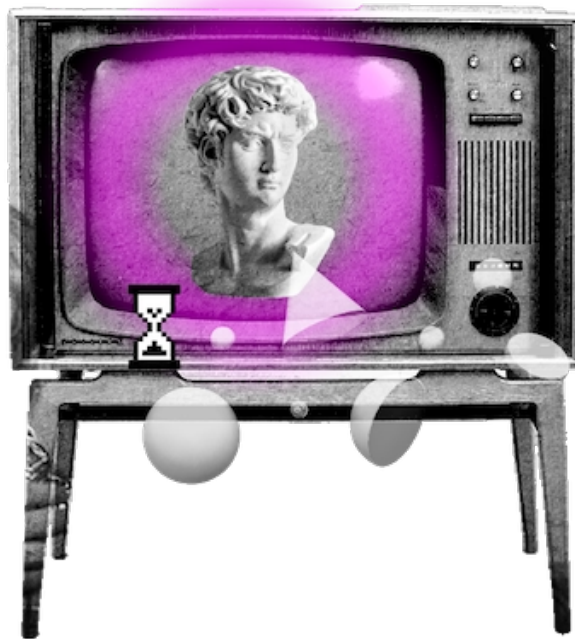


DISEÑO WEB

NUCBA NFT - RESPONSIVE DESIGN



APUNTES





Livecoding: Nucba NFT - Responsive Design

Objetivo de la clase

El objetivo de esta clase es poner en práctica los conceptos que hemos aprendido la clase pasada relacionados al diseño responsivo y el uso de mediaqueries.

Trabajamos sobre la base del proyecto **Nucba NFT** que comenzamos hace algunas clases y lo transformamos en una página web que sea adaptable a todos los dispositivos, incluyendo la confección de un menú de barras, también conocido como menú hamburguesa, que permita adaptar nuestra barra de navegación a los dispositivos más pequeños.

Propiedades ya colocadas que ayudan al responsive design

Si bien el grueso de la clase consta de preparar nuestro menú y aplicar mediaqueries para adaptar nuestra página a dispositivos más pequeños, hay algunas cosas que ya han sido determinadas en las dos clases en las que realizamos el diseño desktop de nuestra página y nos han ayudado en la adaptación.

Max-width

La propiedad **max-width** la hemos utilizado en el diseño realizado en la primera iteración del proyecto para que nuestras secciones tengan un límite de crecimiento en pantallas muy grandes, con el fin de que las mismas no queden demasiado estiradas a lo ancho en dichas pantallas. Para que esto funcione, además de la propiedad **max-width**, hemos centrado los contenidos de cada sección. Esto lo hemos hecho centrando el contenido de la etiqueta **main**, que nos ayuda a centrar todo el contenido principal de la página, limitado en su ancho por el **max-width**.

Flex-wrap

La propiedad **flex-wrap** también la hemos utilizado en la iteración anterior del proyecto, y nos ayuda a adaptar nuestro diseño a pantallas mas pequeñas ya que hace que nuestras **cards**, en caso de no tener espacio para colocarse en la fila en la que están, se coloquen debajo, haciendo que nuestras secciones con cards se vayan acomodando al tamaño del **viewport** colocándose una abajo de la otra en las resoluciones mas pequeñas.



Menú responsive

Un **menú de barras o hamburguesa** es una **forma de presentar opciones de navegación** en un sitio web. Se le llama "**menú de hamburguesa**" porque el icono que representa este tipo de menú consiste en tres líneas horizontales apiladas que se asemejan a las capas de una hamburguesa.

Cuando se hace clic en el icono de la hamburguesa, se despliega un menú desplegable que contiene opciones de navegación para el sitio web. Este menú es comúnmente utilizado en sitios web responsivos que necesitan adaptarse a diferentes tamaños de pantalla, ya que **permite un uso eficiente del espacio en la pantalla, ocultando opciones de navegación que no son esenciales** en un momento dado.

Para realizar el menú hamburguesa solo con html y css, utilizamos una técnica que incluye el **selector de hermanos adyacentes ("+")**, un **input** y su etiqueta **label**.

En el código, creamos un menú que contiene un **input** de tipo **checkbox** vinculado a un label (el atributo for del label es igual al atributo id del input) y las respectivas opciones de navegación (un ul con sus li), justo debajo del input.

```
<nav class="navbar">
  <label for="menu-toggle" class="menu-label">
    
  </label>
  <input type="checkbox" id="menu-toggle" />
  <ul class="navbar-list">
    <li><a href="#">Home</a></li>
    <li><a href="#">Descubre</a></li>
    <li><a href="#">BTC</a></li>
    <li><a href="#">Youtube</a></li>
    <li><a href="#">Contactanos</a></li>
  </ul>
```

Luego, además de aplicar los estilos correspondientes para que el menú al desplegarse se muestre correctamente, hemos colocado algunas propiedades que nos permiten crear este tipo de menús.

En primer lugar, ocultamos nuestro **input** con "**display: none;**", ya que no queremos que el input de tipo **checkbox** sea visible, solo la imagen que está dentro del label mediante el cual manejaremos la aparición y desaparición del menú.



Atributo checked

El atributo **checked** de los inputs de tipo checkbox nos permite saber si un input de este tipo está o no chequeado (seleccionado).

Esto es fundamental para poder hacer nuestro menú hamburguesa solo con HTML y CSS, ya que es lo que nos permitirá mostrar o no el menú desplegable.

```
.navbar-list {  
  /*Estilos del navbar*/  
  display:none;  
}  
  
#menu-toggle:checked + .navbar-list {  
  display: flex;  
}
```

Sabiendo que al apretar en el label que está vinculado a nuestro input oculto se cambiará su estado **checked** , trabajamos la lógica del despliegue del menú.

En estado normal, nuestro menú tiene la propiedad **display:none**; , ya que no queremos que el menú se muestre cuando abramos nuestra página, sino que recién queremos que se vea cuando hagamos click en la imagen de barras de nuestro label.

Finalmente, para que cuando el input este seleccionado se despliegue nuestro menú, utilizamos **#menu-toggle:checked** , que nos permite saber si está o no chequeado nuestro input con id **menu-toggle** , y el selector de hermanos adyacentes que , en caso de que el input esté seleccionado, agregará a nuestro navbar la propiedad **display:flex**; , que es la que hará que nuestro menú se vea. En caso de que se vuelva a clicar el label estando el menú abierto, como el estado de nuestro input no es **checked** , se quitará la propiedad **display:flex**; , y nuestro desplegable se ocultará

Nucba tip: Sigán investigando sobre los temas aquí expuestos. Practiquen, apoyense en la documentación oficial y sobre todo mantengan la constancia y la curiosidad a medida que vayan aprendiendo cosas nuevas.

#HappyCoding 🚀