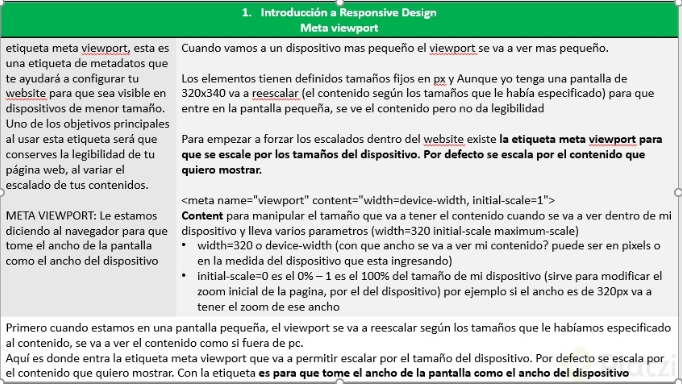
5- Meta viewport

Notas: Para adaptar nuestro web site a cualquier medida de pantalla  
<meta name="viewport" content="width=device-width, initial-scale=1">

La propiedad **initial-scale** controla el nivel de zoom cuando la página se carga por primera vez.  
Las propiedades maximum-scale, minimum-scale, y user-scalable controlan la forma en cómo se permite a los usuarios aumentar o disminuir el zoom en la página.

La propiedad **width** controla el tamaño del viewport.  
Puede definirse con un número en pixeles como width=600 o con un valor especial device-width que es el equivalente al ancho de la pantalla en píxeles CSS en una escala de 100%.



**6- Medidas relativas útiles en Responsive Design**

En esta clase aprenderás las medidas que existen en el campo del responsive design.

Lo primero que debes tener en cuenta es que estas medidas son maleables, en la medida en que dependen de su fuente de origen o medida madre. Entre ellas se encuentran el porcentaje (longitud referente al tamaño de los elementos padre), los em (unidad relativa al tamaño de fuente especificada más cercano), los rem (unidad relativa al tamaño de fuente especificada en el ancestro más lejano, como html o body) y tamaños del viewport vw/vh (longitud relativa porcentual con respecto al viewport).

**Medidas útiles en Responsive Design**

%: Porcentaje, se mide en referencia a la longitud de los elementos padres.

em: Unidad relativa al tamaño de fuente mas cercana, la fuente mas cercana escala primero al elemento mismo, sino lo hace con el elemento padre mas cercano que tenga fuente.

rem: Unidad relativa al tamaño de fuente mas lejana (html o body).

vw/vh: Unidad relativa conceptual en relación al Viewport.

100 vw: 100% del width con respecto al Viewport actual.

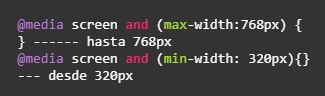
100 vh: 100% del height con respecto al Viewport actual.

**7- Media queries**

Para que logres los resultados que deseas en tus proyectos, es necesario cambiar ciertas propiedades para modificar el tamaño de los textos, contenidos y hojas de estilo; la manera de hacer esto es el media queries.

El media queries es un módulo de css que hace posible al responsive design, éste existe desde el 2010 y se encarga de adaptar la representación del contenido a características del dispositivo.

En esta clase conocerás cómo funciona su estructura, cómo se construye y adquirirás los conocimientos necesarios para trabajar con él, desde tu editor de código.



Gracias a este módulo, existe responsive design

Concepto: Modulo de CSS3, que permite la adaptación del contenido de la webpage a las características del dispositivo en uso.

Composición:

@media media type and (condición) { }

Ejemplos:

@media screen and (max-width: 768px) { }

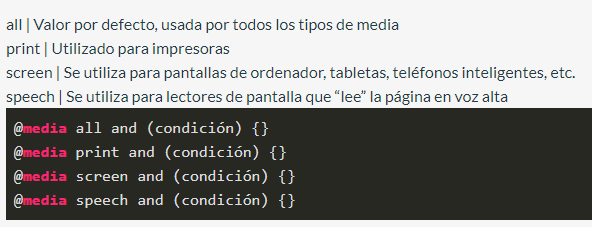
Configuracion responsive para toda pantalla con un ancho máximo de 768 pixeles.

@media screen and (max-width: 768px) and (min-width: 480px) { }

Configuracion responsive para toda pantalla con un ancho entre 480px y 768px.

Como dato importante, si ya tienen la version desktop y quieren aplicar media query siempre va al final del estilo.css.

Por querer tenerlo ordenado lo puse al principio de todo el archivo .css y lo de abajo sobreescribia la media, pase como 30min tratando de lidiar con eso.



**8- Formas de incluir media queries**

En esta clase aprenderás a insertar un media querie en tu proyecto. Para ello, vas a trabajar sobre tu hoja de estilos, utilizando el tag style.

El primer paso para lograr esto será realizar una nueva hoja de estilos en tu proyecto, ésta debe contar, en primer lugar, con la etiqueta link; harás uso de la aplicación de medidas para la pantalla, bordes y colores, entre otras características.

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

10- con display flex(sería al padre) y justify-compain: center también centras el contenido y va directamente al elemento que se cambia-

En display-block y text-alig center, centramos el elemento directamente

Para un diseño responsive con desktop, se comienza desde la resolución más grande hasta la más pequeña….

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

11-

**Pasos para hacer imagenes responsive:**  
I. En los estilos generales  
a. Declarar el width de la imagen y el max-width: 100%;  
b. Declarar el width del contenedor de la imagen en medidas relativas

II. En la Query  
a. Declarar el width: auto en el contenedor de la imagen.

**Pasos para hacer textos responsive:**  
I. En la Query  
a. Declarar un font-size en px al elemento que contiene todos los textos, el cual servirá como guía para el em  
b. Declarar los em para cada texto en la primera query

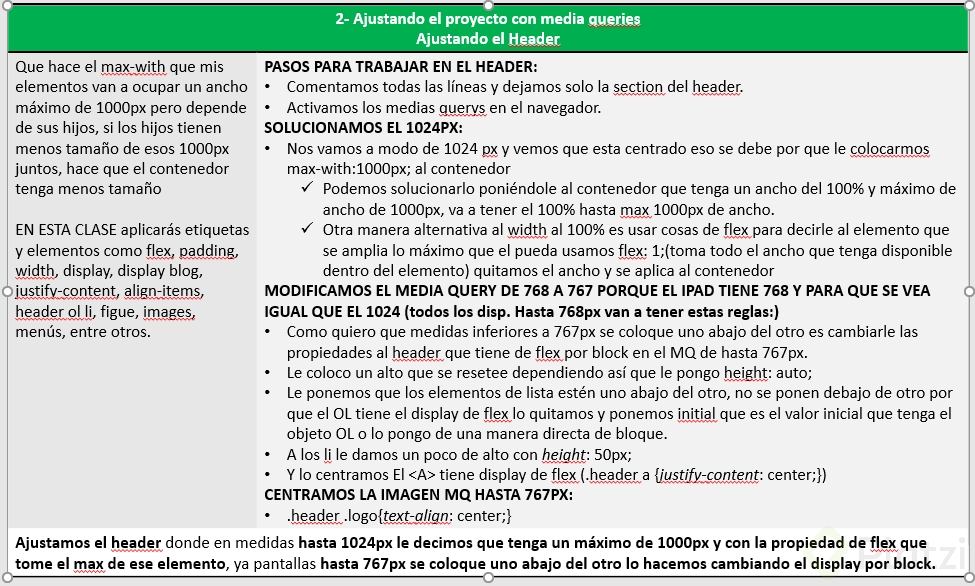
*NOTA*  
a. Al darse la necesidad de redimensionar los tamaños del texto en otra query (o la proxima) el único valor que se deberá cambiar para que todos los textos configurados relativamente se adapten automaticamente es el font-size del contenedor previamente declarado en pixeles.

**Anotaciones:**

**Es recomendable:**

* Usar porcentajes a la hora de distribuir la asignación del ancho de sub elementos de secciones.
* Usar display con valor block para en las media queries.
* Asignar width con valor auto para imágenes.
* Para volver una imagen responsive, se le asigna el 100% del ancho máximo (max-width) del elemento padre.

Se deben ajustar los valores de texto, para ello es indispensable inspeccionar que elementos heredan el Font-size directamente del contenedor y aquellos que no lo hacen. Se deben usar medidas relativas (preferiblemente los em), para ajustar este tipo de tamaños a las fuentes, de la misma forma para las medidas de margin que se deben ir ajustando proporcionalmente con medidas relativas.



-Aquí un repaso rápido de las propiedades de flexbox [css-tricks.com](https://css-tricks.com/snippets/css/a-guide-to-flexbox/).

-Este sitio me gusto muchísimo, un estudiante lo compartió en el curso de Desarrollo web y es de lo mejor  
<https://flexboxfroggy.com/#es>

NOTA: al momento de manejar un elemento como es en Project y quieres que no se salga de la línea y renderice bien en los tamaños, se recomienda el width en auto

NOTA2: img responsive, se da el 100% del padre que es el 50% del contenedor de los 1000px que es el max-width

**Tener en cuenta**



:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

13- anotación sobre las imágenes

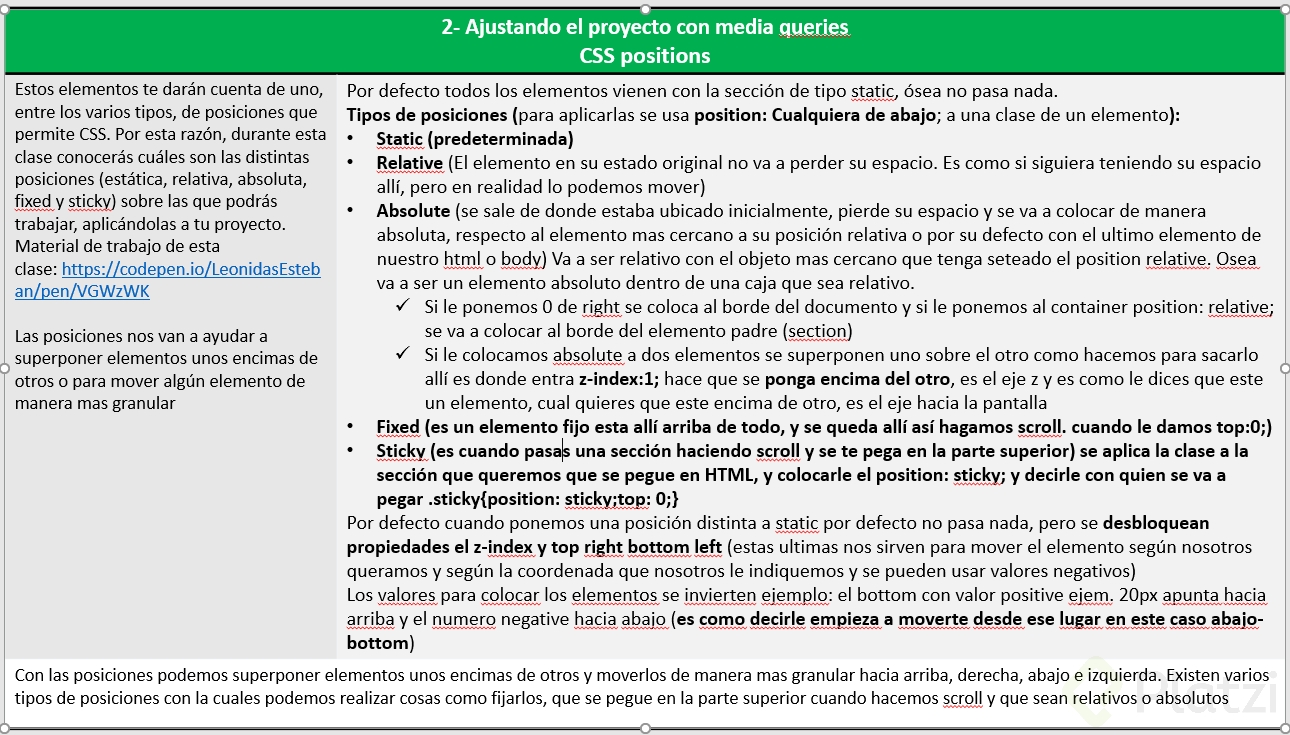
**Anotaciones:**

Cuando hay unidades de márgenes o relleno, es recomendable hacer la sumatoria al momento de ajustar porcentajes a las unidades de ancho. Ejemplo: 48% width (para repartir entre 2 elementos), mas 1% de margin, que en total sumarian 48% + 48% + 1% + 1% +1% +1%.

Cuando agregamos una medida a un selector hijo de etiqueta, y queremos hacer una configuración posterior pero con el selector de clase, el selector de etiqueta prevalece y vale más, por lo que se debe hacer la reconfiguración con el selector de etiqueta.

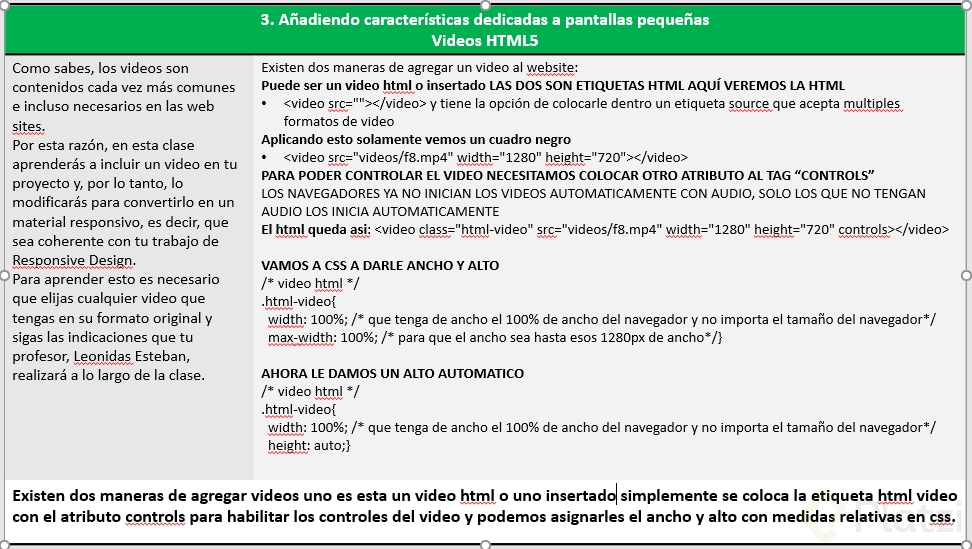
Ej: .event-image valdra en este caso menos que .event img

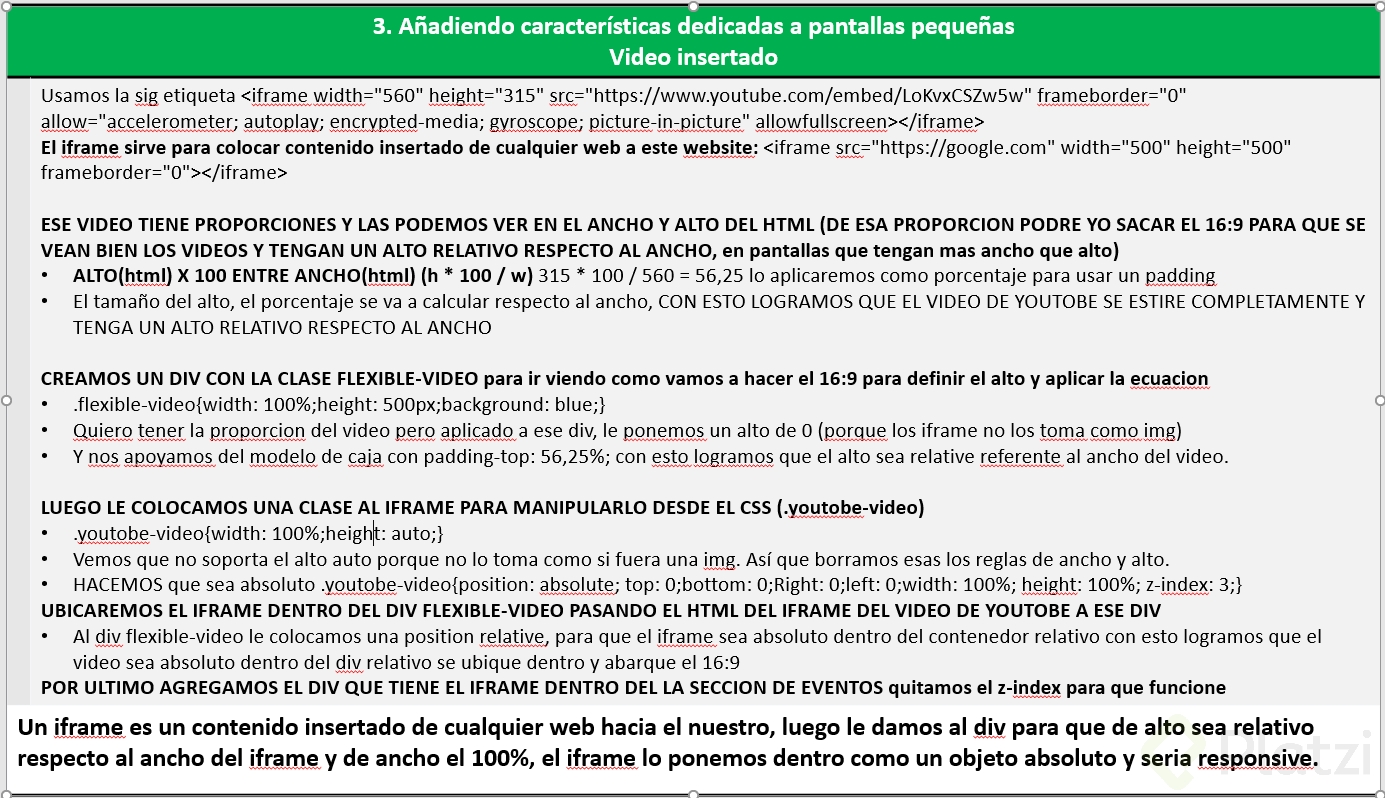
:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

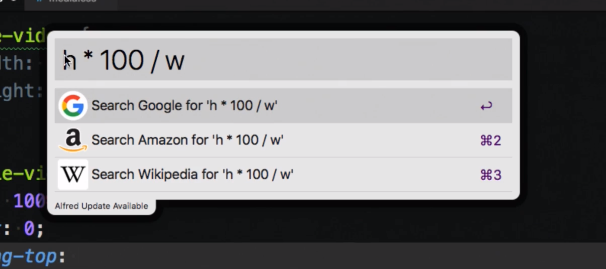
15 – Css positions

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

17- Videos HTML5



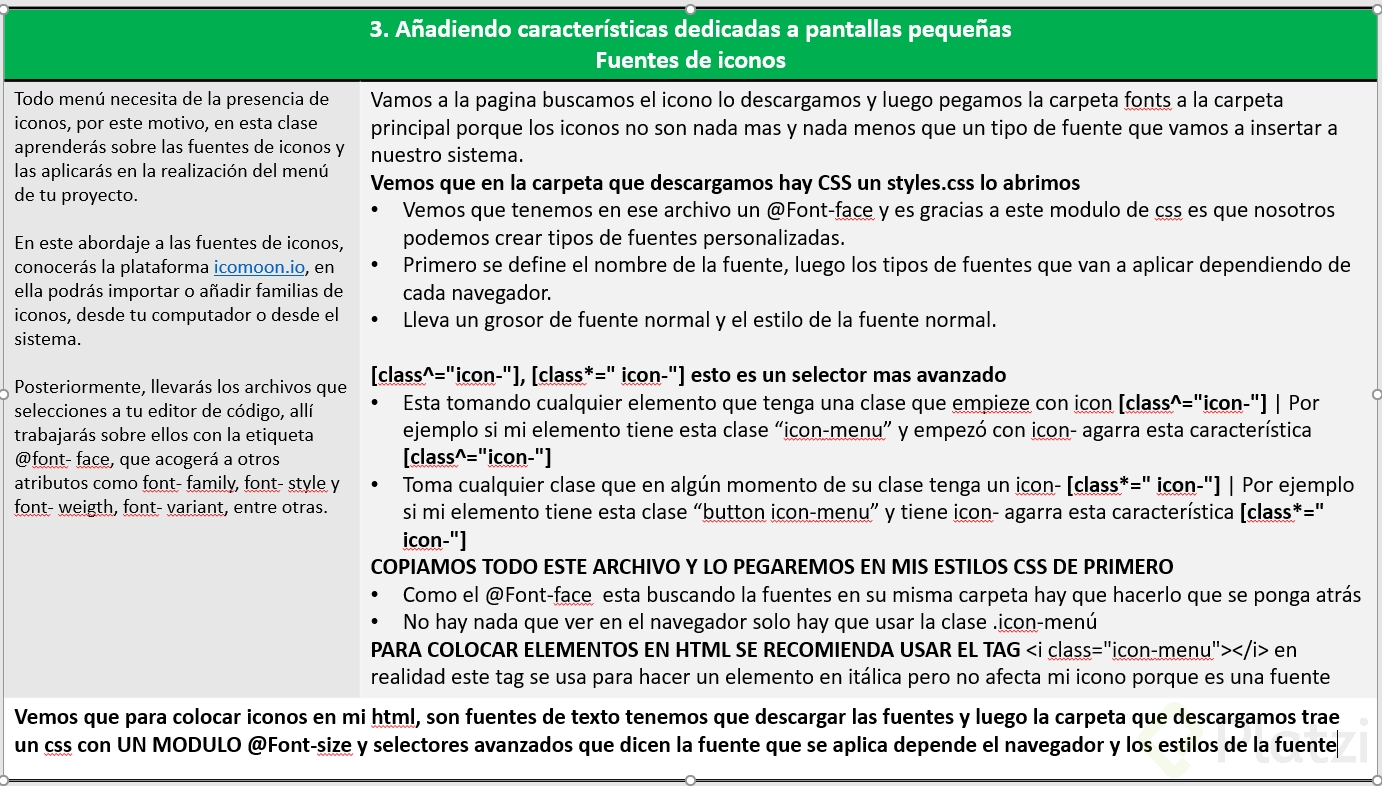
-18 Video insertado



Para hacer responsive una div -los datos se sacan desde la etiqueta ifrem



:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

19- Fuentes de iconos

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

20 Menu hamburguesa

**Menu de hamburguesa**

Una buena manera de ubicar el menú, es con la propiedad de position, con valor fixed, para que el elemento tenga permanencia en el viewport.

Para que el elemento desaparezca mientras estamos trabajando en pantallas grandes debemos ponerle un display con valor none. Solo se debe mostrar cuando trabajemos con pantallas pequeñas. Por lo que debemos configurar a partir del ipad para que sea visible.

Nota: recordemos que al menú de hamburguesa le configuramos un display de block, ya que dejandole el display por defecto de inline, no va a tomar el ancho y alto que le asignemos.

Nota: el icono es un texto se cambi el color con color y tamaño con Font-size

Nota: Para centrar el elemento con display block lo haremos con line-heigth con el mismo alto del height y luego un text aligne center (hay que centrarlo vertical y horizontalmente)

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

21- Posicionando menú

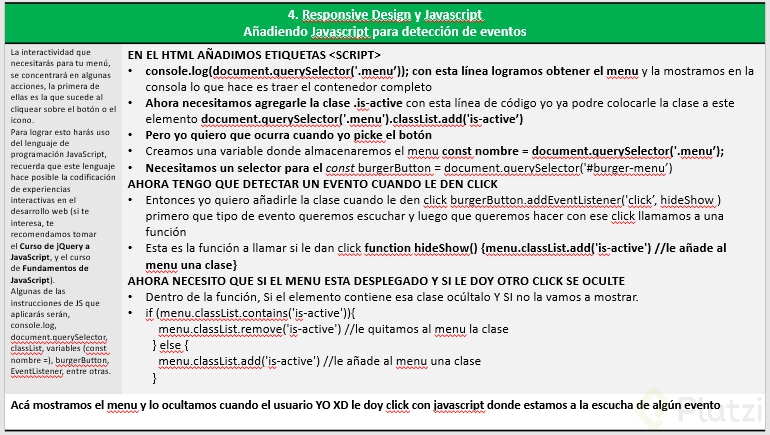
Para poder posicionar el menú, debemos configurarle primeramente la propiedad de position con valor fixed y su respectivo valor en la propiedad z-index (de 3 para que no se superponga al menú de hamburguesa).

Para determinar un color en rgba, podemos ubicarnos en el inspector de elementos, desplegar la propiedad background que tenga en el momento dicho elemento y ubicarnos en su color de rgba.

Otra manera de centrar el texto en el menú aparte de las propiedades de centrado, se puede lograr añadiéndole un valor a bottom, de 0 y quitando la altura (que eventualmente heredara la altura del elemento padre, la cual es auto).

Para añadir una transición a algun elemento en css, podemos usar la propiedad transition y añadirle un valor de tiempo (ejemplo: .3s).

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

22- Añadiendo Javascript para detección de eventos

La interactividad que necesitarás para tu menú, se concentrará en algunas acciones, la primera de ellas es la que sucede al cliquear sobre el botón o el icono.

Para lograr esto harás uso del lenguaje de programación JavaScript, recuerda que este lenguaje hace posible la codificación de experiencias interactivas en el desarrollo web (si te interesa, te recomendamos tomar el Curso de jQuery a JavaScript, y el curso de Fundamentos de JavaScript).

Algunas de las instrucciones de JS que aplicarás serán, console.log, document.querySelector, classList, variables (const nombre =), burgerButton, EventListener, entre otras.

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

23-