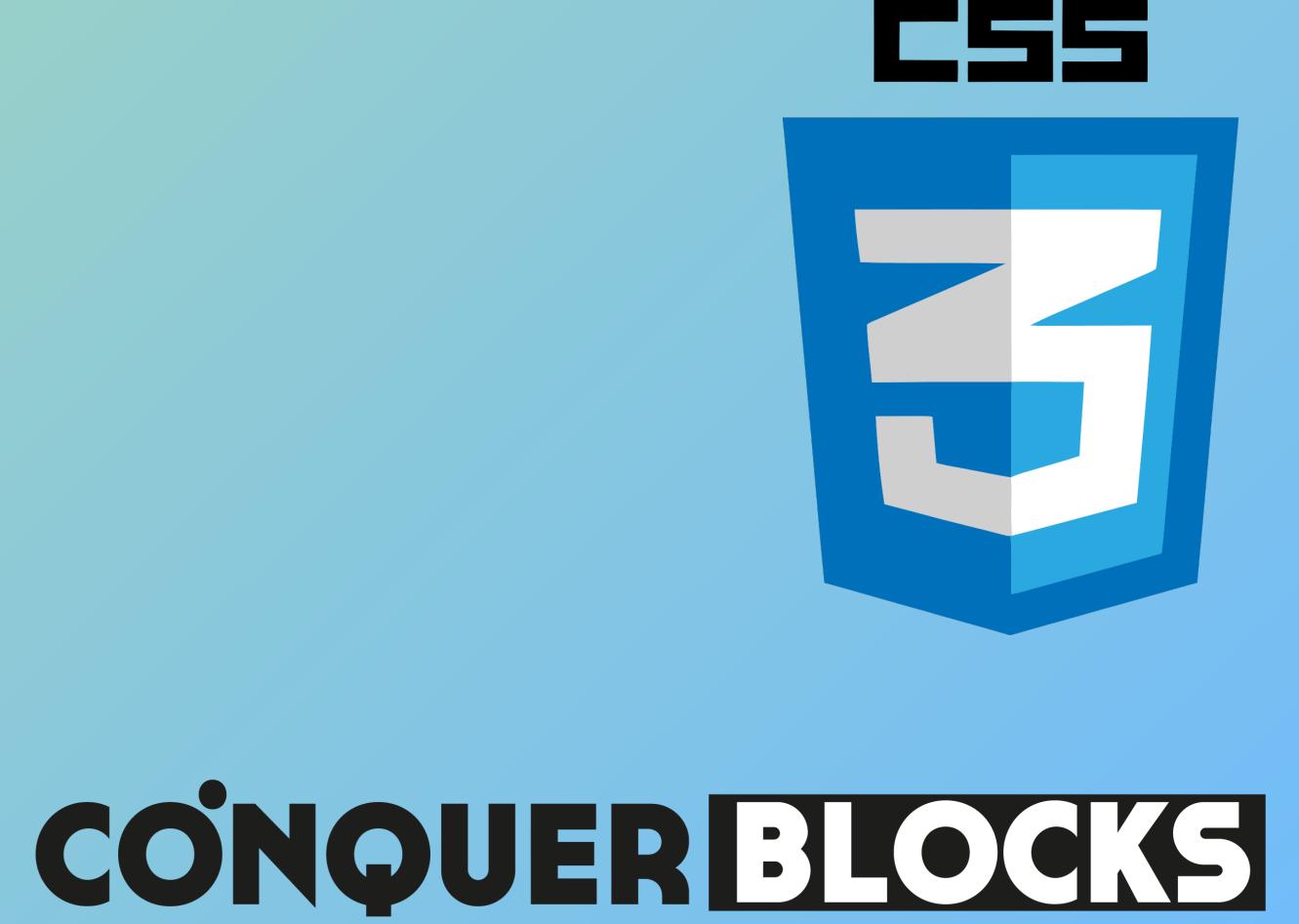


{css}

Clase 17



Clase 17

<índice>

Clase 17: Posicionamiento I

Introducción

Propiedad Display

Propiedad Position

Posicionamiento estático o normal

Propiedad Visibility

Display vs Opacity vs Visibility

Introducción

Introducción

Una de las partes más complejas de CSS sea la creación de layouts, colocación y distribución de los elementos de una página. Todo ha evolucionado en muy poco tiempo

Introducción

Sin embargo, es una parte fundamental dentro de CSS, y es necesario entenderla correctamente para que resulte más fácil de trabajar y crear nuestros diseños.

Introducción

Antes de comenzar, debemos saber que cada etiqueta HTML tiene un «tipo de representación visual» en el navegador, lo que habitualmente se suele denominar el tipo de caja.

Introducción

todos los elementos de una página web deben tener un tipo de representación, que es la forma en la que se va a comportar. En principio, partimos de dos tipos básicos y fundamentales llamados inline y block.

Introducción

¿Qué es inline y block?

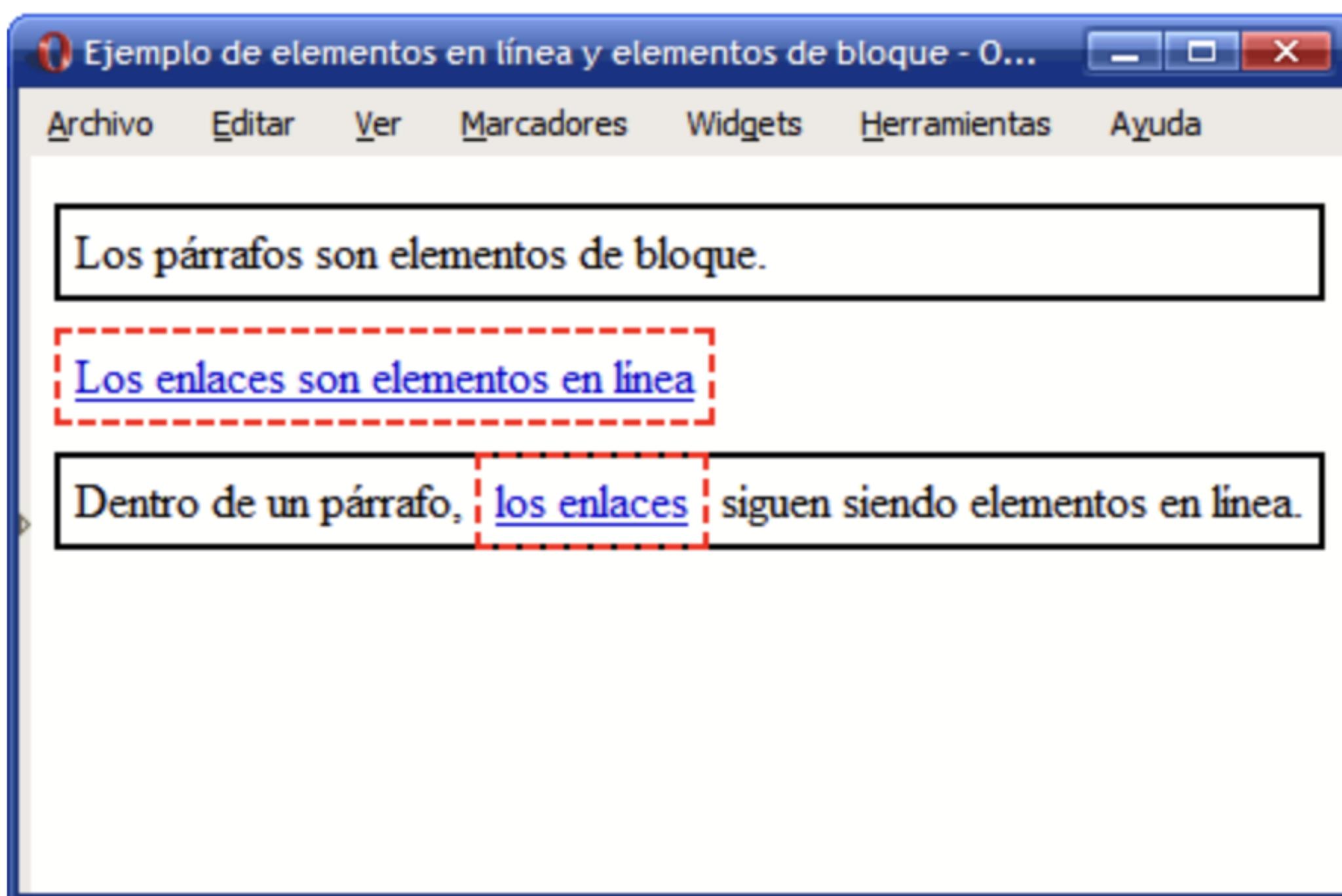


Figura 5.1 Cajas creadas por los elementos de línea y los elementos de bloque

Introducción

Cuando los navegadores descargan el contenido HTML y CSS de las páginas web, aplican un procesamiento muy complejo antes de mostrar las páginas en la pantalla del usuario.

Introducción

Para cumplir con el modelo de cajas presentado en el capítulo anterior, los navegadores crean una caja para representar a cada elemento de la página HTML

Introducción

Los factores que se tienen en cuenta para generar cada caja son:

- Las propiedades width y height de la caja
- El tipo de cada elemento HTML
- Posicionamiento de la caja (normal, relativo, absoluto, fijo o flotante)
- Las relaciones entre elementos (dónde se encuentra cada elemento, elementos descendientes, etc.)
- Otro tipo de información, como por ejemplo el tamaño de las imágenes y el tamaño de la ventana del navegador.

display

display

La propiedad `display` de CSS permite modificar el comportamiento de un elemento HTML, cambiándolo al que le indiquemos, como por ejemplo `inline` o `block` (u otro de los que veremos más adelante)

display

Tipo de caja	Características
block	Se apila en vertical. Ocupa todo el ancho disponible de su etiqueta contenedora.
Versión en línea	
inline	Se coloca en horizontal. Se adapta al ancho de su contenido. Ignora <code>width</code> o <code>height</code> .
inline-block	Combinación de los dos anteriores. Se comporta como <code>inline</code> pero no ignora <code>width</code> o <code>height</code> .

display

block

display

inline

display

inline-block

Si utilizamos la propiedad display con el valor inline-block, conseguiremos un elemento que funcionará como si fuera un elemento inline, pero haciendo caso a las propiedades width y height, las cuales se ignoran si estamos utilizando un display: inline.

display

Demo

display

vertical-align

nos ayudará a establecer el alineamiento vertical
de los elementos de línea

Demo

display

Mediante la propiedad display también podemos utilizar flex, grid, list-item o table.

Los veremos por separado.

display

none

position

position

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.

position

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.

position

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

Valor	Significado
static	Posicionamiento estático. Utiliza el orden natural de los elementos HTML.
relative	Posicionamiento relativo. Los elementos se mueven ligeramente en base a su posición estática.
absolute	Posicionamiento absoluto. Los elementos se colocan en base al contenedor padre.
fixed	Posicionamiento fijo. Idem al absoluto, pero aunque hagamos scroll no se mueve.
sticky	Posicionamiento «pegado». Similar al relativo, usado para pegar menús a la parte superior.

position

Posicionamiento normal o estático

Se trata del posicionamiento que utilizan los navegadores si no se indica lo contrario.

position

Posicionamiento relativo

variante del posicionamiento normal que consiste en posicionar una caja según el posicionamiento normal y después desplazarla respecto de su posición original.

position

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.

position

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.

position

Posicionamiento flotante

se trata del modelo más especial de posicionamiento, ya que desplaza las cajas todo lo posible hacia la izquierda o hacia la derecha de la línea en la que se encuentran

position

Posicionamiento sticky

Hará que en un contenedor con scroll el contenido se quede pegado en la parte de arriba

position

la propiedad position sólo indica cómo se posiciona una caja, pero no la desplaza.

position

Normalmente, cuando se posiciona una caja también es necesario desplazarla respecto de su posición original o respecto de otro origen de coordenadas. CSS define cuatro propiedades llamadas top, right, bottom y left para controlar el desplazamiento de las cajas posicionadas

position

Propiedad	Valor	Significado
top:	auto SIZE	Empuja el elemento una distancia desde la parte superior hacia el inferior.
bottom:	auto SIZE	Empuja el elemento una distancia desde la parte inferior hacia la superior.
left:	auto SIZE	Empuja el elemento una distancia desde la parte izquierda hacia la derecha.
right:	auto SIZE	Empuja el elemento una distancia desde la parte derecha hacia la izquierda.

Posicionamiento normal o estático

Posicionamiento normal

El posicionamiento normal o estático es el modelo que utilizan por defecto los navegadores para mostrar los elementos de las páginas.

Posicionamiento normal

En este modelo, sólo se tiene en cuenta si el elemento es de bloque o en línea, sus propiedades width y height y su contenido.

Posicionamiento normal

Los elementos de bloque forman lo que CSS denomina "contextos de formato de bloque". En este tipo de contextos, las cajas se muestran una debajo de otra comenzando desde el principio del elemento contenedor. La distancia entre las cajas se controla mediante los márgenes verticales.

Posicionamiento normal

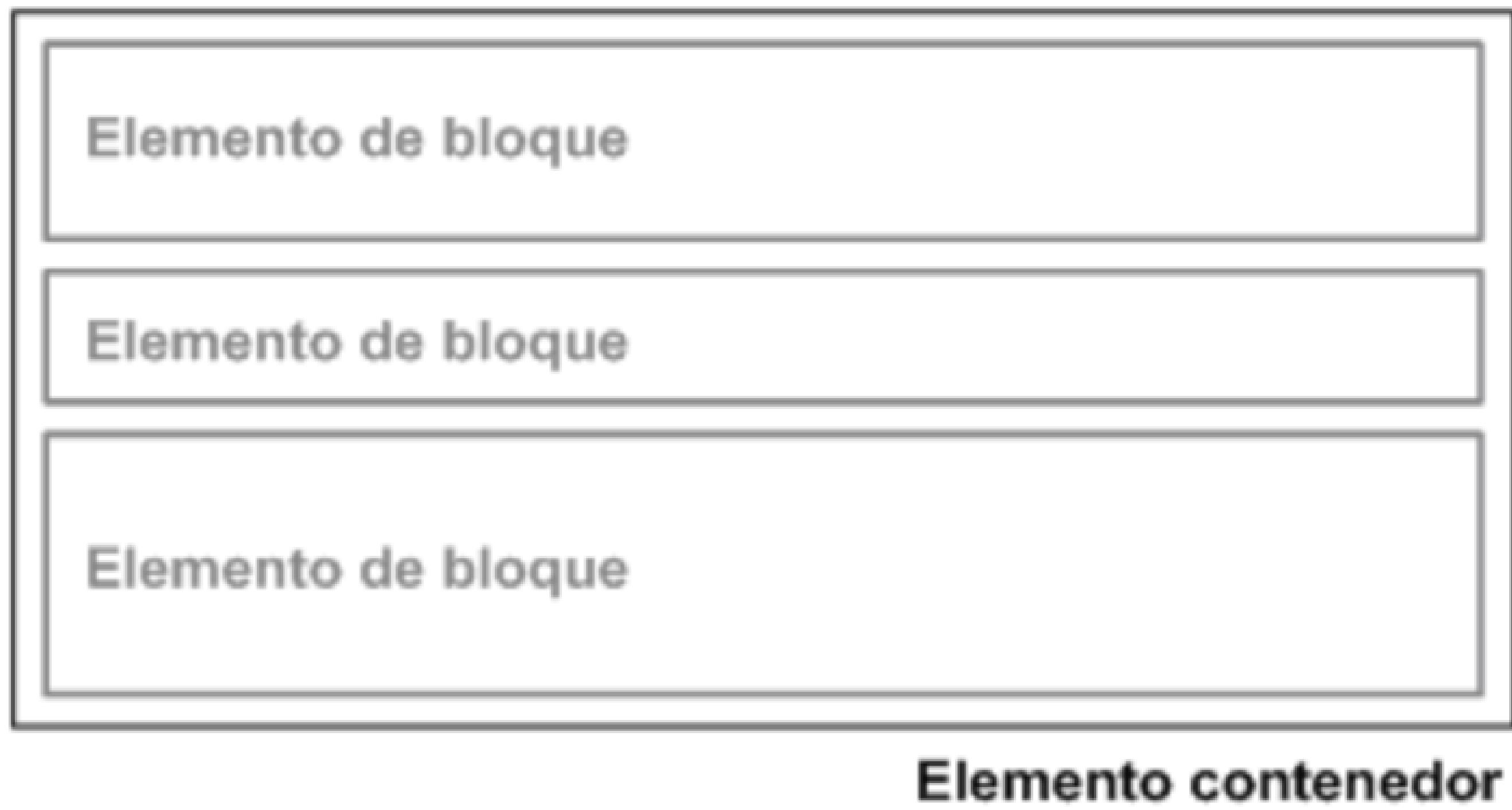
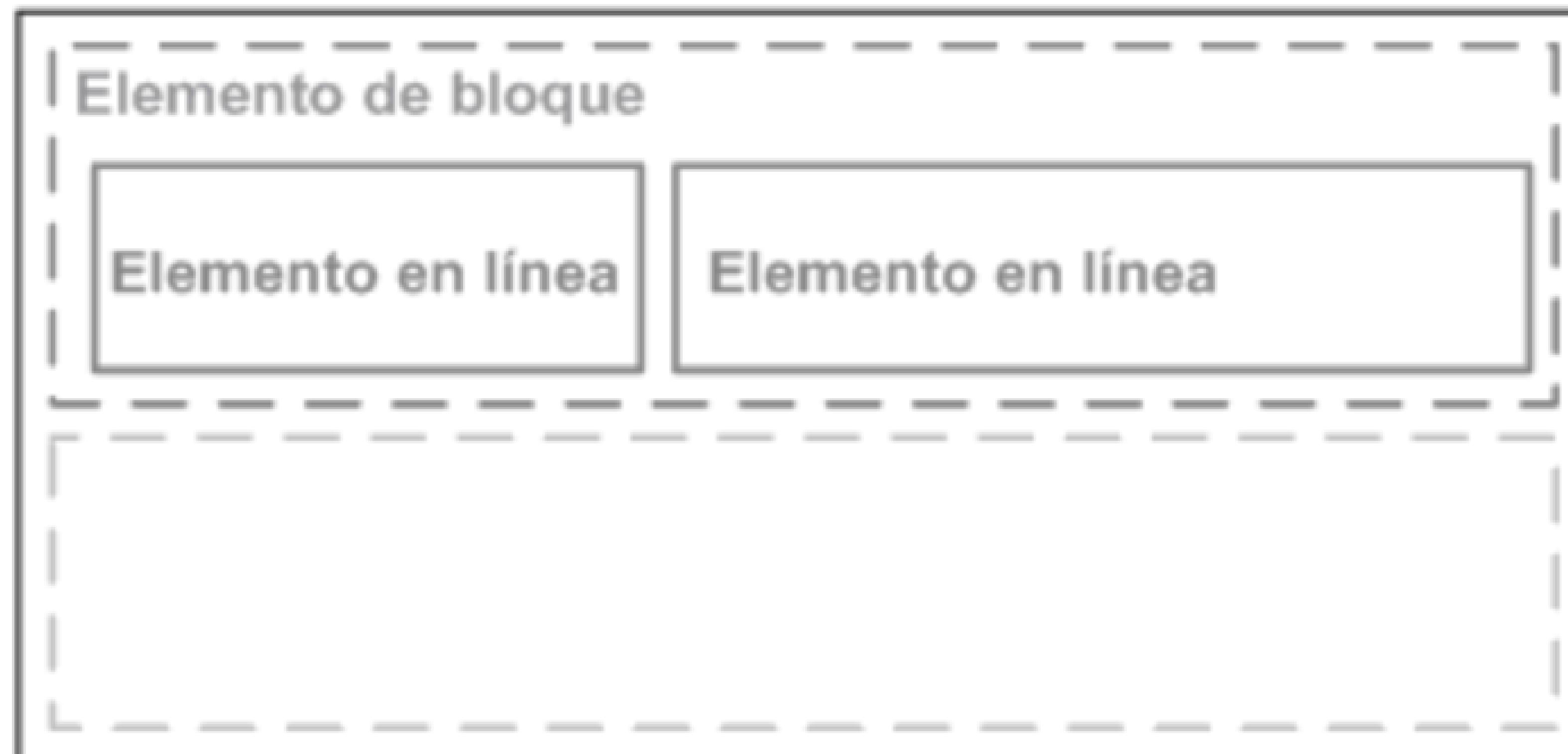


Figura 5.2 Posicionamiento normal de los elementos de bloque

Posicionamiento normal



Elemento contenedor

Posicionamiento normal

Si un elemento se encuentra dentro de otro, el elemento padre se llama "elemento contenedor" y determina tanto la posición como el tamaño de todas sus cajas interiores.

Posicionamiento normal

Si un elemento no se encuentra dentro de un elemento contenedor, entonces su elemento contenedor es el elemento <body> de la página.

Normalmente, la anchura de los elementos de bloque está limitada a la anchura de su elemento contenedor, aunque en algunos casos sus contenidos pueden desbordar el espacio disponible.

Posicionamiento normal

¿Os suena verdad? Es el comportamiento que llevamos viendo desde el principio, pero ahora sabéis exactamente por qué se posicionaban los elementos así

visibility

visibility

Las propiedades `display` y `visibility` controlan la visualización de los elementos. Las dos propiedades permiten ocultar cualquier elemento de la página

visibility

Habitualmente se utilizan junto con JavaScript para crear efectos dinámicos como mostrar y ocultar determinados textos o imágenes cuando el usuario pincha sobre ellos.

visibility

La propiedad `visibility` permite hacer invisible un elemento, lo que significa que el navegador crea la caja del elemento pero no la muestra.

visibility

La propiedad `visibility` permite hacer invisible un elemento, lo que significa que el navegador crea la caja del elemento pero no la muestra.

visibility

La propiedad `visibility` permite hacer invisible un elemento, lo que significa que el navegador crea la caja del elemento pero no la muestra.

visibility

Value	Description
visible	Default value. The element is visible
hidden	The element is hidden (but still takes up space)

visibility

Inicialmente todas las cajas que componen la página son visibles. Empleando el valor hidden es posible convertir una caja en invisible para que no muestre sus contenidos.

visibility

El resto de elementos de la página se muestran **como si la caja todavía fuera visible**, por lo que en el lugar donde originalmente se mostraba la caja invisible, ahora se muestra un hueco vacío.

Display vs Opacity vs Visibility

Display vs Opacity vs Visibility

Siempre existen diferentes formas de
hacer la misma cosa

Display vs Opacity vs Visibility

No siempre tienen el mismo efecto

Display vs Opacity vs Visibility

```
visibility: hidden;  
opacity: 0;  
display: none;
```

Ejemplos

Ejemplos

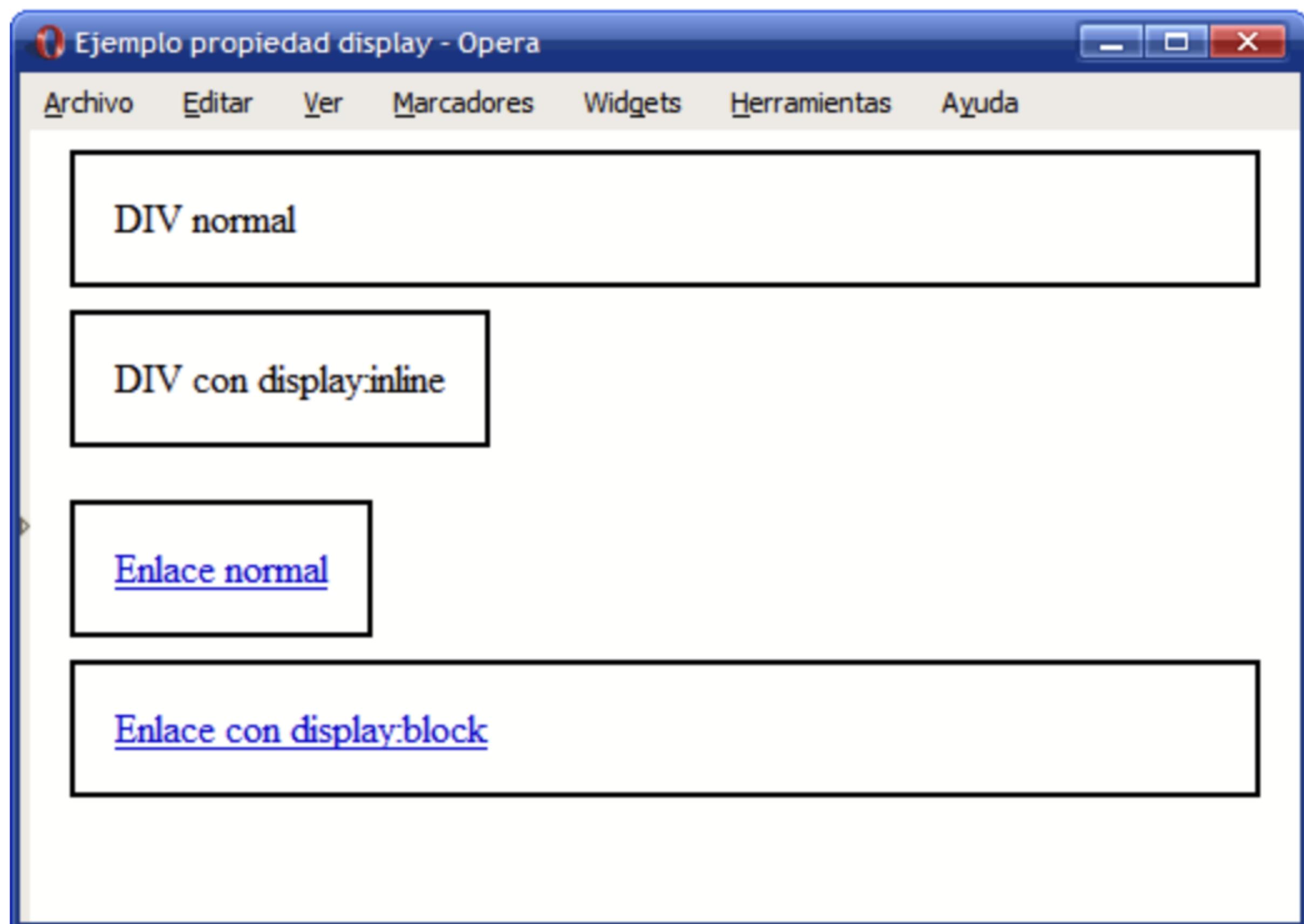


Figura 5.22 Ejemplo de propiedad display

Ejemplos

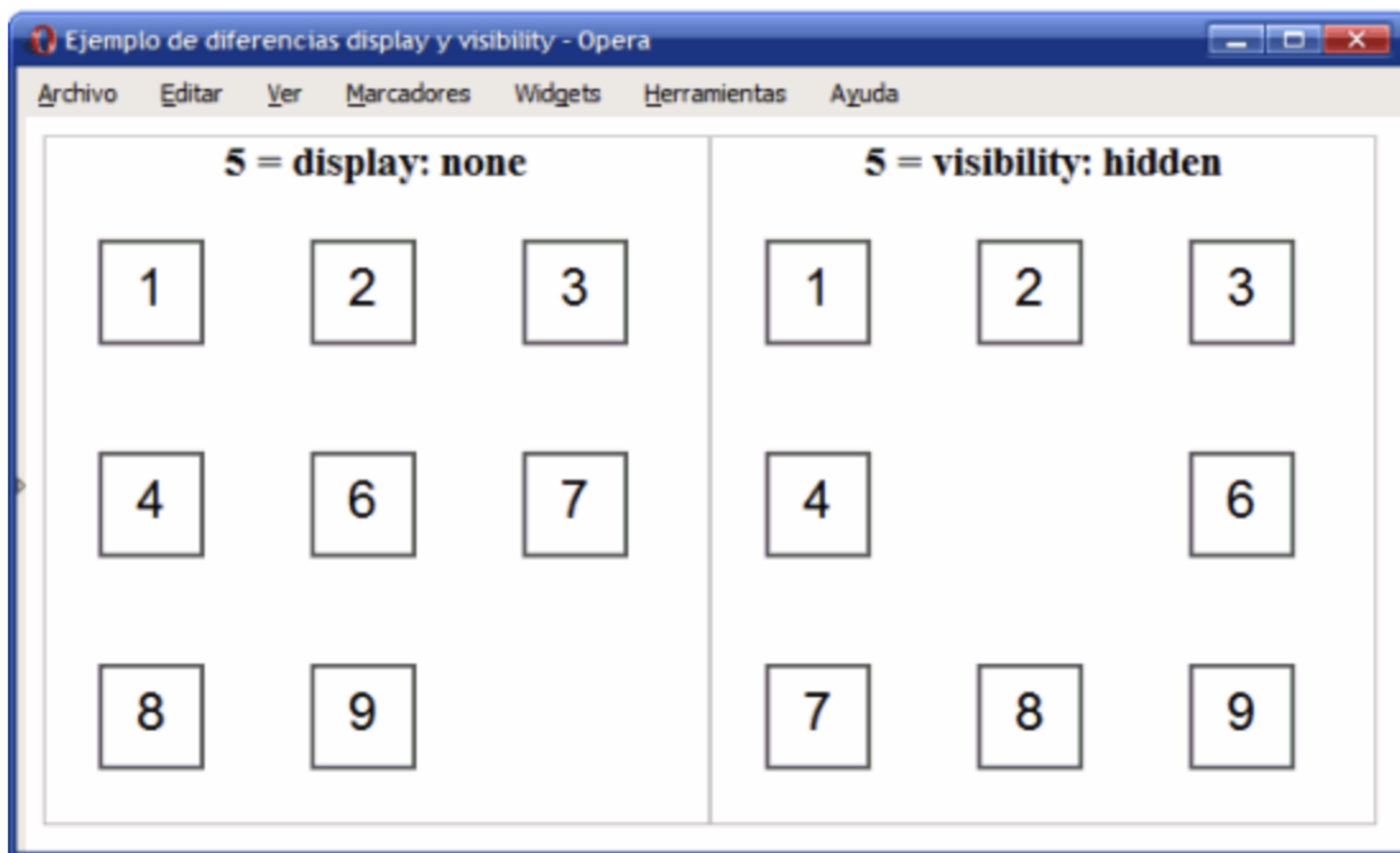


Figura 5.21 Diferencias visuales entre las propiedades display y visibility

<Despedida>

Email

bienvenidosaez@gmail.com

Instagram

@bienvenidosaez

Youtube

youtube.com/bienvenidosaez

CONQUERBLOCKS