedades top, right, bottom y left.

Posicionamiento

Posicionamiento relativo: Ejemplo

```
img.desplazada {  
  position: relative;  
  top: 8em;  
}
```

```

```

```

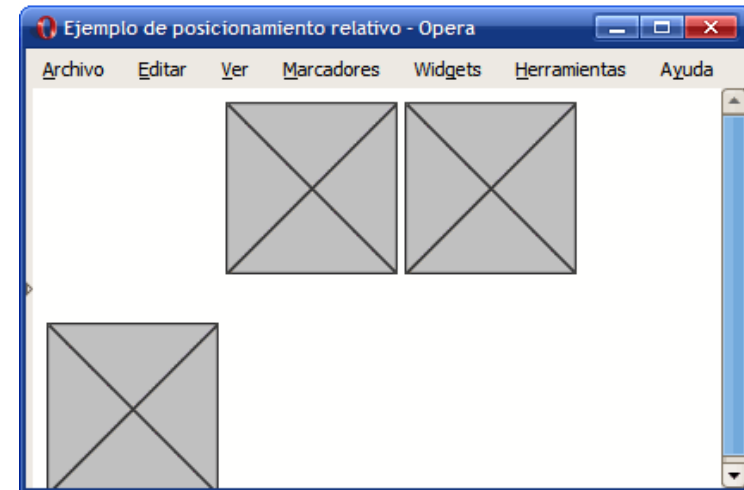
```

```

```



Elementos posicionados de forma normal

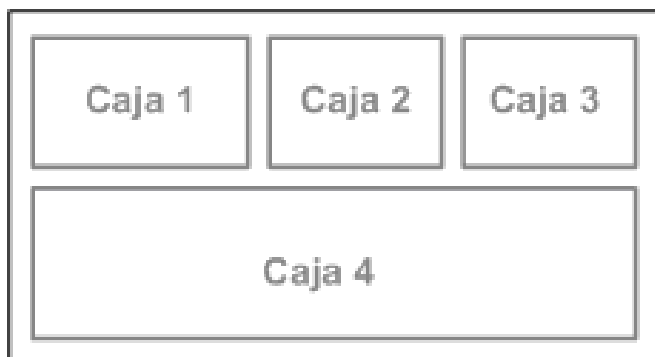


Elemento posicionado de forma relativa

Posicionamiento

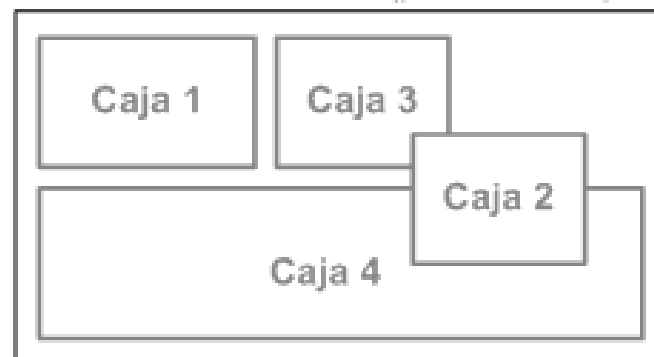
Posicionamiento absoluto

Elemento contenedor



Posicionamiento normal

Elemento contenedor (posicionado)



Posicionamiento absoluto de la caja 2

Cuando una caja se posiciona de forma absoluta, el resto de elementos de la página la ignoran y ocupan el lugar original ocupado por la caja posicionada.

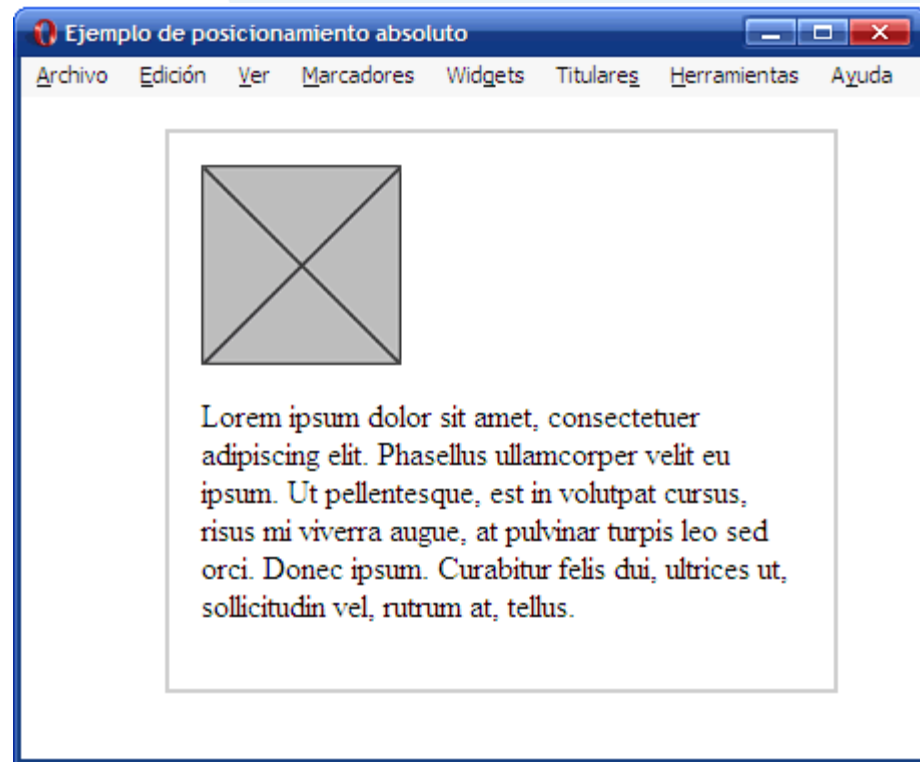
La referencia de los valores de las propiedades top, right, bottom y left es el origen de coordenadas de su primer elemento contenedor posicionado.

Posicionamiento

Posicionamiento absoluto: ejemplo

```
div {  
border: 2px solid #CCC;  
padding: 1em;  
margin: 1em 0 1em 4em;  
width: 300px;  
}
```

```
<div>  
  
<p>Lorem ipsum dolor sit amet, consectetur  
adipiscing elit. Phasellus  
ullamcorper velit eu ipsum. Ut pellentesque,  
est in volutpat cursus, risus  
mi viverra augue, at pulvinar turpis leo sed  
orci. Donec ipsum. Curabitur  
felis dui, ultrices ut, sollicitudin vel, rutrum at,  
tellus.  
</p>  
</div>
```



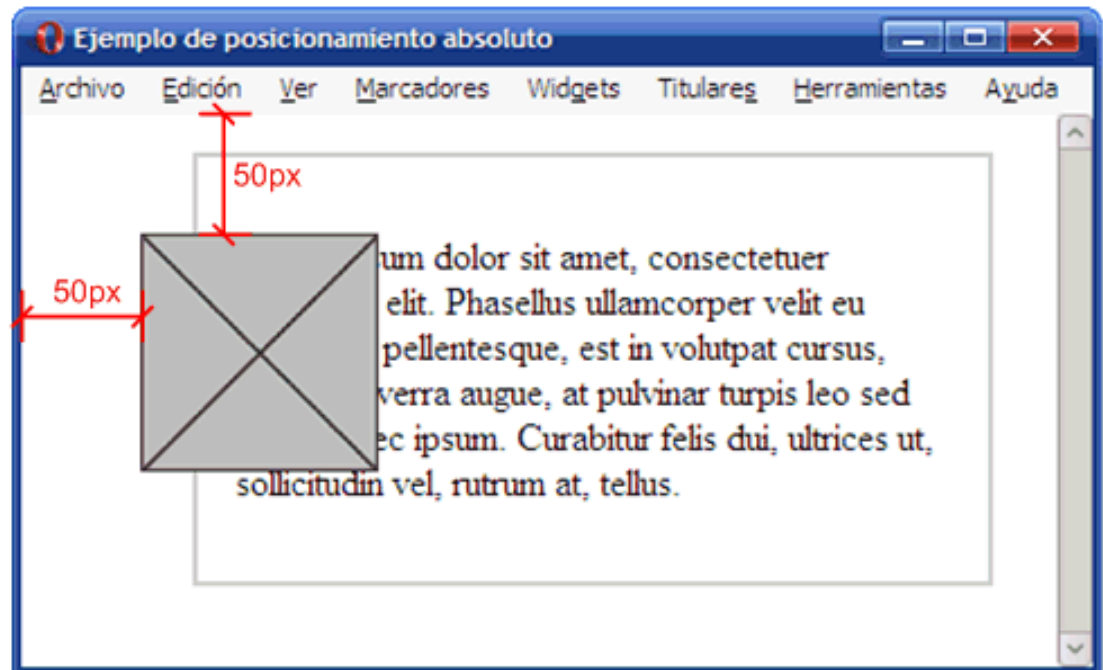
Situación original antes de modificar el
posicionamiento

Posicionamiento

Posicionamiento absoluto: ejemplo

Se posiciona de forma absoluta la
imagen

```
div img {  
position: absolute;  
top: 50px;  
left: 50px;  
}
```



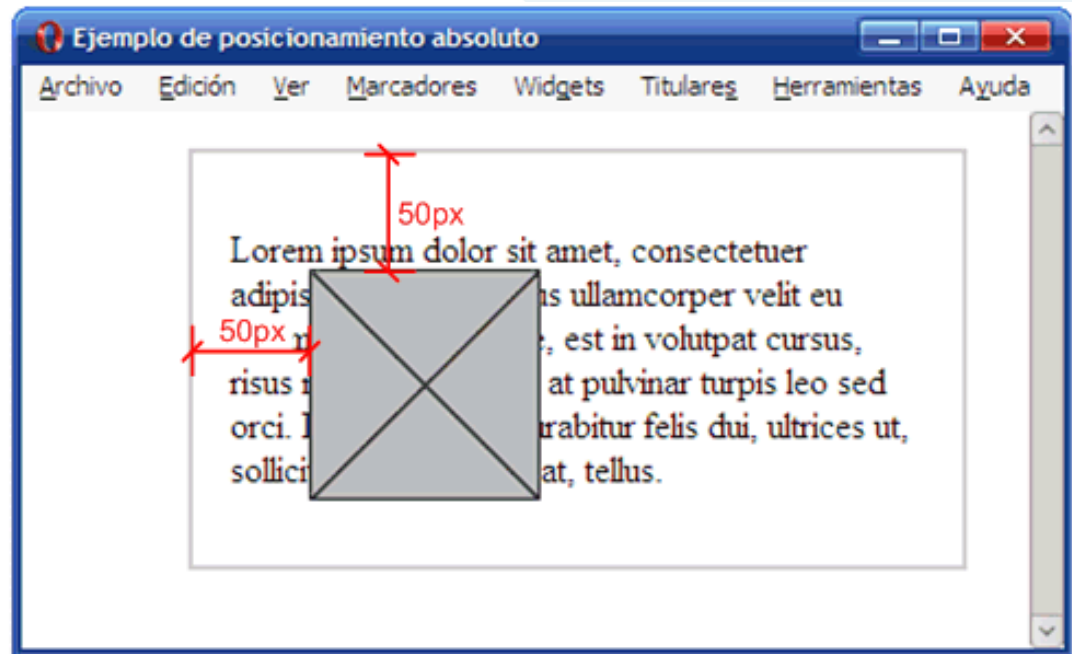
La referencia del posicionamiento absoluto es la
página entera

Posicionamiento

Posicionamiento absoluto: ejemplo

Se posiciona de forma relativa el
contenedor

```
div {  
border: 2px solid #CCC;  
padding: 1em;  
margin: 1em 0 1em 4em;  
width: 300px;  
position: relative;  
}  
div img {  
position: absolute;  
top: 50px;  
left: 50px;  
}
```



La referencia del posicionamiento absoluto es el elemento
contenedor de la imagen

Posicionamiento

Posicionamiento fijo

El posicionamiento fijo es un caso particular del posicionamiento absoluto, ya que sólo se diferencian en el comportamiento de las cajas posicionadas.

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.

Posicionamiento

Posicionamiento flotante

La propiedad position no permite controlar el posicionamiento flotante, que se establece con otra propiedad llamada **float**.

float: left | right | none | inherit

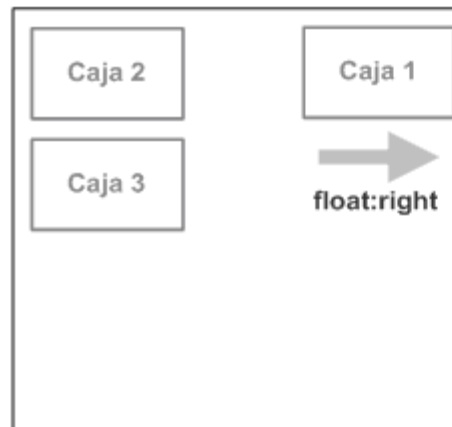
Cuando se posiciona una caja de forma flotante la caja deja de pertenecer al flujo normal de la página, lo que significa que el resto de cajas ocupan el lugar dejado por la caja flotante.

Elemento contenedor



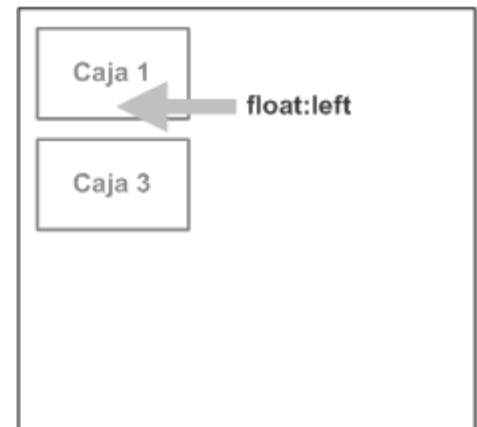
Posicionamiento normal

Elemento contenedor



Posicionamiento float de la caja 1

Elemento contenedor



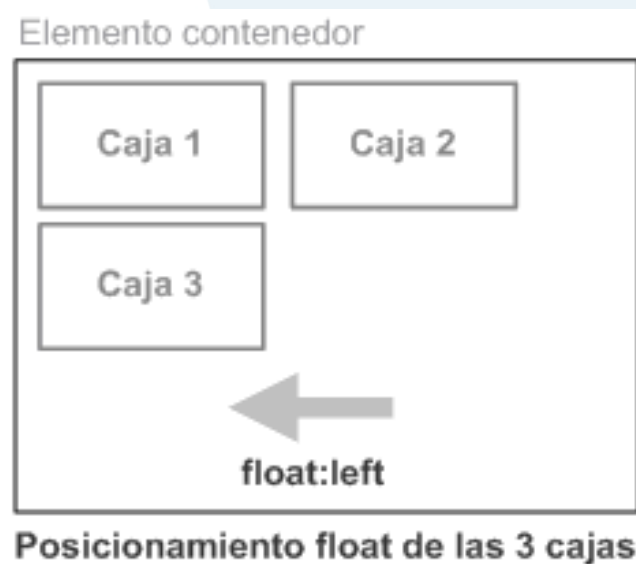
Posicionamiento float de la caja 1

Posicionamiento

Posicionamiento flotante



Ejemplo de posicionamiento
float de varias cajas



Ejemplo de posicionamiento
float cuando no existe sitio
suficiente

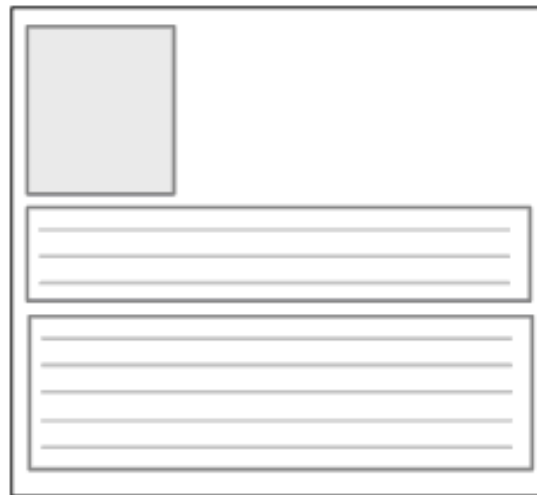
Posicionamiento

Posicionamiento flotante

Los elementos que se encuentran alrededor de una caja flotante adaptan sus contenidos para que fluyan alrededor del elemento posicionado:

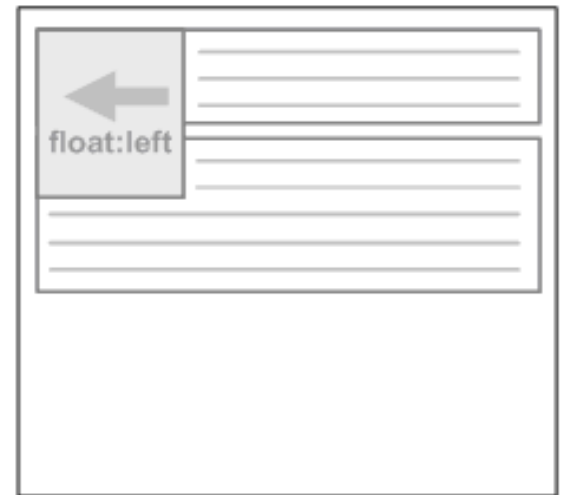
```
img {  
float: left;  
}
```

Elemento contenedor



Posicionamiento normal

Elemento contenedor



Posicionamiento float de la imagen

Posicionamiento

Posicionamiento flotante

Forzar a que un elemento no fluya alrededor de otro elemento posicionado mediante float.

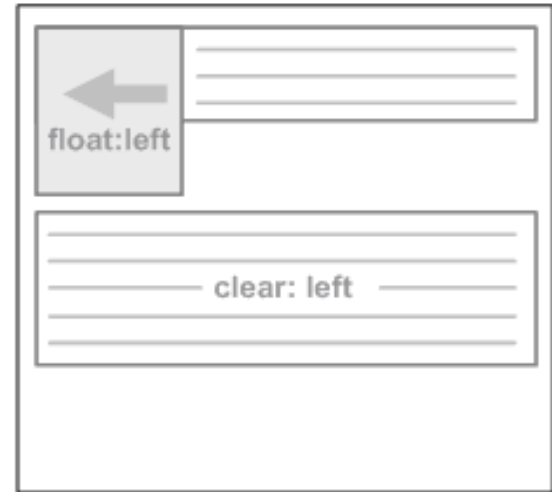
```
p{  
clear: left;  
}
```

Elemento contenedor



Posicionamiento float de la imagen

Elemento contenedor

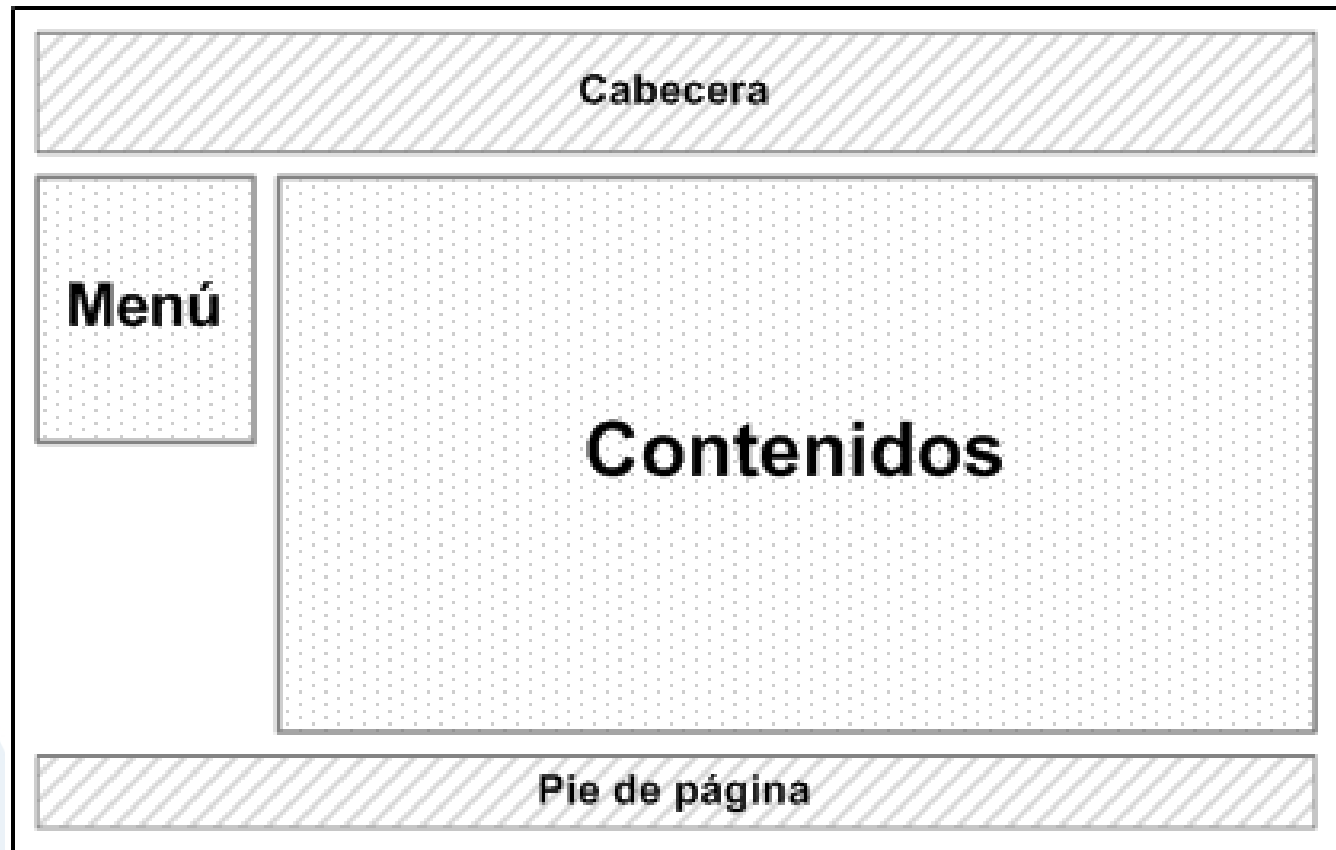


Párrafo "limpio" con clear:left

clear: none | left | right | both | inherit

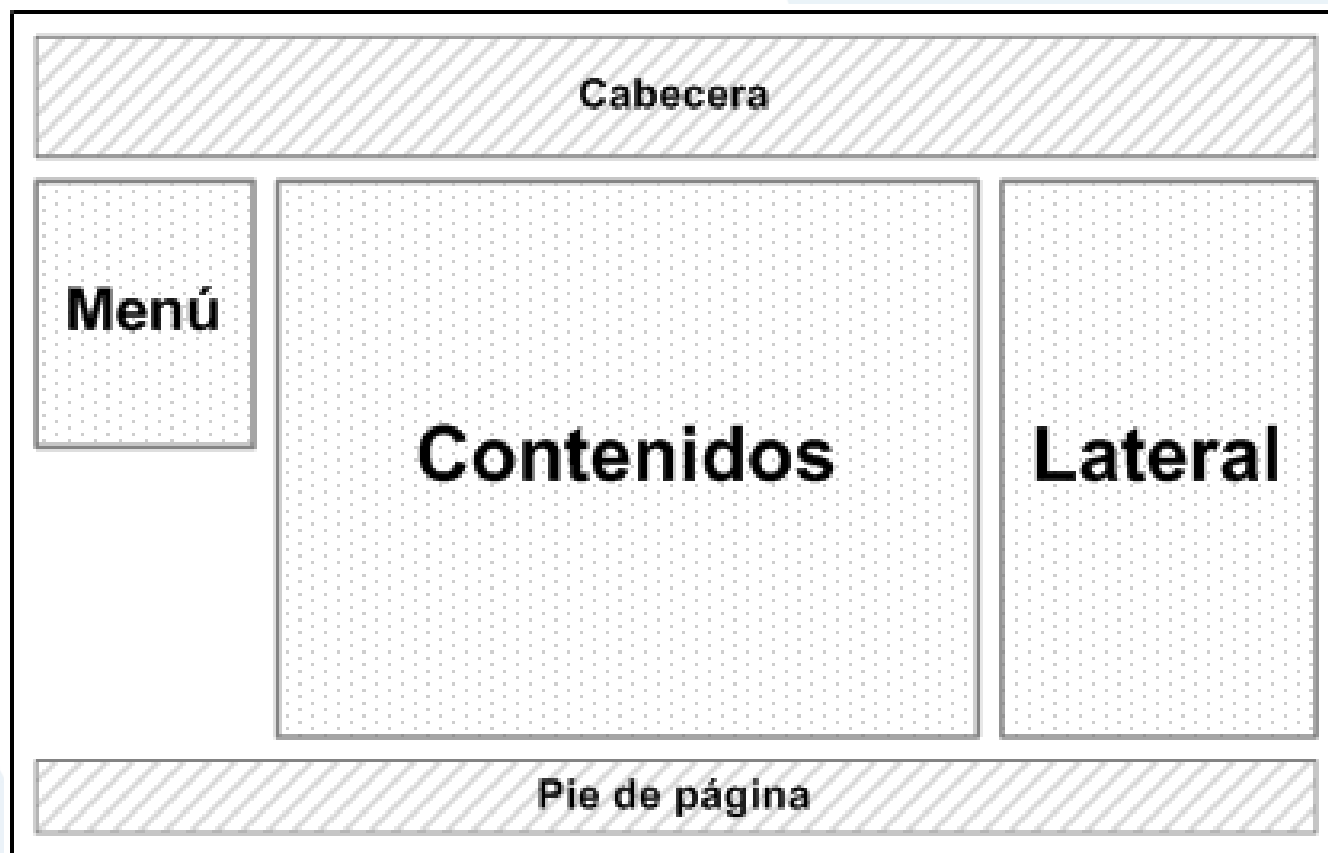
Posicionamiento

Diseño a 2 columnas con cabecera y pie de página



Posicionamiento

Diseño a 3 columnas con cabecera y pie de página



Visualización

Propiedades display

display: inline | block | none |
inline-block

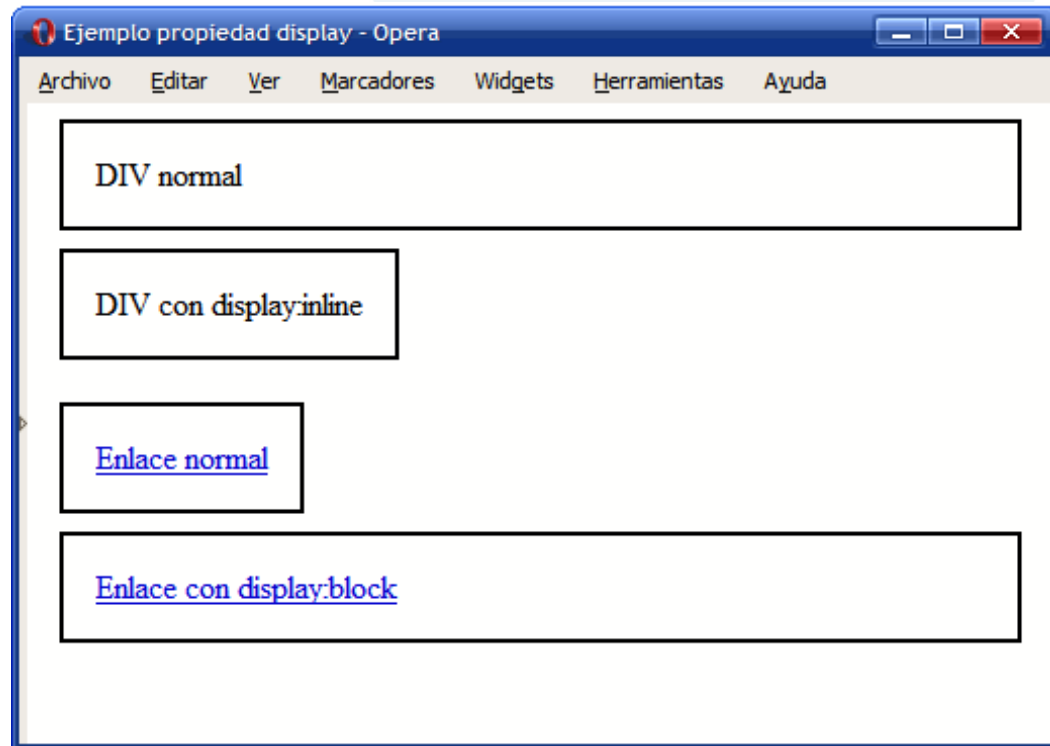
```
<div>DIV normal</div>
```

```
<div style="display:inline">DIV con  
display:inline</div>
```

```
<a href="#">Enlace normal</a>
```

```
<a href="#"  
style="display:block">Enlace con  
display:block</a>
```

El valor **none** oculta un
elemento y hace que
desaparezca de la página.



Visualización

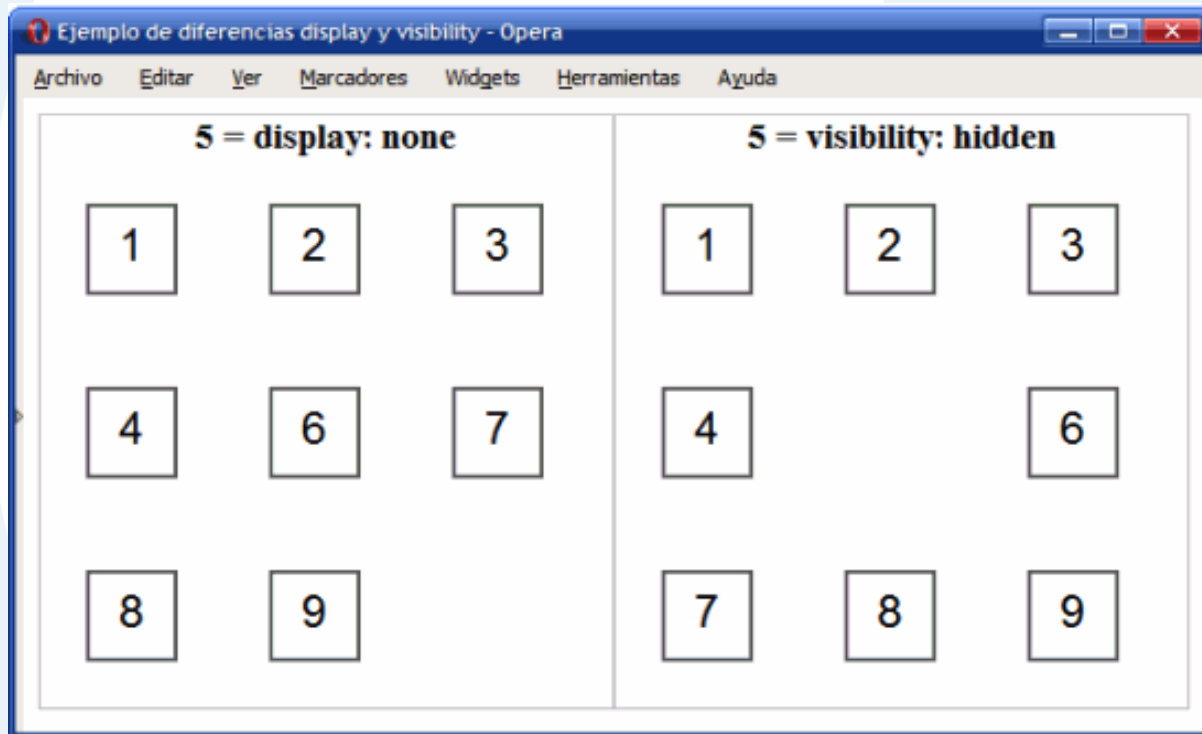
Propiedades visibility

visibility: visible | hidden | collapse | inherit

El valor collapse de la propiedad visibility sólo se puede utilizar en las filas, grupos de filas, columnas y grupos de columnas de una tabla. Su efecto es similar al de la propiedad display con el valor none.

Visualización

Propiedades display y visibility

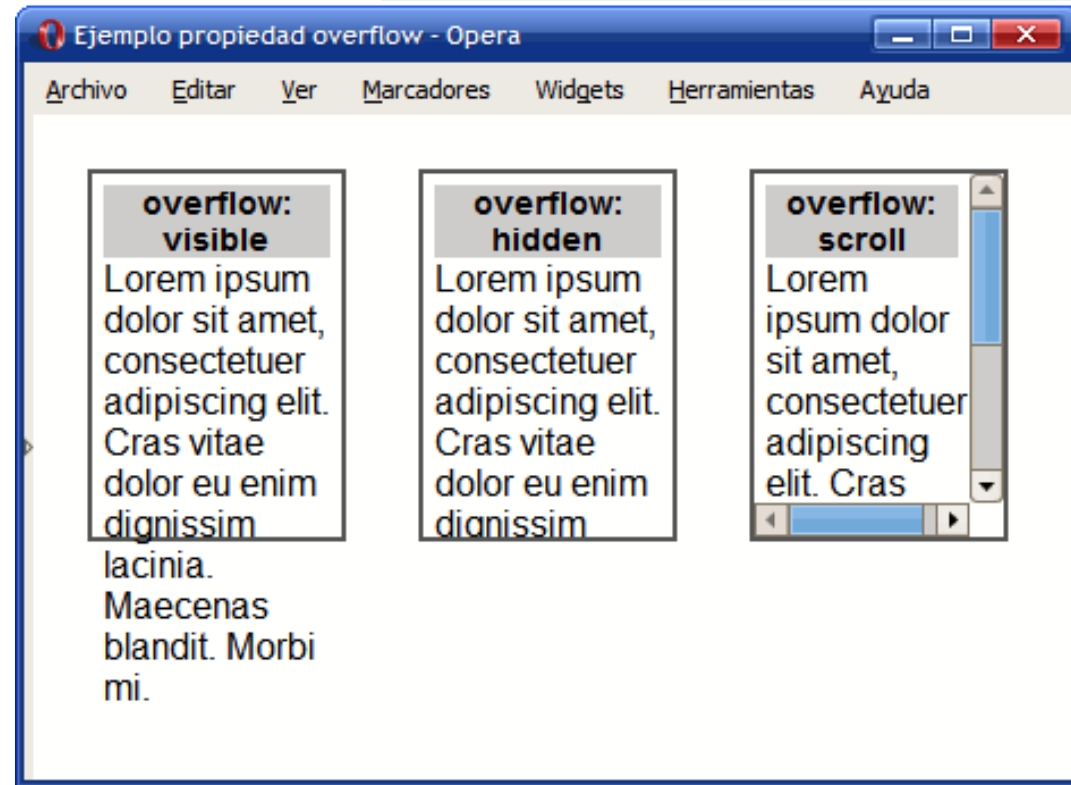


Diferencias visuales entre las
propiedades display y visibility

Visualización

Propiedad overflow

la propiedad overflow controla la forma en la que se visualizan los contenidos que sobresalen de sus elementos.



Ejemplo de propiedad overflow

Texto

CSS define numerosas propiedades para modificar la apariencia del texto. A continuación comentamos las propiedades más comunes:

font-family: *indica la **fuentes** del texto*

font-size: *el **tamaño** del texto*

font-color: *para el **color** del texto*

font-weight:bold : *para poner el texto en **negrita***

font-style: italic : *para **cursiva***

font-decoration: underline : *para **subrayado***

text-align: *Alineación del texto*

Enlaces

CSS también permite aplicar diferentes estilos a un mismo enlace en función de su estado. Para ello introduce un nuevo concepto llamado *pseudo-clases*:

- **:link**, aplica estilos a los enlaces que apuntan a páginas o recursos que aún no han sido visitados por el usuario.
- **:visited**, aplica estilos a los enlaces que apuntan a recursos que han sido visitados anteriormente por el usuario. El historial de enlaces visitados se borra automáticamente cada cierto tiempo y el usuario también puede borrarlo manualmente.
- **:hover**, aplica estilos al enlace sobre el que el usuario ha posicionado el puntero del ratón.
- **:active**, aplica estilos al enlace que está pinchando el usuario. Los estilos sólo se aplican desde que el usuario pincha el botón del ratón hasta que lo suelta, por lo que suelen ser unas pocas décimas de segundo.

*Las pseudo-clases :link y :visited solamente están definidas para los enlaces, pero las pseudo-clases :hover y :active se definen para todos los elementos HTML.

Enlaces

Las pseudo-clases siempre incluyen dos puntos (:) por delante de su nombre y siempre se añaden a continuación del elemento al que se aplican, sin dejar ningún espacio en blanco por delante:

/ Incorrecto: el nombre de la pseudo-clase no se puede separar de los dos puntos iniciales */*

a: visited { ... }

/ Incorrecto: la pseudo-clase siempre se añade a continuación del elemento al que modifica */*

a :visited { ... }

/ Correcto: la pseudo-clase modifica el comportamiento del elemento <a> */*

a:visited { ... }

Las pseudo-clases también se pueden combinar con cualquier otro selector complejo:

a.importante:visited { ... }

a#principal:hover { ... }

div#menu ul li a.primer:active { ... }

Enlaces

Definir las reglas CSS que permiten mostrar los enlaces con los siguientes estilos:

- 1.En su estado normal, los enlaces se muestran de color rojo #CC0000.
- 2.Cuando el usuario pasa su ratón sobre el enlace, se muestra con un color de fondo rojo #CC0000y la letra de color blanco #FFF.
- 3.Los enlaces visitados se muestran en color gris claro #CCC.



Enlaces

Mostrar los enlaces como si fueran botones:

```
a { margin: 1em 0; float: left; clear: left; }
```

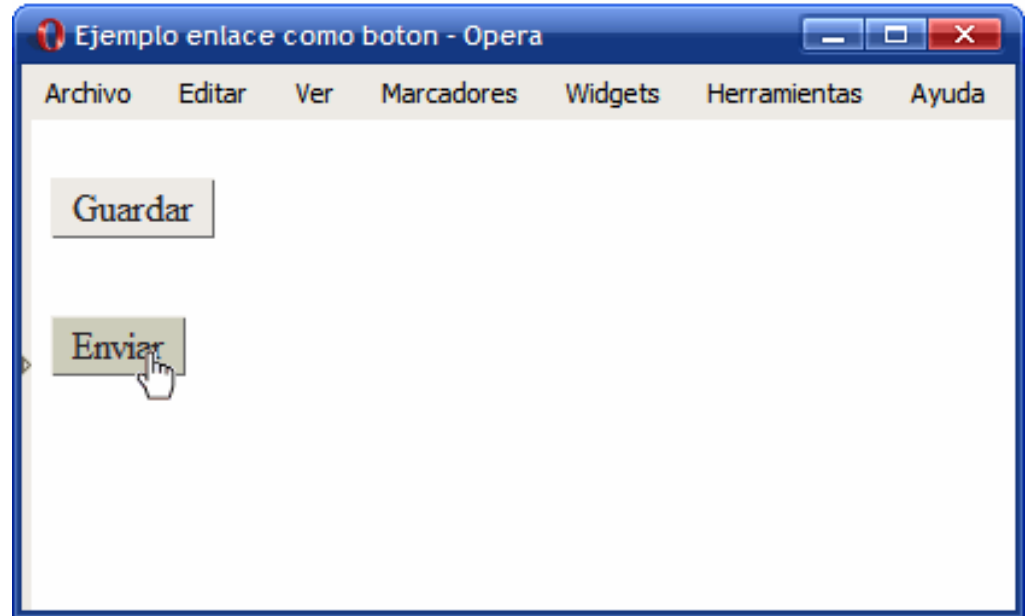
```
a.boton {  
text-decoration: none;  
background: #EEE;  
color: #222;  
border: 1px outset #CCC;  
padding: .1em .5em;  
}
```

```
a.boton:hover {  
background: #CCB;  
}
```

```
a.boton:active {  
border: 1px inset #000;  
}
```

```
<a class="boton" href="#">Guardar</a>
```

```
<a class="boton" href="#">Enviar</a>
```



Imágenes

Establecer la anchura y altura de las imágenes:

```
#destacada {  
width: 120px;  
height: 250px;  
}  

```

El navegador deforma las imágenes y el resultado estético es muy desagradable. Se recomienda establecer la altura/anchura de las imágenes.

```

```

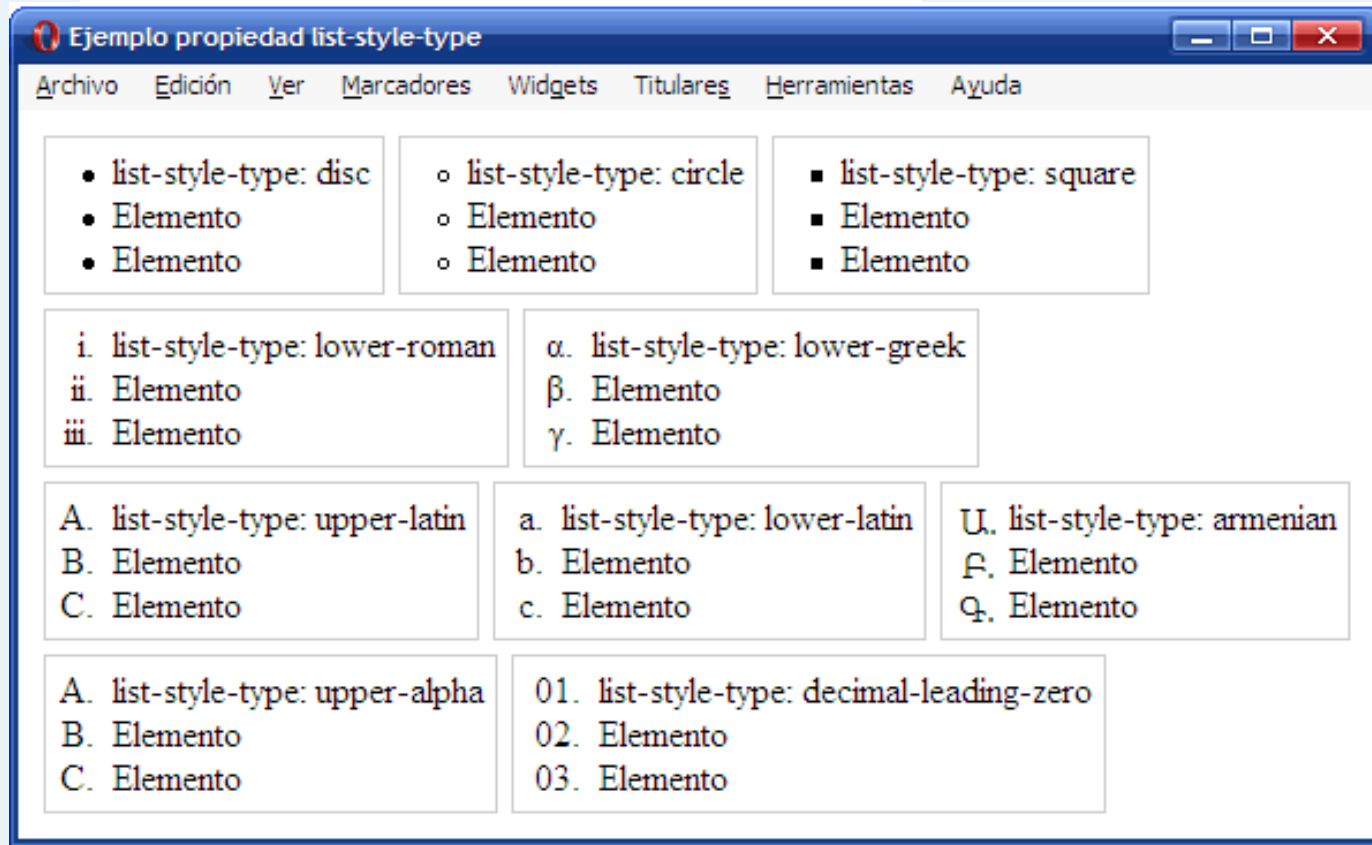
Eliminar el borde de las imágenes con enlaces:

```
img {  
border: none;  
}
```

Ejercicio 5 y 6

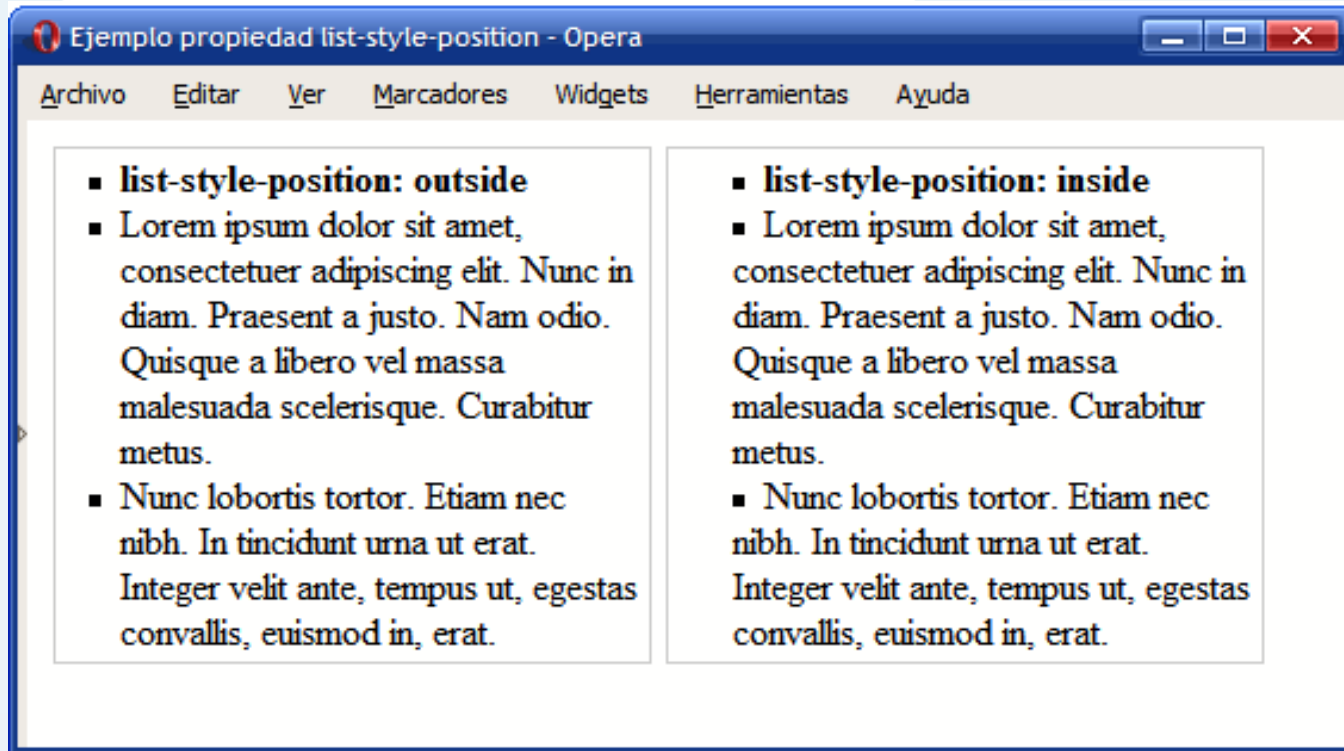
Listas

Viñetas personalizadas: list-style-type



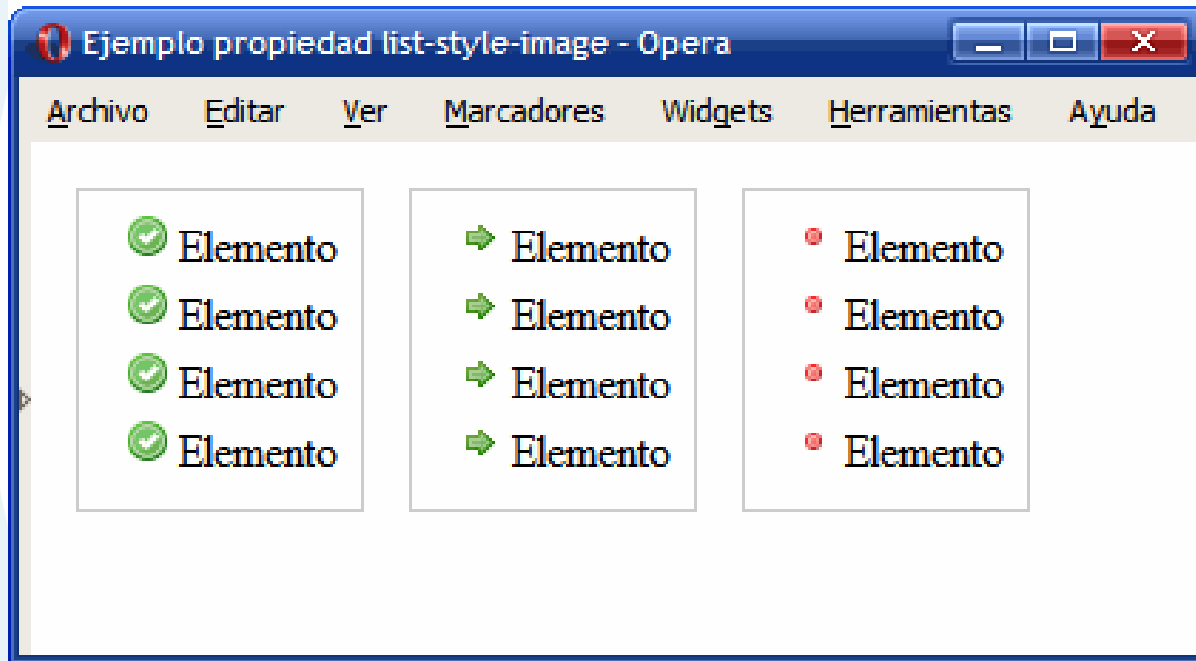
Listas

Viñetas personalizadas: list-style-position



Listas

Viñetas personalizadas: list-style-image



Listas

Viñetas personalizadas: list-style

Permite establecer todas las propiedades de una lista de forma directa.

`list-style: list-style-type || list-style-position || list-style-image`

No se debe mostrar ni viñetas automáticas ni viñetas personalizadas

`ul { list-style: none }`

*Cuando se utiliza una viñeta personalizada, es conveniente indicar la viñeta automática que se mostrará cuando no se pueda cargar la imagen:

`ul { list-style: url(imagenes/cuadrado_rojo.gif) square; }`

Listas

Menú vertical

Transformación de una lista sencilla de enlaces en un menú vertical de navegación.

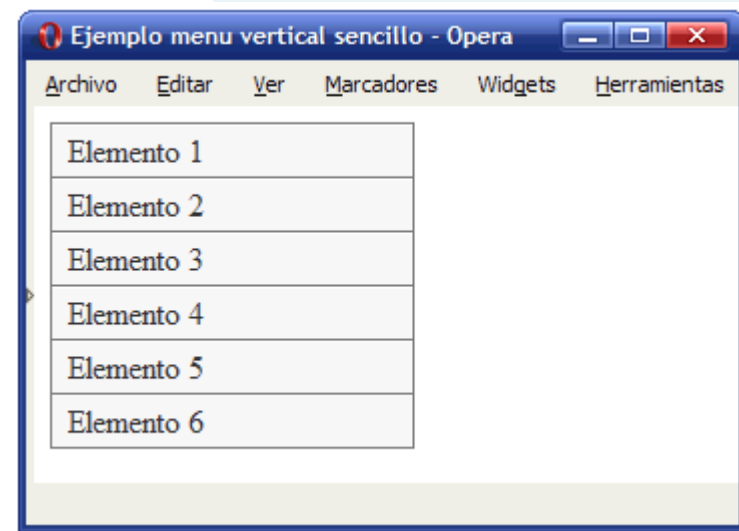
1. Definir la anchura del menú.
2. Eliminar las viñetas automáticas y todos los márgenes y espaciados aplicados por defecto.
3. Añadir un borde al menú de navegación y establecer el color de fondo y los bordes de cada elemento del menú.
4. Aplicar estilos a los enlaces: mostrarlos como un elemento de bloque para que ocupen todo el espacio de cada del menú, añadir un espacio de relleno y modificar los colores y la decoración por defecto

Listas

Menú vertical



Lista de enlaces original



Aspecto final del menú vertical
sencillo creado con CSS

Ejercicio 7

Listas

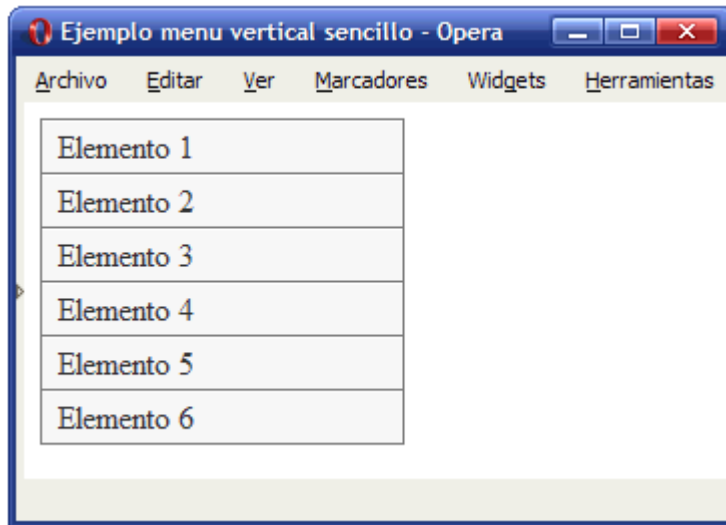
Menú horizontal

Transformación del menú vertical en un menú horizontal:

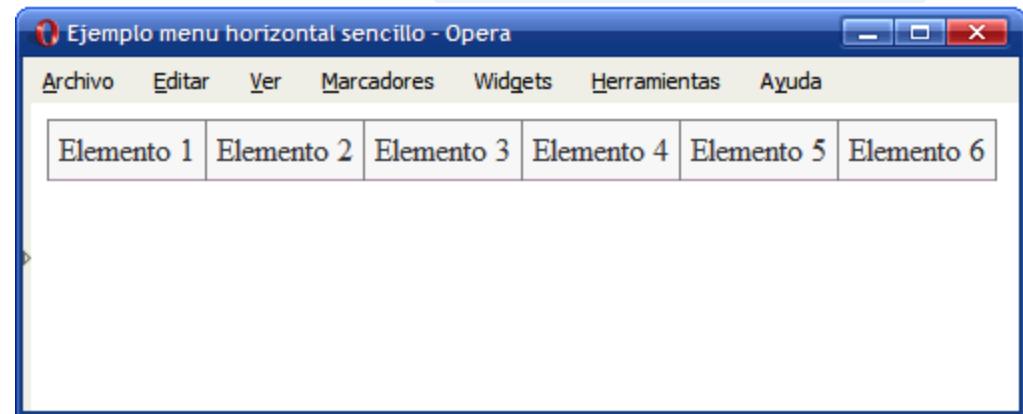
1. Eliminar la anchura y el borde del elemento `` y aplicar las propiedades `float` y `clear`.
2. Modificar la propiedad `float` de los elementos `` del menú.
3. Modificar el `padding` y los bordes de los enlaces que forman el menú.
4. Añade un borde izquierdo en el elemento `` para homogeneizar el aspecto de los elementos del menú.

Listas

Menú horizontal



Menú vertical sencillo creado
con CSS



Aspecto final del menú horizontal sencillo
creado con CSS

Listas

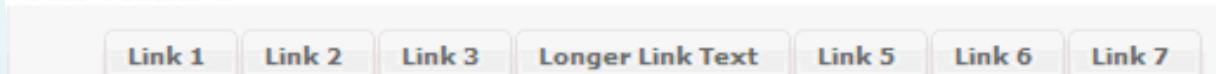
Menú horizontal con pestañas

Modificando los estilos de cada elemento del menú y utilizando imágenes de fondo y las pseudo-clases :hover y :active, se pueden crear menús horizontales complejos, incluso con el aspecto de un menú de solapas o pest

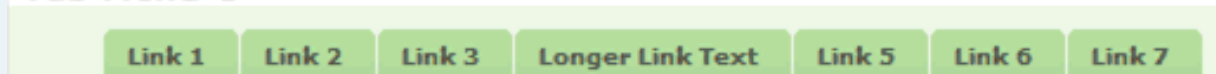
Tab Menu



Tab Menu B



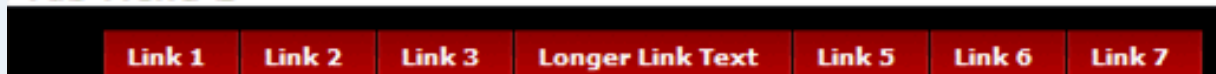
Tab Menu C



Tab Menu D



Tab Menu E



Ejercicio 7
(horizontal)

Tablas

Fusión de bordes

```
.normal {
width: 250px;
border: 1px solid #000;
}
.normal th, .normal td {
border: 1px solid #000;
}
```

```
.normal {
width: 250px;
border: 1px solid #000;
border-collapse: collapse;
}
.normal th, .normal td {
border: 1px solid #000;
}
```

Ejemplo propiedad border-spacing - Opera

A	B	C	D	E
a	1	2	3	4
b	1	2	3	4
c	1	2	3	4
d	1	2	3	4

Ejemplo propiedad border-collapse - Opera

A	B	C	D	E
a	1	2	3	4
b	1	2	3	4
c	1	2	3	4
d	1	2	3	4

Ejercicio 8

Formularios

Mejoras en los campos de texto

```
form.elegante input  
{  
  padding: .2em;  
}
```

Ejemplo padding en input

Archivo Edición Ver Marcadores Widgets Titulares Herramientas Ayu

Formulario sin padding en los input

Nombre
lorem ipsum

Contraseña

Formulario con padding en los input

Nombre
lorem ipsum

Contraseña

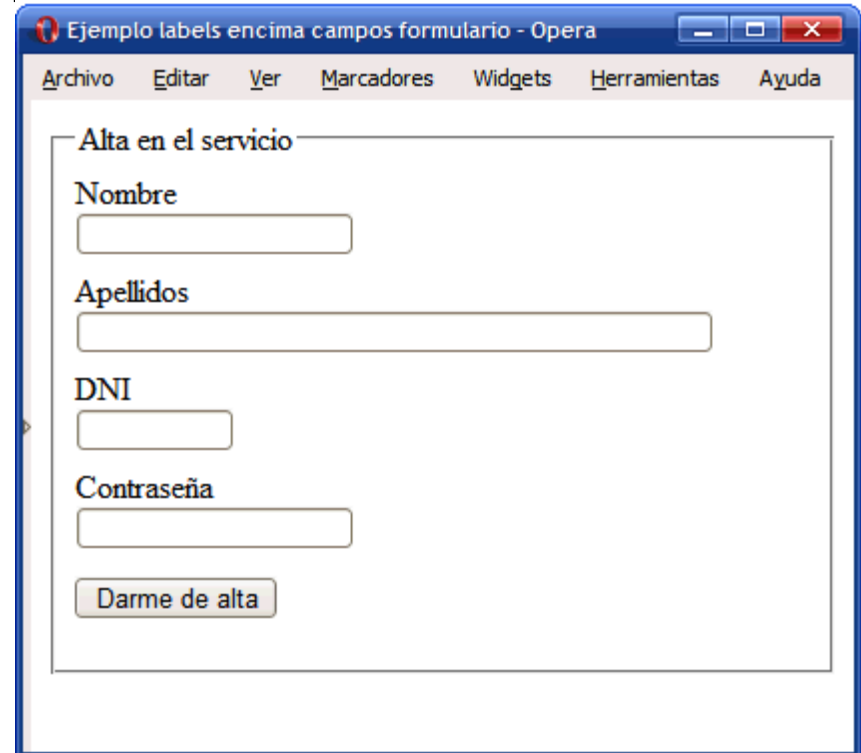
Mejorando el aspecto de los formularios gracias
al padding

Formularios

Labels alineadas y formateadas

```
label {  
display: block;  
margin: .5em 0 0 0;  
}
```

```
.btn {  
display: block;  
margin: 1em 0;  
}
```



Mostrando las etiquetas label encima de los campos del formulario

Formularios

Labels alineadas y formateadas

Ejemplo labels al lado de campos formulario - Opera

Archivo Editar Ver Marcadores Widgets Herramientas Ayuda

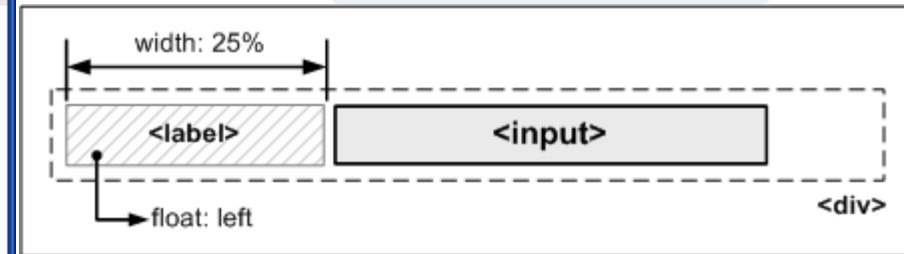
Alta en el servicio

Nombre

Apellidos

DNI

Contraseña



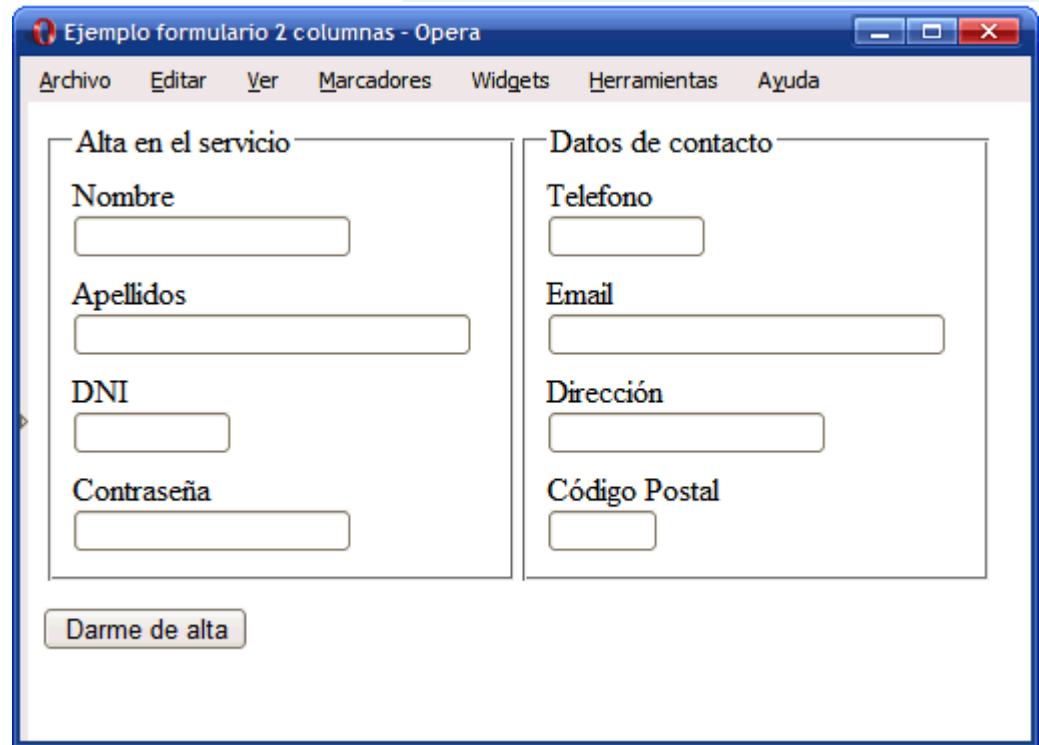
Esquema de la técnica de alineación de etiquetas label y campos de formulario

Mostrando las etiquetas label alineadas con los campos del formulario

Formularios

Formulario en varias columnas

```
form fieldset {  
  float: left;  
  width: 48%;  
}
```



Ejemplo formulario 2 columnas - Opera

Archivo Editar Ver Marcadores Widgets Herramientas Ayuda

Alta en el servicio

Nombre

Apellidos

DNI

Contraseña

Datos de contacto

Telefono

Email

Dirección

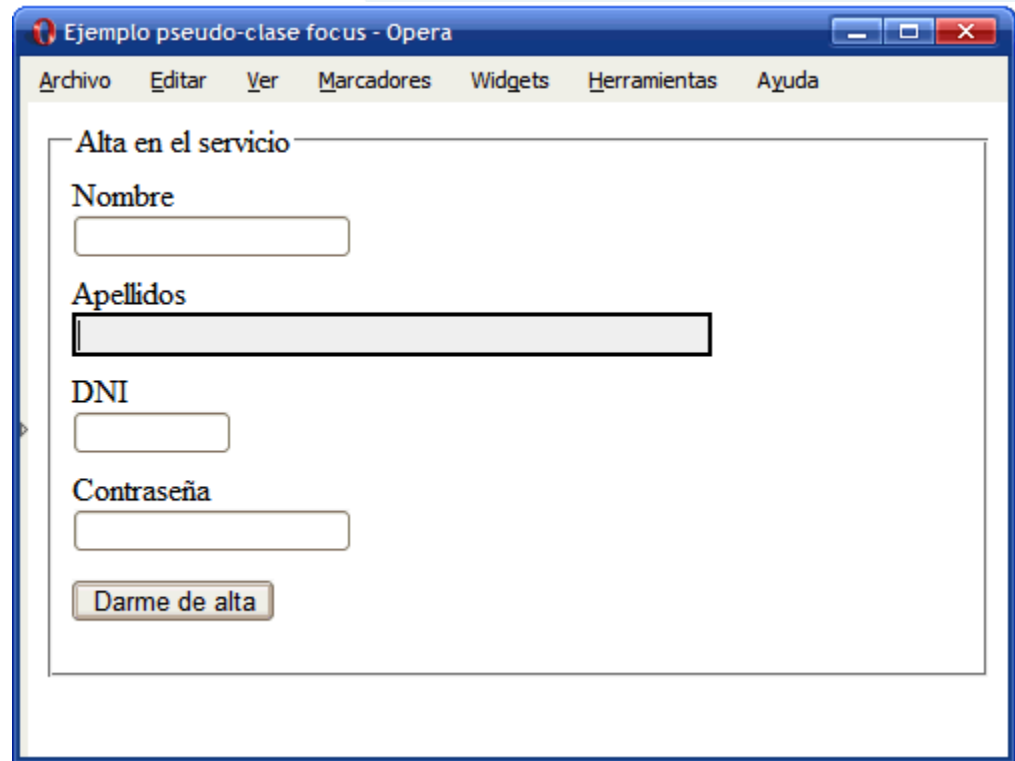
Código Postal

Ejemplo de formulario a dos columnas

Formularios

Resaltar el campo seleccionado

```
input:focus {  
  border: 2px solid  
  #000;  
  background: #F3F3F3;  
}
```



Ejemplo pseudo-clase focus - Opera

Archivo Editar Ver Marcadores Widgets Herramientas Ayuda

Alta en el servicio

Nombre

Apellidos

DNI

Contraseña

Ejemplo de pseudo-clase :focus Ejercicio 9