**HTML**

**Introducción:**

Las Páginas web son actualmente páginas de texto enriquecido con efectos y formas que lo hacen visiblemente atractivo a nuestros ojos. Que necesitamos para ver eso?

1. Una conexión a internet
2. Un navegador.

Un navegador no entiende angular, react js, o php, etc. El navegador solo entiende el lenguaje HTML. Que es HTML?

Hyper Text Markup Language, lenguaje de marcador de contexto.

“Varias tecnologías (como css, js, flash, json, Ajax, etc) pueden usarse para definir los elemento que componen una página web. Sin embargo en el nivel más bajo, una página web se define usando HTML. Sin HTML, no habría páginas web.” Mozilla Developer Network

Historia: Tim berners lee, creo en los 80 creo el protocolo http, y la manera de compartir archivos por medio de una red que se conocería después como la internet. Y creo el lenguaje HTML, y sentó las bases de lo que hoy manejamos y vemos en la red.

**Elementos HTML o Etiquetas: Morfología**

**Etiqueta de apertura**

Inicia con menor que (<), cuenta con un elemento dentro y cierra con un mayor que (>).

**Etiqueta de cierre**

Inicia con menor que (<), seguido de slash (/), cuenta con un elemento dentro y cierra con un mayor que (>).

Observación: Algunas etiquetas no llevan una apertura o cierre específicos, se pueden concatenar.

**Estructura de un archivo HTML**

1. Head (Cabecera): Van los elementos que en un 95% no son visibles en el navegador. En la página web como tal.
2. Body (Cuerpo): Todo lo que ponga es evidente en el navegador.

**Comentarios:** me permiten ordenar mi código y orientar dentro del código. No se hacen visible en la página web. <!-- COMENTARIO-->

**ELEMENTOS DE BLOQUE, FRASE, VACIOS Y DE CITA**

**Encabezados:** me definen la estructura dentro de mi documento, el nivel de títulos, subtítulos por ejemplo. H1, h2, j3, h4…etc Se ven uno debajo del otro, y cada uno letras y estilos predefinidos. **Realizar ejercicio para determinar hasta qué nivel de encabezados existe.**

**Párrafos:** Elementos que nos permiten agrupar de manera coherente y se vea agradable visualmente. <p></p> Entre los diferentes párrafos, siempre va a existir un espacio hacia abajo. **Ejercicio: Agregar varios párrafos y decir que característica especial se ve en el navegador.**

**Elementos vacíos:** no requieren etiquetas de cierre. <br> Salto de línea obligado

<hr>. Línea de separación.

**Elementos frase:** Ayudan a dar formato del texto HTML, sin usar herramientas de diseño.

<strong> negrilla.

<em> Estilo itálico o inclinado

<small> texto de un tamaño más pequeño. Pie de página y elementos de ese tipo

<mark> Resaltar de alguna manera.

<sub> subíndice <sup> superíndice

Anidación, cerrar correctamente los elementos que van por dentro de otros.

Ejercicio: realizar una página que contenga 5 párrafos en donde por lo menos se utilice 4 veces cada uno de estos elementos frases. Aparte debe contar con un título y 3 subtítulos.

**Elementos para citar e indentar:**

<blockquote>, Se usa para citar párrafos. De ahí que su nombre contenga la palabra bloque. Tab a la izquierda y a la derecha.

<q> Permite citar frases.

<address> Direcciones de lugares.

**ATRIBUTOS**

Un atributo es una característica que le doy a mi etiqueta, y cuenta con una palabra clave y un valor. Ejemplo: lang=”es”, lang= “en”, en la cual determino el idioma de mi texto. Chartset es otro ejemplo.

**Ejercicio: Centrar los títulos y justificar los textos del cuerpo. Con los que se cuentan hasta ahora.**

**ENLACES, LINKS, VINCULOS O HIPERVINCULOS**

LINK REMOTO: Me lleva a un sitio que yo desee, fuera de mi sitio. <a href=<https://www.google.com> target=”blank”>TExto</a>

LINK local: <a href=””>.

**Ejercicio:** Crear otro archivo html, donde se evidencie la navegación con un link local.

MAIL: <a href=”mailto:xxx@gdgkds.com”>

LINK ANCLADO: : <a href=”mailto:xxx@gdgkds.com”>

*Atributo title: cambia el link a mostrar*

**IMÁGENES**

FORMATOS: JPG-GIF- PNG

Crear carpeta llamada “images” y almacenar 2 o 3 imágenes.

Etiqueta: <img src=”” alt=”No encontró la imagen”>

**Ejercicio: Unir una etiqueta link con una etiqueta imagen.**

**LISTAS**

Ordenadas:<ol><li></li></ol>, type, números, letras mayus, minus, romanos mayus, minus etc. start=7,

No Ordenadas: <ul>, type: disc, circle, square

Lista de definición: Glosario o diccionario. Dt: definition term, dd: definition data, dl: definition list.

**TABLAS**

**DIV Y SPAN- ELEMENTOS CONTENEDORES**

div, me permite agregar varios elementos en forma de bloquey asignares características especiales por medio de css. Si no se les aplica nada va a pasar con nuestros documentos.

Span hace la misma funcionalidad del div pero por líneas.

**IFRAMES**

Me permite visualizar elementos de otras páginas en mi página HTML.

Ejemplo google maps

**Ejemplo: mostrar html inicial, mediante un iframe.**

**CARACTERES ESPCIALES**

**FORMULARIOS**

**Form, etiqueta**

Atributos: action: me almacena los datos ingresados en el formulario, method, es la manera en cómo se van a enviar los datos. GET cuando la data es corta, y se envía por medio de la url (publico). (Maximo 8192 bytes). POST son datos privados y soportan mayor tamaño de datos.

**Elementos de texto: input, textarea, types.**

**Elementos checkbox, radio button.**

**Elementos de menú de selección y listas de selección:** select para menus de selección,

**Label:** etiqueta para un elemento en especifico

**Fieldset:** A nivel visual me enciarro uno o muchos elementos

**Legend:** titulo de la sección marcada por el fieldset

**Elemento submit:** Boton de enviar

**SCRIPT**

Permiten interactuar con la página que estoy creando, tienen una función específica.

**CSS**

**Que es CSS:** Cascading styled sheets/ hojas de estilo en cascada. Es un mecanismo simple que permite controlar todo el diseño de nuestros proyectos. Hacer las cosas bonitas, atractivas al cliente o público en general.

**Sirve para dar orden ya que maneja el estilo y contenido por aparte.**

En pocas palabras, el HTML es la parte estructural de un proyecto web, lo pesado, fuerte pero que sirve para que nada se derrumbe. Mientras que el CSS es la parte de pintura, estética que da distinción.

Existen 4 clases:

1. Inline
2. Incrustadas
3. Vinculadas
4. Importadas

Tienen una jerarquía, por eso van es cascadas. En ese orden se determina la prioridad.

**Sintaxis: selector {declaración (propiedad:valor)}, ejemplo: p {color: red}**

1. Estilo inline: style, sobre la etiqueta o elemento sobre el cual quiero ejercer el diseño.
2. Estilo incrustado: Se definen en la cabecera del documento con la etiqueta style y allí se ponen todos los elementos que van a afectar a mi body.
3. Estilo Vinculado: Son similares o funcionan igual a las incrustadas. Pero su ubicación es externa y se enlazan en la cabecera. Rel=”stylesheet” type=”text/css” href=”ubicación”
4. Estilo importadas: Enlazo múltiples documentos css para no repetir código. @import url(nombreArchivo)

**Comentarios:** /\*\*/

**Selectores:** Existen

1. Universales (\*) => \* {}
2. Tipo o de etiqueta (especificar el elemento ejemplo p, h1. H2 etc)
3. Class= . algo. No caracteres especiales ni espacios, class en el html para llamr
4. Id= #algo algo. No caracteres especiales ni espacios, id en el html para llamar. Solo se puede aplicar una vez en el archivo html.
5. Pseudo-class= :link :visible :active :hover
6. De atributo: [href]

**COLOR**

El color se puede expresar en rgb, hexadecimal: ejemplo #fffff o (255,0,0)

**PROPIEDAD DE FUENTE**

**Font- family**: Me permite seleccionar la familia de la página. Puedo seleccionar una sola o toda la familia.

**Font-size**: me permite cambiar la unidad de medida: pixeles, puntos, em, ml, ct, etc. Se recomienda px, em(pixeles/16) y %.

**Font-weight:** normal, bold o un valor 100, 500. En el ancho de la letra.

**Font-style:** Modifical el esilo de la fuente normal, italic.

**Font-variant**: aspecto de la fuente small-caps, normal

**Line-height=** interlineado de las líneas porcentaje, 1.5, número (36 px).

**PROPIEDAD DE TEXTO**

**Text-align:** permite alinear el texto según desee, centro, derecha, izquierda, etc.

**Text-indent:** Indentado francés, correr el texto a la derecha según las líneas con las que cuente un texto.

**Text-transform:** cambia el estado de un texto a otro.

**Text-decoration:** Me permite poner una línea en un texto.

**Vertical-align**: Permite alinear con respecto al eje vertical los textos.

**Letter-spacing:** espacio entre letras.

**Word-spacing:** espacio entre palabras.

**PROPIEDAD DE FONDO**

**Backgroung-color:** define el color de fondo de un elemento.

**Background-image:** url() y lo setea de fondo del docs.

**Background-repeat:** me permite repetir la imagen que he colocado de fondo.

**Backgroung-postition:** Me permite especificar la posición de mi background.

**Background\_attachment:** me permite definir cómo se va a comportar mi fondo, al realizar acciones como scroll.

**PROPIEDAD DE CUADRO**

**Width**: ancho que quiero utilizar en mi elemento.

**Height**: altura que quiero utilizar en mi elemento.

**Background:** fondo

**Overflow:** scroll, hidden(oculta el contenido), auto el navegador decide

**Padding:** del borde, hacia dentro del elemento

**Margin:** del borde, hacia afuera del elemento.

**Border**: width, style, border-color, border-top, border-right, border-left, border-botton.

**PROPIEDAD DISPLAY**

**Display:** inline- elemento de línea(en la misma fila), inline-block elemento en bloque pero en línea. Block: no pueden tener nada al lado, none: se desaparece.

**Visibility:** hidden- invisible pero conserva el espacio.

**PROPIEDAD DE LISTA**

**List-style-type:** Cambia la viñeta de la lista

**List-style-image:** No va de la mano, la de arriba. Me permite subir la imagen de la viñeta.

**List-style-position:** offsite, inside. Permite entrar la viñeta o sacarla de la lista.

**PROPIEDAD DE TABLAS**

**Empty-cells:** hide, show. Me permite ocultar o mostrar las celdas que se encuentren vacias.

**Border-spacing:** espacio entre celdas

**Border-collapse:** collapse, permite que los bordes estén pegados, omite el border spacing.

**POSICIONES**

Todos los elementos del html, tienen por defecto tienen asignada una posición estatica.

**Static:** Respeta la posición en que se encuentran en el html.

**Relative:**  si, yo no le indico algo diferente me pinta todo como si fuera estatico. Para especificar esto, uso las coordenadas, Top, bottom, left, right. Estás me hacen mover mi elemento, sin importar si se sobre pone a otros elementos.

**Absolute:** Cuando se especifica esta características, pierden las referencias previas y debo registrar las coordenadas y demás atributos. Esto es según la posición de su padre. Siempre y cuando este tenga position diferente a static. De lo contrario toma la del navegador.

Me permite especificar una posición, pero se pierde el ancho del elemento y su flujo en su navegador. Se desplaza según las coordenadas que yo le especifique.

**FIxed:** Al igual que la position absolute, se pierde el orden de sus elementos, y se ubica siempre con las medidas del navegador. A diferencia que este elemento no se mueve al realizar scroll.

**PROPIEDAD Z-INDEX:**

Con posición static, no se puede efectuar esta propiedad. Y permite poner sobre el eje z un orden en los elementos en pantalla según el navegador.

Las coordenadas de ubicación deben ser positivas. Se recomienda que haya diferencia en la propiedad z-index entre los elementos al menos de 10.

**PROPIEDAD FLOAT**

Float, me permite ubicar elementos de manera especifica en algún lugar de la pantalla. Y los demás elementos se ubican alrededor de este.

**AGRUPAR SELECTORES:** h1,h2,h3{}