

EL DISEÑO WEB CON
display : flex;

La distribución de los contenidos

display: block;

flex0.html

display: inline;

flex0.css

display:inline-block;

¿Cómo funciona el desbordamiento en las cajas?

Propiedad *overflow*

¿Qué es display:flex?

Flexbox es un módulo de diseño en CSS3 para mejorar la forma en que distribuimos los contenidos en un sitio web.

Con FlexBox facilitamos la forma en que posicionamos elementos porque es más simple y usamos menos código.

FlexBox está constituido por un padre (Contenedor - Flex) y sus hijos (Elemento - Flex).

Compatible con los navegadores

<https://caniuse.com/#feat=flexbox>

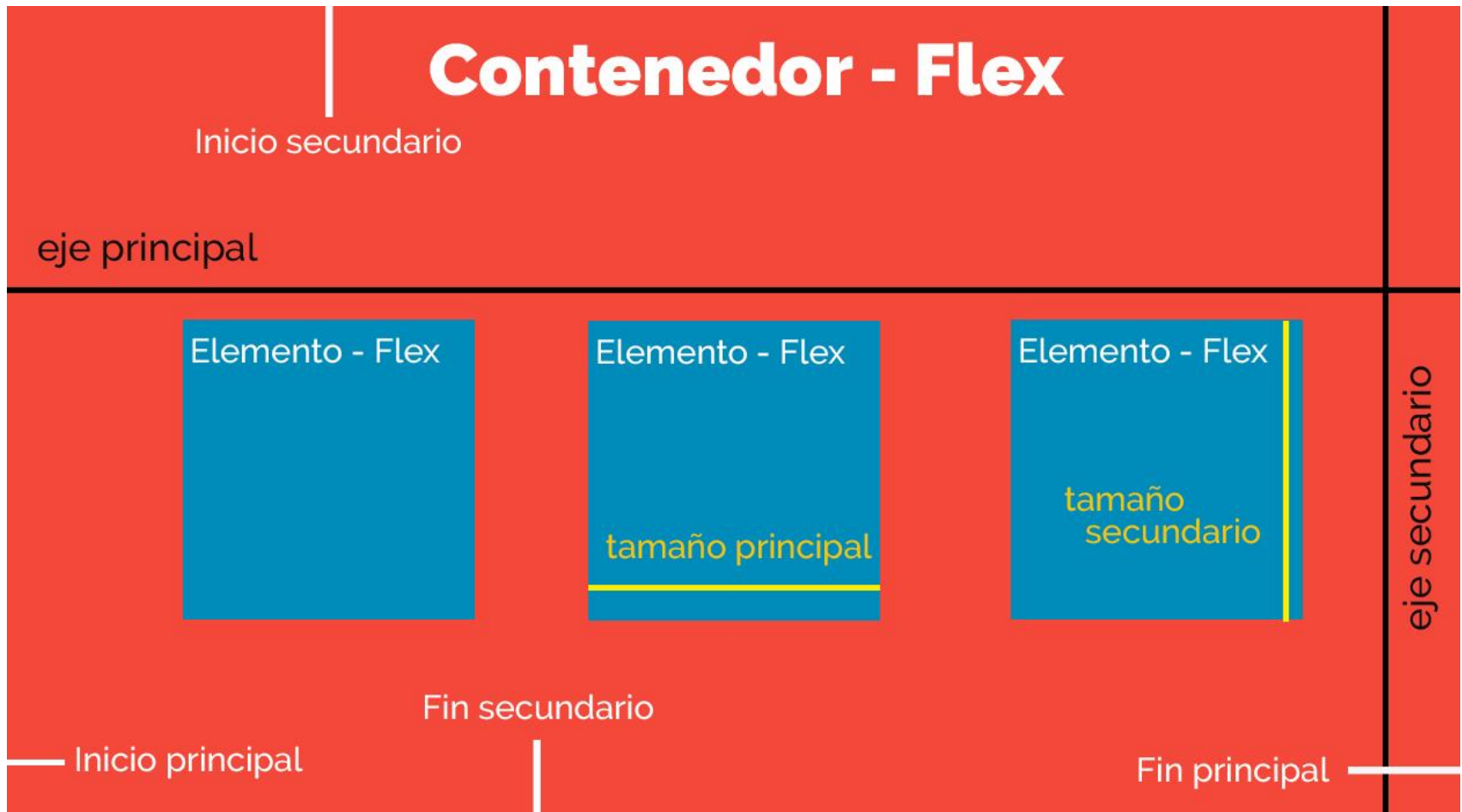
La estructura.

```
<section class="padre">  
</section>
```

```
.padre {  
    display: flex;  
}
```

Es lo único que necesitamos para configurar el contenedor principal y automáticamente todos sus hijos se comportarán como elementos hijos **flex**.

La estructura.



flex1.html flex1.css

display: flex / inline-flex

FLEX CONTENEDOR (PADRE)

flex-direction: row/ column; se adapta al ancho/alto del padre.

Posibilidad de reverse

flex-wrap MULTILINEA. Si el tamaño de la caja no es suficiente usará varias líneas.

* propiedad *shorthand* **flex-flow** puedes incluir *direction* y *wrap*

justify-content alinea elementos del eje principal (por defecto, horizontal)

align-items alinea elementos del eje secundario (por defecto, vertical)

align-content sólo para el caso de tener flex-wrap activado
MULTILINEA

* propiedad *shorthand* **place-content** puedes incluir

justify-content y *align-content*

Propiedades del contenedor - flex (Padre)

ALINEACIONES

flex-direction. Define las direcciones del eje principal que será el horizontal si trabajamos con filas y vertical si trabajamos con columnas.

flex2.html flex2.css

row | row-reverse | column | column-reverse

flex-wrap. Controlamos si el contenedor flex mueve sus elementos - flex en líneas individuales o múltiples

flex3.html flex3.css

nowrap | wrap | wrap-reverse

Propiedades del contenedor - flex (Padre)

ALINEACIONES

justify-content. Alinea los elementos - flex a lo largo del eje principal de la línea actual (contenedor - flex).

flex-start | flex-end | center | space-between | space-around | space-evenly

flex4.html flex4.css

Propiedades del contenedor - flex (Padre)

ALINEACIONES

align-items. Similar a la propiedad justify-content, pero en la dirección del eje secundario.

stretch | flex-start | flex-end | center | baseline

flex5.html flex5.css

Propiedades del contenedor - flex (Padre)

ALINEACIONES

align-content. Alinea las líneas de un contenedor - flex cuando los elementos no utilizan todo el espacio disponible en el eje secundario

flex-start | flex-end | center | space-between | space-around

flex6.html

flex6.css

Esta propiedad solo funciona cuando el contenedor - flex tiene varias líneas de elementos - flex.

Propiedades del contenedor - flex (Padre)

ALINEACIONES

GAP, huecos separación entre hijos

`gap: 10px /* misma separación */`

`gap: 5px 10px; /* vertical horizontal */`

`flex7.html`

`flex7.css`

Propiedades de los elementos - flex (Hijos). Cada hijo puede llevar una misma propiedad con distinto valor.

ALINEACIONES

align-self. Permite la alineación por defecto (o la específica por align-items) para ser anulados por **elementos - flex** individuales.

auto | flex-start | flex-end | center | baseline | stretch

flex8.html

flex8.css

*float, clear y vertical-align no tienen efecto en un **elemento - flex**.*

Propiedades de los elementos - flex (Hijos). Cada hijo puede llevar una misma propiedad con distinto valor.

order. Controlamos el orden de los elementos - flex que están dentro del contenedor - flex.

Hay que poner un entero (puede ser negativo).
Por defecto es 0.

flex9.html

flex9.css

Propiedades de los elementos - flex (Hijos). Cada hijo puede llevar una misma propiedad con distinto valor.

flex-grow. Especifica el factor que determina hasta que punto un **elemento - flex** crecerá en relación con el resto de los **elementos - flex**.

Hay que poner un número. *Por defecto es 0.*

La propiedad **flex-grow** actúa en situaciones donde:

- Hay un **flex-basis** definido.
- Los hijos cubren el tamaño total del contenedor flex padre.

flex10.html

flex10.css

Ejemplos de estructuras flex

Combinación de *display:flex* y propiedades en padre e hijos con *transform* para mostrar dinamismo.

flex_layout1.html

flex_layout1.css

Reorganizamos en función del tamaño de la pantalla
@media

flex_layout2.html

flex_layout2.css