

Al escribir un simbolo de exclamacion ¡ en VSC ya carga las básicas del HTML.

Se denomina ‘Emmet’

<!DOCTYPE html>

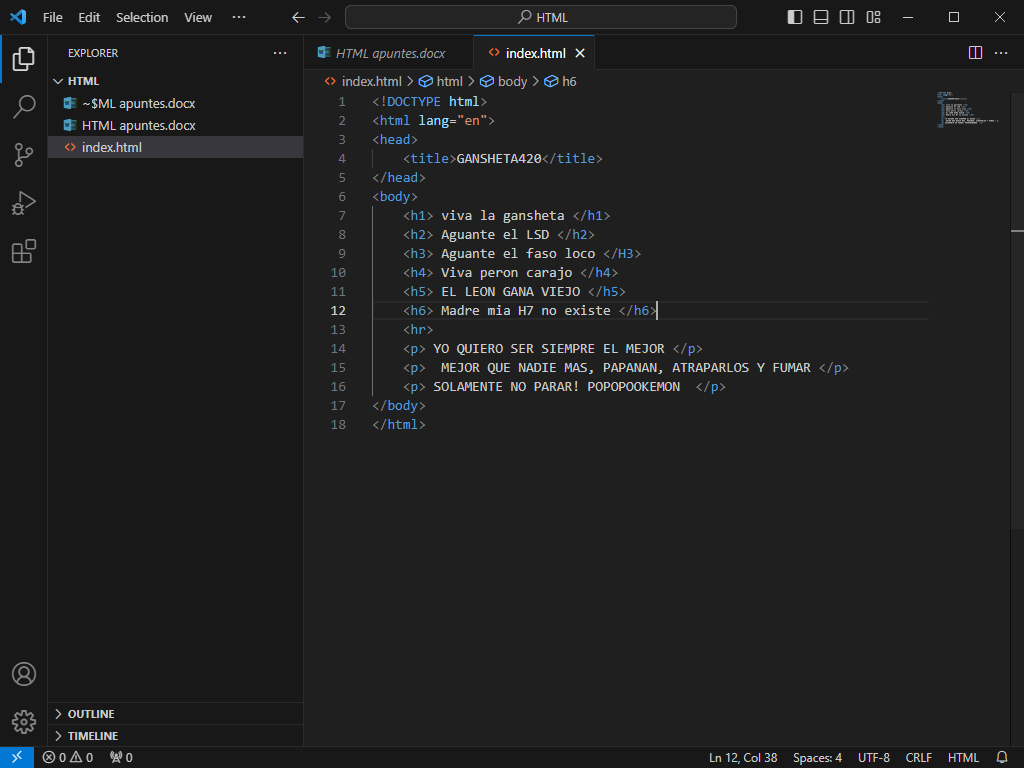
El código presentado significa que es de HTML 5 o moderno si esto no esta, quiere decir que HTML esta en versiones pasadas como por ejemplo de los 90’ por ende no sirve.

Tal vez se modifique en el tiempo pero cuando creamos etiquetas ejemplo <h1> siempre DEJAR SANGRIA!   
Algo muy importante, en términos de desarrollo web es que los H1 o headings no es para demostrar que se va a detallar en el sitio web sino para el motor de busqeuda en general.

Osea los headings definen bastante si te aparece o no en Google.

Solo se llega hasta 6 Headings o títulos mas de eso it’s not posible.

Para escribir párrafo, se usa <p>

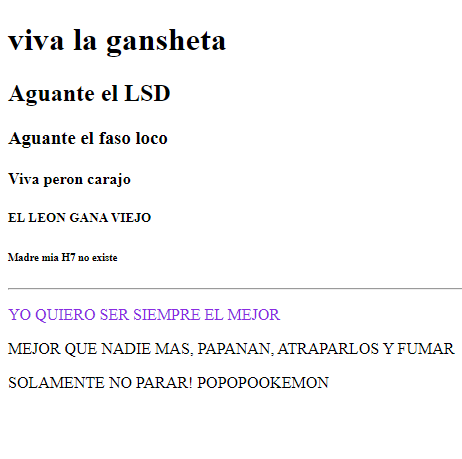
<hr> se utiliza para la famosa línea que divide el texto, separador digamos. 

Para comentarios entre DEVS

 <!-- Comentarios Falopita -->

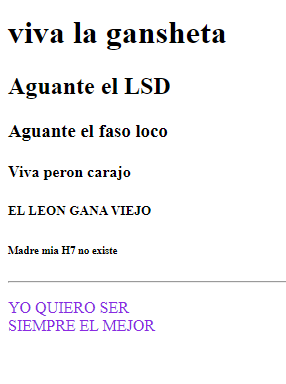
Si tenemos en un párrafo y queremos identificar o remarcar o seleccionar algo especifico se utiliza <span> el cual es CSS y modifica el texto.

<p> <span style="color: blueviolet;"> YO QUIERO SER SIEMPRE EL MEJOR </span> </p>



<p> <span style="color: blueviolet;"> YO QUIERO SER <br> SIEMPRE EL MEJOR </span> </p>

<br> Hace un salto de línea digamos ósea mete un ‘enter’ o un punto final y sigue abajo el texto.



Etiquete de hipervinculo o link.

 <a href="https://www.leanmolina.com/"> Mejor coaching ever</a>

La unfcion es <a> y la denotación con href…

**Ahora esto te crea el link directamente a la pag pero no Crea una pestania.**

Ahora safe for dummies y que al hacer click se abra la ventana, se usa el siguiente comando.

 <a target="\_blank" href="https://www.leanmolina.com/"> Mejor coaching ever</a>

El target es la función seguida por el blank.

Para agregas imágenes se utiliza la siguiente función

<img src = > pero la misma no tiene por que cerrarse como las demás al final.

  <img src="C:\Users\Shampeta\Desktop\Fullstack\HTML\Shampapic.jpg" />

Por ejemplo se cargo la imagen en VSC, se indico el directorio como bien se explica y la imagen se muestra correctamente, lógicamente existen varias maneras de uploadear imágenes y subirlas muy didáctico.

 <img src="C:\Users\Shampeta\Desktop\Fullstack\HTML\Shampapic.jpg" width="400" />

La podes escalar la imagen con Width y Hight functions.

Podemos pedirle al usuario que ingrese información o mejor dicho como hacer un formulario con las siguientes funciones

 <form>

        <input/>

    </form>

Input lógicamente como dice su termino y form de formulario, pero vamos a necesitar etiquetas.

  <form>

        <label> Nombre y Apellido </label>

        <input/>



Ahora algo menor pero no menos importante, sin identificar o manipular el label no toma el input de una para retrasados, por eso se hace lo siguiente.

 <label for="Nombre y Apellido" > Nombre y Apellido </label>

        <input id="Nombre y Apellido" />

Eso hace que cuando se clique ‘nombre y apellido’ ya aparece para inputear.

Ahora necesitamos como bien sabemos ponerle un nombre para poder reconocer al formulario y mas tarde asignarle un lugar o una dirección.

   <input id="Nombre y Apellido" name="Nombre" placeholder="Nombre y Apellido" />

Vease propiedad Name como le asignamos un nombre para identificar y placeholder hace sombreo para retrasados:



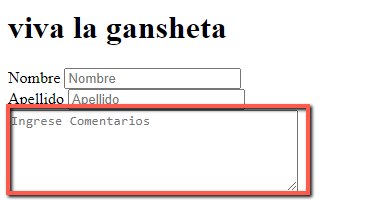
**Funcion importante “type” le agrega el tipo de función dependiendo del valor ingresado.**

 <input type="button" id="Nombre y Apellido" name="Nombre" placeholder="Nombre y Apellido" />



Por ejemplo existen tipos de type usuales alo: Email , Password, file para que suban archivos.

<textarea id="Comentarios"name="comentarios" placeholder="Ingrese Comentarios"></textarea>



Funciones como ‘Row’ y Col’ agregan filas y columnas.

  <Label> Comentarios </Label>

        <label for="comentarios"></label>

        <textarea cols="50" rows="10" id="Comentarios"name="comentarios" placeholder="Ingrese Comentarios"></textarea>

        <input type="submit" />

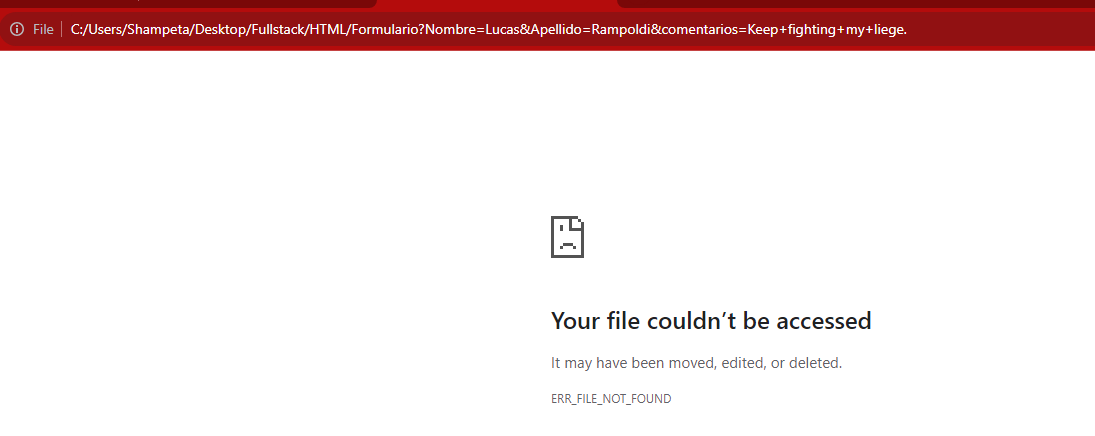


Estoy peleando por que comentarios quede arriba. ☹

 <form action="Formulario" method="get">

Ahora fijémonos, que pasa cuando ingreso información al escribir las funciones previamente nombradas.



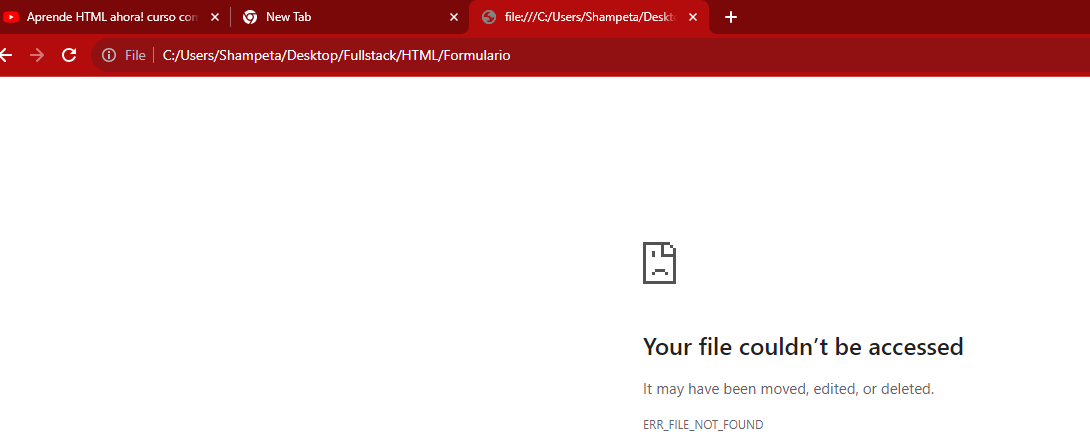


[file:///C:/Users/Shampeta/Desktop/Fullstack/HTML/Formulario?**Nombre**=Lucas&Apellido=Rampoldi&comentarios=Keep+fighting+my+liege](file:///C:/Users/Shampeta/Desktop/Fullstack/HTML/Formulario?Nombre=Lucas&Apellido=Rampoldi&comentarios=Keep+fighting+my+liege).

Por ejemplo en negrita se remarco Nombre por la función en si del codeo.

Si yo voy y cambio de nombre en la función ‘Name=’ se vería reflejado.

**Ahora que pasa si cambiamos la función previamente mostrada ‘Get=’**



**Ambas funciones sirven tanto Get como POST es mas cuestión persona.**

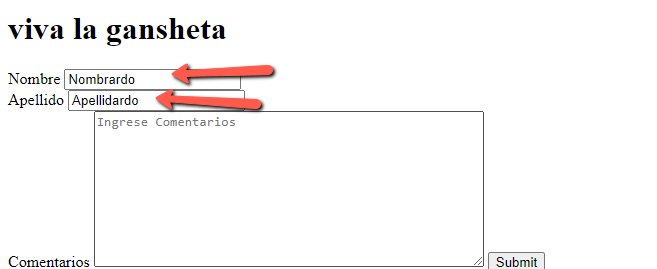
**Generalemente POST se usa para mas cantidad de datos que GET.**

Con la función ‘Value’ lógicamente le agregamos ‘valor’ a una función.

Por ejemplo, véase la utilización en el código y pagina.

 <input value="Nombrardo" type="text" id="Nombre" name="Nombre" placeholder="Nombre" />

<input value="Apellidardo" type="text" id="Apellido" name="Apellido" placeholder="Apellido" />



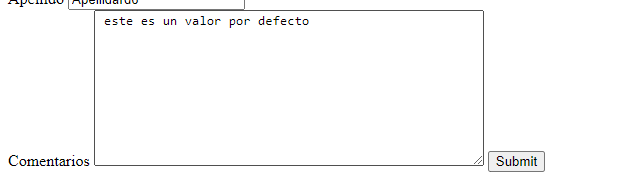
Ya figura asignado el valor como muestran en los campos de Nombre y Apellido.

<textarea value="Este es un valor por defecto" cols="50" rows="10" id="Comentarios"name="comentarios" placeholder="Ingrese Comentarios"></textarea>

El value con textarea no funciona asi, no lo explica el gordito anda a saber por que.

 <textarea cols="50" rows="10" id="Comentarios"name="comentarios" placeholder="Ingrese Comentarios"> este es un valor por defecto </textarea>

Appropiate coding.



BOTONES EN HTML

<input type="submit" />

El tema de como esta hecho es que no podes ponerle etiquetas o label.

Es importante cuando especificas un botón en HTML escoger su tipo, para esto se usa ‘type’

Que tiene 3 propiedades

* Button se epsecifica con JS el comportamiento que este tenga.
* Reset resetea de los valores prestablecidos.
* Submit Va a enviar la info en el form asignada junto con POST O GET.

Ejemplo Submit: <form action="Formulario" method="post">

<button type="button"> Enviar </button>

Bueno como no esta terminado no hace nada el botón pero se entiende.

Ahora reset restabelce los datos que estaban ya en un principio ósea como vemos anetrioramente ya en nombre apellido y comentarios venia cargado info, si apretamos reset luego de escribir vuelve al estado normal.

Boton vs Submit el botón permite mas label y customización.

LISTAS EN HTML

Empezamos con la funcion UL (Lista no ordenada) Quiere decir que cada vez que nosotoros agregemos elemenots a esta lista, estos van a parecer con un puntito en negro.

Funcion <LI> List item

    <ul>

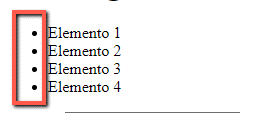
        <li> Elemento 1 </li>

        <li> Elemento 2 </li>

        <li> Elemento 3 </li>

        <li> Elemento 4 </li>

    </ul>

 Los puntos negros de la función UL.

Como toda en la vida ahora vemos la función OL (Organized List)

   <ol>

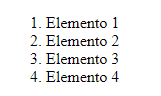
        <li> Elemento 1 </li>

        <li> Elemento 2 </li>

        <li> Elemento 3 </li>

        <li> Elemento 4 </li>

    </ol>

 Como vemos mas que oganizada o no diría enumerada pero bueno bien.

Con ‘Value’ le agregamos valores, descripción grafica.

<ol>

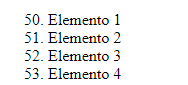
        <li value="50"> Elemento 1 </li>

        <li> Elemento 2 </li>

        <li> Elemento 3 </li>

        <li> Elemento 4 </li>

    </ol>

 Ver para creer.

Vamos a ver como ANIDAR listas El anidamiento (llamado nesting en inglés) es la práctica de incorporar llamadas (calls) a funciones o procedimientos (unas) dentro de otras,

        <li>

            <ol>

                <li> Sub elemento 1 </li>

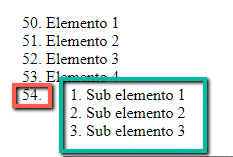
                <li> Sub elemento 2 </li>

                <li> Sub elemento 3 </li>

            </ol>

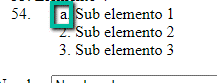
        </li>

    </ol>



La lista siguió de 53 a 54 y después se subdividió pero sigue perteneciendo a la misma.

<li style="list-style-type: lower-alpha;"> Sub elemento 1 </li>



Style cambiando el tipo.

PERO NO PARA EL RESTO DE LOS ELEMENTOS WACHIN, POR QUE SE LO PUSISTE SOLO A 1 NO? MALARDO

                <li style="list-style-type: lower-alpha;"> Sub elemento 1 </li>

                <li style="list-style-type: lower-alpha;">  Sub elemento 2 </li>

                <li style="list-style-type: lower-alpha;"> Sub elemento 3 </li>

Ahí si funca paja subir la imagen.

TABLAS!

Funcion <table> creadora de las mismas <tr> “Table Row” también <th> table header.

    <table>

        <tr>

            <th> Producto </th>

            <th> Precio </th>

        </tr>

    </table>

No pongo screen por que es al pedo ya que se cargo headers nomas y nada mas en la table, figura en la pagina Producto Precio como header.

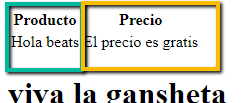
Por otro lado la función <td> la misma etiqueta se usa para definir elemento estanderes que se encuentra en la tabla.

        <tr>

            <td> Hola beats </td>

            <td> El precio es gratis</td>

        </tr>



El senior explica como esta anclada la función TH con la TD véase la imagen.

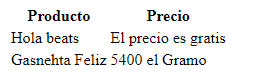
Ahora para que todo sea ordenado y prolijo vamos a tener que entender de columnas y filas.

        <tr>

            <td> Gasnehta Feliz</td>

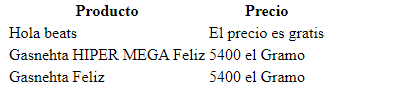
            <td> 5400 el Gramo </td>

        </tr>



Ahora fijemosnos que Producto Precio al usar TH de header se alinean bien y se detalla abajo prolijamente, obviamente al escribir gansheta feliz o palabras largas queda mas engorroso.

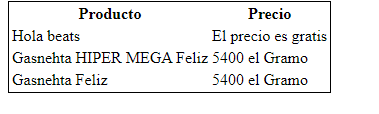
Agregeamos algo de ese estilo



Producto y precio al ser headers están alineados, pero ya podemos empezar a ver a que nos referimos, ahora vamos a hacer que los elementos dentro de la tabla tengan una especie de separación para que sea mas fácil e ordenado dstingirlos.

Funcion que modificaremos en <table>

 <table style="border: 1px solid black;">



**ATENCION ahora vamos a hacerlo para todas las filas, PEQUENIA INTRO A CSS.**

Pero la función que vamos a ver no va dentro de Body, como va la mayoría de funciones, la misma la vamos a utilizar en <head>

</head>

    <style>

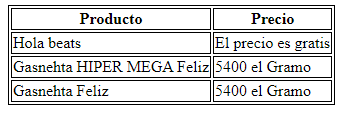
        table, th, td {

            border: 1px solid black;

        }

     </style>

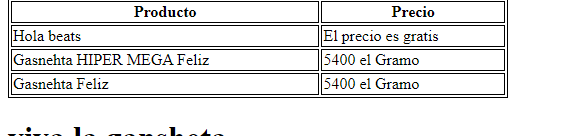
Volvieron los corchetes si! Jsjsjsjs lpqmp, miremos como table, th, td se asignan. AHORA CORTAMOS NO COPIAMOS donde estaba el border 1px solid black y lo pegamos ahí OJALDRE!.



Ahora, como todo esta mas pegado ,ordenemoslo aun mas.

SIEMPRE QUE AGREGAMOS <STYLE> VA PUNTO Y COMA!

<table style="width: 500px;">

Vease que lo pusmos en table. 

GOD. Ahora nos falta ver como separamos de manera logica, para indicarle al explorador que lo se encuentra al comienzo es una cabecera y el resto cuerpo.

Aca entra la funcion <thead> esto sirve para agrupar las cabeceras que se encuentran en nuestras tablas.

Otra etiqueta que vamos a ver es <tbody> que sirve para identificar y agrupar todas las filas que peretenzacn al cuerpo de nuestra tabla, para eso voy a recortar todo lo que este adentro de <tr> digamos la infod e los productos y precios.

<body>

    <table style="width: 500px;">

        <thead>

        <tr>

            <th> Producto </th>

            <th> Precio </th>

        </tr>

       </thead>

       <tbody>

        <tr>

            <td> Hola beats </td>

            <td> El precio es gratis</td>

        </tr>

        <tr>

            <td> Gasnehta HIPER MEGA Feliz</td>

            <td> 5400 el Gramo </td>

        </tr>

        <tr>

            <td> Gasnehta Feliz</td>

            <td> 5400 el Gramo </td>

        </tr>

    </tbody>

    </table>

Ahora, no subo la foto por que se ve igual, PERO que paso? Que cambio? Por que esto sirve para que algunos exploradores web habilitan un scroll de contenido separado de la cabezera, y el pie de nuestra tabla (que todavia no agregamos)

Continuaremos agregando nuestra etiqueta de footer, y ahí entra la funcion <tfoot>

    </tbody>

        <tfoot>

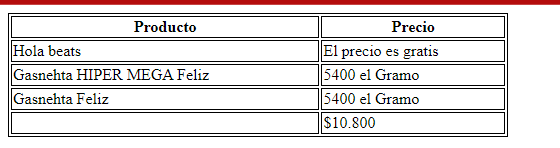
            <td></td>

            <td> $10.800 </td>

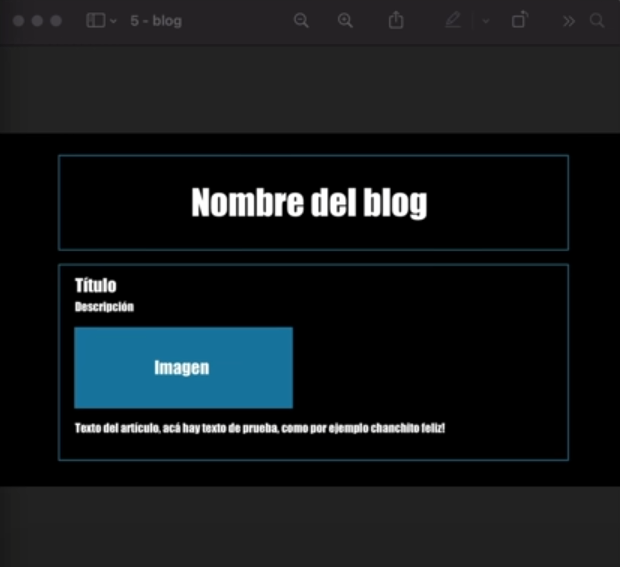
        </tfoot>

    </table>

Fijemosnos como al final de la tabla y el codeo mismo.



Bueno a continuación nos va a enseniar el mister a hacer un blog etc.



La función <header> se utiliza para poder agrupar links también alguna imagen o logo de esta pagina y lo mas importante es que podemos crear tantas etiquetas de Header que queramos MIENTRAS se cumplan las siguientes reglas.

1. No podemos colocar una etiqueta de header dentro de otra etiqueta de header
2. No podemos colocar una etiqueta de header dentro de una etiqeuteqa de footer.

AHORA veamos la etiquta <Footer> se utiliza para colocar el pie de pagina de nuestra web.

Podemos tener multiples etiquetas de footer dentro de la app y por lo general contienen copyright, info de contacto, como navegar en el sitio etc, (Esto no es una regla obivo es un ejemplo)

La otra etiqueta es <section> como lo dice el nombre define secciones de la app.

Ahora estructuramos nuestra pagina desde lo superior header – section y footer.

    <header> </header>

    <section> </section>

    <footer> </footer>

Otra etiqueta es <article> se usa para definir contenido completamente independiente en este caso, funciona muy bien para definir los post de un blog. O para una historia de noticias etc.

<body>

    <header> </header>

    <h1> Foro de la Gansheta Feliz </h1>

    <section> </section>

    <article>

        <h2> Titulo de mi post</h2>

        <p> Descripcion del post</p>

        <img src="Shampapic.jpg" />

        <p> texto del articulo que hay aca perro</p>

    </article>

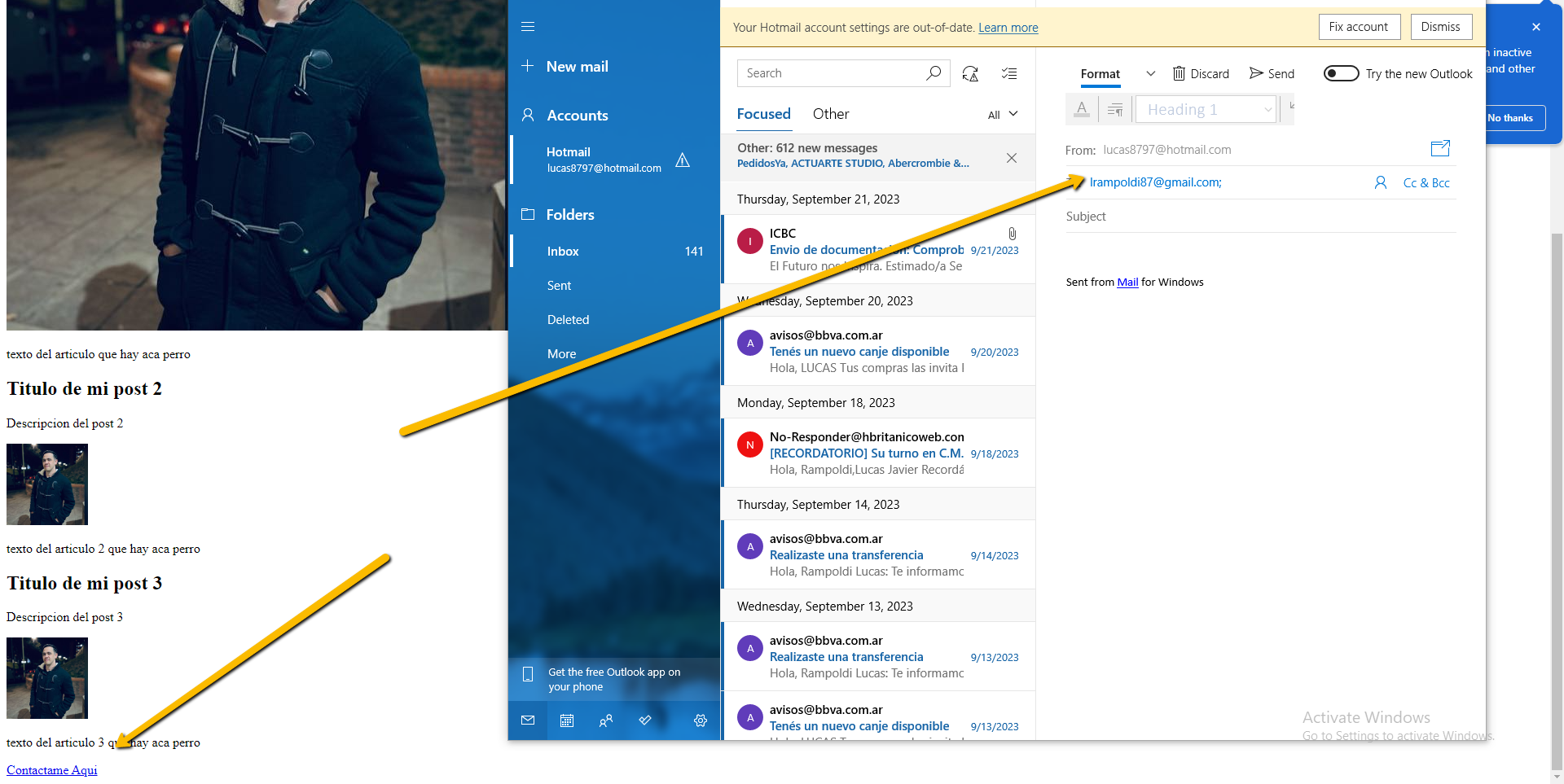
    <footer> </footer>

</body>



Bueno terminando el video nos muestra como dejar para que te envíen un mail.

 <a href="mailto:lrampoldi87@gmail.com"> Contactame Aqui </a>



**Etiqueta importantísima <div> en HTML**

Bueno DIV en si tiene la misma función que <section> se utiliza para poder dividir el contenido o agrupar en secciones. Era mas usada antes que salgan section y article.

Cambiando de tema.

Van a existir momentos donde vamos a querer darle un identificador único o genérico en nuestras etiquetas HTML ya sea mediante JS o asignarle algún estilo para que cambie la forma del design.

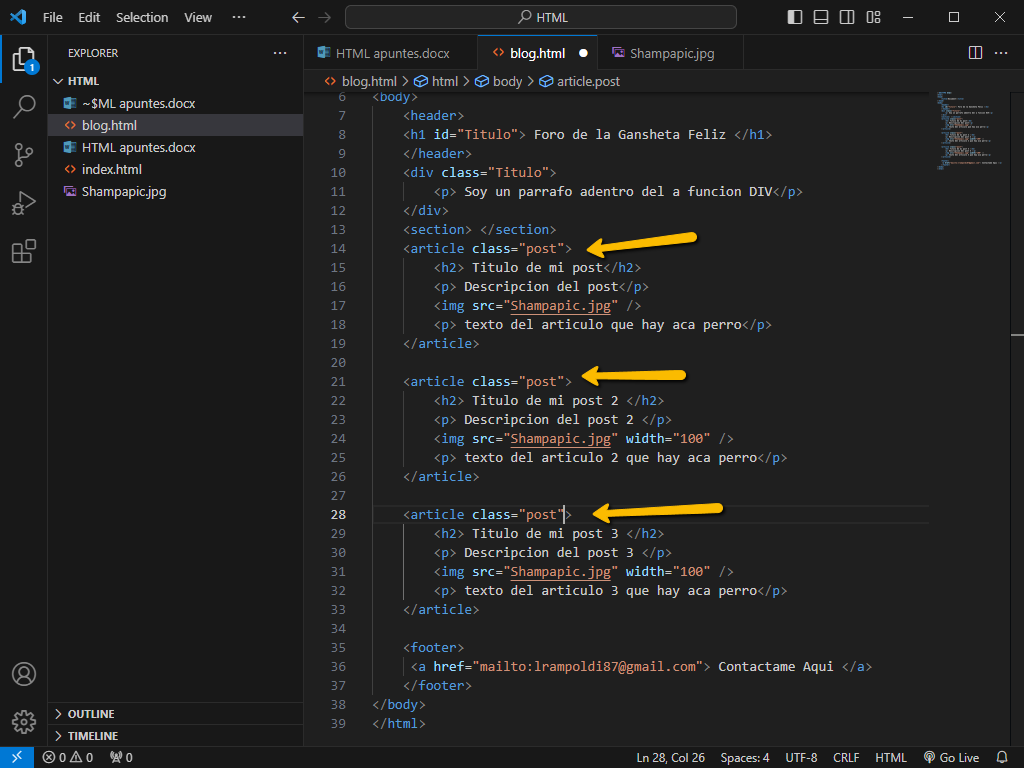
 <h1 id="Titulo"> Foro de la Gansheta Feliz </h1>

Esto no va a cambiar en lo absoluto en el comportamiento del titulo, pero esto nos permite interactuar con estos elementos ya sea mediante Java script para asignarle un elemento o lo que sea.

Ahora, ponele que queres usar Titulo como ID pero ya lo estas usando, por LOGICA no deberías replicarlo y que existan 2 o mas id=titulo. Aca entra en escena la siguiente función <class>

 <div class="Titulo">

Como aprendimos lo mejor es usar nombres distintos.



Ahora fijemosnos como justo en article pusimos class post, para eso nos sirve en CSS Y JS Para diesniar los posts.

Bueno vamos a estudiar ahora la etiqueta <a> (ANCHOR ADD O NO SE QUE BOSTA), no la vimos antes por que no habíamos visto ID

<a href="#Titulo"> Ir al comienzo </a>

Bueno el comando href=#titulo es un hipervínculo que en este caso te lleva, miremos como tomamos el nombre del ID titulo y nos lleva al principio de pagina.



<b> es negrita o también existe <strong> xD - <i> italic o cursiva. <em> es énfasis.

<blockquote> se usa para citar a alguien o algo.

<code> Gansheta Codigo </code>

Aparece la letra como en forma de código.

<pre> de preformataedo. <small> es para texto pequienio.

<u> es para subrayar

<text área> es un cuadrado de texto modificable para el usuario.

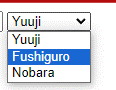
Bueno ahora entra <select> y <option> como bien indican los términos se usa para darle al

Usuario opciones y que elija.

                <option value="Yuuji"> Yuuji </option>

                <option value="Fushiguro"> Fushiguro </option>

                <option value="Nobara"> Nobara </option>



<label> etiqueta pero tricky no?

Bueno veamos como funca mas que nada.

 <label> Edad </label>

            <input type="number">

            <br />

            <label> Contrasenha </label>

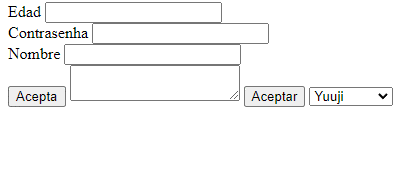
            <input type="password">

            <br / >

            <label> Nombre </label>

            <input type="text" />

            <br / >



Ojo algo nuevo <fieldset> pone un marco parecido a lo que vimos antes.

Y algo peculiar <legend> es el titulo del fieldset digamos

        <fieldset>

            <legend> Formulario Canabico </legend>

        <form>

            <label> Edad </label>

            <input type="number">

            <br />

            <label> Contrasenha </label>

            <input type="password">

            <br / >

            <label> Nombre </label>

            <input type="text" />

            <br / >

            <input type="submit" value="Acepta" />

            <textarea></textarea>

            <button> Aceptar </button>

            <select>

                <option value="Yuuji"> Yuuji </option>

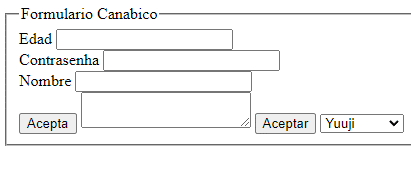
                <option value="Fushiguro"> Fushiguro </option>

                <option value="Nobara"> Nobara </option>

            </select>

        </form>

    </fieldset>



<form action =” (destino de la pagina)> sirve para enviar la información de un formulario a otra pagina.

