Bluee Digital Fullstack

Clase 1

(Ver lo de slack)

Que es la web?

Es un lugar donde podemos encontrar muchos documentos (imagenes, texto, videos, etc.).

Es como un libro. Ya que tiene paginas, imagenes, texto, informacion. Entonces la web es mas completo que un libro.

Esta constituido por paginas, donde cada pagina puede tener informacion en forma de texto, video, audio, imagenes.

Las web se encuentran dentro de los navegadores (google, edge, mozilla) que a su vez se encuentran dentro de sistemas operativos y son el programa principal de la PC. El SO nos sirve para conectar el hardware con el software (Windows, macOS Monterey, Linux).

Windows nombra a sus versiones con numeros, mientras que Monterey con nombres. Linux permite que cualquier programador pueda manipular el codigo fuente de si mismo (es de codigo abierto). Muchos programadores, partiendo de Linux, pudieron crear otros SO (Red Hat, Debian, ArchLinux, SUSE, Gentoo Linux, etc.).

Como decia, dentro de los SO pueden haber distintas aplicaciones hechas por programadores, entre ellas Word, Zoom, etc y tambien navegadores web. Estos ultimos nos sirven navegar sobre la internet red de redes. Estos ultimos nos permite escribir un dominio o texto para buscarlo. Si buscamos en este, nos va a mostrar un listado de posibles dominios.

Un dominio es una manera de encontrar algo en internet. Es un DNI, en la cual identificamos un objeto en algun lugar. En el mundo de la internet, seria un numero IP. Una IP es el conjunto de numeros que nos permite identificar una computadora en un momento dado. La IP es como el DNI de la computadora. Entonces, el dominio es la IP y hay dos tipos, publica y privada.

En conclusion, cuando buscamos un dominio en el navegador, nos muestra un listado de IPs de computadoras de todo el mundo.

NOTA: La DNS (sistema de nombre de dominios). Convirte dominios (nombre) a IP (numeros).

Los proveedores de internet, que pagamos para que nos bride internet. Ellos nos brindan un modem que nos da el internet por cable. La diferencia con el router es que nos da internet por wifi. Puede haber modem-router, que es la combinacion de las dos.

Nos brindan esta maquina que está conectada al proveedor de internet que esta concetado a su vez a un monton de computadoras con modem-routers. El proveedor nos hace posible conectar a estos montones de computadoras entre si y estos proveedores a su vez se conectan con otros proveedores.

Cuando buscamos en el navegador un dominio, le estamos haciendo un pedido al modem para que busque esa IP, ese pedido tambien pasa al proveedor y buscara esa computadora que yo estoy buscando y encontrar esa informacion. Estas ultimas computadoras son servidores, las cuales son computadoras mas potentes. Este servidor le responde al pedido dando archivos (html y css por ejemplo) a su modem, esa se la da al proveedor, y el proveedor se las pasa a nuestro modem, y nuestro modem a nuestro cpu. Y el navegador es el encargado de leer los archivos (el html y css, etc).

VIDEO Servidores.

Los servidores son CPUs conectados entre si en la misma torre/estructura. Se alojan en la Data Center.

El unico fin de estas torres tienen es guardar los archivos y mandarlas a nuestras computadoras. Se suelen usar Linux en estos. Hay diferentes tipos de servidores y de hosting.

Empresas hosting: donweb.com - hostinger.com - wiredhosting.com

Estas tienen las torres capaces de guardar la informacion que pedimos (html, css, etc.).

Comprar un dominio se puede en nic.ar. Y se pagan anualmente diferencia de los anteriores que son mensuales.

En las paginas anteriores, podemos buscar algun dominio para saber si esta disponible para usar o no.

Todos los dominios estan conformados por su nombre y extencion. Si es .com significa que es comercial. El .org es de organizaciones. Etc. Con este dominio (nombre y extencion) vamos a buscar la pagina.

2° Parte de la clase

HTML (Lenguaje de marcado hipertexto). Es el lenguaje por el cual nos comunicamos con los navegadores y que nos entienda que queremos mostrar.

Fue creado por Tim Beners-Lee que fue el padre ed la web (no del internet). Creo las reglas para que podamos hacer nuestras paginas web. El creó la www (triple w) World Wide Web.

PowerPoint:

Clase 2

HTML e introduccion en CSS.

Recordemos un poco lo de la clae anterior. Las paginas Web son como un libro. Y que pasaban los pedidos de datos por servidores y nos desvolvian archivos (como por ejemplo html y css). Esta informacion viaja entre continentes gracias a los servidores y red de redes (cables submarinos). Estas computadores tienen hardware y software(linux por ejemplo).

Vimos que el dns transforma el numero ip a dominio.

Tambien vimos que es un lenguaje y por quien fue creado el desarrollo web.

HTML no es un lenguaje de programacion, es un lenguaje de marcado. Es decir de estructura de paginas web. Al igual que CSS, no es de programacion, sino que es de estilos.

Formas de escribir nombres archivos html. Que sean minusculas y los espacios con guion (-) o guion bajo (\_). Tambien podemos usar el camelCase (Mayuscula por cada inicial de palabra menos la primer palabra).

Etiquetas meta:

La etiqueta <meta con la propiedad y valor name=”description” content=”Descripcion de la pagina...”> es la descripcion que va a salir abajo del titulo cuando busquemos esa pagina en el navegador (Se recomienda que no sea muy largo aprox solo dos renglones). Cada archivo HTML va a tener su descripcion ya que cada uno va a tener un fin diferente.

La etiqueta <meta name=”keywords” content=”cursos, curso, digital, bluee”> Para agregar palabras clave (separadas por coma), en la cual relacionara con las palabras que la persona busque en el navegador y asi posicionarnos. En este ejemplo si buscamos en el navegador alguna de estas palabras, hay mas chances de que aparezca primero en el navegador. Podemos poner las palabras clave que queramos y conviene que sean las palabras clave mas relacionadas a lo que se refiere este html en puntual.

La etiqueta <meta name=”author” content=”Agustin Perea”> Para decir quien es el autor de la pagina.

La etiqueta <meta name=”robots” content=”index”> Para que el navegador pueda indexar los dominios a traves de los robots. Los robots buscan con las palabras clave, todas las paginas relacionadas. Y en otras palabras, si yo quiero que este archivo sea posible buscarlo por los navegadores, tenemos que usar esta etiqueta. El index significa que va a poner mi pagina en el indice de mi navegador.

La etiqueta <meta name=”robots” content=”noindex”> Para que no sea posible buscarlo por los navegadores.

Algunas etiquetas del body:

<section> Es para dividir los contenidos de la pagina

<article> Son pequeñas porciones de la pagina que se repiten en codigo aunque cambien el contenido.

<div> Es un contenedor generico y no tienen valor semantico. Significa division, aunque se le puede dar muchos usos.

Clase 3:

La etiqueta <span> sirve para darle texto, pero que sea de largo como mucho de una sola linea. Sino usar <p>.

<small> Es para escribir un parrafo con tamaño reducido.

<ol> voy a usar en un menu cuando se que va a ser fijo (que no voy a cambiar nunca). <ul> si sabemos que va a cambiar va a ser <ul>.

Tambien en la clase hablamos sobre seguridad y el protocolo http.

Mas info aca: [(1) ¿Qué es y para qué sirve el protocolo HTTP? - YouTube](https://www.youtube.com/watch?v=VROImVct7YA)

Clase 4

Url metodo para viajar a archivos de otras computadoras.

Podria buscar imagenes o videos de yandex para poner en mi pagina web. Pero si sobrecargo la pagina con url eternas, puede que vaya mas lento. Ademas de que depende de que no se caiga el link. Mejor siempre trabajar con archivos locales de tu computadora.

Si quiero referenciar un archivo local con el href, si esta ubicado en la misma carpeta donde esta ubicado el html que le hace referencia, solo se pone el nombre del archivo para buscarlo. Sino, hay que poner el nombre de la carpeta, luego la barra, y despues el nombre del archivo.

Si se pone ../ para retroceder una carpeta para atras del archivo html. ./ es para ir a la carpeta donde esta el html.

Las anclas es poner en los href, los id de la misma pagina.

One Page: Como su nombre dice, son una pagina de una sola pagina.

Clase 5

0.26mm es igual a 1px. (basandose en la resolucion de pantalla).

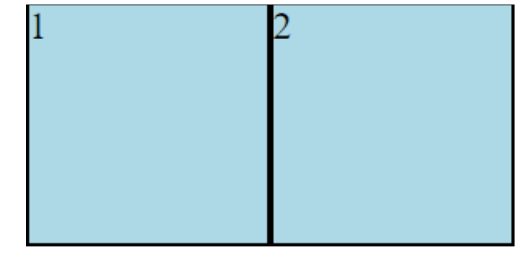
Por defecto. Para representar el tamaño de fuente standar de la etiqueta ‘p’ es de 16px.

1em equivale a 16px en PC de escritorio. 1em equivale a 8px en Tablet. 1em equivale a 4px en Celular.

:root es un selector de CSS que selecciona todo nuestro proyecto y le aplica los estilos. Sirve para dar cambios generales. Si ponemos un font-size:20px, lo que hace es cambiar el valor standar de la fuente de este proyecto (de 16px por defecto a 20px). Ahora globalmente 1rem son 20px. Ya que el rem viene el root.

Clase 6

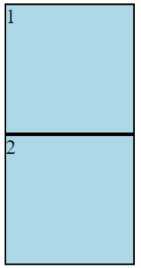
Filmina:

Al utilizar la propiedad float, se permite comenzar el proceso de encolumnado de elementos en mi html. Si usamos float: left; podemos ubicar los elementos desde la izquierda a derecha. 

Si usamos float:right va a ir de la derecha hacia la izquierda.



Si usamos float:none; es el valor predeterminado y no permite que ningun elemento se encolumne.



La propiedad clear:right; permite que los elementos indiquen que lado no permitirá elementos flotantes. En este caso indicamos que los elementos no se puedan ubicar a la derecha.

Tambien puede ser left (que no haya valores flotantes a la izquierda).

Tambien puede ser both que no permite que los elementos flotantes se puedan ubicar en ambos lados.

Flex

Trabajar con float es simple pero tiene como limitacion que es dificil controlarlo al qurer hacer una interfaz responsiva.

Por eso se usa flexbox, que bribda una flexibilidad que float no tiene.

Flex es el valor de la propiedad display. Y display permite decir como se visualizaran los elementos, siendo flex uno de sus tantos valores.

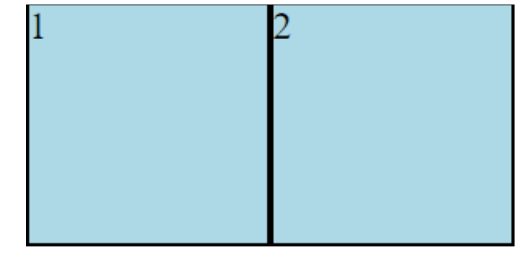
Display:none; no se visualizará el elemento, lo esconderá.

Display: block e inline ya las conozco.

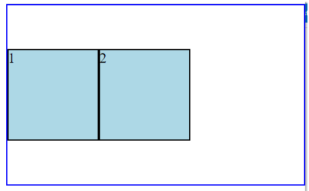
Display o visibility? La diferencia es que visibily:hidden ademas de hacer desaparecer el elemento, el espacio queda vacio en la interfaz . display:none; es como que borra el elemento y permite que otros elementos ocupen ese lugar.

Display:flex; Se aplica en el contenedor de los elementos que estan encolumnados.

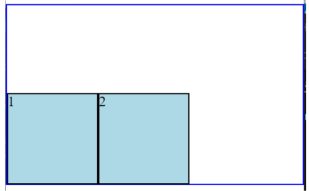
Si le damos esa propiedad al contenedor que usamos en los anteriores ejemplos, se vería así.



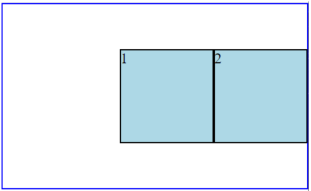
La propiedad align-items permite que se ubiquen verticalmente los elementos afectados por el flex. Si agregamos align-items:center; el resultado será:



Tambien puede ser align-items:flex-start o end; flex-start lo posiciona al inicio, el flex-end al final. Como por ejemplo:



La propiedad justify-content permite trabajar con alineacion horizontal de los elementos. Tiene casi los mismos valores de align-items, por ejemplo justify-content:flex-end; (Para mandarlo a la derecha) y align-items:center;(Para centrarlo verticalmente).



Los valores space-around (espacios alrededor de cada elemento) y space-between (espacios entre los mismos elementos).

Display:grid

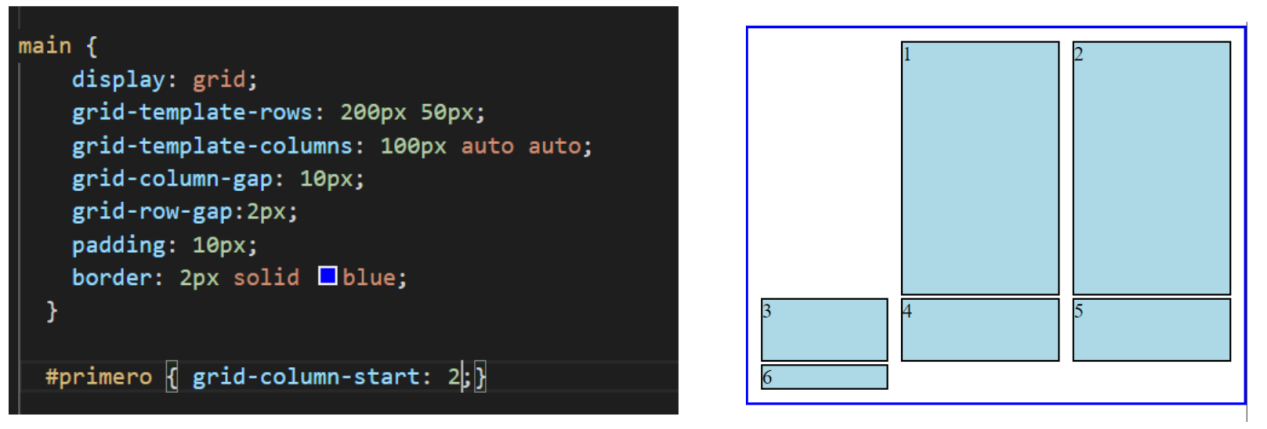
Nos permite encolumnar elementos en forma de grilla. Y para poder trabajarlo, ademas debemos agregar grid-template-column:100px auto auto. Que permite indicar el tamaño de cada columna.

Grid-template-rows:200px 50px 10px; indica el alto de cada fila.

Grid-column-gap:10px; permite indicar el espacio entre columnas.

Grid-row-gap:2px; permite indicar el espacio entre filas.

Ejemplo:



Como flex, grid tambien trabaja horizontalmente con justify-content y align-content verticalmente con los mismos valores de justify-content. Se puede utilizar estas como alternativa al gap.

Clase vivo Notas(3/11/22)

Con la propiedad css text-indent:20px. Lo que hago es dar la separacion a la primera palabra de un texto.

El font-family: serif, este tiene angulos en las esquinas. El sans-serif significa sin serifa, es decir, sin los angulos.La cursive, quita la serifa y redondea las letras. El inherit significa ‘en herencia’, es decir, que conserva la tipografia del contenedor padre de la caja que seleccionamos. El initial, es usar la fuente por defecto. El monospace saca la serifa y trata de que cada caracter tenga el mismo tamaño de ancho. Arial es la fuente por defecto del navegador.

En font-family, se utiliza la ‘,’ para separar los distintos tipos de fuente. Siempre le pondrá la primera, si en otro navegador no la tiene, se pone la siguiente fuente. Es como tener una fuente de respaldo en caso de que no funcione la primera.

Podemos combinar por ejemplo ‘arial serif monospace’ sin poner la coma y le aplica una combinacion de cada propiedad. Recordar ponerlas entre comillas simples.

Cuidado al subir una pagina a un hosting. Si en el SO de la PC que hace el host no esta la fuente que utilizamos en nuestros textos de la pagina subida, utilizará otra que si tenga esa computadora.

Para traer fuentes externas a nuestro archivo CSS:

Reglas de CSS: Empiezan con ‘@’ y van fuera de los selectores.

Con este podemos definir/crear una fuente:

@font-face{

font-family:miFuente; //nombre de mi fuente. Le puedo poner cualquiera.

src: url(‘carpFont/fuente.ttf’); //src incrusta contenido directamente. La url busca un archivo utilizando una ruta

}

Las fuentes que descargue, las tengo que poner en la misma carpeta donde esta nuestro proyecto.

Se recomienda mejor siempre descargar la fuente en lugar de usarlas online.

Para traer fuentes externas a nuestro archivo CSS con Google Fonts (forma Offline):

Elegimos cualquier fuente. Le damos el ‘+’. En el menu la podemos descargar abajo. Y luego utilizamos los mismos pasos que hicimos arriba con la Regla.

Para traer fuentes externas a nuestro archivo CSS con Google Fonts (forma Online):

Vamos a tener que usar la regla @import url(‘...’).

Esta la podemos copiar y pegar directamente desde google fonts. Y ademas tenemos que pegar la regla de CSS donde corresponde.

En lugar de usar @import, podemos usar el <link> y copiar y pegar al archivo html.

Clase 7

Filmina:

Los formularios permiten el intercambio de informacion con el usurario. Con este intercambio de informacion podemos comunicarnos con el servidor. Por lo tanto, los formularios se complemntan con lenguajes de programacion de servidor como php o node.js para conseguir esa interaccion.

Todo formulario se crea con la etiqueta htm form y adentro le pondremos inputs. El campo de texto contiene atributos necesarios para trabajar. Entre los mas importantes se encuentra name. Este permite que el campo participe de la interaccion con el codigo php. Entonces, por cada valor de name se encontrará una variable en php por ejemplo.

PHP es programacion en el servidor. Le habla al servidor y de esa forma permite conectarnos. Gracias al atributo name, enviamos esa informacion a nuestro codigo y este nos brinda la respuesta desde el servidor.

Existen otros lenguajes como node.js que cumplen la misma funcion. Va segun gustos cual usar.

El trabajar con ambos es ismple y permite interactuar para obtener estas respuestas e intercambio de informacion que hasta ahora solo teniamos con el frontend y no el backend.

El mérito de la conexion entre php y el formulario, se lo debemos a name, pero el atributo action determina con que archivo se vinculó el formulario de forma especifica. Tambien podemos usar el atributo method, que es necesario para saber que metodo utilizar de intercambio de informacion utilizar. Podemos ver un ejemplo:

<form action=”formulario.php” method=”post”>

<label>Nombre</label>

<input type=”text” name=”nombre” placeholder=”Nombre”>

<label>Email</label>

<input type=”email” name=”email” placeholder=”info@dominio.com”>

<label>Edad</label>

<input type=”number” name=”edad” min=”12” max=”45”>

<label>Sugerencia</label>

<textarea name=”nombre” rows=”5” cols=”20”></textarea>

</form>

-La etiqueta label le indica al usuario cual es el contenido del campo a completar, siendo el texto del label lo que el usuario puede entender si debe colocar su nombre, apellido, etc.

-El placeholder es un atributo que permite mas alla del label, indicar dentro del propío campo como debe completar la informacion.

-El campo de correo validará el contenido si tiene el @. Y podriamos programarlo para que si el usuario no lo pone, que salga un mensaje de error.

-El campo numero solo me permitirá poner numeros y podriamos agregar rangos a traves de atributos como se muestran el el ejemplo.

-El usuario no puede usar un campo de texto para hacer comentarios, ya que este es muy pequeño Por esta razon usamos con el campo de comentarios con rows y cols podemos indicar el tamaño del campo. Aunque es mas recomendable hacerlo con width y height en css.

- Pueden haber otros campos como password, color, archivo, las diferentes variantes para fechas, botones.

-Utilizaremos el method post en situaciones donde el usuario cambia informacion del servidor, por ejemplo añadir un comentario o una nueva direccion de correo.

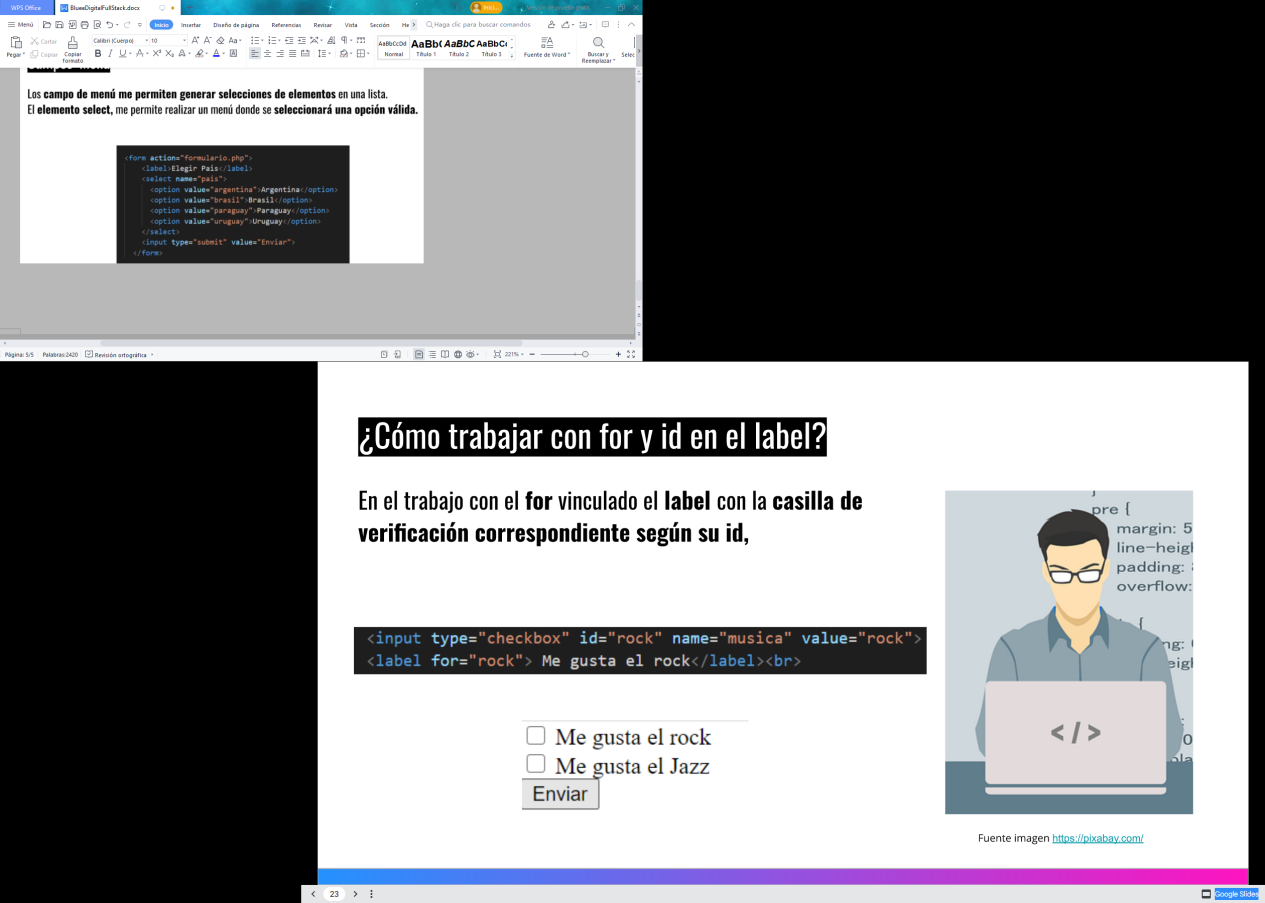
El metodo get debe utilizarse cuando el usuario no cambia nada, sino simplemente obtenga informacion dada por un servidor, como el buscador Google.

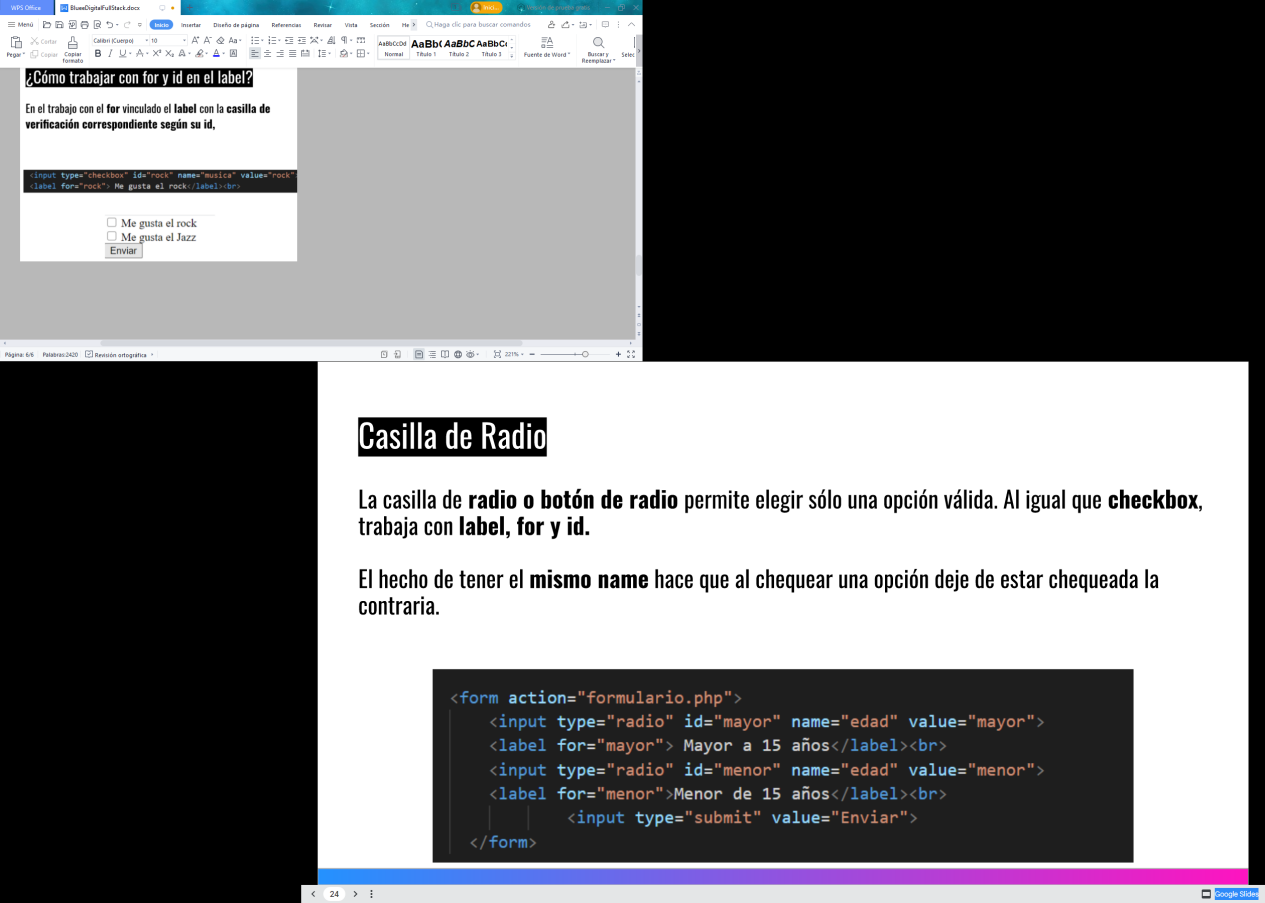
Podemos notar que se utiliza el metodo get cuando la informacion se procesa dentro de la misma url quedando esos datos visibles y sin seguridad. Por ej: .google.com.ar/webhp?source=hp&ei=YGhyY7q...

Para crear un campo menu en donde nos pernite generar selecciones de elementos en una lista. El elemento select me permite realizar un menu donde se seleccionará una opcion valida.





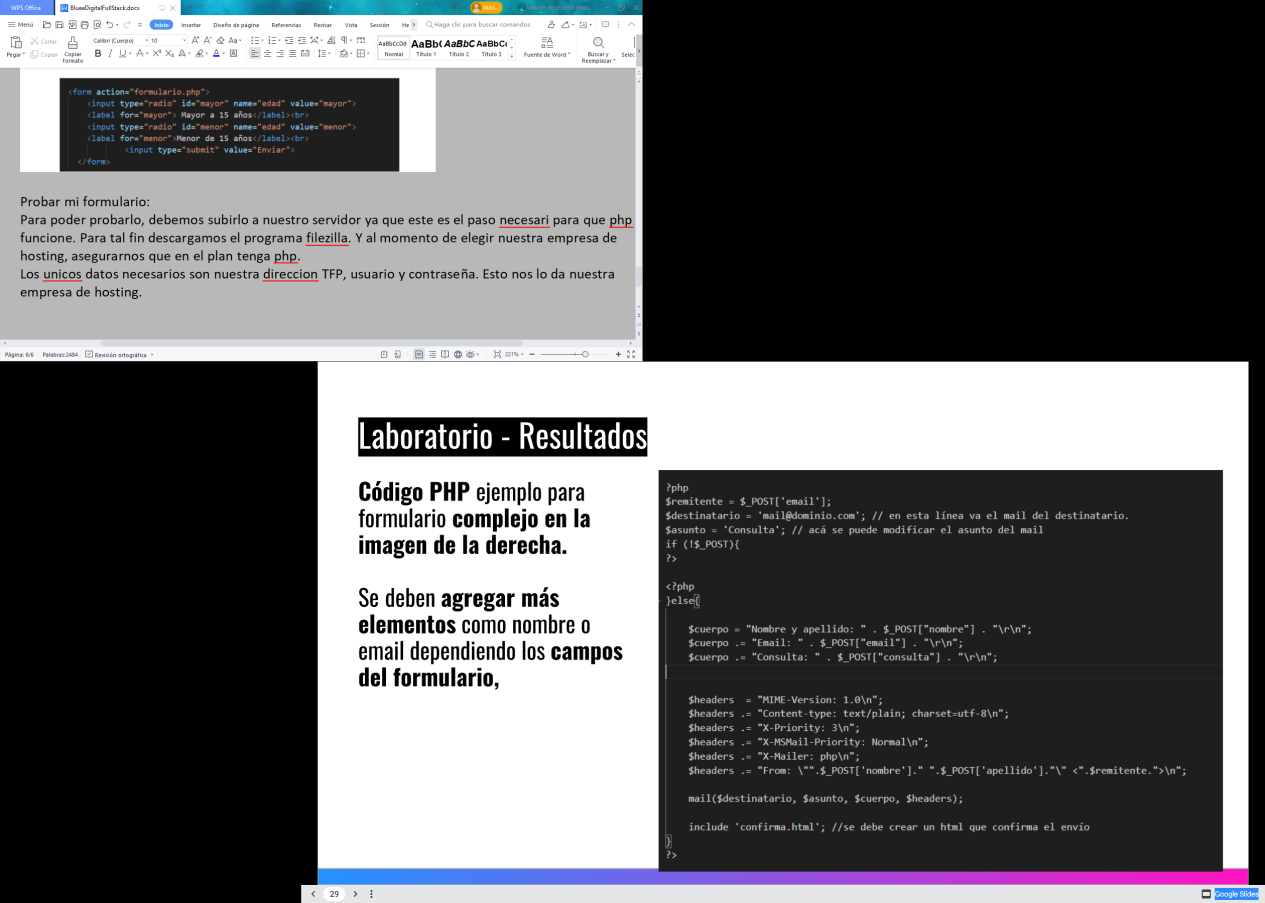




Probar mi formulario:

Para poder probarlo, debemos subirlo a nuestro servidor ya que este es el paso necesari para que php funcione. Para tal fin descargamos el programa filezilla. Y al momento de elegir nuestra empresa de hosting, asegurarnos que en el plan tenga php.

Los unicos datos necesarios son nuestra direccion TFP, usuario y contraseña. Esto nos lo da nuestra empresa de hosting.



Notas Grabacion clase (10/11/2022)

(

Notita de calse anterior:

font-family: serif (Serifa)

Sans-serif (No aplicar serifa)

Cursive (Saca serifa, redondeo y negrita)

Inherit (La fuente es heredada de la etiqueta padre)

Initial ( usando fuente inicial)

Monospace (Sacar la serifa y le da el mismo ancho a todas las letras)

)

Propiedad outline. Es el borde de seleccion que se muestra por fuera del borde y no ocupa lugar. Por lo tanto hay que tener cuidado por si tapa a algun otro elemento. Y sus valores son los mismos que con border.

CLASE 8

Multimedia

Filmina:

Cuando ponemos una etiqueta html para video, tenemos que poner ‘controls’ para que se puede interactuar con el.

Ademas podriamos poner ‘muted’ para que al cargar la pagina, el video este muteado.

El atributo poster=”imagen.jpg”, permite poner una imagen que aparecerá antes de que el usuario decida reproducir el video.

El atributo loop indica que el video se reproducirá nuevamente al finalizar.

Podemos utilizar width y height para dar tamaño al video.

Preload=”” permite definir si se cargará el video o no cuando se inicie la pagina. Si está activo el atributo ‘autoplay’ aparte, se ingonará el preload.

Entonces preload=”auto”, cargará el video ENTERO cuando la pagina carga. preload=”metadata”, cargarpa solo la metadata cuando la pagina carga. preload=”none” NO carga el video cuando la pagina carga.

Ademas, dentro del bloque video, podriamos agregar etiquetas source y poner el mismo video en varios formatos para que el navegador seleccione la opcion mas indicada para el usuario. Por ejemplo:



Formatos de video

El ‘.ogg’ utilizan el formato de audio comprimido Ogg Vorbis, ofrecen mejor calidad que el mp4. Se puede generar desde premiere o utilizar un convertidor online como [https://www.google.com/url?q=https://audio.online-convert.com/es/convertir-a-ogg&sa=D&source=editors&ust=1669122996533544&usg=AOvVaw06GrkCmO1s\_SZYZ39nF57y](https://www.google.com/url?q=https:/audio.online-convert.com/es/convertir-a-ogg&sa=D&source=editors&ust=1669122996533544&usg=AOvVaw06GrkCmO1s_SZYZ39nF57y)

Por otro lado, WebM es de codigo abierto aunque no da tan buena calidad como mp4.

Formato de audio

Es similar al caso de video. El ‘.mp3’ es de menor calidad que el ‘.ogg’, pero es mas compatible qcon algunos navegadores, por eso es necesario trabajar con multiples origenes para mejor accesibilidad.

Convertidor online de formato de audio:

[https://www.google.com/url?q=https://www.zamzar.com/es/convert/ogg-to-mp3/&sa=D&source=editors&ust=1669122996543625&usg=AOvVaw14lx4f98JtCf4D\_4Gu3m3t](https://www.google.com/url?q=https:/www.zamzar.com/es/convert/ogg-to-mp3/&sa=D&source=editors&ust=1669122996543625&usg=AOvVaw14lx4f98JtCf4D_4Gu3m3t)

Para escuchar musica desde una pagina, Debemos trabajar con la etiqueta <audio> como ya hicimos con video, tambien puede tener la propiedad controls y darles varios source dentro para mejorar accesibilidad.

Iframe para video

Ofrece la posibilidad para mostrar contenido de paginas externas en nuestra pagina.

Como por ejemplo con youtube. Luego de elegir el video, debemos hacer click en la opcion compartir. Y hacemos click en la opcion ‘insertar’. Ahí nos aparecerá el iframe que podemos copiar y pegar.

Ademas de youtube, podriamos agregar iframes de comentarios de Facebook.

(COMPLETAR TABLAS ..............................................................................................)

VIDEO (15/11/2022)

Las extensiones de un archivo son pequeñas abreviaciones que hace referencia al tipo de archivo, ya que nuestras PCs contienen muchos tipos de archivos diferentes.

Extencion .jpeg es para imagens/fotos comprimidas y perderan un poco de calidad para su rendimiento y que pesen menos. Los .png no son comprimidas y tienen buena calidad (pueden no tener fondo).

El formato WebP combina lo mejor de jpg/jpeg y png. Que lo que hace es bajarle el peso a la imagen sin perder calidad.

El formato AVIF combina todo lo visto hasta ahora. No baja absolutamente nada de calidad y comprime la imagen. Pero no es soportado por todos los navegadores.

CLASE 9

FILMINA: INTRODUCCION A LA WEB

Position relative, trabaja con valores respecto a su posicion inicial.

Position absolute actúa como las capas de photoshop. Estas se pueden superponer. Y tambien es importante saber que fijan su movimiento en base a su elemento contenedor.

Para decir que capa se verá primero, utilizaremos la propiedad z-index que comienza por defecto en el mismo nivel de su contenedor.

Fixed permite generar elementos fijos en el interfaz. Es muy utilizados para publicidades o botones o menues. Y se ubican en relacion a la ventana.

Sticky fija elementos en el momento en que el scroll lo activa. Si no hay barra de scroll, el elemento queda en relative. Muy usado para barras de navegacion.

Opacity. Si es 1 se ve completamente. Cuanto menor sea, mas transparente se ve. Si llega a 0 es completamente transparente.

Colores especiales: Para poder trabajarlos utilizamos hsla(), basado en la matiz, saturacion y luminosidad del color.

Selectores avanzados:

Con el simbolo ‘+’ en selector de css indicamos que todos los elementos luego del elemento que marcamos se les aplicará la propiedad. Por ejemplo en el siguiente, todos los elementos con clase .texto se veran afectados por el color rojo sólo si estan ubicados luego de un nav.



Los selectores ::after y ::before indican que se insertrá contenido ya sea de texto o de imagen antes o despues de algun elemento de nuestra estructura.

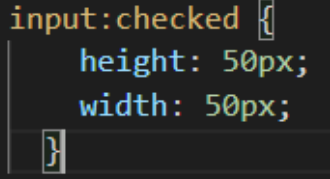
Por ejemplo que se incluya un asterisco luego de cada párrafo del documento.



O si queremos que se muestre una imagen antes de un enunciado.



Tambien hay selectores avanzados para formularios. El selector :checked indica propiedades para elementos con casillas de verificacion o botones de radio. Podemos usar :disabled o :enabled para cuando para los elementos que tengas el atributo disabled o no aplicado.



El selector :required ayuda a la validacion para que el usuario entienda que el elemento es obligatorio.

Tambien si un elemento no es valido, aparecen :valid o :invalid para dar a entender que hay elementos que estan mal al completar el formulario.

Podemos usar selectores :first-child o :last-child para determinar si seleccionamos el primer o el ultimo del contenedor padre.

El :nth-first-child(n), aplica la propiedad al elemento ubicado en esa posicion que determinamos en el n.

:first-line o :first-letter para dar caracteristicas a la primera linea o letra.

::selection afecta al elemento que ha sido seleccionado por el usuario con el cursor del mouse. No admite todas las propiedades pero si de color, de tipografia, fondo, outline, cursor donde con el valor pointer podemos generar el icono de mano o cualquier otro cuando el cursor está seleccionando un elemento.

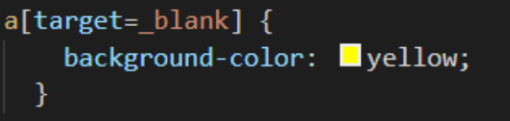
:hover mouse por arriba.

Input:focus Podemos hacer foco en componentes como los campos de texto. Y este selector sirve mas que nada para que el usuario comprenda en que elemento está focalizado.

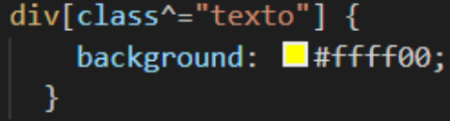
:visited indica propiedades para enlaces que ya han sido visitados por el usuario.

:link es para aquellos enlaces que no han sido visitados.

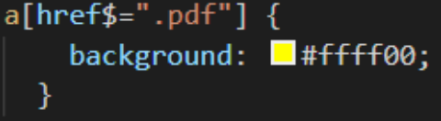
Los selectores de atributo aplican propiedades a elementos que contengan ciertos atributos o valores en el mismo. Por ejemplo, los elementos ‘a’ que contengan el atributo ‘target=\_blank’ tendrán color de fondo amarillo.



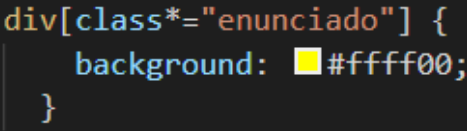
Podemos especificar atributos aclarando que cierto atributo empieza o termina con determinado valor. Por ejeplo, que todos los elementos cuya clase comience con la palabra texto, mas allá de contener otros elementos, les dara el color de fondo amarillo.



Si queremos indicar que por tener texto o contenido al final de un atributo, indicaremos com $. Por ejemplo dar cierto color de fondo a vinculos cuyo destino sean un elemento o archivo .pdf.



Si queremos indicar que el valor estará compuesto por cierta palabra sin importar si esta al inicio o al final, será con ‘\*’ asterisco. Por ejemplo, si hay un dic con una clase cuyo nombre en alguna parte tenga el texto enunciado, será suficiente para darle el color de fondo amarillo.



CLASE 9 GRABACION (17/11/2022)

Cuando queramos insertar un audio con la etiqueta <audio>, tendremos que utilizar varias propiedades, como por ejemplo ‘controls’ para mostrar la barra de sonido.

Con ‘autoplay’ se reproduce automaticamente cuando se entra a la web.

Con ‘muted’ el video esta muteado por defecto al cargar la pagina, manualmente tendremos que activarla.

El ‘volume=”20” ’ se asigna el valor de volumen por defecto del audio desde 0% a 100%. Se podria modificar con JS.

La etiqueta <iframe> sirve para traer e incrustar archivos o paginas externas o locales con url. Si en el src le damos una pagina web, nos lo mostrara en la pagina. Muchas webs no pueden ser traidas con iframe por su lentitud y y tamaño.

Podemos usar ifrmae para traer videos de youtube. Esto lo haremos con la opcion que nos dá el mismo video cuando damos a compartir e insertar.

Esto tambien pasa con Google Maps.

CLASE 10 FILMINA

MAQUETACION AVANZADA TRANSITION

Transition: es una propiedad que permite realizar un pasaje entre un valor y otro de una misma propiedad.

Una transicion comienza desde un valor inicial. Este valor cambia en el tiempo.

Podemos activar una transicion a traves de pseudo calses (hover, focus, etc.).

-(1)Para comenzar la transicion, debemos integrar la propiedad transition-property. Permite llamar a la transicion desde el punto inicial. Con escribir transition es suficiente, ya que es su shorthand.

-(2)Hay que determinar el tiempo, la transicion se ubica en 0 segundos si el tiempo no se especifica. Por lo tanto, omitir esta propiedad implica que no haya transicion. Para hacerlo debemos utilizar transition-duration. Los valores pueden ser ms o s.

-(3)Indicamos en la pseudo clase el cambio final. En este caso, que cambie a color azul.

-(4) Para hacer mas interesante la transicion, podemos usar el transition-delay. Que permite demorar la transicion en s o ms.

De todas formas lo unico obligatorio es indicar el duration, ya que sino no se realiza la transicion.

-(5) Podemos dar distintos tipos de ritmo de transicion con transition-timing-function. Puede ser linear o ease-in o ease-out. Linear indica que la transition se hará desde el comienzo hasta el final con la misma velocidad. El ease permite que la transition comience lento, luego vaya rapido y termine lento o viceversa.

Tambien está el cubic-bezier, que imprime los tiempos de transicion en cuatro puntos. El primer y ultimo punto significan el inicio y el final, van desde 0 a 1.

Ejemplo:

div{

width: 100px; //Propiedades div

height:100px;//Propiedades div

background-color:pink;//Propiedades div

transition-property: background-color;//(1)

Transition-duration: 10s;//(2)

Transition-timing-function: cubic-bezier(0.5, 1, 0.2, 1);//(5)

transition-delay:0.5s;//(4)

}

div:hover{ background-color:blue; //(3)

}

Trabajar en una linea, esto puede hacerse con transition. Va en orden como vimos los elementos anteriores.

Transition: property duration timing-function delay;

Ejemplo:

Transition: background-color 10s linear 0.5s;

Trabajar con el foco. Podemos modificar elementos como los campos de texto para indicar que el usuario está haciendo foco en ellos. Es casi lo mismo que hicimos con :hover.

Input{width: 100px;

Transition: width 2s linear 1s;}

Input:focus{ width: 300px;}

Con transition podemos trabajar multiples propiedades al mismo tiempo. Se hace indicando las propiedades modificadas en la transicion de forma especifica, tanto su tiempo como otras propiedades de transicion.

Ejemplo:

div{

width: 100px;

height:100px;

background-color:indianred;

transition-property: background-color, width;//Aqui he puesto las 2 propiedades a modificar

Transition-duration:2s, 4s;//La duracion de propiedad

}

div:hover{width:200px; //Propiedades finales

background-color:yellowgreen; //Propiedades finales

}

En el caso anterior, queriamos modificar alguna propieades y no todas. Pero si queremos modificar todas, contamos con la opcion all y hacerlo de manera mas rapido.

Utilizando el ejemplo anterior sería:

div{

width: 100px;

height:100px;

background-color:indianred;

transition: all 2s 4s linear ;//Aqui he puesto todas las propiedades a modificar y duracion

}

div:hover{width:200px; //Propiedades finales

background-color:yellowgreen; //Propiedades finales

}

Gradientes: Pueden ser lineales o radiales. Se les puede agregar transparencia y colores en diferentes formatos. Lo importante es entender que son como imagenes de fondo, no son colores aplicables ni a bordes ni a tipografia.

Pueden contener dos o mas colores, marcamos su inicio y final segun lo indiquemos y pueden contener transparencias. El primer elemento indica la direccion (to right, to left, to top, to bottom).

Div{background-image: linear-gradient(direction, color1, color2);}

Si queremos determinar un origen hacia la derecha con color rojo que termine en amarillo.

Div{background-image: linear-gradient(to right, red, yellow);}

Implementar transparencia. Puede ser así. Con esto, el primer color es mas translucido que el segundo. 0 es invisible y 1 es el color solido sin transparencia.

Div{background-image: linear-gradient(to right, rgba(100,200,200,0.5), yellow);}

Las gradientes pueden ser diagonales

Div{background-image: linear-gradient(to bottom right, red, yellow);}

Tambien pueden tener grados de inclinacion.

Div{background-image: linear-gradient(45deg, red, yellow);}

Gradientes radiales. Son similares a las lineales. La unica diferencia es que estos parten desde el centro hacia afuera a traves de un circulo o elipse. Por ejemplo

Div{background-image: radial-gradient(red, yellow);}

Podemos indicar si usamos circulo o elipse como origen. Si no indicamos ninguno, por defecto es elipse. Es importante ajustar el objeto para notar la diferencia entre un valor y otro.

Div{width:200px;

Height:100px;

background-image: radial-gradient(circle, red, yellow);}

Ademas, podemos elegir la posicion donde se encuentra ese origen. Se hace con farthest-side. Puede ser con left,top,right,bottom o con determinar cada eje.

//Con lados

Div{width:200px;

Height:100px;

background-image: radial-gradient(farthest-side circle at 60px 70px, red, yellow);}

//O con ejes

Div{width:200px;

Height:100px;

background-image: radial-gradient(farthest-side circle at left, red, yellow);}

CLASE 10 GRABACION (24/11/2022)

Frameworks, nos van a ayudar a hacer el trabajo mas facil con html y css.

La etiqueta <label> es una etiqueta de texto que solo funciona dentro del <form>. Se asocia con algun campo input. Permite ingresar texto. Al darle click a ese texto, se selecciona el input. Pero al leber hay que darle un for=”nombreCualquiera”, y al input le damos un id=”nombreCualquiera” para que se asocien el texto y el input.

Checkbox es para multiples opciones. Radio es solo para seleccionar una opcion.

Los radio de ese formulario, deberian tener el mismo name=”...” para que solo se seleccione una opcion de los radio.

El input de type=”file” permite abrir la carpeta del disco local y poder seleccionar un archivo. Se le puede agregar la propiedad multiple para que te deje seleccionar varios y no solo uno.

La etiqueta <select> es una etiqueta contenedora de otras etiquetas. Dentro de esta, pueden ir etiquetas <option>, en la que dentro se le puede poner un texto en el cual la persona va a poder elegir.

Se recomienda que siempre la primera option sea algo como <option value=”0”>Seleccionar una opcion<option> Esto para que sea lo primero que aparezca en el menu.

Y como vimos en el value, le podemos asignar numeros para luego utilizar en el backend.

Dentro de la etiqueta de apertura de form, puedo agregar un action para que se ejecute cuando se le de click al submit para enviar el formulario. Ejemplo: <form action=”enviar.php”>.

Si a algun input le pongo la propiedad disabled, no permite que el usuario modifique el campo. No podra escribir ni seleccionar ni borrar nada en ese input. El readonly solo se puede leer y lo puedo leer desde JS.

Ahora vamos a usar CSS.

CLASE 1.11

FILMINA MAQUETACION AVANZADA

ANIMATION

Una animacion es como una transicion pero con muchos mas elementos en juego, como por ejemplo la posibilidad de cambiar el valor de una propiedad.

Esto permite que los elementos puedan tener distintos valores durante un tiempo.

Para trabajar una animacion, primero debe crearse. Las animaciones a diferencia de las transiciones, se activan al cargar la pagina. No necesitamos que el usuario haga una accion para que se active. Con entrar a la pagina es suficiente.

Para comenzar la animacion, tenemos que trabajar con la regla @keyframe nombreAccion {}. Puede tener cualquier nombre minetras no tenga espacios y que no comience con numeros. Y dentro de las keyframes, tenemos que decir un comienzo y final, que puede ser con from y to, o con 0% a 100%. Dentro de cada una, podemos nombrar los elementos que queremos modificar a lo largo de la animacion.

Para llamar la animacion, se utiliza una propiedad llamada animation-name: nombreAccion;. Esto se pone en el elemento que hará la animacion.

Tambien tenemos que indicar el tiempo de animacion con animation-duration:5s; si no ponemos esta propiedad, la animacion nunca iniciará.

La animacion tambien puede tener el animation-delay, animation-timing-function como en las transition.

Para usar cubic-bezier de manera simple, podemos utilizar la siguiente herramienta online: https://www.google.com/url?q=https://cubic-bezier.com/%23.17,.67,.83,.67&sa=D&source=editors&ust=1670353840898832&usg=AOvVaw1GKDeYDdMXRnD\_iBU-ll50

Animation-direction, puden ser ‘normal’(hacia adelante), ‘reverse’(comenzará por el final), ‘alternate’ (hace un ciclo: 0%->100%->0%->), ‘alternate-reverse’(hace un ciclo: 100%->0%->100%->).

Animation-iteration-count sirve para indicar cuantas veces se producirá la animacion. Por defecto es 1.

El valor puede ser ‘infinite’ para que sea en bucle.

Animation-play-state para indicar que una animacion esta pausada con el valor ‘paused’. Podriamos usar javascript para que el usuari decida cuando y donde la animacion iniciará y no cuando carga la pagina.

Animation-fill-mode indica como se completaran aquellos espacios sin animacion. Por ejemplo, sería al principio (antes de que la animacion empiece) y al final (luego de que termina la animacion).

Con ‘backwards’ se mantiene los estilos del principio de la misma. Con ‘forwards’ mantiene los estilos finales de la animacion. Con ‘both’ extiende las caracteristicas de ambos lados de la animacion. Acá son cruciales los detalles adicionales de cada parte para tener resultados.

Agregarle tranform a la animacion, complementa el trabajo con transiciones y animaciones. Las transformaciones modifican los elementos para poder cambiarlo en su posicion, eje y tamaño.

Podriamos rotarlo con transform:rotate() segun la cantidad de grados. Positivo hacia las agujas del reloj, negativo al contrario de las agujas.

Tambien podemos trabajar con los ejes del elemento, y hacer una rotacion en 3 dimensiones. Aqui por ejemplo lo hacemos con eje X, Y y Z.

Transform: rotateX(130deg) rotateY(40deg) rotateZ(360deg);

NOTA: el rotate() que vimos clases atras, puede ir entre 0° y 360° o negativos de eso. Mientras que los de X,Y y Z pueden ir de 0° a 180° o negativos de eso.

Con transform: translate(100px) para mover el elemento en eje X, Y y Z. El eje Z marca la profundidad del elemento.

Con transform:scale(1.5) podemos duplicar o reducir la escala del elemento. Si ponemos 2, duplicaremos su tamaño original. O con 0.5 seria la mitad de la medida inicial. El 1.5 seria como el 150% total.

Con transform:skew(15deg, 15deg) gira un elemento en el plano 2d.

Transform:matrix() trabaja con operaciones matematicas. Para hacerlo podemos usar la siguiente pagina:

[https://www.google.com/url?q=http://angrytools.com/css-generator/transform/&sa=D&source=editors&ust=1670353840928189&usg=AOvVaw2ZwoG7vvbLA5umppW3S8is](https://www.google.com/url?q=http:/angrytools.com/css-generator/transform/&sa=D&source=editors&ust=1670353840928189&usg=AOvVaw2ZwoG7vvbLA5umppW3S8is)

GRABACION (29/112022)

Tamaños recomendados para adaptar cualquier proyecto web a diferentes dispositivos.

PC de escritorio:

Max-width: 1920px

Max-width: 1400px

Max-width: 1200px

Tablets:

Max-width: 922px

Max-width: 768px

Celulares:

Max-width: 576px

Para la orientacio de los celulares vertical o volteado. Se utiliza

@media (orientation: portrait){} //Es el que es por defecto (en vertical)

@media (orientation: landscape){} //(volteado en horizontal)

Recordar que se pueden utilizar operadores and u or para determinar que se apliquen las propiedades css a esas condiciones. Por ejemplo

@media (orientation: portrait) and (max-width: 700px){...}//Para que se palique el css cuando el celular esté en vertical y que la pantalla tenga como maximo 700px

((((((ANOTAR SOBRE @media print para cuando se quiere imprimir la pagina))))))))))

Para que se usan los div, section, article?????????????

Nota: article esta compuesto por mas de una etiqueta visual. Por ejemplo adentro de un article hay una imagen y un titulo.

En section pueden haber uno o varios articulos. Se puede usar directamente section si adentro va a haber una etiqueta visual.

Div divide grupo de etiquetas.

Puedo usar float para mover elementos hacia el respectivo lado que señalice.

CLASE 1.12

FILMINA

GRABACION (1/12/2022)

Cuando ponemos @ alteramos reglas de css que vienen por defecto, por ejemplo la fuente de texto, el viewport, la media, etc.

((((((((Buscar breakpoint css))))))

Si utilizo el float para ordenar varios elementos, cada uno los hace flotar sin importar el orden de la estructura en el archivo html.

(Raro de usar)La propiedad ‘clear:left’ en css por ejemplo, no permite posicionar otros elementos a la izquierda del selector seleccionado.

Con el position relative en el css ((ocupa lugar)Recordar que se basa en la posicion original del elemento en la estructura html). El left se usa junto con el top, y el right con el bottom. Si no uso los pares que corresponden, las propiedades no se aplican. Si se ponen los cuatro, primero siempre va a aplicar el valor de top y left. Recordar, usar top y left, o right y bottom.

En otras palabras, el relative empuja los elementos.

Absolute (No ocupa lugar). Puede solapar los elementos uno encima de otro.

Aquí no se usa top,left,right,bottom. Aqui se utiliza margin-top,margin-right,etc.

Podemos darles el orden de la capa con z-index.

CLASE 1.13

FILMINA

CLASE GRABACION(6/12/2022)

El position:fixed lo voy a usar para mostrar un popup fijo. O un boton de whatsapp. Lo que hace esto es fijar un elemento en la pagina. Tiene que estar acompañado de sus coordenadas, ya sea top,left,right,bottom.

El position:sticky es lo mismo que el fixed pero siguiendo el scroll. Esto se va ausar mas que nada para los menu. Y tambien hay que marcarle la posicion de sus coordenadas.

El left y right va a servir para cuando el scroll es horizontal. Y el top y bottom para cuando el scroll sea vertical.

Visibility: visible es el por defecto para visualizar el elemento. El visibility: hidden es para hacer invisible el elemento y al mismo tiempo ocupa su lugar.

El display sirve para indicar si un elemento se muestra o no. Con display:none no se muestra el elemento y tampoco ocupa lugar.

Display: flex se indica en un contenedor para decir que los elementos dentro de esta se acomoden uno al lado del otro. En las herramientas del desarrolador puedo probar los distintos tipos de alineacion y posicionamiento flex.

El flex-wrap:wrap sirve para que si un texto se sobresale de una caja, lo acomoda dentro. El no-wrap hace sobresalir el texto de la caja.

Si estamos trabajando en una flex-direction:column y le indicamos una altura. El wrap hará que se crea una columna al costado con el contenido que sobresalía. Siempre manteniendo la medida.

Justify-content:center centra los elementos al medio. Justify-content, alinea los elementos horizontalmente.

Align-items sirve para alinear los elementos verticalmente.

RECORDAR: El align-content solo funciona cuando la columna y el wrap estan activados.

CLASE 1.14 FRAMEWORK AVANZADO

FILMINA

GRABADO (13/12/22)

Recordar, position relative, solo puede ser top y left o right y bottom. Y mantiene su lugar en el flujo.

El position absolute lo vamos a querer usar cuando queremos que el elemento se pueda mover de manera independiente, libremente por donde quiera. Y se maneja de acuerdo al viewport con top y left etc como el anterior (NO USAR). Y para moverlo dentro de su contenedor lo movemos con margin y padding. Y solo aquí se usa z-index para definir cual elemento va arriba de otro. Recordemos que el absolute está en movimiento libre.

El position fixed. Lo deja siempre fijo en la pantalla. Se puede usar todas las top, right, bottom, left.

El position sticky. Es muy parecido al fixed, pero la diferencia es que sigue al scroll.

Visibility: hidden. Hace invisible al elemento. Sigue ocupando su espacio aunque no se vea.

Display: none. Oculta al elemento. No se ve y no ocupa espacio.

Display:inline. Convierte elementos de bloque a linea. Y viceversa con block

Acordarse de usar la herramienta de desarrollador de la pagina.

Display:flex. Este contenedor se tratará como flex (en fila y columna).

Flex-direction: row. Elementos hijos en fila. Column hijos en columna.

Flex-direction: row-reverse/column-reverse. Elementos en fila/columna invertidas.

Flex-wrap:wrap. Que si se supera el ancho/alto (segun si es row o colum) del contenedor, se desplazarán los elementos que superen el ancho/alto.

Alinear elementos flex:

(solo funciona con el wrap activado.) con align-content.

(alinear de manera horizontal) Justify-content : flex-star(apoya a la izquierda), flex-end(apoya a la derecha), center(Mueve todo horizontalmente al centro), space-between(Espaciado entre elementos), space-around(Espaciado alrededor y entre los elementos), space-evenly(Da un espaciado uniforme a cada elemento).

(Alinear elementos verticalmente) aling-items: center (alinea verticalmente al centro), felx-start(Mueve hacia arriba), stretch(estirar los elementos para que ocupen todo el alto).

Grid:

Display: grid. Vamos a trabajar en el contenedor como grid (en grilla). Se basa en columnas, NO en filas.

Posicionamiento con grid:

(Posicionamiento vertical de las grillas)Align-content

(Posicionamiento horizontal de elementos)Justify-content:center, start, end, space-between, space-around, space-evenly.

(Posicionamiento vertical)Align-items: center,start,end,stretch

(Posicionamiento horizontal de las grillas)Justify-items:center, start,end,stretch

(definir el ancho de cada columna)Grid-template-column: 350px 500px (columna1 columna2)

Por lo general no modificar el height. Dejarle el height en auto, que lo maneje el navegador.

Podriamos hacer varios grid-template-columns para hacer otras filas abajo (RECOMENDABLE)

(Definir el espaciado horizontal entre columnas, como un margin:right) Grid-column-gap: 10px. (No recomendado usar, mejor usar margin y padding)

(Definir cuantas filas y su alto (NO RECOMENDADO USAR))(Tratar de no usar row)

Grid-template-row: 400px 200px (fila1 fila2)

(Definir el espaciado vertical entre filas, como un margin:bottom) Grid-row-gap: 10px (No recomendado usar, mejor usar margin y padding)

Orden de las columnas y filas

(alineamos la columna de la izquierda y posicionamos)Grid-column-start:2 (En este caso los elementos comienzan alineados a la izquierda en la columna 2)

(Alineamos la columna de la derecha y posicionamos)Grid-column-end:3 (En este caso los elementos comienzan alineados a la derecha en la columna 3)(Funciona si los elementos dentro estan en display:inline)

CLASE 2.1 Conceptos iniciales de JS

GRABACION (15/12/2022)

CLASE PRACTICA

Que preguntar?:

Donde conseguir trabajo|||||||||

Tips para estudiar |||||||||||

Que practicar Envato market theme forest

Hacer dominio||||||||||||||||

Tipos de sitios web:

-Home Page

Intenta mostrar todo lo que tiene una institucion dentro de UNA sola pagina.

Es una pagina que está compuesta por varias secciones. Puede tener o no menú.

Gracias al menu con las anclas, se puede navegar con el scroll en esta misma pagina.

De todas formas, puede contener enlaces que manden a otro dominio. Como tener enlaces hacia redes sociales o algo de “trabaja con nosotros”. Por tener enlaces hacia subdominios o dominios diferentes, sigue siendo Home Page. Ya que estas direcciones son aparte a la que estamos viendo.

La pagina no debería ser muy larga, ya que lo puede volver lento

- Landing Page

Es UNA sola pagina con el fin de convencer o vender lo de esa pagina. Suscribirse para luego en el futuro mandar mensajes o emails a los clientes. No tienen menu. No tiene mas de una seccion. Por lo general tienen un formulario o boton de Whatsapp.

- Institucionales

A diferencia de las Home Page. Navegamos entre distintas paginas/distintas carpetas del mismo proyecto o base de datos.

Si nuestra pagina es muy extensa y tiene muchos contenidos, estamos obligados a que con algun menu/boton mandar a otro html del proyecto con esa informacion.

- Ecomerce

Tiene como objetivo centrarse en los productos de la empresa y no sobre una empresa/institucion (Los institucionales es al reves. El ecommerce te vende algo a primera vista desde que la abrís, mientras que la institucional no).

Son aplicaciones web que sirven exclusivamente para vender algo. Te ofrecen los productos de manera directa. Y te intenta incentivar para que lo compres. No te explica de la mision, vision, metas de la empresa, etc.

(La pagina de digital Bluee es institucional porque nos explica mas sobre que se trata la empresa. No nos vende directamente los productos cuando entramos a la pagina. No nos incita de entrada a comprar el producto por mas que venda algo)

(Mercado libre es una market place)(El market place podemos como usuarios registrarnos y que seamos vendedores y ofrecer nuestros propios productos. Mientras que el ecommerce, es una empresa que vende su producto).

(Youtube es una aplicacion Web/mobile, ya que su unica funcion es ver videos)

- Paginas puntuales

Empezar con las paginas mas simples.

No empezar por el index. Ya que es la mas completa y compleja. Mejor hacer las otras primeros, como la de contacto, preguntas frecuentes, etc. Y podriamos empezar a crear el header y footer para copiar y pegar en cada html del proyecto.

Tambien podriamos usar float si queremos mandar primero alguna seccion a algun lado. Pero con estono puedo hacer que queden uniformemente distanciados.

Si queremos hacer el footer con flex para darle varias secciones dentro, no podriamos manipular el tamaño de cada columna dentro de esta.

Con grid si puedo decidir el tamaño de cada columna. Podriamos en html dividir las varias columnas dentro de un div cada una. Y luego con css aplicarle grid.

CLASE 2.3 CREEEEEEEEEEEEEEEEEEEEEEO

FILMINA

GRABACION(29/12/2022)

CLASE NO SEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE

Filmina

GRABACION (12/1/2023)

Tipos de errores, son 3.

Error de sintaxis en el IDE cuando escribimos mal algo y no respetamos las reglas que tenemos que seguir. El mismo IDE nos dice que está mal incluso antes de ejecutar el codigo y nos lo muestra con subrayado rojo por lo general.

Error que se muestra durante el tiempo de ejecucion. Está correctamente escrita, pero durante su ejecucion nos da un error y no puede continuar. Esto puede deberse a algun error porque el programa intenta una operacion que es imposible que se lleve a cabo. como por ejemplo llamar una funcion que no creaste (el editor no lo toma como error de sintaxis porque de todas formas lo intenta buscar), si esta funcion no existe, el programa intenta una operacion que es imposible que se lleve a cabo.

Error de lógica. El codigo puede dar bien o mal, pero hay un error conceptual nuestro. Cuando uno espera tener un resultado pero al final no es lo esperado. Suele suceder cuando se tiene un gran volumen de codigo.

Try/catch

Try, intenta ejecutar el codigo que tiene. Si se encuentra un error en tiempo de ejecucion aquí, el catch lo toma y ejecuta algo. Luego sigue con el codigo normalmente.

Si no hay error en el try, continua normalmente.

((((La etiqueta script en html se pone arriba cuando eso no tiene nada que ver con el body))))

FILMINA

GRABACION(17/1/2023)

Funciones callback. Son funciones que se llaman entre si.

Funcion anonima. Se ejecuta inmediatamente despues de su creacion.

FILMINA

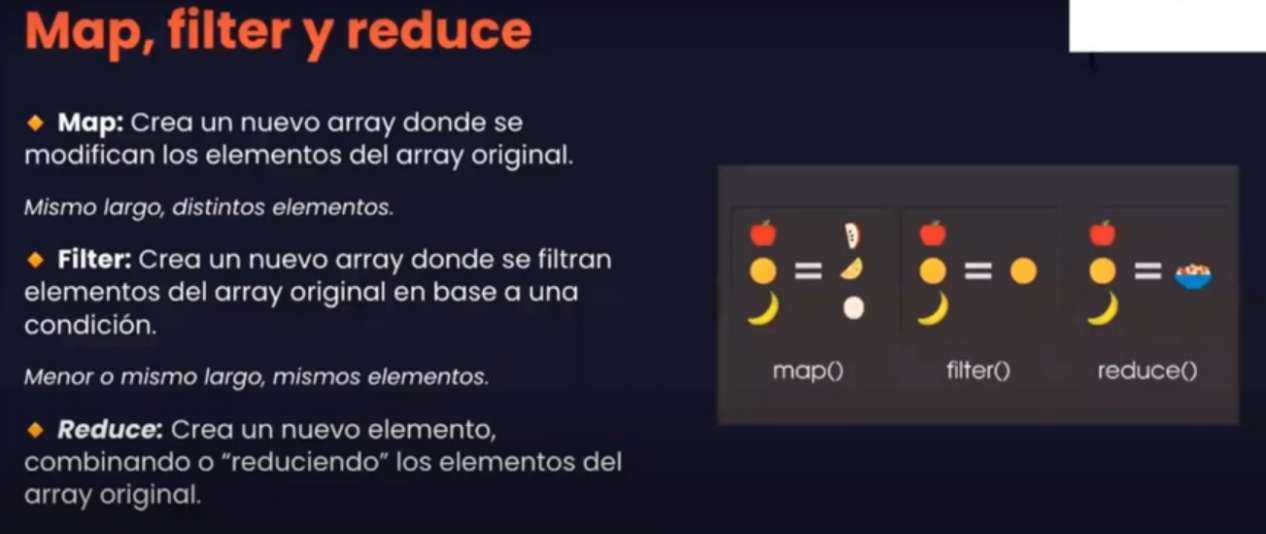
GRABACION (19/1/2023)

FALTAN VARIAS PARA AGREGAR. PERO LAS CLASES GRBADAS Y LAS FILMINAS ESTAN DESINCRONIZADAS DE CONTENIDO

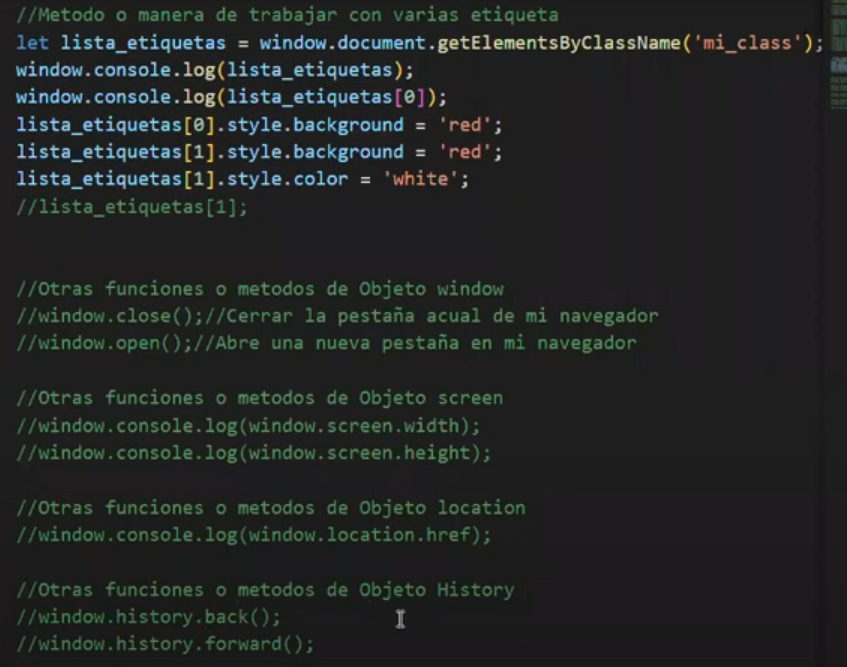
CLASE 2.14

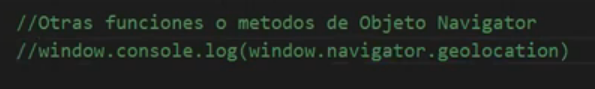
FILMINA

GRABACION (7/2/2023)

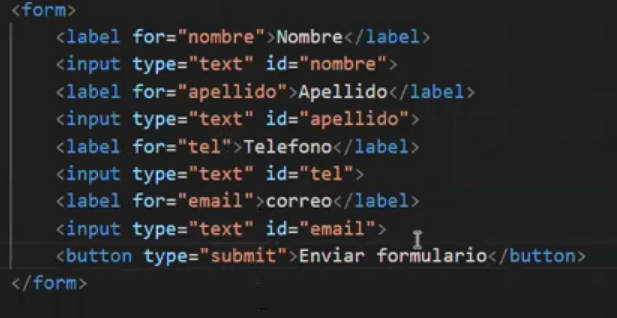


Dom y Bom

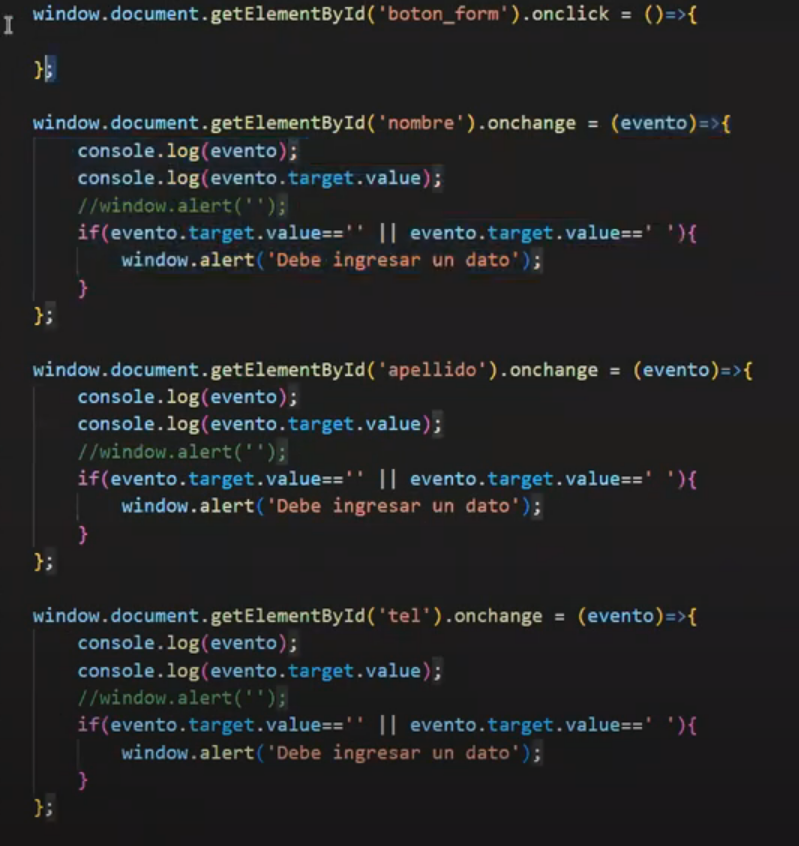




Formulario



Ejemplo de como validar los inputs del formulario de manera manual con JS :



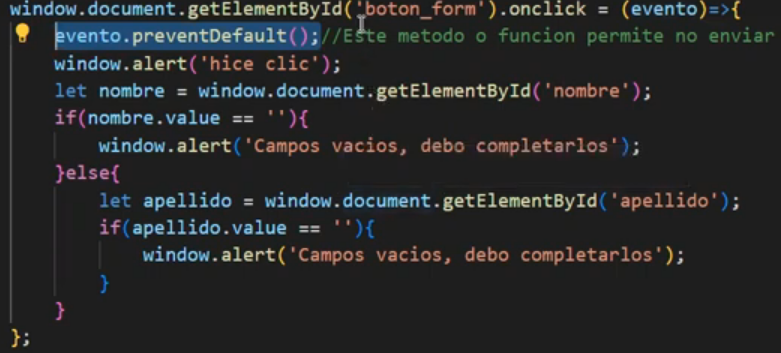
En la proxima se verá como prevenir el comportamiento por defecto del submit entre otras cosas mas como que todas las validaciones queden dentro de ese boton submit.

CLASE 2.15

FILMINA

GRABACION(14/2/2023)

El preventDefault() es un metodo que se le aplica al evento de un addEventListener por ejemplo para formularios. Gracias a eso, no se reinicia la pagina automaticamente, no va a otra pagina automaticamente y permite no enviar el formulario automaticamente aunque esten mal los datos luego de apretar el submit.



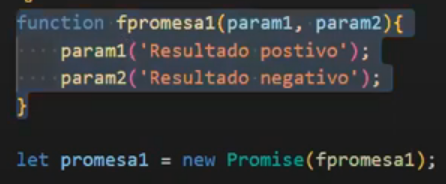
Promesas

Para utilizarlas, surgen de la necesidad de un “mal” uso de las callbacks. En el cual suceden momentos en donde se este esperando a que se termine un metodo para activar otro.

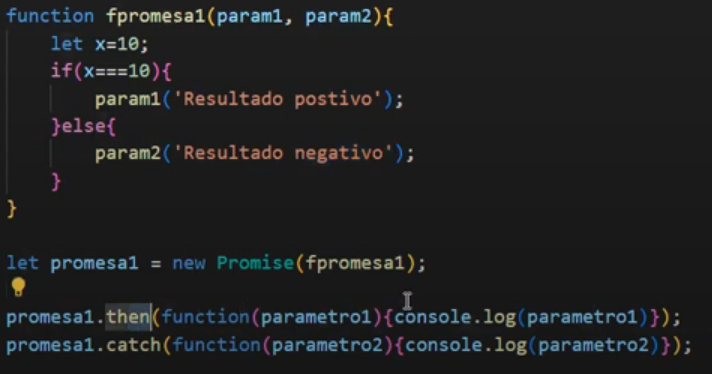
Las promesas son clases de JS que podemos utilizar para hacer tareas asincronas o tareas en segundo plano. Así no se tiene que parar el codigo por esperar algunas funciones. Simplemente sigue con otra linea de codigo.

Al crear la promesa, determinamos que va a haber una funcion asincrona.

En el parametro del new Promise le vamos a pasar cual funcion queremos que sea asincrona

el fpromesa1() es el metodo que queremos que sea asincrono y tiene como parametro un metodo positivo y un negativo, dependiendo el resultado del positivo ejecutará el negativo o no.

Cuando la promesa da como resultado algo positivo, podemos usar el .then() para determinar que hará satisfactoriamente. Y sino el catch() para demostrar que hubo un error.



Conclusion: Tenemos un bloque de codigo el cual queremos que tenga un resultado positivo o negativo (param1 y param2).

Este bloque de codigo para que sea asincrono, lo tenemos pasar por una Promesa. Y gracias a esta, segun el resultado que nos de podemos usar el then para tratar la respuesta positiva. Y sino catch para tratar la respuesta negativa.

CLASE 3.1 React

FILMINA  
React es utilizada por los desarrolladores para crear interfaces graficas en una sola pagina y se basan en componentes en una sola pagina. Es de codigo abierto y tiene mucha documentacion.

SPA (Simple Page Aplication) se refiere a que nuestra pagina va a requerir una sola pagina donde estaría el HTML y los componentes. No es necesario conectar con un servidor para recargar, solo renderiza los componentes.



Para utilizarlo, vamos a requerir el VSC, la extension Live Server, Instalar NodeJs, Git y cuenta de GitHub.

La CDN es la red de distribucion de contenido.

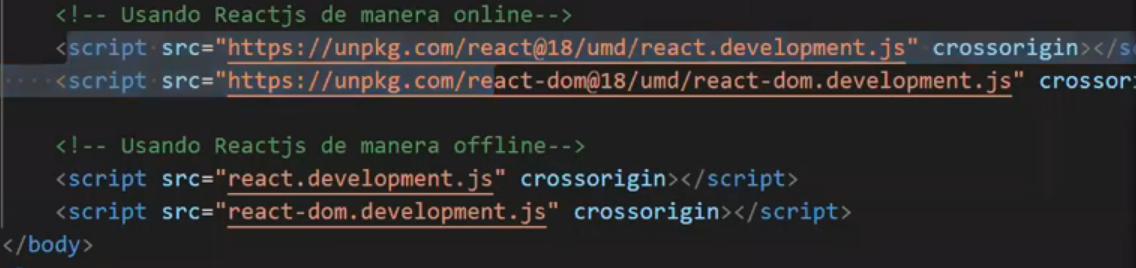
GRABACION (16/2/2023)

Biblioteca/libreria. Conjunto de archivos JS creadas por otras personas para distintos propositos. Tienen metodos propios. Las puedo descargar y usar en el codigo sin necesidad de crearlas yo desde cero.

Asi como en una biblioteca real, tenemos libros conformados por textos en algun lenguaje. En el mundo del software, tenemos carpetas con textos en algun lenguaje.

NoseJS sirve para ejecutar achivo JS sin nencesidad de tener html. Y gracias a esta herramienta tambien podemos manipular la computadora utilizando JS.

Para incorporar React a nuestro proyecto podemos:



La manera online es utilizando esas dos etiquetas que podemos copiar desde la pagina oficial de React. De manera offline lo podemos hacer descargando los archivos de los links mostrados.

Para pasar nuestro codigo de lenguaje React a JavaScript para que el navegador lo entienda, vamos a utilizar la libreria Babel.

Para incorporarlo a nuestro archivo, podemos hacer lo mismo que como hicimos con React.

La extension de un archivo React es .jsx.

Cuando se quiera vincular el archivo React, para que Babel tome el archivo React y lo convierta a JavaScript, hay que hacer lo siguiente con la propiedad type.



CLASE 3.2

FILMINA

GRABACION 23/2/2023

Metodos de React estan mas explicados en el mismo archivo de ejercicio

React está basado en componentes. Los componentes son una agrupacion de etiquetas. Por ejemplo, un carrucel es un componente ya que es un sector de la pagina conformado por su html,css y js que hace una funcionalidad.

Una funcion que me retorne html es un componente.

Las variables en el lenguaje jsx permiten cada una tener solo una etiqueta html. Mientras que las funciones permiten tener un conjunto de etiquetas html agrupadas dentro de una etiqueta padre.

Los componentes simples, guardan y retornan una sola etiqueta. Mientras que los compuestos un conjunto de etiquetas.

IMPORTANTE: En un archivo jsx, TODAS las etiquetas html tienen que tener una etiqueta de cierre.

Asi que hay que prestar atencion a las etiquetas que no tienen cierre como en html (ej la tiqueta img).

CLASE 3.3

FILMINA

GRABACION 2/3/2023

Props. Significa propiedades. Va de la mano con los componentes. Todo componente va a tener propiedades.

Si yo quiero pasar datos a esas componentes, tendré que hacerlo gracias a los parametros.

En React cada componente solo puede tener UN parametro props.

Pero si pasamos muchos datos a traves de un componente padre a varios componentes hijos, se vuelve ineficiente. Para eso es efectivo el Redux. Que mediante una base de datos llama a los componentes para cada uno de los datos.

CLASE 3.4

FILMiNA

GRABACION 7/3/2023

Bootstrap

CLASE 3.5

FILMINA

GRABACION 9/3/2023

Cuando vemos un codigo minificado, significa que se les borra los espacios en blanco, optimizando bastante el codigo. Es algo que voy a tener que hacer a lo largo del tiempo.

Estado. Es una props que nos va a servir para dar una funcionalidad extra a las que tienen las props.

La props es que los estados son datos inmutables y los Estados son mutables.

Esto es mas efectivo que usar metodos para modificar el DOM como haciamos con JS