

Maquetación y animaciones Web mediante HTML5 y CSS3

Modelos de cajas

Módulo: Desarrollo de Interfaces Web

CFGS Desarrollo de Aplicaciones Web

Cajas - Dimensionables

- La propiedad `resize` está en CSS3 pero no está soportado por todos los navegadores (por ejemplo, Internet Explorer).
- Para utilizar esta propiedad el atributo `overflow` debe tomar un valor diferente de `visible` (`hidden`, `scroll` o `auto`).
- Los valores que puede tomar `resize` son: `none`, `both`, `horizontal` y `vertical`.
- Las propiedades `height` y `width` determinan el alto y el ancho del área de contenido.

Cajas - Dimensionables

La propiedad `box-sizing` permite modificar el comportamiento de las cajas. Los valores que puede tomar son:

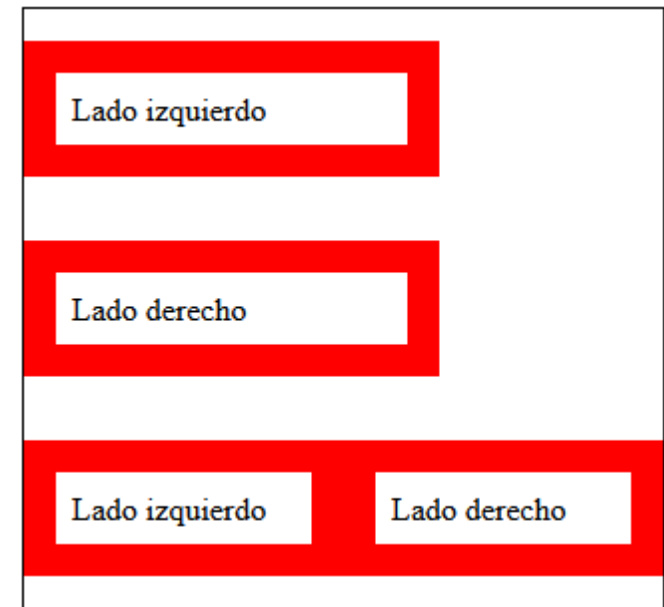
- ✓ `content-box`. Valor por omisión. La caja se comporta como en el modelo de caja tradicional.
- ✓ `border-box`. Las medidas hacen referencia a las medidas del contenido, el padding y el borde.
 - A la propiedad `width` se le resta el ancho del borde y del padding.

www.w3.org/TR/css3-ui/#pseudo-validity

Ver el manual de la w3schools: http://www.w3schools.com/css/css3_box-sizing.asp

Cajas - Dimensionables

```
div.box{width:20em;
  border:thin black solid;}
.box p:nth-of-type(1), .box p:nth-of-type(2){
  width:50%;
  padding:0.5em;
  border:1em solid red;
  float:left;}
.box p:nth-of-type(3), .box p:nth-of-type(4){
  -moz-box-sizing:border-box;
  box-sizing:border-box;
  width:50%;
  padding:0.5em;
  border:1em solid red;
  float:left;}
...
<div class="box">
  <p>Lado izquierdo</p>
  <p>Lado derecho</p>
  <p>Lado izquierdo</p>
  <p>Lado derecho</p>
  <div></div>
</div>
```

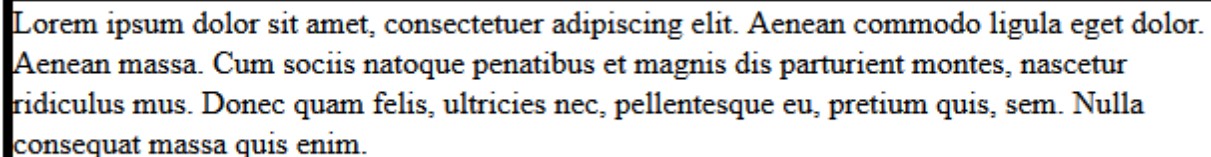


Cajas - Contornos

La propiedad outline permite trazar contornos alrededor de un elemento. En este caso, el contorno no computa dentro de las dimensiones del elemento.

En CSS3 se incluye una propiedad outline-offset que permite separar el contorno del borde.

```
p.conreborde{  
    margin:2em;  
    border: black solid thick;  
    outline-offset: 1em;  
    outline: red solid thick;  
}
```



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.

Cajas - Efectos sobre texto

Cuando el texto no cabe en una caja, la renderización depende de la propiedad `overflow`. En este caso, puede desbordarse, recortarse o mostrar barras de desplazamiento.

La propiedad `text-overflow` permite añadir nuevos comportamientos con los valores:

- `clip`. Corta el texto. Igual que `overflow:hidden`.
- `ellipsis`. Añade puntos suspensivos al texto antes de cortarlo.
- `cadena`. Añade cadena al texto antes de cortarlo. No soportado por todos los navegadores.

Cuando una palabra no cabe en el contenedor, la palabra desborda la caja. La propiedad `word-wrap` con el valor `break-word` hace que la palabra se corte.

www.w3.org/TR/css3-text/

Cajas - Efectos sobre textos

Revisar el manual de la W3Schools sobre los efectos sobre textos:

http://www.w3schools.com/css/css3_text_effects.asp

Cajas - Flexbox

El modelo de cajas flexible permite crear un layout flexible de la página sin necesidad de utilizar las propiedades width, float, etc. Es decir, permite colocar los elementos de una página para que se comporten de forma predecible cuando el diseño de la página debe acomodarse a diferentes tamaños de pantalla y diferentes dispositivos.

www.w3.org/TR/css3-flexbox/

Puede ser un sustituto de los framework CSS basados en retículas. El problema es que sólo es soportado por las últimas versiones de los navegadores.

La propiedad display tiene ahora dos nuevos valores:

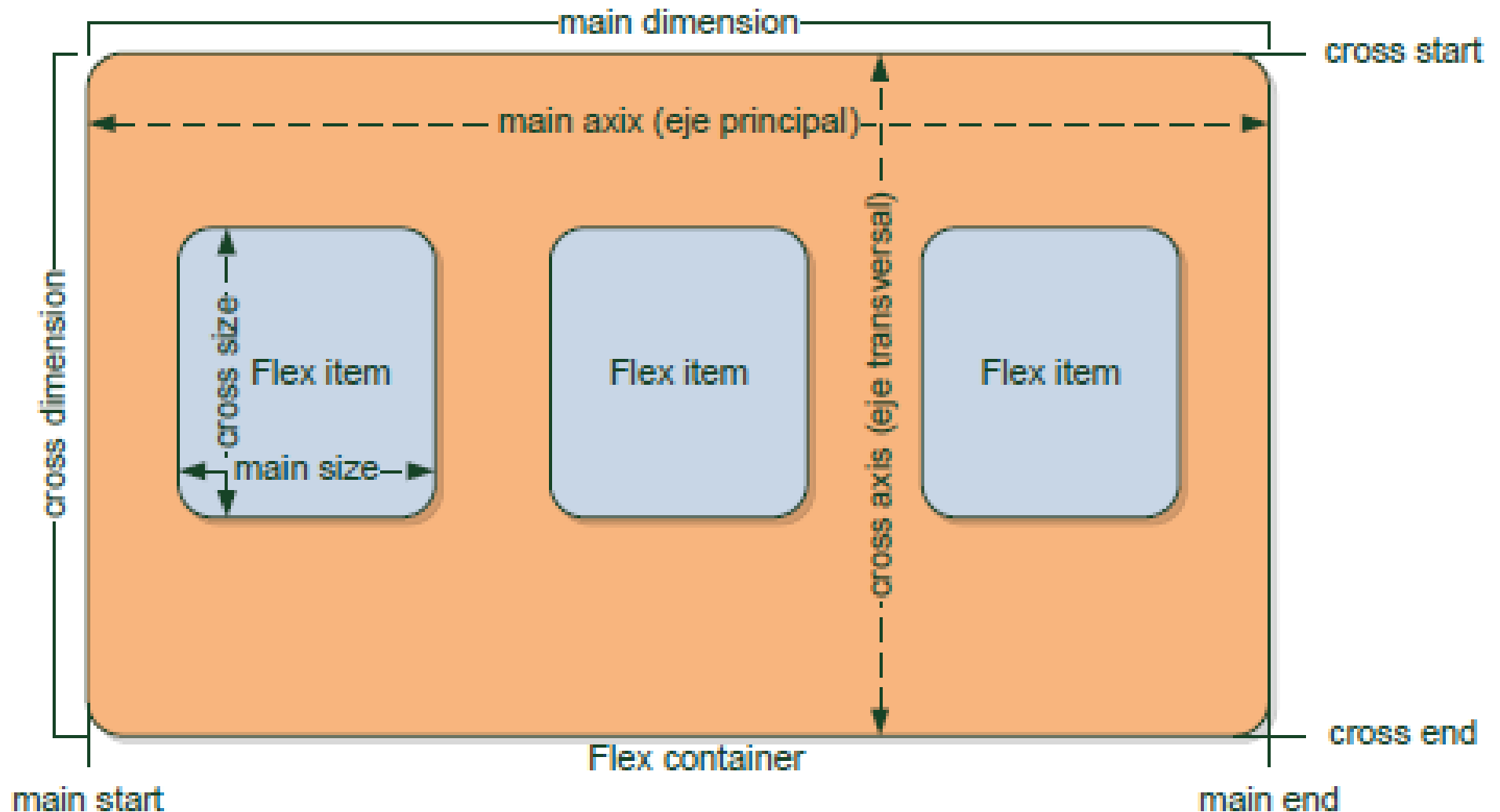
- flex. Genera una caja flexible como elemento de bloque.
- inline-flex. Genera una caja flexible como elemento en línea.
- Sus dimensiones se adaptan al tamaño de la ventana.

Cajas - Flexbox

- Los elementos hijos serán lo que se llama flex items que se adaptarán al contenedor padre.
- Los elementos hijos pueden colocarse en cualquier dirección y puede tener dimensiones flexibles para adaptarse al espacio visible.
- En los elementos con la propiedad display a block o inline, el posicionamiento se calcula a partir de las direcciones del flujo de los elementos de bloque o en línea.
- En el modelo de cajas flexible, el posicionamiento se calcula a partir de las direcciones del flujo flexible.
- Se puede redefinir el sentido del flujo de los elementos (hacia arriba, hacia abajo, hacia la izquierda o hacia la derecha).
- Alienar los elementos respecto al padre o respecto a sus hermanos.

Cajas - Flexbox

En el modelo de cajas flexible, el posicionamiento se calcula a partir de las direcciones del flujo flexible.



Cajas - Flexbox

Contenedor flexible (flex-container): El elemento "padre" que contiene los elementos flexibles. Un contenedor flexible se define usando los valores flex o inline-flex en la propiedad display.

Las propiedades de columnas no se aplican al flex-container.

Elemento flexible (flex-items): Cada hijo de un contenedor flex se convierte en un elemento flexible. Si hay texto directamente incluido en el contenedor flexible, se envuelve automáticamente en un elemento flexible anónimo. Además a los flex-items se les puede asignar display:flex con lo que se definiría un nuevo contexto en cada uno para sus hijos.

Las propiedades float, clear, vertical-align, no se aplican a los flex-items.

Cajas - Flexbox

Ejes: Cada diseño de "caja flexible" sigue dos ejes. El eje principal (**main axis**) es el eje a lo largo del cual los elementos flexibles se suceden unos a otros. El eje secundario (**cross axis**) es el eje perpendicular al eje principal.

Direcciones: Los lados inicio principal/fin principal (main start/main end) e inicio secundario/fin secundario (cross start/cross end) del contenedor flexible describen el origen y final del flujo de los elementos flexibles.

Dimensiones: Los términos equivalentes a "altura" y "anchura" usados en los elementos flexibles son tamaño principal (main size) and tamaño secundario (cross size), que respectivamente siguen al eje principal y al eje secundario del contenedor flexible. Si el eje principal fuera el horizontal, el main size sería el width y el cross size el height; si el eje principal fuera el vertical, las propiedades anteriores se intercambiarían.

Cajas - Flexbox

Para crear una caja flexible hay que crear un contenedor con la propiedad `display a flex` o `inline-flex`.








- Crean un contenedor en el que la propiedad `float` de los hijos no interfieren en el posicionamiento y los márgenes del contenedor no se mezclan con los del contenido.

Dentro del contenedor se situarán sus elementos flexibles.

- Cada hijo será un elemento flexible.
- En los elementos flexibles, las propiedades `min-width` y `min-height` admiten un valor `auto`. El valor `auto` asegura que el elemento sea lo suficientemente largo como para que encaje su contenido.

Cajas - Flexbox

Dirección y ajuste

- La propiedad flex-direction del contenedor establece el sentido en que se colocan los elementos flexibles. Puede tomar los valores:
 -  row (valor por defecto). Los elementos se disponen en línea.
 -  column. Los elementos se disponen en columna.
 -  row-reverse. Los elementos se disponen en línea en sentido inverso.
 -  column-reverse. Los elementos se disponen en columna en sentido inverso.
- La propiedad flex-wrap del contenedor establece el ajuste de los elementos. Puede tomar los valores:
 -  nowrap (valor por defecto). No se realiza el ajuste. Si los elementos no caben, sobrepasan la caja del contenedor.
 -  wrap. Si los elementos no caben se ajustan en varias filas (o columnas).
 -  wrap-reverse. Los elementos hacen ajuste, pero comienzan a ajustarse de bajo a arriba.



Cajas - Flexbox

El contenedor tiene flex-direction:row

Flex item 1

Flex item 2

Flex item 3

El contenedor tiene flex-direction:row-reverse

Flex item 3

Flex item 2

Flex item 1

Cajas - Flexbox

El contenedor tiene flex-direction:column

Flex item 1

Flex item 2

Flex item 3

El contenedor tiene flex-direction:column-reverse

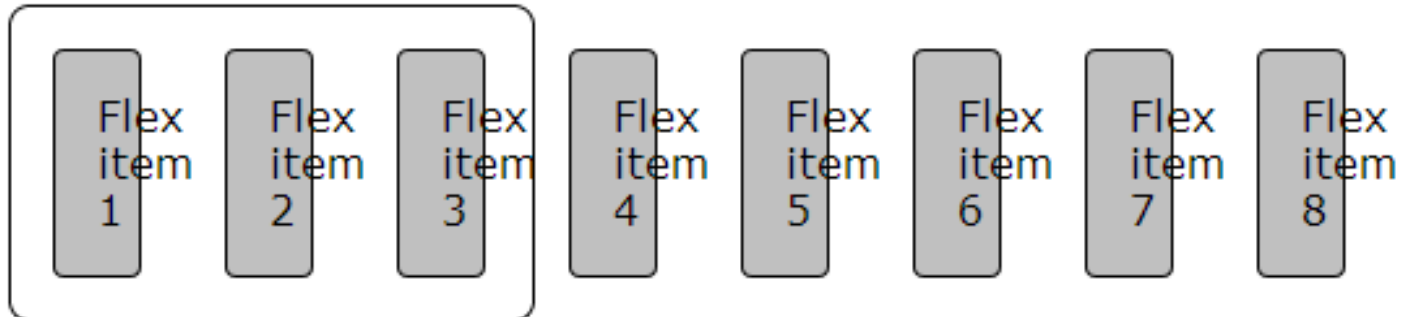
Flex item 3

Flex item 2

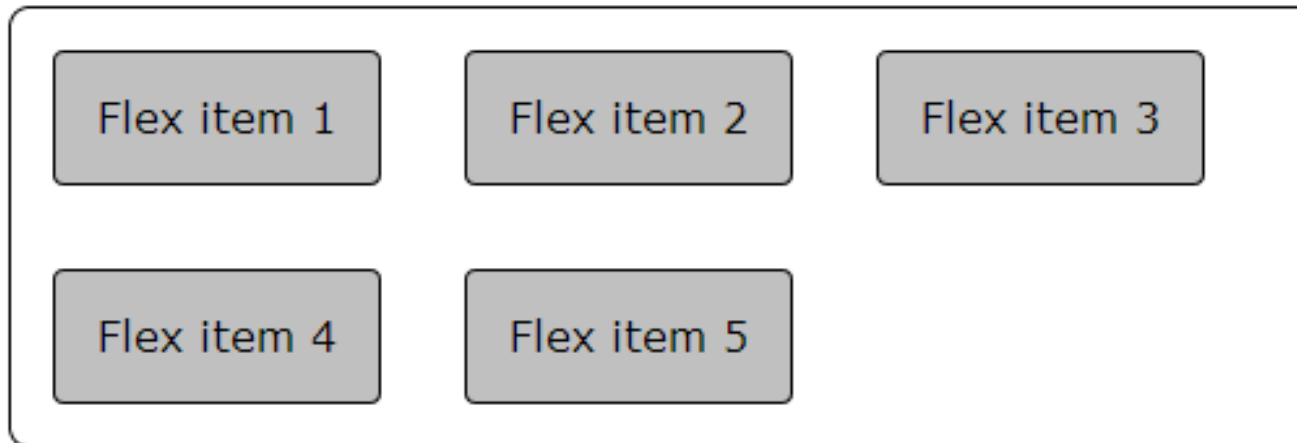
Flex item 1

Cajas - Flexbox

El contenedor tiene flex-wrap: nowrap

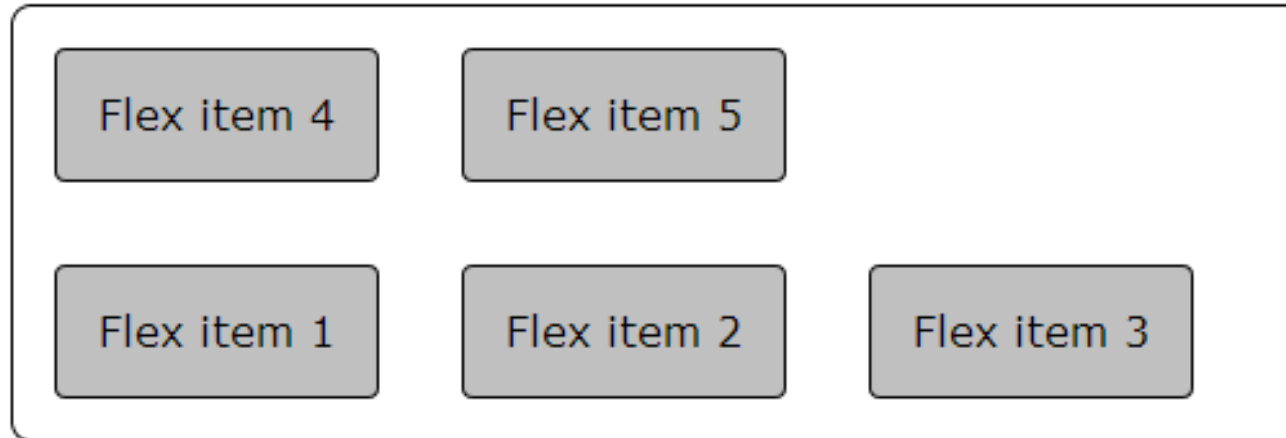


El contenedor tiene flex-wrap: wrap

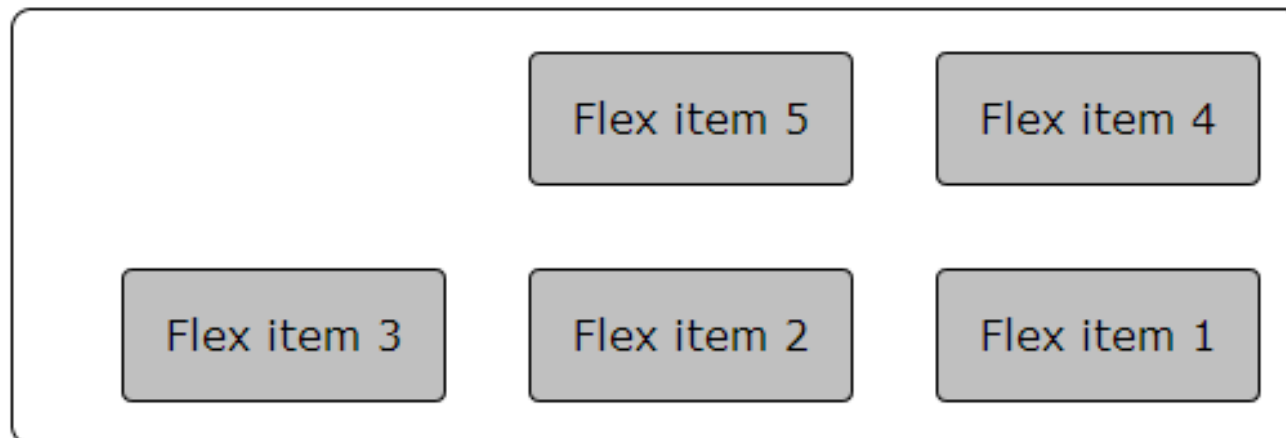


Cajas - Flexbox

El contenedor tiene flex-wrap: wrap-reverse



El contenedor tiene flex-wrap: wrap-reverse y flex-direction: row-reverse



Cajas - Flexbox

El orden en el que aparecen los elementos es igual al orden del código (en uno u otro sentido).

La propiedad **order** permite cambiar el orden, aunque solo afecta a los medios visuales.

- ✚ No afecta a la salida de, por ejemplo, un lector de pantalla.
- ✚ Toma un valor entero.
 - Un valor negativo hace que “retroceda” de su posición actual.
 - Un valor 0 hace que se quede dónde le corresponde.

Se ha cambiado la propiedad order de los flex item

Flex item 2
order: -1

Flex item 1
order: 2

Flex item 4
order: 3

Flex item 5
order: 4

Flex item 3
order: 5

Cajas – Flexitems - Flexibilidad

La flexibilidad (el factor de aumento o disminución) de los flex items la controla la propiedad flex de dichos elementos. Tiene tres argumentos opcionales:

flex: factorAumento factorDisminución tamañoInicial

- *factorAumento* indica la proporción respecto a los otros elementos en que crecerá cuando sobre espacio en el contenedor. También lo controla la propiedad flex-grow.
- *factorDisminución* indica la proporción respecto a los otros elementos en que disminuirá cuando falte espacio en el contenedor. También lo controla la propiedad flex-shrink.
- *tamañoInicial* indica el tamaño inicial del elemento antes de distribuir el espacio. También lo controla la propiedad flex-basis.

Cajas – Flexitems - Flexibilidad

Si el contenedor tiene 800px de ancho y cada elemento tiene un ancho inicial de 100px (más 2 píxel del borde), el espacio sobrante serán 494 píxel.

Si todos los elementos tienen la propiedad flex-grow a 1, cada ítem ocupará:
 $494/3 + 100 = \sim 264\text{px}$.

Si todos los ítem 1 y 3 tienen la propiedad flex-grow a 1 y el ítem 2 a 2, el espacio sobrante se dividirá entre 4.

- El ítem 1 tendrá $(494/4) * 1 + 100 = \sim 223,5\text{px}$.
- El ítem 2 tendrá $(494/4) * 2 + 100 = \sim 347\text{px}$.
- El ítem 3 tendrá $(494/4) * 1 + 100 = \sim 223,5\text{px}$.

El contenedor tiene 800px

Flex item 1
flex: 1 1 100px;

Flex item 2
flex: 2 1 100px;

Flex item 3
flex: 1 1 100px;

Cajas – Flexitems - Flexibilidad

Si el contenedor tiene 800px de ancho y cada elemento tiene un ancho inicial de 400px (más 2 píxel del borde), faltarán 406 píxel para situar los elementos.

Si todos los elementos tienen la propiedad flex-shrink a 1, cada ítem ocupará: $400 - (406/3) = \sim 264,4\text{px}$.

Si todos los ítem 1 y 3 tienen la propiedad flex-shrink a 1 y el ítem 2 a 2, el espacio sobrante se dividirá entre 4.

- El ítem 1 tendrá $400 - (406/4) * 1 = \sim 298,5\text{px}$.
- El ítem 2 tendrá $400 - (406/4) * 2 = \sim 197\text{px}$.
- El ítem 3 tendrá $400 - (406/4) * 1 = \sim 298,5\text{px}$.

El contenedor tiene 800px

Flex item 1

flex: 1 1 400px;

Flex item 2

flex: 1 2 400px;

Flex item 3

flex: 1 1 400px;

Cajas – Flexitems - Flexibilidad

Un único valor numérico en la propiedad flex indica el factor de aumento del elemento. Sería lo mismo que incluir sólo la propiedad flex-grow.

Divide el contenedor en 4 partes. Flex item 2 siempre ocupará el doble

Flex item 1
flex: 1

Flex item 2
flex: 2

Flex item 3
flex: 1

Divide el contenedor en 4 partes. Flex item 2 siempre ocupará el doble

Flex item 1
flex: 1

Flex item 2
flex: 2

Flex item 3
flex: 1

Cajas – Flexitems - Flexibilidad

La propiedad `flex-basis` puede tomar los valores:

- Medida o porcentaje.
- `initial`. Trata de mantener el valor inicial de la caja si hay espacio suficiente. Si no se ha establecido una propiedad `width`, mantiene el ancho del contenido.
- `auto`. Mantiene el elemento flexible, aumentado o disminuyendo su ancho según sobra espacio en el contenedor.
- `none`. Es similar a `initial`, pero mantiene el tamaño si hay desbordamiento.

El elemento 2 trata de mantener su tamaño de 200px

Flex item 1
`flex: 1`

Flex item 2
`width: 200px; flex: initial`

Flex item 3
`flex: 1`

El elemento 2 trata de mantener su tamaño de 200px

Flex item 1
flex: 1

Flex item 2
width: 200px; flex: initial

Flex item 3
flex: 1

El elemento 2 modifica su tamaño, los otros tienen el ancho del contenido

Flex item 1
flex: initial

Flex item 2
flex: auto

Flex item 3
flex: initial

El elemento 2 modifica su tamaño, el elemento 1 mantiene el ancho del contenido

Flex item 1
flex: none

Flex item 2
flex: auto

Flex item 3
flex: initial

El elemento 2 modifica su tamaño, los otros tienen el ancho del contenido

Flex item 1
flex: initial

Flex item 2
flex: auto

Flex item 3
flex: initial

Cajas – Flexbox - Alineación

Una vez el contenedor flexible ha calculado la posición y el tamaño de los elementos se hace la alineación. La alineación con la propiedad `margin`.

- El valor `auto` en el eje principal separa los elementos en varios bloques.

[Inicio](#) | [Productos](#) | [Carrito](#)

[Ayuda](#)

```
nav ul{
  ...
  display:flex;
  nav>ul>li#ayuda{margin-left:auto;}
<nav>
<ul>
<li><a href="#1">Inicio</a></li>
<li><a href="#2">Productos</a></li>
<li><a href="#3">Carrito</a></li>
<li id="ayuda"><a href="#4">Ayuda</a></li>
</ul>
</nav>
```

Cajas – Flexbox– Alineación en el eje principal

La propiedad `justify-content` distribuye los elementos a lo largo del eje principal. Puede tomar los valores:

- `flex-start`. Alinea los elementos flexibles al comienzo del eje principal.
- `flex-end`. Alinea los elementos flexibles al final del eje principal.
- `center`. Centra los elementos en el eje principal.
- `space-between`. Alinea los elementos a lo largo del eje principal. El primero y el último se alinean en los extremos del eje, dejando el espacio sobrante entre ellos.
- `space-around`. Alinea los elementos a lo largo del eje principal, dejando el espacio sobrante alrededor de ellos.

El contenedor tiene la propiedad `justify-content`: `space-between`

Flex item 1

Flex item 2

Flex item 3

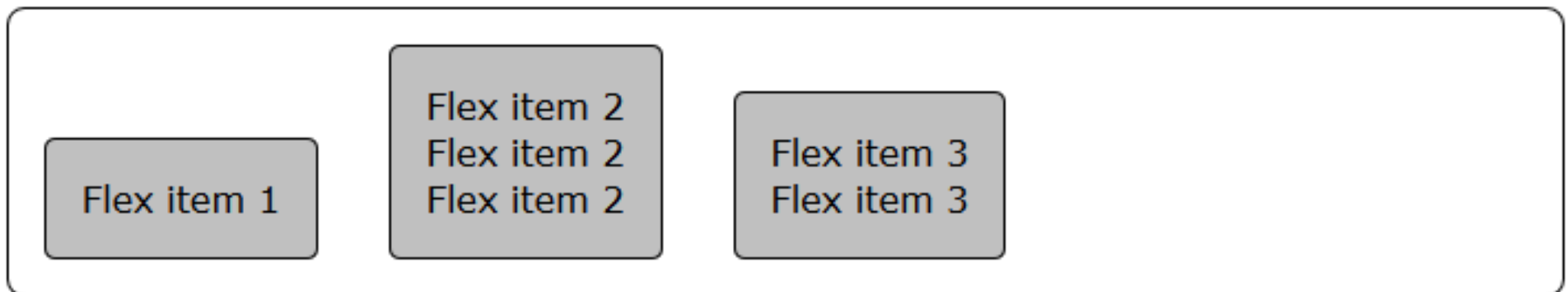
Cajas – Flexbox– Alineación en el eje transversal

La propiedad align-items permite alinear los elementos en el eje transversal.

Admite los valores:

- flex-start. Alinea los elementos flexibles al comienzo del eje trasversal.
- flex-end. Alinea los elementos flexibles al final del eje transversal.
- center. Centra los elementos en el eje transversal.
- baseline. Alinea los elementos a partir de la altura de los caracteres de la primera línea respecto al eje transversal.
- stretch. Alarga los elementos para todos ocupen la totalidad del eje transversal. Es la opción por omisión.

El contenedor tiene la propiedad align-items: flex-end

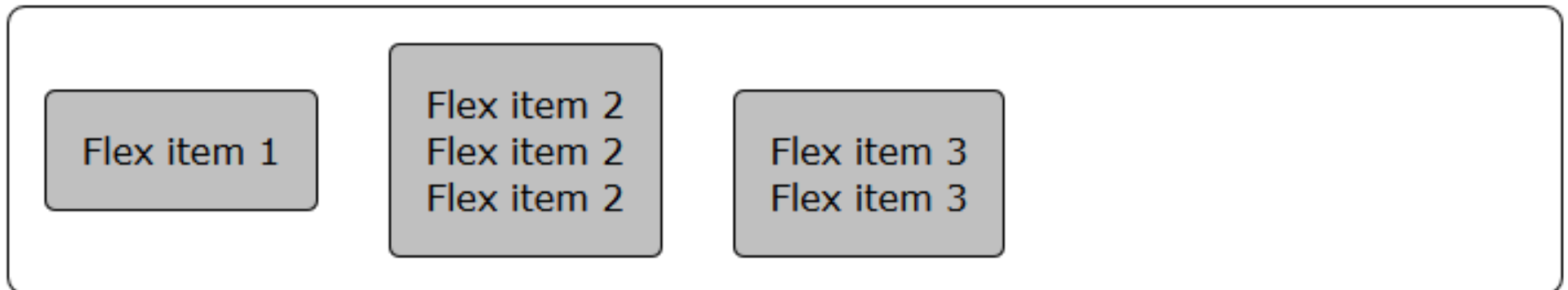


Cajas – Flexbox– Alineación en el eje transversal

La propiedad align-items establece la alineación de todos los flex items de un contenedor.

- La propiedad align-self de un flex item permite romper esa alineación para un elemento concreto.
- Admite los mismos valores, más el valor auto que establece el mismo valor que el align-items del padre o stretch si no está establecida.

El contenedor tiene la propiedad align-items: flex-end; el primer elemento tiene align-self: center

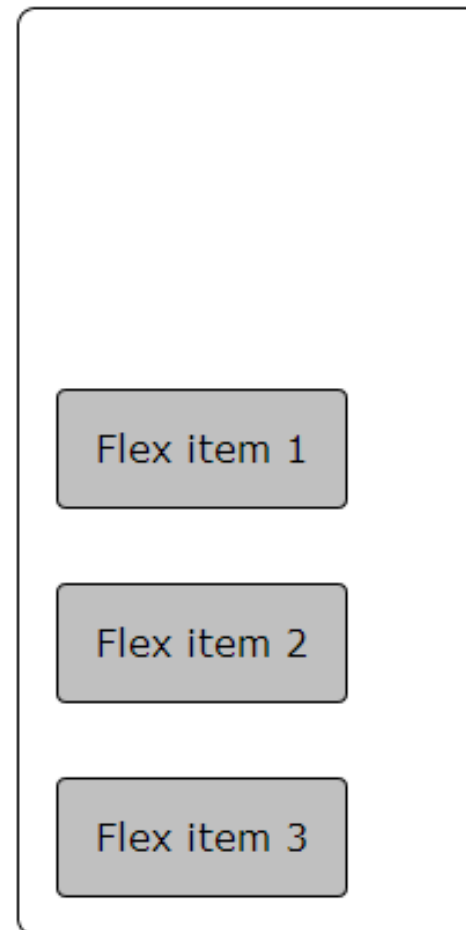


Cajas – Flexbox– Alineación en el eje transversal

La propiedad align-content hace la misma función que justify-content, pero en el eje vertical.

- Sólo tiene efecto si los flex item ocupan más de una línea.
- Toma los mismos valores que justify-content (flex-start, flex-end, center, space-between, space-around) más el valor stretch. Con stretch los elementos ocupan todo el contenedor.

El contenedor tiene la propiedad flex-wrap: wrap y align-content: start-end (sólo funciona en IE y Chrome)



Cajas - Flexbox

Revisar el manual de la W3Schools sobre los efectos sobre Flexbox:

http://www.w3schools.com/css/css3_flexbox.asp