

# Posicionamientos

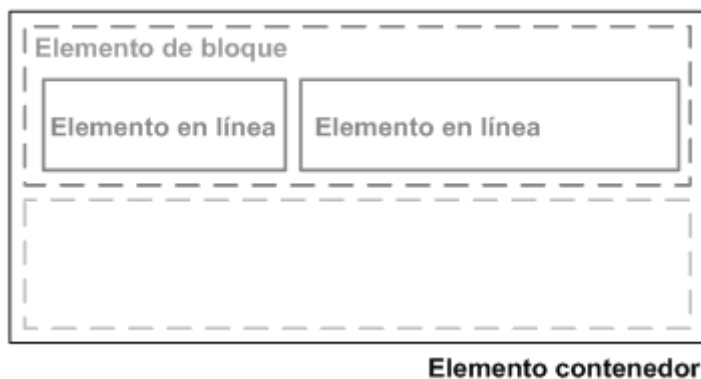
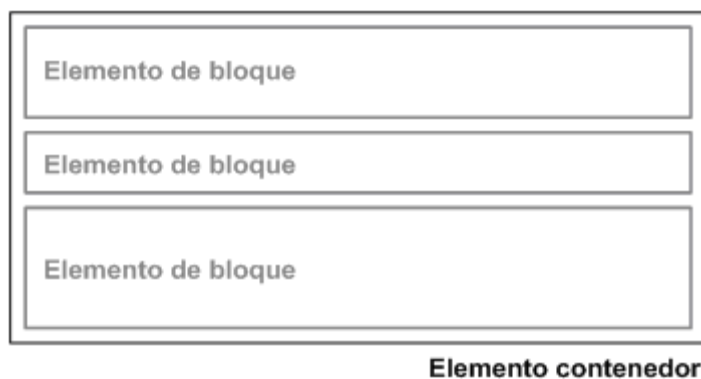
## Definición

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, permitiendo 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:

### **position: static** (dentro del flujo)

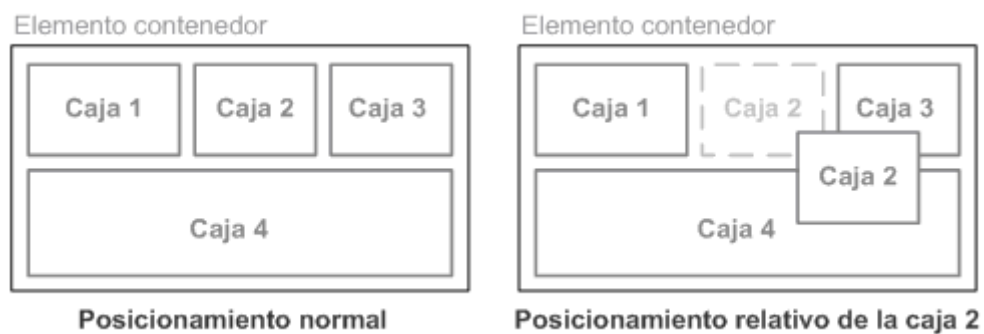
Corresponde al posicionamiento normal o estático, que responde al flujo de HTML. No admite las propiedades top, right, bottom y left.



## position: relative (dentro del flujo)

El desplazamiento de la caja se controla con las propiedades top, right, bottom y left, y a pesar de producir un desplazamiento, la posición inicial que debiera ocupar en el plano se mantiene intacta.

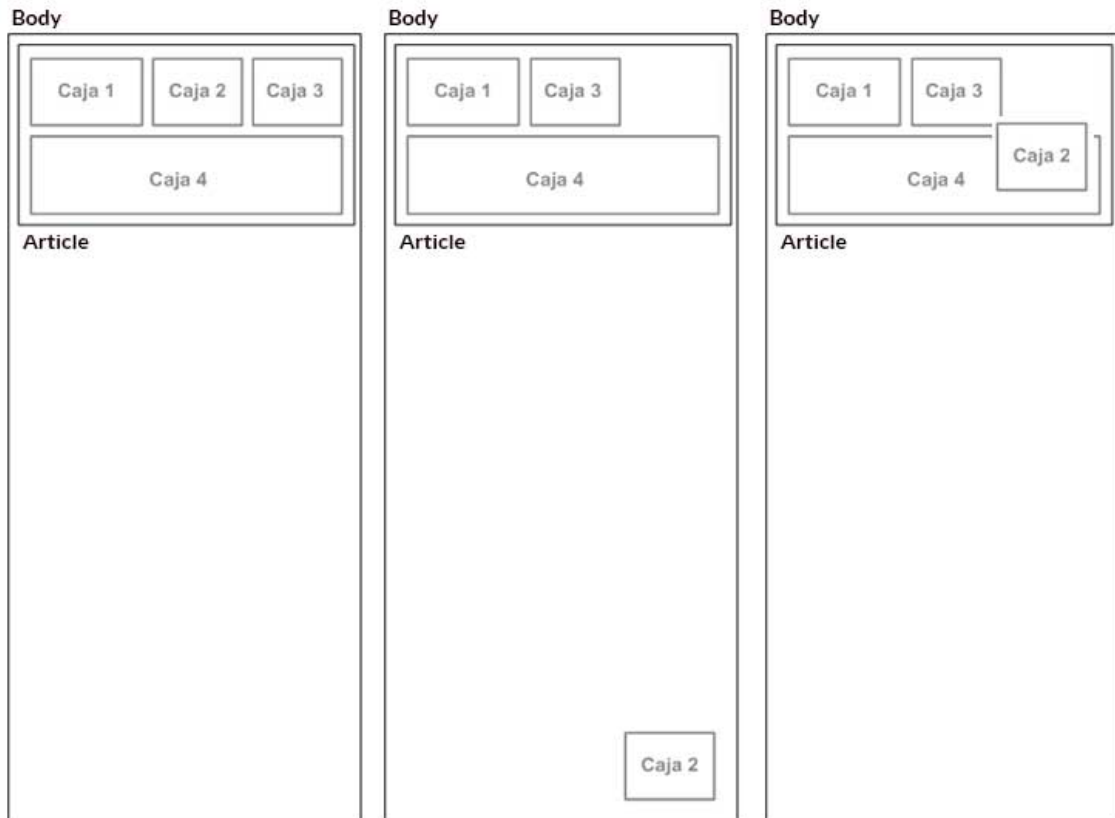
```
.caja2 {  
    position: relative;  
    top: 50px;  
    left: 60px;  
}
```



Hace de contenedor de referencia para los elementos en posición absoluta.

## position: absolute (fuera del flujo)

La posición exacta (no el desplazamiento) de la caja se controla con las propiedades top, right, bottom y left, pero su interpretación es mucho más compleja, ya que el origen de coordenadas del desplazamiento depende del posicionamiento de su elemento contenedor, siendo el <body> por defecto, o de haberlo, su elemento contenedor en posición relativa.



```
.caja2 {  
    position: absolute;  
    bottom: 50px;  
    right: 50px;  
}
```

```
.caja2 {  
    position: absolute;  
    bottom: 50px;  
    right: 50px;  
}  
  
article {  
    position: relative;  
}
```

**position: fixed**      (fuera del flujo)

El desplazamiento se establece de la misma forma que en el posicionamiento absoluto, pero en este caso el elemento permanece inamovible en la pantalla, omitiendo el scroll.

## float: left / right (fuera del flujo)

Cuando una caja se posiciona con el modelo de posicionamiento flotante, automáticamente se convierte en una caja flotante, lo que significa que se desplaza hasta la zona más a la izquierda o más a la derecha de la posición en la que originalmente se encontraba.

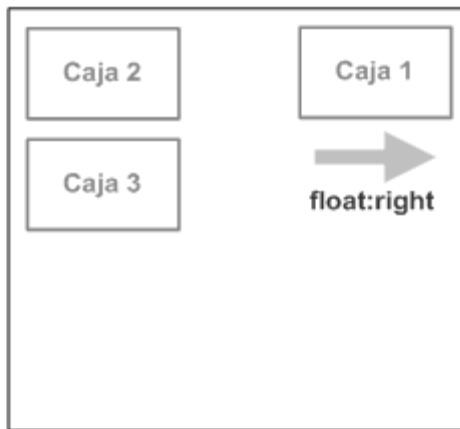
```
.caja1 {  
    float: right;  
}
```

Elemento contenedor



Posicionamiento normal

Elemento contenedor



Posicionamiento float de la caja 1

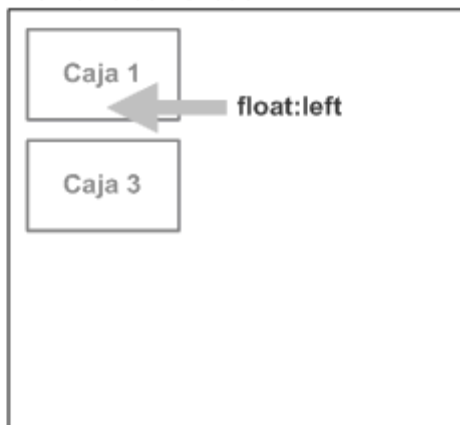
```
.caja1 {  
    float: left;  
}
```

Elemento contenedor



Posicionamiento normal

Elemento contenedor

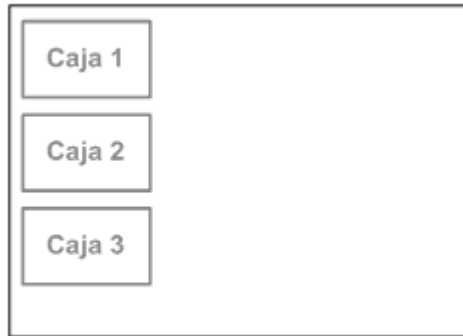


Posicionamiento float de la caja 1

Las cajas no se superponen entre sí porque las cajas flotantes tienen en cuenta las otras cajas flotantes existentes:

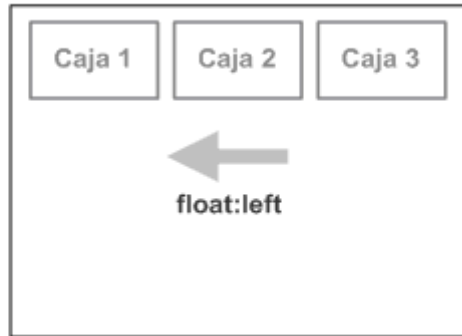
```
.caja1, .caja2, .caja3 {  
    float: left;  
}
```

Elemento contenedor



Posicionamiento normal

Elemento contenedor



Posicionamiento float de las 3 cajas

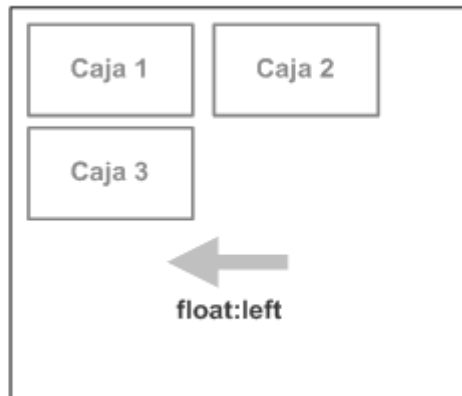
Si no existe sitio en la línea actual, la caja flotante baja a la línea inferior hasta que encuentra el sitio necesario para mostrarse lo más a la izquierda o lo más a la derecha posible en esa nueva línea:

Elemento contenedor



Posicionamiento normal

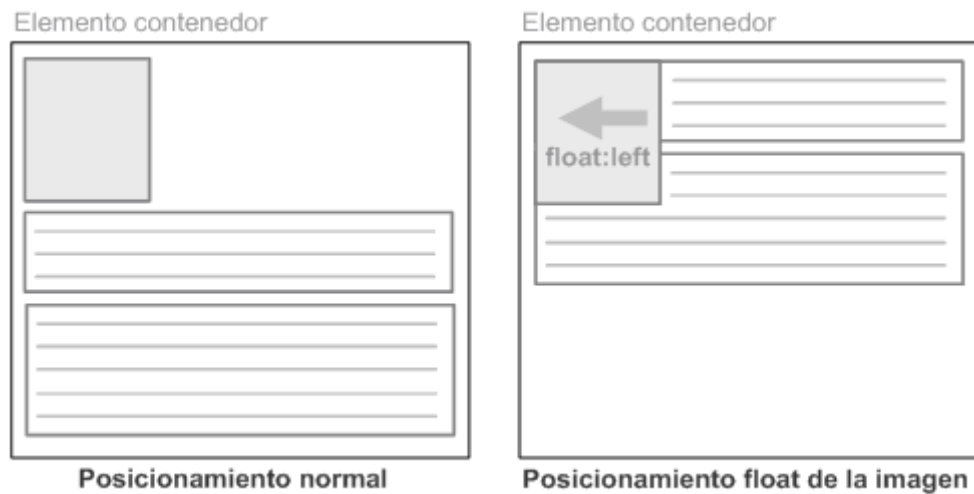
Elemento contenedor



Posicionamiento float de las 3 cajas

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

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



La salida de los elementos flotantes del flujo supone un problema conocido con la altura de sus contenedores:

```
.anterior {  
    float: left;  
}  
.siguiente {  
    float: right;  
}
```

[« Anterior](#)

[Siguiente »](#)

Una de las formas de resolverlo es a través de la propiedad overflow, con su valor hidden:

```
.anterior {  
    float: left;  
}  
.siguiente {  
    float: right;  
}  
.contenedor {  
    overflow: auto;  
}
```

« [Anterior](#)

[Siguiente](#) »