

Programación Estática y Laboratorio web

CSS

Parte IV



Universidad Nacional del Comahue
Facultad de Informática



Visualización: Display

display	Visualización de un elemento
Valores	inline block none list-item run-in inline-block table inline-table table-row-group table-header-group table-footer-group table-row table-column-group table-column table-cell table-caption flex inherit
Se aplica a	Todos los elementos
Valor inicial	inline
Descripción	Permite controlar la forma de visualizar un elemento e incluso ocultarlo

- Los más utilizados son inline, block, none y flex.
- inline permite mostrar cualquier elemento como si fuera de línea.
- block permite mostrar cualquier elemento como si fuera de bloque.
- none hace como si el elemento no existiera
- flex lo veremos a continuación

[Display: flex]

- Es una propiedad del contenedor (parent).
- Sirve para posicionar los elementos internos (children).
- Vamos a profundizar en parent/children más adelante.
- Convierte al contenedor en un contenedor flexible.
- Objetivo: alinear y distribuir los elementos de manera sencilla y eficiente
- Se usa en conjunción con varias propiedades más, algunas de ellas en el contenedor y otras en los elementos contenidos

[Display: flex]

```
<div id="contenedor">

  <div id="d1">
    <h1>Div 1</h1>
  </div>

  <div id="d2">
    <h1>Div 2</h1>
  </div>

  <div id="d3">
    <h1>Div 3</h1>
  </div>

  <div id="d4">
    <h1>Div 4</h1>
  </div>
</div>
```

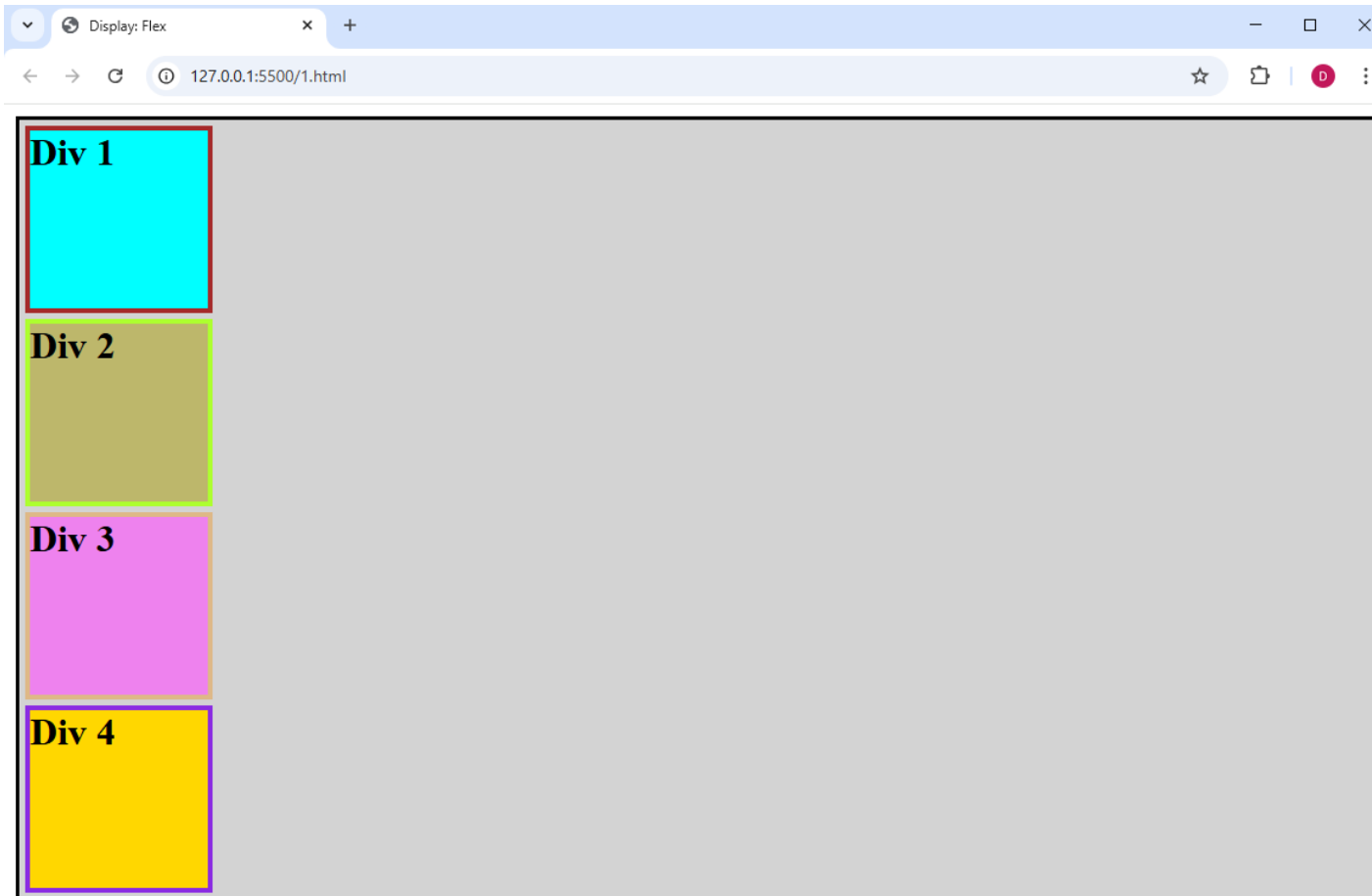
```
#contenedor
{
  background-color: lightgray;
  margin: 10px;
}

.d_inner
{
  width: 150px;
  height: 150px;
  border-width: 4px;
}

#d1
{
  border-color: brown;
  background-color: aqua;
}

#d2
{
  border-color: greenyellow;
```

[Display: flex]

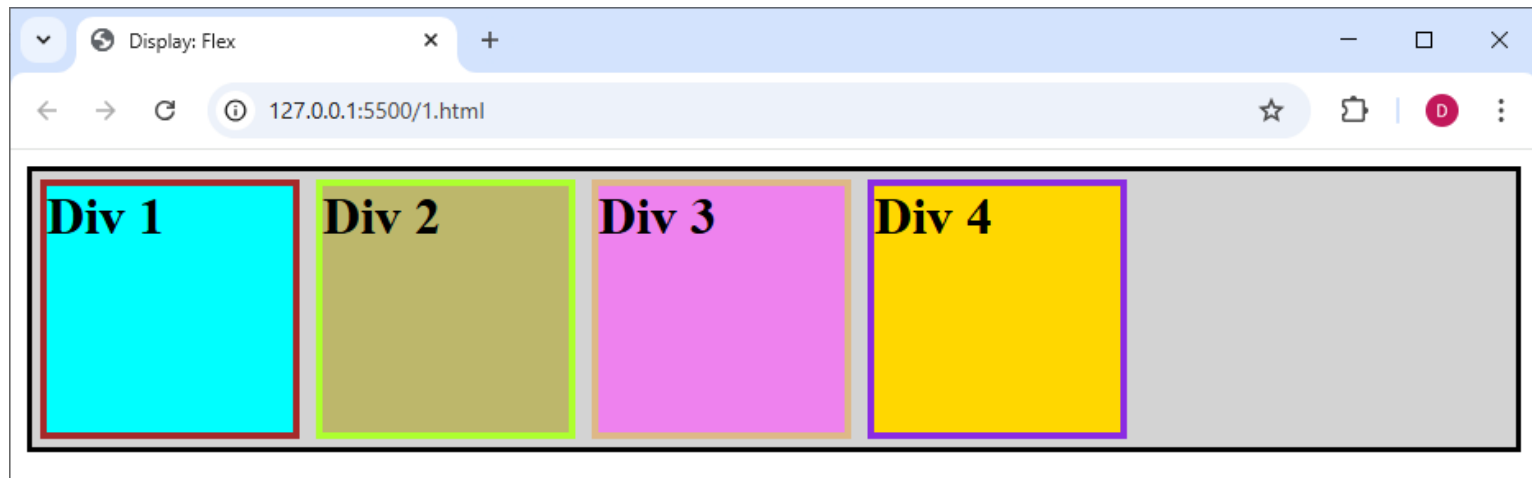


[Display: flex]

■ Agregando display: flex al contenedor

```
#contenedor
{
  display: flex;

  background-color: lightgray;
  margin: 10px;
}
```

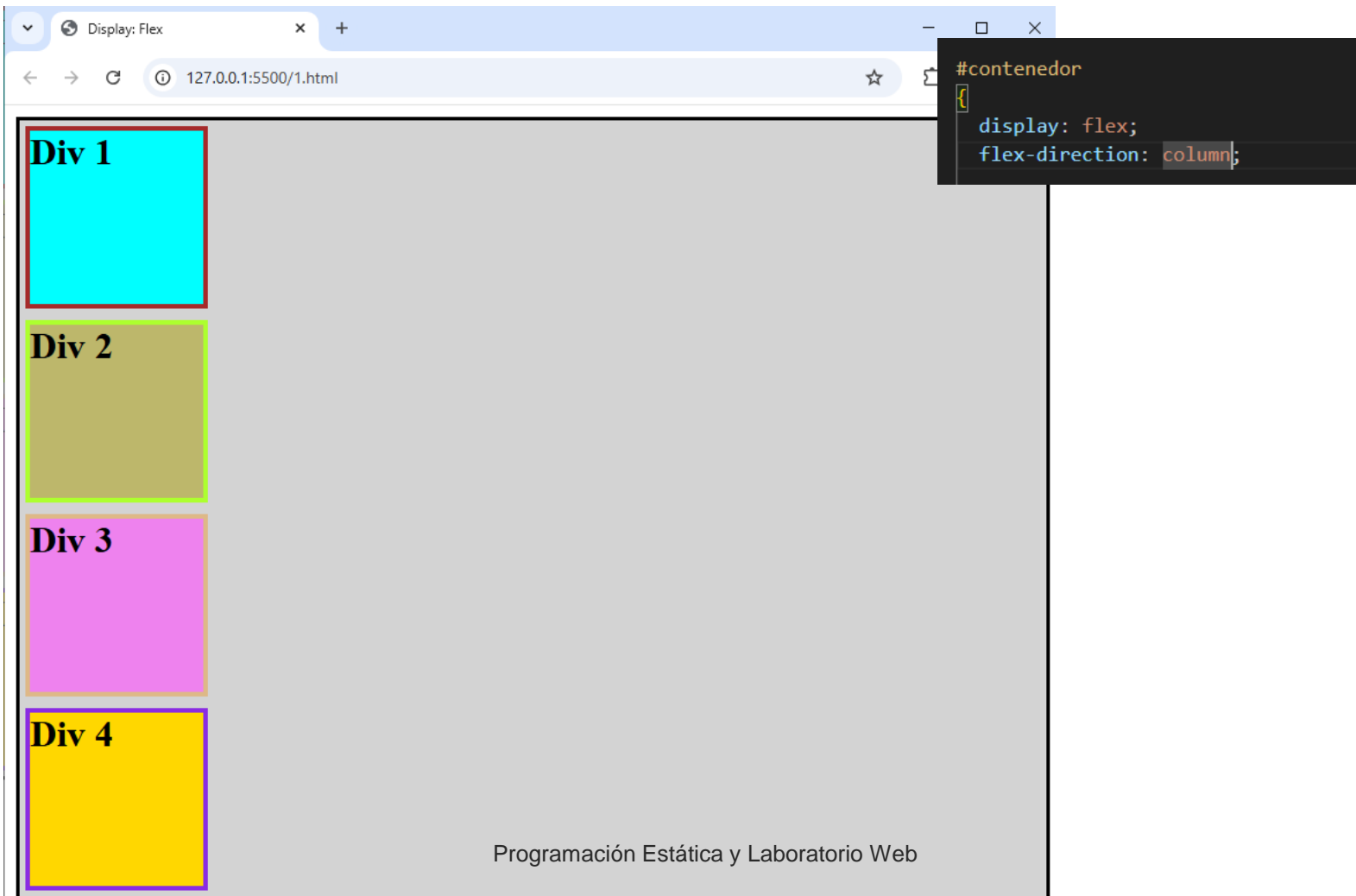


[Display: flex]

- Por defecto, ordena de izquierda a derecha agrupado a la izquierda.
- Conceptos: flex-direction
- flex-direction indica si ordenar de manera flexible horizontal o verticalmente.
- La dirección indicada es el eje principal, y la perpendicular es el eje secundario



Display: flex



Visualización: flex-direction

flex-direction	Dirección principal del flujo de posicionamiento
Valores	row row-reverse column column-reverse
Se aplica a	Elementos contenedores con display:flex
Valor inicial	row
Descripción	Permite controlar al orden de posicionamiento de los elementos hijos dentro de un contenedor display:flex

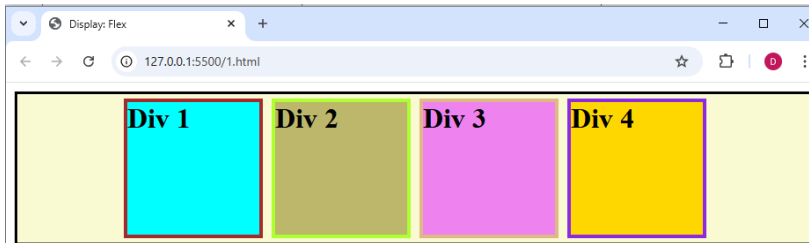
- row: ordena en fila, de izquierda a derecha
- column: ordena en columna, de arriba hacia abajo
- row-reverse: fila de derecha a izquierda
- column-reverse: columna de abajo hacia arriba

Visualización: justify-content

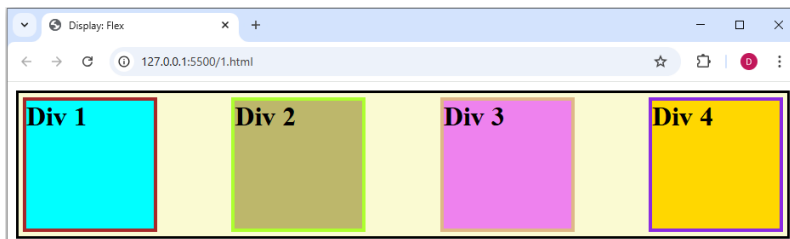
justify-content	Alineación del contenido a lo largo del eje principal (definido por flex-direction).
Valores	flex-start flex-end center space-between space-around space-evenly
Se aplica a	Contenedores con display:flex
Valor inicial	flex-start
Descripción	Controla cómo se distribuyen los elementos hijos dentro del contenedor.

- Resuelve el problema que siempre nos volvía locos: la alineación de elementos internos
- flex-start: Alinea todos los elementos al inicio del eje principal.
- flex-end: Alinea al final del eje principal.
- center: Centra los elementos.
- space-between: Primer y último elemento pegados a los bordes, el resto se reparte el espacio.
- space-around: Espacio igual alrededor de cada elemento.
- space-evenly: Espacio exactamente igual entre todos los elementos, incluso entre los bordes y los extremos.

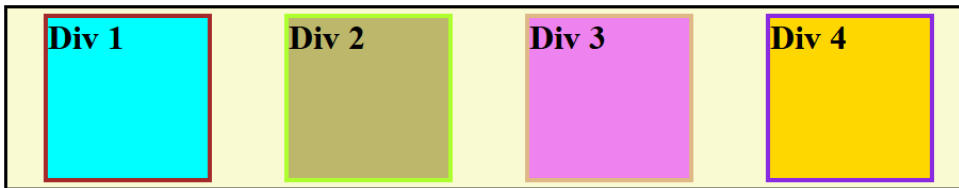
Visualización: justify-content



```
#contenedor  
{  
  display: flex;  
  flex-direction: row;  
  justify-content: center;  
}
```



```
#contenedor  
{  
  display: flex;  
  flex-direction: row;  
  justify-content: space-between;  
}
```



```
#contenedor  
{  
  display: flex;  
  flex-direction: row;  
  justify-content: space-around;  
}
```



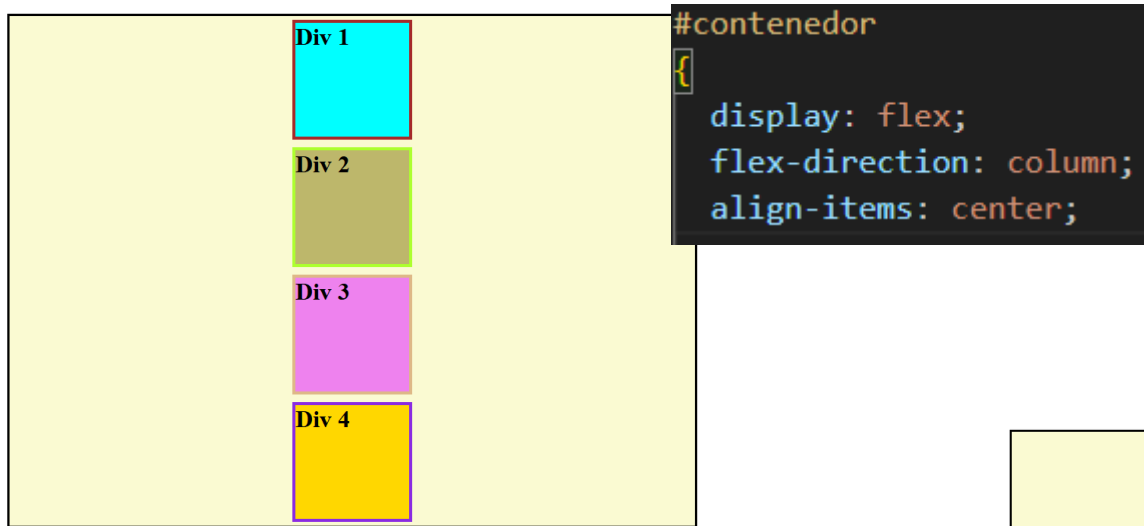
```
#contenedor  
{  
  display: flex;  
  flex-direction: row;  
  justify-content: space-evenly;  
}
```

Visualización: align-items

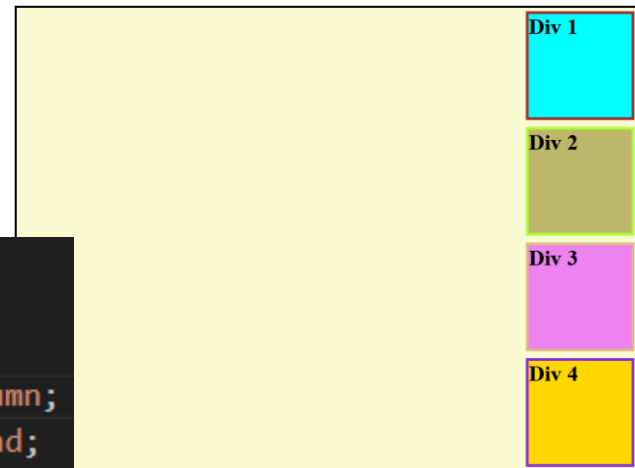
align-items	Alinea los elementos hijos a lo largo del eje secundario (el perpendicular al principal)
Valores	stretch flex-start flex-end center baseline
Se aplica a	Contenedores con display: flex
Valor inicial	stretch
Descripción	Controla cómo se alinean los elementos dentro del contenedor en el eje transversal

- stretch: (por defecto) Estira los elementos para que ocupen todo el alto/largo disponible del eje secundario.
- flex-start: Alinea los elementos al inicio del eje secundario.
- flex-end: Alinea al final del eje secundario.
- center: Centra los elementos en el eje secundario.
- baseline: Alinea todos los elementos según su línea base de texto.

Visualización: align-items



```
#contenedor  
{  
  display: flex;  
  flex-direction: column;  
  align-items: flex-end;  
}
```



Visualización: flex-wrap

flex-wrap	Controla si los ítems deben mantenerse en una línea o pueden saltar a otra línea (hacer "wrap").
Valores	nowrap wrap wrap-reverse
Se aplica a	Contenedores con display: flex
Valor inicial	nowrap
Descripción	Indica si los ítems deben permanecer en una línea o distribuirse en múltiples líneas.

- nowrap: (por defecto) Todos los ítems van en una sola línea, aunque se deformen.
- wrap: Los ítems saltan a una nueva línea si no entran.
- wrap-reverse: Como wrap, pero la nueva línea aparece arriba en lugar de abajo.

Visualización: flex-wrap

#contenedor

```
{  
  display: flex;  
  flex-direction: row;
```



#contenedor

```
{  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;
```

Visualización: align-items

align-items	Distribuye el espacio entre las líneas del contenido cuando hay múltiples filas o columnas.
Valores	stretch flex-start flex-end center space-between space-around space-evenly
Se aplica a	Contenedores con display: flex
Valor inicial	stretch
Descripción	Alinea el conjunto de líneas en el eje secundario (vertical si row, horizontal si column).

- Importante: eje secundario

Display: flex para elementos “hijos”

Propiedad

flex-grow

flex-shrink

flex-basis

flex

align-self

order

Qué hace

Hace que el ítem crezca para llenar espacio disponible

Permite que el ítem se achique si el espacio es insuficiente

Define el **tamaño inicial base** del ítem antes de aplicar grow o shrink

Atajo que combina grow, shrink y basis: ejemplo → flex: 1 1 auto

Permite que un ítem individual **sobrescriba** align-items

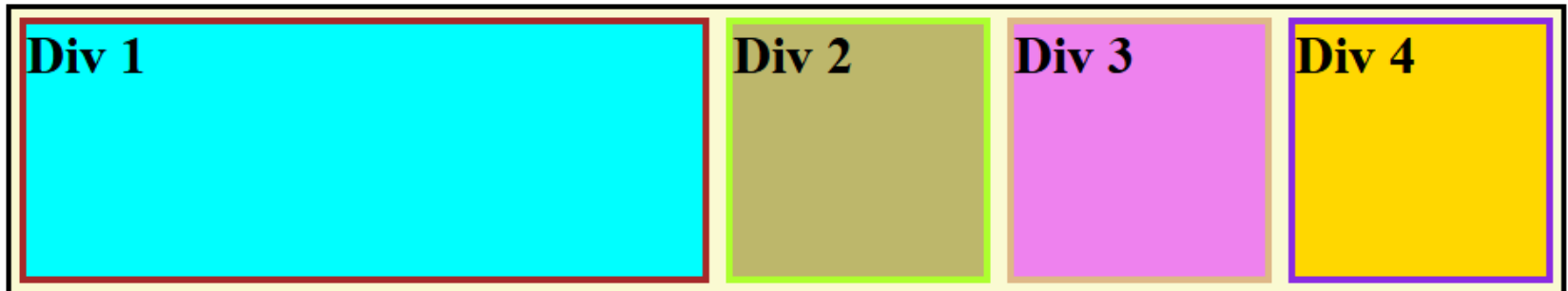
Cambia el orden visual de los ítems sin alterar el HTML

Visualización: flex (para hijos)

flex	Atajo que define cómo se comporta un ítem dentro del contenedor flex
Valores	none auto initial · números personalizados como 1, 2 1 0, etc
Se aplica a	Elementos hijos (flex items) dentro de un contenedor con display: flex
Valor inicial	0 1 auto
Descripción	Combina tres propiedades: flex-grow, flex-shrink y flex-basis.

- flex: 1 El ítem ocupa todo el espacio disponible de forma proporcional
- flex: 0 El ítem no crece ni se achica, mantiene su tamaño base
- flex: 1 1 0 Crece y se achica, pero parte de base cero
- flex: auto Usa el tamaño del contenido pero puede crecer y achicarse si hace falta

Visualización: flex (para hijos)



```
#contenedor
{
  display: flex;
  flex-direction: row;
```

```
#d1
{
  flex: 1;

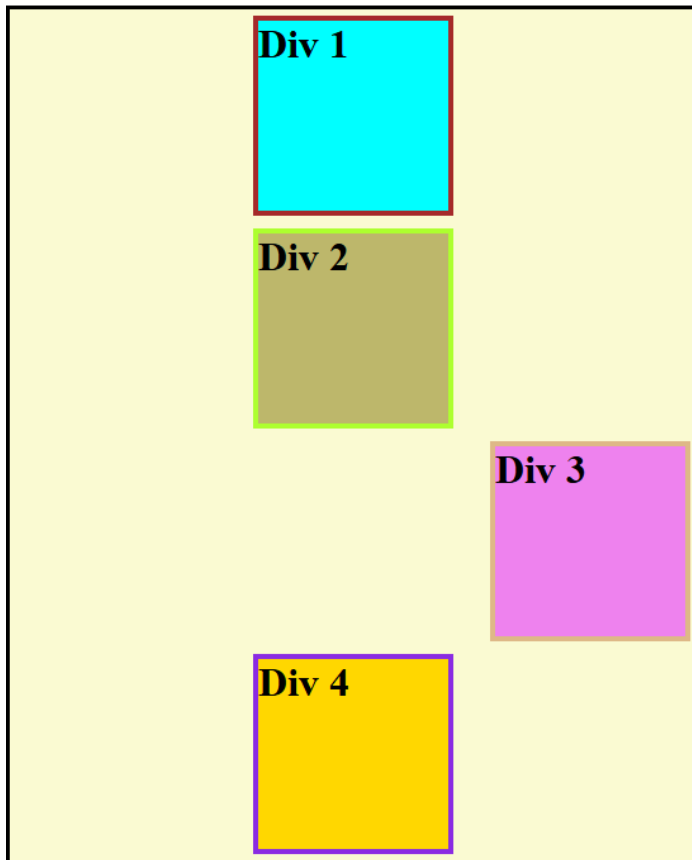
  border-color: ■ brown;
  background-color: ■ aqua;
}
```

Estilos. fondos: align-self

align-self	Permite que un ítem individual tenga su propia alineación en el eje secundario
Valores	auto · flex-start · flex-end · center · baseline · stretch
Se aplica a	Elementos hijos (flex items) dentro de un contenedor con display: flex
Valor inicial	auto (toma el valor de align-items del contenedor)
Descripción	Sobrescribe la alineación definida por align-items, solo para ese ítem .

- Permite hacer excepciones al alineamiento general del contenedor

Estilos. fondos: align-self



```
#contenedor
{
  display: flex;
  flex-direction: column;
  align-items: center;
}
```

```
#d3
{
  align-self: flex-end;
  border-color: burlywood;
  background-color: violet;
}
```

[¿Y entonces que usamos?]

display: flex

Alinear, distribuir y organizar elementos en fila o columna

Layouts modernos: menús, cards, centrar contenido, formularios, etc.

float

Hacer que un elemento "flote" y el texto lo rodee

Solo en casos puntuales (ej: imagen con texto al costado). No para layouts.

position: relative

Mover un elemento respecto a su posición original. Define contexto de posición

Tooltips, íconos superpuestos, globos de texto, ajustes finos en la ubicación.

table

Mostrar datos en forma de filas y columnas (estructura tabular)

Tablas de datos reales: reportes, precios, horarios. No para estructurar páginas.

Vínculos

- Introducción a CSS.
<http://www.librosweb.es/css/>
- Ejemplo de CSS:
<http://www.csszengarden.com/tr/espanol/>
- <http://www.csszengarden.com> (en inglés)