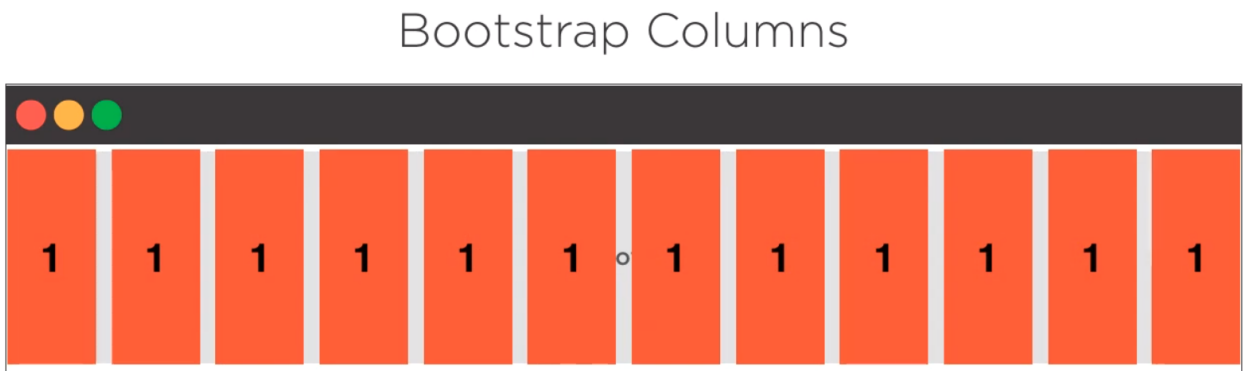
Creating Flexbox Layouts

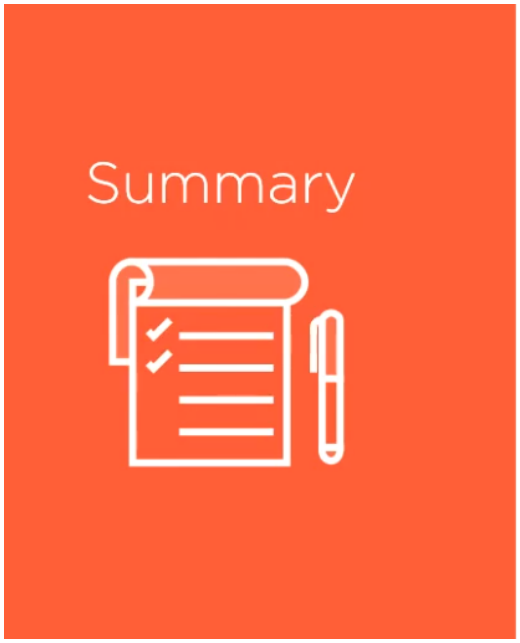
Cada fila en bootstrap se puede dividir en 12 columnas diferentes



Aquí se muestra un ejemplo donde usamos 7 columnas para texto y 5 para imagen.

Cuando se requiere que en una página sean menos columnas y al dividirse sea necesario tomas de 1.5 columnas, es donde se vuelve difícil, es por eso que FlexBox, ha sido de gran ayuda.
Flexbox no define el número de columnas, flexbox es un nuevo modo de diseño en css3.
El diseño de flexbox distribuye el espacio a lo largo de una sola columna o fila que no se limita a tamaños de columnas específicos, es similar a los diseños flotantes, pero mejor!.
Flexbox se puede dividir de la forma que se desee y se puede definir uno para que ocupe más espacio que el otro.
Una de las cosas buenas de flexbox es el hecho de que no siempre se tiene que definir el tamaño y es tan flexible que solo ocupara la cantidad de espacio disponible.
Flexbox permite crear filas y columnas (vertical y horizontalmente), para ello se usa flex-row ay flex-column.
Es buena práctica agregar flex-row para no tener problemas con los navegadores.
Flex-row-revert, te muestra tu fila del lado contrario, esta aplica a las columnas de igual manera (flex-column-revert).

Resumen



Getting started

Explored starter and sample files

Flex containers

Row vs column containers

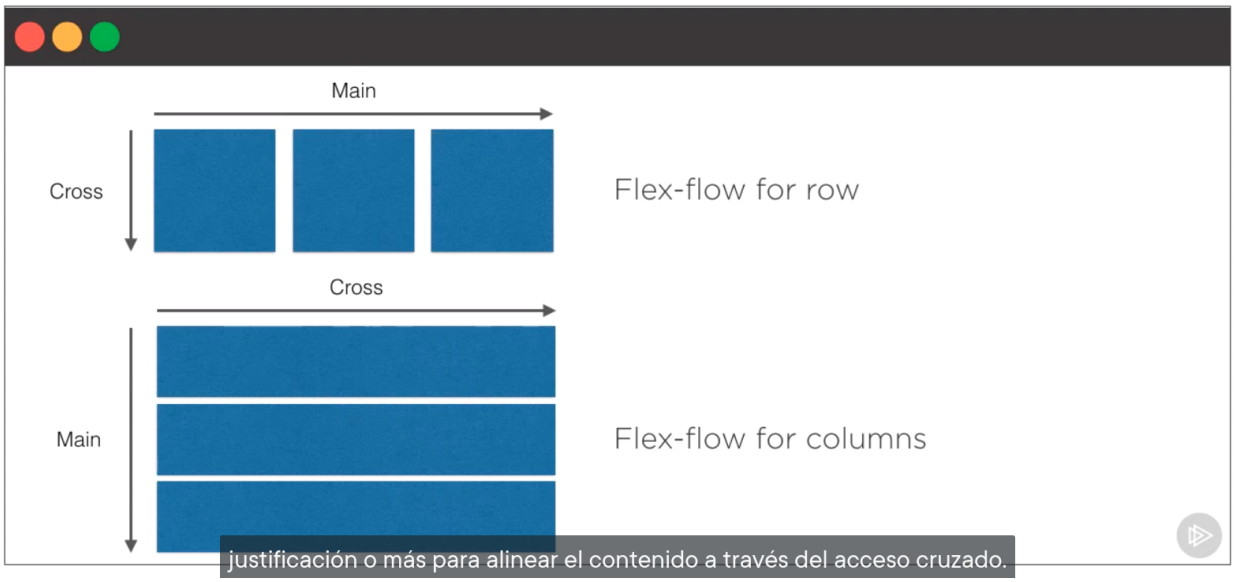
Container direction

Combining vertical & horizontal containers

Bootstrap sizes

Adjusting Alignment, Margins, and Display Order

Main Axis and Cross Axis



Toda la información acerca de justify content, se encuentra aqui:
https://getbootstrap.com/docs/5.0/utilities/flex/

Margin y Padding

Donde la propiedad es una de:
m - para clases que establecen margin
p - para clases que establecen padding

Donde lados es uno de:
t- para clases que establecen margin-top o padding-top
b- para clases que establecen margin-bottom o padding-bottom
s- (inicio) para clases que se establezca margin-left o padding-left en LTR, margin-right o padding-right en RTL
e- (fin) para clases que se establezca margin-right o padding-righten LTR, margin-left o padding-left en RTL

x- para clases que establecen tanto *-lefty*-right
y- para clases que establecen tanto *-topy*-bottom
en blanco: para clases que establecen a margin o padding en los 4 lados del elemento
Donde el tamaño es uno de:

0- para clases que eliminan el margino paddingconfigurándolo en0
1- (por defecto) para las clases que establecen el margino paddingen$spacer * .25
2- (por defecto) para las clases que establecen el margino paddingen$spacer * .5
3- (por defecto) para las clases que establecen el margino paddingen$spacer
4- (por defecto) para las clases que establecen el margino paddingen$spacer * 1.5
5- (por defecto) para las clases que establecen el margino paddingen$spacer * 3
auto- para clases que configuran el marginen automático

Fill Grow and shrink

*Use **flex-fill** en una serie de elementos hermanos para forzarlos a anchos iguales a su contenido (o anchos iguales si su contenido no supera sus cuadros de borde) mientras ocupa todo el espacio horizontal disponible.
*Utilice **flex-grow-***para alternar la capacidad de un elemento flexible de crecer para llenar el espacio disponible. En el siguiente ejemplo, **flex-grow-1** utiliza todo el espacio disponible que pueden, mientras que permiten que los dos elementos flexibles restantes tengan el espacio necesario.
*Utilice **flex-shrink-***para alternar la capacidad de un elemento flexible de encogerse si es necesario. En el siguiente ejemplo, el segundo elemento flexible **flex-shrink-1** se ve obligado a ajustar su contenido a una nueva línea, "encogiéndose" para permitir más espacio para el elemento flexible anterior con .w-100.

Auto Margins

Flexbox puede hacer cosas increíbles cuando mezcla alineaciones flexibles con márgenes automáticos. A continuación se muestran tres ejemplos de control de elementos flexibles mediante márgenes automáticos:v mx-auto(da un margen automático a los 2 lados).

Adapting Content

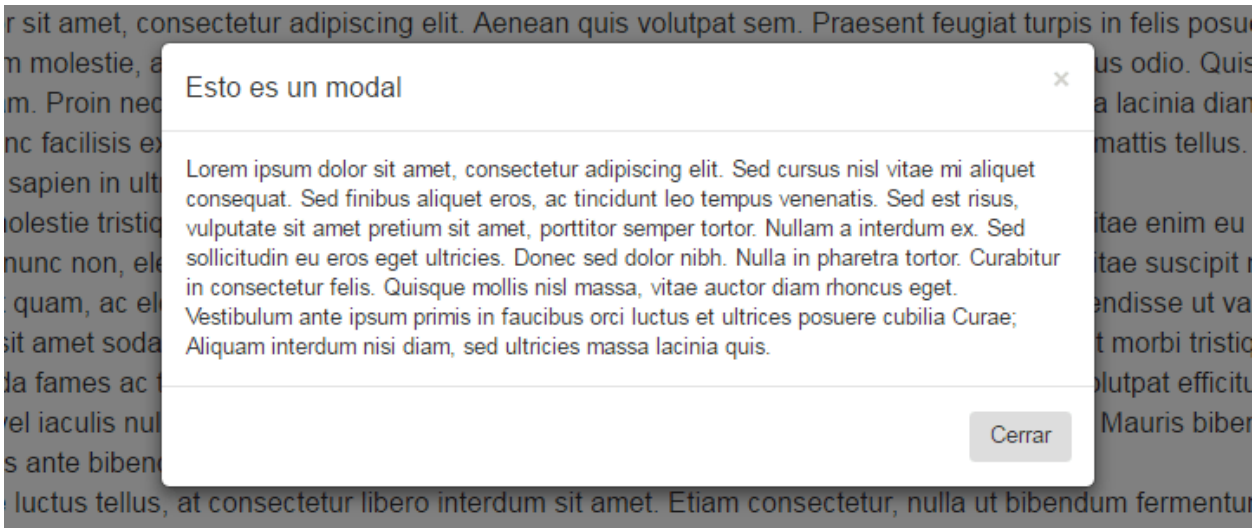
Cambia cómo se envuelven los artículos flexibles en un contenedor flexible. Elija entre no envolver en absoluto (el navegador predeterminado) con **flex-nowrap**, envolver con **flex-wrap** o envolver al revés con **flex-wrap-reverse**.
Cuenta con elementos responsivos agregando:
sm
md
lg
xl
xxl
De la siguiente manera:
flex-sm-row(cuando la pantalla sea pequeña, pondrá ciertos elementos como fila).

Adding in Other Bootstrap Layout Utilities

Modal

Seguro que, en algún proyecto, el cliente te ha sugerido mostrar un mensaje o cierta información al usuario que destaque respecto al contenido de la página. Ya sea cuando pulsamos un botón o automáticamente cuando el usuario lleve cierto tiempo en el artículo o página de la susodicha web. Esto nos da una idea de lo que conocemos por modal.

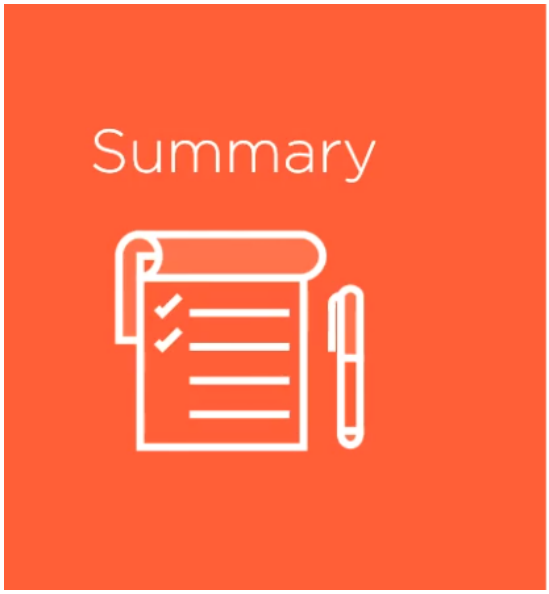
Es, por tanto, es una capa que se superpone a las demás quedando en primer plano de nuestra web y dejando en un segundo plano el resto de la página. Para que el usuario no capte la atención en otro componente que no sea el modal, se suele crear este dentro de otra capa que ocupa el 100% del ancho y del alto de la ventana con un mínimo de opacidad y tono oscuro.



Mas informacion aqui :
https://blog.gtoeurope.es/crear-modal-en-bootstrap/

Changing images and visibility based on size

Resumen



Finalizing Our Project Before Deployment

JSON

JavaScript Object Notation (JSON) es un formato basado en texto estándar para representar datos estructurados en la sintaxis de objetos de JavaScript. Es comúnmente utilizado para transmitir datos en aplicaciones web (por ejemplo: enviar algunos datos desde el servidor al cliente, así estos datos pueden ser mostrados en páginas web, o viceversa).
En el caso del juego web que se está creando, el JSON nos sirve para mandar las preguntas y respuestas a nuestras cards.
Con un Archivo JS se procesan los datos de dicho JSON para poder enviar lo necesario a nuestro HTML.

Resumen

JSON data

Showing questions

Finalizing your files