It is a new module in CSS3 design to improve the way we do Responsive Design, thus avoiding the use of float (Do not use float to responsive design). Thanks to FlexBox facilitate how we position elements, is simpler and use less code

Flexus made to improve the items align, directions and order in the container even when they are with dynamic or even unknown. thus more complex layouts can be achieved with less code thus solving common responsive web design problems easily.The prime characteristic of the flex container is the ability to modify the width or height of its children to fill the available space in the best possible way on different screen sizes.

layout mode providing for the arrangement of elements on a page such that the elements behave predictably when the page layout must accommodate different screen sizes and different display devices. Forgot the classes like xs, sm, lg, portrait, landscape etc. You will not have to declarate classes for each device or resolution anymore. Just write one class and you will have a full-responsive website. Technology means progress, see how easy CSS layout will become once Flexbox turns mainstream.

or flexbox, is a layout mode providing for the arrangement of elements on a page such that the elements behave predictably when the page layout must accommodate different screen sizes and different display devices.

That means with flex box, we can freely arrange the layout and blocks with sets of property from aligning, spacing to setting the order of the blocks. Sounds really cool but what about the browser compatibility.

Buttons, tags, tables, forms, images with

multiple styles and much more!

Flexbox grid ideas

-Utilizar una grilla basada en flex.basis, esta propiedad hace como si fuera col de boostrap pero manejado en pixeles o rems, y hace un breakpoint sin necesitdad de utilizar medias queries,

-dejar de utilizar width

---pixeles-----

-flex: 300px = flex 1 1 300px

solo usar:

flex-basis: 300px

(ocupa el ancho restante)

flex 0 1 300px

(deja a otro elemento ocupar todo el ancho y solo deja 300px para el

nunca usar flex shrink en 0

flex-basis cambia a height en lugar de width cuando el contenedor es column

ultizar mejor 1-1-100% 0 0-1-200px

en lugar de utilizar column

si quieres utilizar flex-basis con pixeles tienes que aplicarlesela al padre, si el padre tiene solo "flex-3" por ejemplo los hijos aunque tengan flex:1-1-300px no respetaran los 300px para que hagan el breakpoint

nunca poner display flex al contenedor directo de los botones, en cualquier caso encerrarlo en un contenedor . Y ese contenedor padre dentro de otro contenedor abuelo donde sera display flex. asi podras moverlo a tu antojo justify y align

si un elemento padre es column todos sus elementos hijos directos tendran un largo extra,

----

Con estas propiedades puedes redimencinar al 100% una imagen, y su contenedor con propiedad auto para chrome

min-width: 100%;

width: 150px;

para firefox y los demas asignarle column al contenedor y flex-1-1-150

y a la imagen darle un width y heigth 100%

prueba a poner column con la propiedad auto al contenedor y a la imagen min-width de 100% y en width el tamaño que se requiera.

Asi lograrias que la imagen nose encogiera ni se desproporcionara.

colores: TED #009688

green: #4CAF50

---

Si tienes una imagen con un width determinado y le pones una clase flex-0-1-200 puedes controlar el tamaño de una imagen con eso, siempre y cuando el contenedor padre de la imagen tenga las clases flex y algun tipo de width como del 100% o flex-1 o flex-1-1-250,

si al padre le quitas la clase flex la imagen hija quedara se centrara

imagen responsiva con flex:

contenedor.

<div class="flex-0-1-150">

y a la imagen se le da un w100(width 100%)  
pd: no poner flex al contenedor)

3 opcion:

column flex-0-1-400 al padre

puedes crear un column sin flex y sin que se inviertacon:  
column > div=width:100%

y a los divs aplicarles clases normales como flex-0-1-500

pd: encontrar un nuevo nombre para column como flex-column para flex column y column para column con 100%;

puedes crear un cuadrado con flexus dandole al padre column y flex-1-1-200, enseguida al hijo un flex-1-1-200 y tienes tu cuadrado

PD. el papa de column no tiene que ser column, tiene que ser flex o en su defecto a column sustituirlo por flex

----

Al poner al papa Column los hijos se ocupan el 100% pero tu puedes corregir eso haciendo que solo ocupen la cantidad de espacio que necesitan poniendo junto con column align-start o end al padre asi no ocupara el 100 y siempre se mantendran en columnas, nunca se iran a compartir el espacio de arriba aunque quepan por que son hijos de column, pD si solo quieres aplicarlo a un solo elemento puedes aplicar self-start a un hijo.

---------------

si quieres cambiar una palabra de color de un parrafo usa la etiqueta p como padre y span en el texto que quieres cambiar

Simple. Robust. Responsive. Flexible.

Juiced is built on Flexbox which works for over 94.5% of the Internet users across the globe. It is better than using floats bar none. Plus, we've added a few things we think you'll like. see just how easy CSS layout will become once Flexbox becomes mainstream.

Technology means progress, so people should use more recent browsers.

Works on virtually everything

— mobile phones, tablets, notebooks,

desktops, retina and big screens

Piensa dos veces antes de empezar a programar, o terminarás programando dos veces antes de empezar a pensar.

backgrounds:

background: linear-gradient(to top right, #34b3a0 0%, #204056 100%);

ejemplo de not y hover juntos:

[class\*="btn"].bg-moon:not(.btn-tag-text):hover

-imagenes sin distorciones

imagenes con medida en pixeles

auto para firefox aunque puede omitirse