



# TALLER DE FLEXBOX PROFESIONAL


@diana\_aceves\_

## ¿QUÉ VAMOS A VER EN ESTE TALLER?

- ▶ Aplicaciones conocidas básicas:
  - Alineación vertical de elementos
  - Columnas de igual altura
  - Grid flexibles
  - Cambiar orden de elementos
  - Holy Grail Layout
  - Sticky footer... PERO TAMBIÉN VAMOS A VER...

## ¿QUÉ VAMOS A VER EN ESTE TALLER?

► Otras aplicaciones:

- Items tipo app
- Layouts en zig-zag
- Gráficas sencillas
- Inputs con iconos 
- Headers con múltiples zonas
- Filtros de igual tamaño con número desconocido de elementos

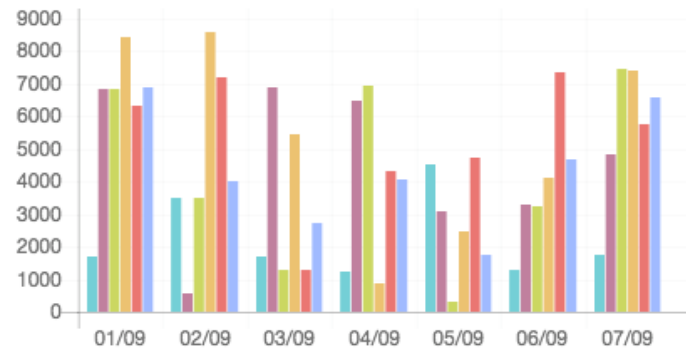
## Mi facturación

Tamoil España S.A.

● TOTAL FACTURADO 95.430,00€  
Beneficios Mundo 123 120,00€

DESGLOSADO

ACUMULADO



Nº COMERCIO

DIRECCIÓN / TPV

<span style="color: teal;">●</span> 23425322	Serrano 141, Madrid TPV fijo con línea ADSL	>
<span style="color: purple;">●</span> 34563465	C/ Astrofísico Carlos Sánchez Magro 141, 28085 Talavera de la Reina TPV Móvil GPRS	>
<span style="color: green;">●</span> 85678567	Grán Vía 63, Madrid TPV fijo con línea ADSL	>
<span style="color: orange;">●</span> 23456789	Grán Vía 63, Madrid TPV fijo con línea ADSL	>
<span style="color: red;">●</span> 09876546	Grán Vía 63, Madrid TPV fijo con línea ADSL	>
<span style="color: blue;">●</span> 55580356	Grán Vía 63, Madrid TPV fijo con línea ADSL	>

## Smart Keyboard for iPad Pro

Type. On a totally new type of keyboard.

Whichever iPad Pro you choose, there's a Smart Keyboard that fits.<sup>2</sup> By combining new technologies and materials, we created a portable keyboard that's unbelievably thin and even doubles as a lightweight yet durable iPad cover. And thanks to the Smart Connector, there are no cords, pairing, or plugs. Just attach and start typing.

[Learn more about the Smart Keyboard >](#)



## Apple Pencil for iPad Pro

Completely familiar.  
Entirely revolutionary.

Apple Pencil feels incredibly responsive from the moment you pick it up.<sup>2</sup> It's sensitive to both tilt and pressure, allowing you to create a range of artistic effects. And with its pixel-perfect precision, Apple Pencil opens up new creative possibilities.

[Learn more about Apple Pencil >](#)

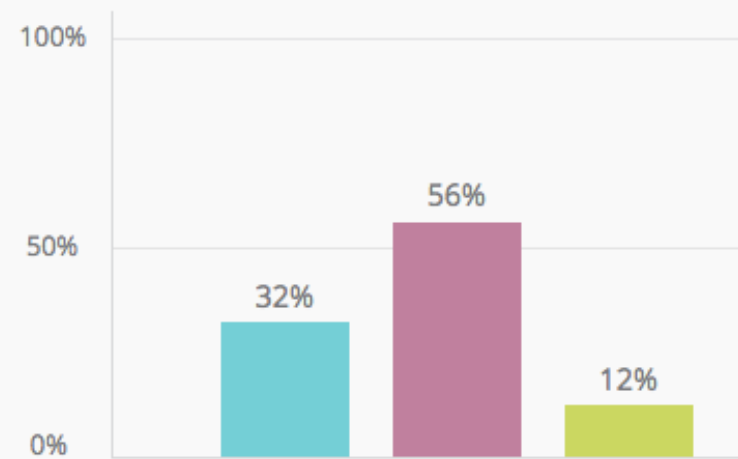


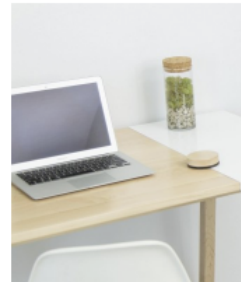
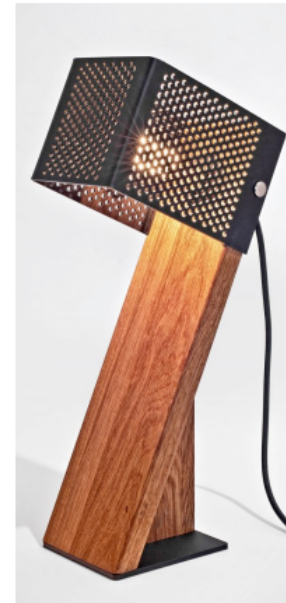
0042513762 | C. CTRA. N-III, 30 HONRUBIA

### ¿EN QUÉ TRAMO DEL DÍA COMPRAN MÁS?

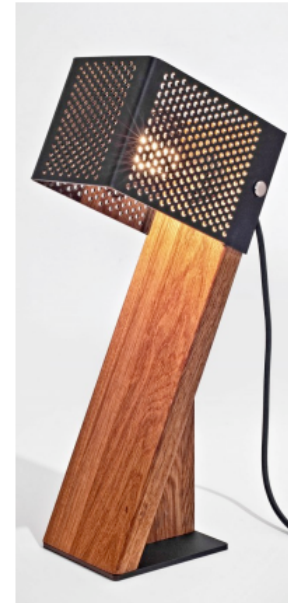
- Mañana (07:00-15:00)
- Tarde (15:00-21:00)
- Noche (21:00-07:00)

EN MI COMERCIO



**THRONE**[HOME](#)[ABOUT](#)[CLIENTS](#)[PORTFOLIO](#)[SERVICES](#)[BLOG](#)[CONTACT](#)[PAGES](#)[BUY NOW!](#)[All](#)[Dishes](#)[Lamps](#)[Others](#)[Tables](#)



**THRONE**[HOME](#)[ABOUT](#)[CLIENTS](#)[PORTFOLIO](#)[SERVICES](#)[BLOG](#)[CONTACT](#)[PAGES](#)[BUY NOW!](#)[All](#)[Chairs](#)[Creative](#)[Dishes](#)[Lamps](#)[Others](#)[Tables](#)



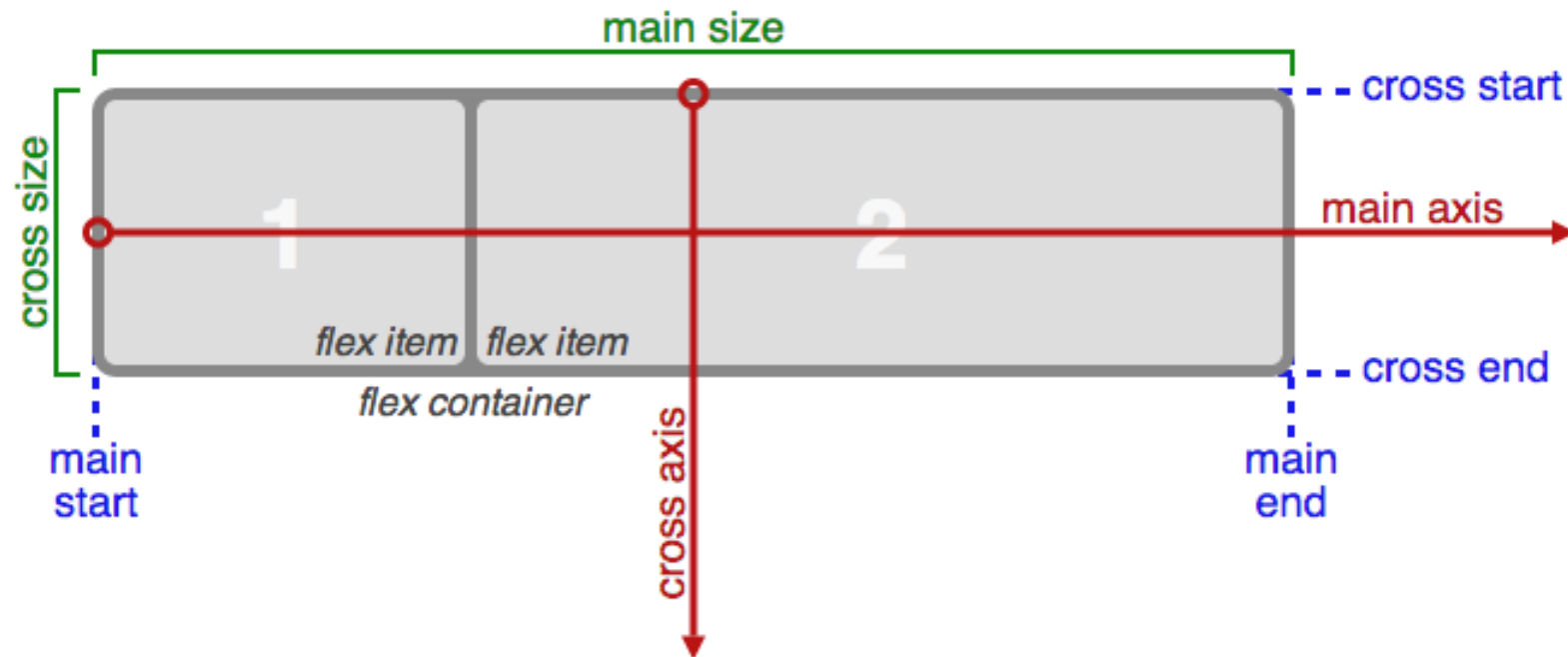
## ¿QUÉ ES FLEXBOX?

- ▶ FLEXBOX es un módulo completo de layout, es decir define cómo se muestran los elementos y cómo se relacionan con el resto.
- ▶ El contenedor va a poder modificar las dimensiones y el orden de los items para acomodarlos según nuestras indicaciones.
- ▶ ¿Qué cosas podemos hacer con Flexbox que antes no podíamos hacer de manera fácil?
  - ▶ Alineación vertical
  - ▶ Columnas de igual altura independientemente del contenido
  - ▶ Cambiar el orden en el que se MUESTRAN los elementos sin que cambie el HTML
  - ▶ ...Y MUCHAS MÁS

- ▶ **SÍ** - Salvo que tengas la mala suerte de tener que dar soporte a IE antiguos
- ▶ **AUTOPREFIXER** - Preprocesadores, PostCSS, gulp, grunt...

[illegible]

## LOS EJES



- ▶ Guía muy completa:

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

## EJEMPLOS CODEPEN CURSO

- ▶ CODEPEN COLLECTION: "Taller Flexbox profesional"  
<http://codepen.io/collection/nYWkGe/>

## EJEMPLOS y EJERCICIOS DE APLICACIÓN (en orden de aparición)

EJEMPLO 1: display:flex vs display:inline-flex

EJEMPLO 2 : Resumen de propiedades de la caja

EJEMPLO 2.1 : Resumen de propiedades de la caja aplicado al ejemplo de inline-flex

### EJERCICIO 1: Totales

TOTAL FACTURADO	95.430,00€
-----------------	------------

### EJERCICIO 2: Items tipo app sencillos

Cómo son	>
Cuándo y dónde compran	>

## EJEMPLO 3: Centrado total (horizontal y vertical) - Caja centrada con contenido también centrado



## EJERCICIO 3: Header Throne



**THRONE**

[HOME](#)

[ABOUT](#)

[CLIENTS](#)

[PORTFOLIO](#)

[SERVICES](#)

[BLOG](#)

[CONTACT](#)





## ¿CÓMO SE RESPETAN LAS DIMENSIONES DE LOS ITEMS?

- ▶ **CROSS SIZE: (EJE SECUNDARIO - CROSS AXIS)**
  - ▶ Si se ha definido (por width o height), ese tamaño se respetará.
  - ▶ Si no se ha definido, se utilizará todo el espacio disponible (STRETCH).
  - ▶ Si no se ha definido y se utiliza un valor diferente de stretch para align-content o align-items en el contenedor, se tomará el tamaño de su contenido.
- ▶ **MAIN SIZE: (EJE PRINCIPAL - MAIN AXIS)**
  - ▶ Si no se ha definido el tamaño, se calculará según el contenido.
  - ▶ Si se ha definido (por width o height) esté podría respetarse, podría encogerse, o crecer.



## ESPACIO OCUPADO Y ESPACIO DISPONIBLE

- ▶ **ESPACIO OCUPADO:** es la suma de los main-size de todos los flex-items en la misma línea, más los márgenes si los tienen.
- ▶ **ESPACIO DISPONIBLE:** es la diferencia entre el tamaño del contenedor y el espacio ocupado. Puede ser **positivo** (sobra espacio) o **negativo** (los items suman más que el contenedor).
- ▶ **Si NO HAY ESPACIO SUFICIENTE**, por defecto los items **ENCOGEN** para caber dentro del contenedor. (Los márgenes nunca colapsan)
- ▶ **Si HAY ESPACIO SUFICIENTE**, por defecto **NO CRECEN**, porque Flexbox quiere que le digamos **CÓMO QUEREMOS QUE CREZCAN**.

## FLEX-BASIS

- ▶ Flex-basis es el tamaño base que se considera para los cálculos, NO EL TAMAÑO DEFINITIVO.

Es decir, que podrá variar (crecer o encogerse), según los valores de flex-grow, flex-shrink y el tamaño del contenedor.

- ▶ IMPORTANTE:
  - ▶ Flex-basis siempre gana a width o height.
  - ▶ Si no se define flex-basis o se establece en auto, se tomará en cuenta el valor de width o height, o el tamaño del contenido si no se han definido. ESO NO SIGNIFICA QUE SE RESPETE, SINO QUE COMO FLEX-BASIS SE TOMARÁN ESOS.

## FLEX-BASIS <-> WIDTH

- ▶ Flex-basis no siempre controla el ancho, en **flex-direction:column**, flex-basis controla el ALTO.
- ▶ Flex-basis solo funciona sobre el main-axis, de modo que si estoy trabajando con **flex-direction:column**, tendré que especificar **width** si quiero establecerlo.
- ▶ Flex-basis gana -> si utilizo la propiedad **FLEX** que es el shorthand de **[flex-grow] [flex-shrink] [flex-basis]** , sobrecribiré **width** sin darme cuenta.
- ▶ En **responsive** es fácil que pase de **flex-direction: row** a **column**, si establezco **width** tendré problemas.

## FLEX-BASIS <-> WIDTH



Así es que, queridos niños, aunque nos cueste al principio, usemos

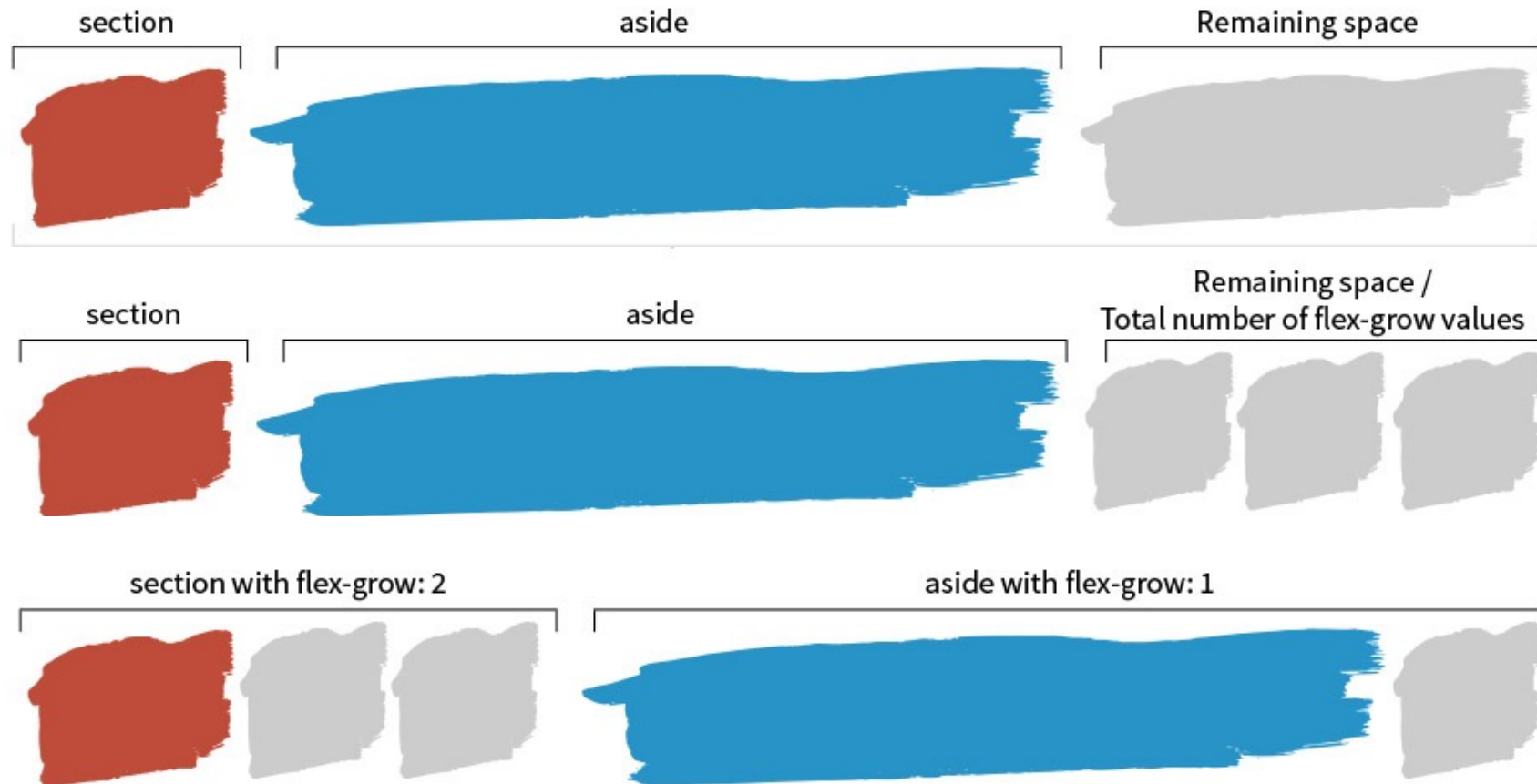
**FLEX-BASIS**

## FLEX-GROW (CRECIMIENTO)

- ▶ Controla cuánto crece un elemento para rellenar el espacio sobrante. Por defecto NO CRECEN (FLEX-GROW:0)
- ▶ Sólo se aplica si hay espacio disponible
- ▶ Es un número POSITIVO → UNIDADES QUE CRECERÁ

**UNIDAD = ESPACIO DISPONIBLE / SUMA DE FLEX-GROWS EN LA MISMA LÍNEA**

## FLEX-GROW (CRECIMIENTO)



<https://css-tricks.com/flex-grow-is-weird/>

EJEMPLO 4: FLEX-GROW



## FLEX-SHRINK (ESTRECHAMIENTO)

- ▶ Si el espacio disponible es negativo (el tamaño del contenedor es menor a la suma de los tamaños de los items), por defecto los items se encogen en proporciones iguales (FLEX-SHRINK:1) para caber en una sola línea , pero respetando el contenido o si tiene establecido min-width o min-height.

UNIDAD = ESPACIO DISPONIBLE (SERÁ NEGATIVO) / SUMA DE FLEX-SHRINK EN LA MISMA LÍNEA

### EJEMPLO 5: FLEX-SHRINK