

UNIVERSIDAD CATÓLICA DE SANTA MARÍA
PROGRAMA PROFESIONAL DE INGENIERÍA DE SISTEMAS

SESIÓN N° 02:

CSS

I

OBJETIVOS

- ❖ Conocer los fundamentos de configuración web en aspecto y visualización.
- ❖ Añadir las características que habitualmente incorpora a una página web codificada en CSS.
- ❖ Diseñar interfaces de usuario en entorno web.

II

TEMAS A TRATAR

- ❖ Introducción.
- ❖ HyperText Markup Language
- ❖ Formularios con HTML
- ❖ ASP.NET
- ❖ Resumen

III

MARCO TEORICO

1. INTRODUCCIÓN

En el mundo de la internet se hace uso de páginas web, para su construcción se utiliza el lenguaje HTML, pero las características gráficas y de presentación se hacen a través del lenguaje CSS, la forma como se hace el trabajo con ambos lenguajes es diferenciada y aparte a pesar de que se integran para darle estructura y presentación visual adecuada al documento web que se recibe en el navegador. [1]

Las características gráficas o expandir a los requerimientos modernos de manejo de información permitiendo múltiples características de aspectos gráficos y manejo de recursos modernos que permiten una presentación de documento visualmente atractiva funcional y que se adapta a las características del dispositivo que se utiliza para visualizar el documento web. [2]

Actualmente se trabaja con la versión del lenguaje CSS 3, el cual ha sido modernizado para representar las características gráficas y de animación que son necesarias en la presentación de cualquier documento web moderno, debemos tener en cuenta que el trabajo con el lenguaje CSS es diferenciado de lo que se hace con HTML a pesar de que son los elementos HTML los que van a acabar capturando las características gráficas de presentación que se establecen a través del lenguaje CSS. [1]

2. ¿ QUÉ ES CSS

Es un lenguaje que define las características gráficas y de presentación de los elementos definidos en HTML dentro de un documento web. CSS significa «hojas de estilo en cascada» (Cascading Style Sheets). mediante su uso vamos a poder darle características gráficas a cada elemento incluido dentro de un documento web, la importancia del uso de CSS uno a los elementos HTML podremos manipular los colores las dimensiones de los elementos podemos agregar tamaños de fuentes tipos de fuentes específicas y podemos construir de esta manera

a través del código qué significa estilos en cascada a través de esta invitando vamos a poder radica en que se está haciendo una separación entre lo que es la presentación visual y la estructura del documento que se está confeccionando. [3]

A. REGLAS CSS

La regla CSS se compone de la propiedad que vamos a configurar separando con: el conjunto de propiedades que se aplica a dicha propiedad, las propiedades son agregadas a los elementos que van a compartir la configuración. [1]

B. FORMAS DE DECLARAR CSS

Existen 3 formas para hacer declaración de reglas CSS, las cuales veremos a continuación:

a) EN LÍNEA

Se incluye dentro de la declaración del mismo elemento HTML que tenemos en el código como se muestra a continuación: [2]

```
<p style="color: blue">Hola Mundo!!</p>
```

Para hacer la especificación de las características de presentación se agrega el atributo style y a continuación se definen las características que afectarán a dicho elemento, las características definidas de esta manera prevalecen sobre las definiciones hechas en cualquier otro punto del documento web o un documento vinculado. [2]

b) EN LA CABECERA DEL DOCUMENTO WEB

Una segunda forma mucho más práctica y funcional es agregar las declaraciones de estilo en la cabecera del documento web dentro de la etiqueta style, de esta manera no estamos haciendo un formateo de cada elemento por separado y cuando se presenten modificaciones que afectan a todos los elementos del mismo tipo las actualizaciones serán rápidas y funcionales: [1]

```
<style>
  h1 {
    color: bisque;
  }
</style>
```

c) EN UN ARCHIVO VINCULADO

Una característica aún mejor de configuración de reglas CSS es en un archivo de configuración aparte el cual puede ser vinculado al documento web que estamos configurando, de esta manera se hace mucho más extensible las configuraciones ya que pueden ser aplicadas a muchos documentos diferentes solamente indicando la ruta de ubicación del archivo a través del comando de vinculación: [2]

```
/******
ARCHIVO      : main.CSS
AUTOR       : ANGEL MONTESINOS
FECHA       : 21/07/2022
DESCRIPCIÓN : Definición de un archivo CSS con reglas
******/
h1 {
  color: black;
  font-size: larger;
```

```
}  
h2 {  
    color: brown;  
    font-family: 'Times New Roman', Times, serif;  
}  
p {  
    color: #000;  
    font-weight: 100;  
    background-color: bisque;  
}
```

Las declaraciones que hemos hecho en el archivo anterior son aplicables a un documento web que especifique el enlace con dicho archivo esto se hace a través de la siguiente instrucción:

```
<!--ARCHIVO      : index.HTML      -->  
<!--AUTOR       : ANGEL MONTESINOS -->  
<!--FECHA       : 21/07/2022      -->  
<!--DESCRIPCIÓN : Archivo HTML ejemplo para llamar a una hoja de estilos CSS -->  
  
<!DOCTYPE HTML>  
<HTML lang="es">  
  
<head>  
    <meta charset="UTF-8">  
    <meta http-equiv="X-UA-Compatible" content="IE=edge">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>Documento</title>  
</head>  
  
<body>  
    <h1>Titulo de la pagina Web</h1>  
    <h2>Sub titulo de la pagina</h2>  
    <p style="color: blue">Hola Mundo!!</p>  
  
    <link rel="stylesheet" href="./main.CSS">  
</body>  
  
</HTML>
```

Siendo esta última la forma más adecuada para hacer las configuraciones con reglas CSS de los elementos de un documento web. [1]

C. SELECTORES

Nos permite identificar a los elementos HTML sobre los cuales se van a aplicar las reglas CSS. existen diversos tipos de selectores los cuales nos va a permitir hacer una gestión global y práctica de la configuración de los elementos HTML de una página web. Los electores se encuentran clasificados de la siguiente manera: [2]

a) SELECTORES SIMPLES:

Selecciona elementos a través de su nombre de etiqueta:

a.1) POR SU NOMBRE:

Sólo especifica el nombre de la etiqueta:

```
h1 {  
  color: black;  
  font-size: larger;  
}
```

b) POR SU CLASE:

Se agrega el nombre de una clase al elemento para poder seleccionarlo, el nombre de clase se puede compartir entre conjuntos de elementos; para especificar la clase se utiliza el carácter(.). [2]

```
<p style="color: blue">Hola Mundo!!</p>  
<p class="parrafo"> el texto que aparece </p>
```

Para especificar la regla asociada a la clase definida dentro del elemento HTML se utiliza como prefijo al carácter(.)

```
.parrafo {  
  color: crimson;  
  font-style: italic;  
}
```

c) POR SU ID:

Se agrega una id al elemento HTML para poder seleccionarlo, dicho nombre de id es único y no se puede volver a utilizar dentro de un documento HTML; para especificar el id se utiliza el carácter (#). [2]

```
<p class="parrafo"> el texto que aparece </p>  
<div id="division">  
  <p>Elemnto subordinado</p>  
</div>
```

La especificación de la regla CSS para el elemento al cual se le agregó id sería la siguiente:

```
#division {  
  color: blueviolet;  
  font-weight: 200;  
}
```

D. SELECTORES DE ATRIBUTO

Estos electores se utilizan para dar formato a los elementos de un documento HTML según los atributos que posee el elemento o inclusive según el valor de los atributos. [2]

```
<div>  
  <a href="http://www.ucsm.edu.pe"></a>  
  <br>  
  <a href="http://www.mininter.gob.pe"></a>  
  <br>  
  <a href="http://www.udemy.com"></a>  
  <br>  
</div>
```

Utilizaremos el código anterior para especificar diferentes tipos de configuración de acuerdo al tipo de atributo y al valor asignado al atributo.

```
[href] {  
    color: green;  
}  
[href="http://www.udemy.com"] {  
    color: red;  
}
```

E. PSEUDO CLASES

Es un selector en relación con el estado del elemento, por ejemplo, existen atributos que se refieren al estado del elemento como hover o focus:

Quitar algo joven vieron y se activa la célula se le aplica las reglas es hola indicado las activa cuando el estado cambia por ejemplo elementos. [2]

```
a:hover {  
    text-decoration: none;  
}  
input:focus {  
    background-color: blanchedalmond;  
}
```

También se pueden usar otros como nth-child para seleccionar a un elemento subordinado a otro:

```
ul li:nth-child(2) {  
    color: red;  
}
```

Con esta definición estamos dándole un formato al segundo elemento de la lista en la lista no ordenada. se puede utilizar este tipo de pseudo clases para diversas variantes de nth. [2]

F. PSEUDO ELEMENTOS

Su comportamiento se basa en seleccionar un elemento a partir de su situación en relación a otro elemento, se especifican con 2 veces 2 puntos (::): [2]

```
p::first-letter {  
    color: #000;  
    font-size: large;  
}
```

G. SELECTORES COMBINADOS Y MÚLTIPLES

Combinaciones: no son en sí selectores, por el contrario, son formas de combinar 2 o más selectores de manera práctica para una selección especial de un elemento. por ejemplo, para seleccionar elementos a continuación de títulos o elementos de división. [2]

Descendientes:

```
li a {  
    text-decoration: none;
```

```
color: red;
}
```

Adyacencia:

```
ul + p {
  color: red;
  font-family: 'Times New Roman', Times, serif;
}
```

Hijos directos:

```
div#container > ul {
  border: 1px solid black;
}
```

Hermanos:

```
h2 ~ p {
  margin: 10px 0;
}
```

Múltiples: Tampoco son selectores por sí mismos, es posible agrupar selectores en una misma regla CSS separándolos por comas para aplicarlos a una de las declaraciones o a todos de manera directa. [2]

```
/* El estilo se aplica a todos los elementos "div", "p", "span" y "ul" */
div, p, span, ul {
  text-decoration: underline;
}

/* El estilo se aplica a todos los elementos "ul" que se encuentran dentro
de "div p span"*/
div, p, span, em {
  text-decoration: underline;
}
```

H. SELECTORES REGEX

Son los que están basados en patrones de coincidencia. [1]

[class^="palabra"]: selecciona los atributos que coincidan en su inicio con palabra.

[class\$="palabra"]: selecciona los atributos que coincidan en su finalización con palabra.

[class*="palabra"]: selecciona los atributos que contengan como su cadena a palabra.

```
a {
  color: blue;
}

/* Enlaces internos, iniciando con "#" */
a[href^="#"] {
  background-color: gold;
}
```

```

/* Enlaces con "cAsE" en cualquier lugar del URL,
with matching capitalization */
a[href*="cAsE" s] {
    color: pink;
}

/* Enlaces que termina con ".org" */
a[href$=".org"] {
    color: red;
}

/* Enlaces que inician con "https" y terminan en ".org" */
a[href^="https"] [href$=".org"] {
    color: green;
}

```

I. HERENCIA

La herencia también debe entenderse en este contexto: algunos valores de propiedad CSS establecidos en los elementos principales son heredados por sus elementos secundarios y otros no. [2]

Por ejemplo, si establece un color en un elemento, todos los elementos dentro de él también tendrán un estilo con ese color y fuente, a menos que les haya aplicado valores de color y fuente diferentes directamente.

J. CASCADA

Determina la forma como el motor de CSS va a seleccionar los elementos según tres factores:

Orden de origen: Aplica la última reglas CSS especificada en el archivo CSS, si se encuentran al mismo nivel de especificación. Es importante únicamente cuando las reglas se encuentran al mismo nivel de especificidad. [2]

Especificidad: se refiere a cuán específica es la regla, mientras más lo sea más prioridad se tendrá sobre otras reglas. La cantidad de especificidad que tiene un selector se mide utilizando tres valores (o componentes) diferentes, que se pueden considerar como columnas ID, CLASE y ELEMENTO en el lugar de las centenas, decenas y unidades:

- Identificadores : marque uno en esta columna para cada selector de ID contenido dentro del selector general.
- Clases : puntúe uno en esta columna por cada selector de clase, selector de atributo o pseudoclase contenido dentro del selector general.
- Elementos : puntúe uno en esta columna por cada selector de elemento o pseudoelemento contenido dentro del selector general.

Selector	Identificadores	Clases	Elementos	Especificidad total
h1	0	0	1	0-0-1
h1 + p::first-letter	0	0	3	0-0-3
li > a[href*="en-US"] > .inline-warning	0	2	2	0-2-2
#identifier	1	0	0	1-0-0
button:not(#mainBtn, .cta)	1	0	1	1-0-1

Importancia: Hay una parte especial de CSS que puede usar para anular todos los cálculos

anteriores, incluso los estilos en línea: la bandera !important. Sin embargo, debe tener mucho cuidado al usarlo. Esta marca se usa para hacer que una propiedad individual y un valor par sean la regla más específica, anulando así las reglas normales de la cascada, incluidos los estilos en línea normales. [2]

K. UNIDADES DE MEDIDA

Absolutas: no cambian en relación con nada.

Unidad	Nombre	Equivale a
cm	Centímetros	1cm = 96px/2,54
mm	Milímetros	1mm = 1/10 de 1cm
Q	Cuartos de milímetros	1Q = 1/40 de 1cm
in	Pulgadas	1in = 2,54cm = 96px
pc	Picas	1pc = 1/6 de 1in
pt	Puntos	1pt = 1/72 de 1in
px	Píxeles	1px = 1/96 de 1in

Relativas: cambian en función a otro elemento.

Unidad	Relativa a
em	Tamaño de letra del elemento padre, en el caso de propiedades tipográficas como font-size, y tamaño de la fuente del propio elemento en el caso de otras propiedades, como width.
ex	Altura x de la fuente del elemento.
ch	La medida de avance (ancho) del glifo "0" de la letra del elemento.
rem	Tamaño de la letra del elemento raíz.
lh	Altura de la línea del elemento.
vw	1% del ancho de la ventana gráfica.
vh	1% de la altura de la ventana gráfica.
vmin	1% de la dimensión más pequeña de la ventana gráfica.
vmax	1% de la dimensión más grande de la ventana gráfica.

L. COLORES

Los colores se pueden configurar bajo diferentes esquemas, existen 165 nombres para colores que pueden ser utilizados para su configuración, otra forma es hacerlo a través de valores hexadecimales para lo cual tenemos 16.2 millones de colores disponibles, lo mismo sucede para las combinaciones de colores basadas en rgb= red, green, blue. [2]

Existe otra forma de hacer combinaciones de colores que se basa en la TONO(HUE), SATURACIÓN y LUMINOSIDAD(HSL), la cual tiene un cuarto valor para indicar transparencia.

Name	Hexadecimal	Modo RGB	Modo HSL
Black	#000000	rgb(0%,0%,0%)	hsl(0,0%,0%)
Blue	#0000FF	rgb(0%,0%,100%)	hsl(240,100%,50%)
Blue Violet	#8A2BE2	rgb(54%,17%,89%)	hsl(271,76%,53%)
Brown	#A52A2A	rgb(65%,16%,16%)	hsl(0,59%,41%)
Chocolate	#D2691E	rgb(82%,41%,12%)	hsl(25,75%,47%)
Coral	#FF7F50	rgb(100%,50%,31%)	hsl(16,100%,66%)
Crimson	#DC143C	rgb(86%,8%,24%)	hsl(348,83%,47%)
Cyan	#00FFFF	rgb(0%,100%,100%)	hsl(180,100%,50%)
Dark Blue	#00008B	rgb(0%,0%,55%)	hsl(240,100%,27%)
Gray	#808080	rgb(50%,50%,50%)	hsl(0,0%,50%)
Indigo	#4B0082	rgb(29%,0%,51%)	hsl(275,100%,26%)
Magenta	#FF00FF	rgb(100%,0%,100%)	hsl(300,100%,50%)
Orange	#FFA500	rgb(100%,65%,0%)	hsl(39,100%,50%)
Purple	#7F007F	rgb(50%,0%,50%)	hsl(300,100%,25%)
Red	#FF0000	rgb(100%,0%,0%)	hsl(0,100%,50%)
Silver	#C0C0C0	rgb(75%,75%,75%)	hsl(0,0%,75%)
Sky Blue	#87CEEB	rgb(53%,81%,92%)	hsl(197,71%,73%)
Violet	#EE82EE	rgb(93%,51%,93%)	hsl(300,76%,72%)
White	#FFFFFF	rgb(100%,100%,100%)	hsl(0,0%,100%)
Yellow	#FFFF00	rgb(100%,100%,0%)	hsl(60,100%,50%)

3. RESUMEN

La configuración del aspecto visual de un sitio web se hace por separado utilizando el lenguaje CSS, por razones de arquitectura y de ingeniería del software las especificaciones estructurales del sitio web, así como sus características visuales se mantienen por separado; esto nos permite una mayor productividad y una gestión mucho más profesional del producto.

la configuración de los elementos de un documento web se hace a través de reglas CSS hay 3 formas diferentes de agregar estas reglas dentro de un documento HTML; en línea que tiene la mayor prioridad y preponderancia, en la cabecera de la página web dentro de etiquetas style y y finalmente como un documento adjunto que es la forma más práctica y profesional de hacerlo debido a que se puede usar el mismo conjunto de definiciones para otros documentos web sin ningún problema. [2]

Los elementos escritos en HTML que están dentro de un documento web se referencian mediante Selectores, básicamente existen 3 tipos: de clase, por nombre y y por ID, existen otras formas mucho más elaboradas de hacer uso de los selectores como los selectores de

atributo que se refieren a características específicas del elemento; las pseudo clases que se encargan de seleccionar elementos según estados específicos del mismo; los pseudo elementos, se seleccionan en función a las características de otro elemento; los selectores combinados y múltiples; que no son en sí selectores, pero, permiten seleccionar elementos en función de las características de los elementos del entorno; los selectores regex, que se basan en una regla; la herencia y cascada, que se refiere a como las características gráficas serán aplicadas a los elementos subordinados; las unidades de medida que nos especifican la forma de establecer dimensiones dentro de los elementos de un documento web y finalmente los colores que determinan el aspecto cromático de los elementos incluidos dentro de la página web. Con todos estos recursos se puede hacer todo un conjunto bastante grande de configuraciones para determinar el aspecto gráfico de un documento web a la hora de presentarlo al usuario. [2]

IV

(La práctica tiene una duración de 2 horas) ACTIVIDADES

1. EXPERIENCIA DE PRÁCTICA N° 01: APLICAR REGLAS CSS

Conforme un grupo tal como lo indica el docente.

Crear una cuenta de GitHub donde creará y registrará todos sus proyectos:

a) APLICAR REGLAS CSS EN LÍNEA:

1. Abra el VS Code.
2. Crear una carpeta de proyecto.
3. Agregar un archivo HTML, escriba en él HTML:5 y presione enter. Aparece la estructura básica de un proyecto de página Web.
4. Escriba una página Web con todos los elementos y recursos semánticos y no semánticos necesarios para visualizar información sobre cualquier tipo de tema.
5. Estructure de forma adecuada la página web agregando los componentes y recursos necesarios para mantener una estructura uniforme, mantenible y adecuada para la presentación de información.
6. Crear 2 archivos HTML aparte y copiar el contenido de la página Web desarrollada a cada uno de ellos.
7. En la primera página web cambiar el título de la página a “en línea”.
8. Comience a seleccionar los elementos y aplicar diferentes tipos de reglas CSS a cada elemento en línea.
9. Luego de configurar completamente con las reglas CSS el documento web grábalo y cargue el documento en el navegador Web para ver los resultados.
10. Tomar nota de cualquier inconsistencia o mala visualización, cierre el navegador y a continuación corrija los errores e inconsistencias detectados
11. Vuelva a visualizar el documento web en el navegador.
12. Diseñe cambios sobre las reglas CSS utilizadas para configurar el aspecto visual de la página Web.
13. Haga una descripción de qué tan fácil fue hacer los cambios en los elementos especialmente cuando las mismas reglas se aplican al mismo tipo de elementos de forma separada.

b) APLICAR REGLAS CSS EN LA CABECERA DEL DOCUMENTO WEB

1. Abrir el segundo documento HTML creado.
2. Cambiar el título del documento a “CSS en cabecera”.
3. En la cabecera del documento abrir la etiqueta `<style> </style>`.
4. Copie todas las reglas CSS que introdujo en los elementos del documento web.
5. Revise y estructure las reglas que utilizó en el documento.
6. Grabe el archivo y visualícelo en el navegador web.
7. Cierre el navegador y vuelva a editar las reglas CSS que utilizó.
8. Haga una descripción de qué tan fácil fue hacer los cambios a las reglas CSS en comparación al método anterior de configuración del documento web.

c) APLICAR REGLAS CSS CON UN ARCHIVO VINCULADO

1. Abrir el tercer documento HTML creado.
2. Cambiar el título del documento a "CSS vinculado".
3. Crear un documento CSS con el nombre de main.
4. copiar todas las reglas CSS que fueron escritas en la cabecera del documento 2.
5. Vincular al archivo CSS con el documento web declarando el link en la cabecera del documento web.
6. Grabe el archivo y visualícelo en el navegador web.
7. Cierre el navegador y vuelva a editar las reglas CSS que utilizó.
8. Haga una descripción de qué tan fácil fue hacer los cambios a las reglas CSS en comparación a los dos métodos anteriores de configuración del documento web.

2. EXPERIENCIA DE PRÁCTICA N° 02: USO DE SELECTORES

a) USO DE SELECTORES POR SU NOMBRE

1. Diseñar un documento HTML nuevo, debe contener varias secciones para hacer la aplicación de reglas CSS.
2. Crear un nuevo documento para que le llamaremos nombre.CSS.
3. Agregue reglas que usen únicamente selectores de nombre para la configuración del documento web.
4. Grabar ambos documentos y visualizar la página web en el navegador.
5. Establecer un registro de la forma cómo aparecen los elementos seleccionados y si éstos cumplen con las especificaciones CSS establecidas.
6. Corregir los elementos que no aparecen de la forma deseada.
7. Grabar y visualizar en el navegador web, establezca las conclusiones de hacer uso de selectores por nombre.

b) USO DE SELECTORES POR SU CLASE

1. Hacer 2 copias del documento web anteriormente editado.
- 2.
3. Crear un nuevo documento para que le llamaremos clase.CSS.
4. Agregue reglas que usen únicamente selectores de clase para la configuración del documento web.
5. Grabar ambos documentos y visualizar la página web en el navegador.
6. Establecer un registro de la forma cómo aparecen los elementos seleccionados y si éstos cumplen con las especificaciones CSS establecidas.
7. Corregir los elementos que no aparecen de la forma deseada.
8. Grabar y visualizar en el navegador web, establezca las conclusiones de hacer uso de selectores de clase.

c) USO DE SELECTORES POR SU ID

1. En la segunda copia agregar ID a los elementos HTML, los cuales deben ser establecidos de acuerdo con el tipo de elemento y su uso.
2. Crear un nuevo documento para que le llamaremos nombre.CSS.
3. Agregue reglas que usen únicamente selectores con ID para la configuración del documento web.
4. Grabar ambos documentos y visualizar la página web en el navegador.
5. Establecer un registro de la forma cómo aparecen los elementos seleccionados y si éstos cumplen con las especificaciones CSS establecidas.
6. Corregir los elementos que no aparecen de la forma deseada.
7. Grabar y visualizar en el navegador web, establezca las conclusiones de hacer uso de selectores por nombre.

3. EXPERIENCIA DE PRÁCTICA N° 03: UTILIZACIÓN DE

SELECTORES DE ATRIBUTOS, PSEUDO CLASES Y PSEUDO ELEMENTOS

a) UTILIZACIÓN DE SELECTORES DE ATRIBUTOS

1. Crear un nuevo documento CSS con el nombre de atributos.CSS.
2. Establecer reglas CSS para elementos basados en sus atributos y valores de sus atributos.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. volver a cargar el documento web y luego guardar ambos archivos.

b) UTILIZACIÓN DE SELECTORES DE PSEUDO CLASES

1. Crear un nuevo documento CSS con el nombre de pseudoclases.CSS.
2. Establecer reglas CSS para elementos basados en sus pseudo clases, debe tener en cuenta que debe usar por lo menos 10 de todos los que admite el lenguaje.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. volver a cargar el documento web y luego guardar ambos archivos.
7. Hacer una diferenciación entre pseudo clases y atributos.

c) UTILIZACIÓN DE SELECTORES DE PSEUDO ELEMENTOS

1. Crear un nuevo documento CSS con el nombre de pseudoelementos.CSS.
2. Establecer reglas CSS para elementos basados en sus pseudo elementos, debe tener en cuenta que debe usar por lo menos 7 casos.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. volver a cargar el documento web y luego guardar ambos archivos.
7. Hacer una diferenciación entre pseudo clases y pseudo elementos.

4. EXPERIENCIA DE PRÁCTICA N° 04: UTILIZACIÓN DE SELECTORES COMBINADOS, MÚLTIPLES, REGEX HERENCIA Y CASCADA

a) UTILIZACIÓN DE FORMAS COMBINADAS Y MÚLTIPLES

1. Crear un nuevo documento CSS con el nombre de combinadosymultiples.CSS.
2. Establecer reglas CSS para elementos basados en combinaciones en cuenta que debe usar por lo menos 7 casos.
3. Establecer reglas CSS para elementos basados en casos múltiples en cuenta que debe usar por lo menos 5 casos.
4. Grabar el documento CSS.
5. Cargar el documento web en el navegador y visualizar los resultados.
6. Corregir los elementos que no tengan el aspecto visual deseado.
7. volver a cargar el documento web y luego guardar ambos archivos.

b) UTILIZACIÓN DE SELECTORES REGEX

1. Crear un nuevo documento CSS con el nombre de regex.CSS.
2. Establecer reglas CSS para elementos basados en reglas regex tener en cuenta que debe usar por lo menos 5 casos.
3. Grabar el documento CSS.

4. Cargar el documento web en el navegador y visualizar los resultados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. volver a cargar el documento web y luego guardar ambos archivos.

c) UTILIZACIÓN DE SELECTORES POR HERENCIA Y CASCADA

1. Crear un nuevo documento CSS con el nombre de hereciaycascada.CSS.
2. Establecer reglas CSS para elementos basados en herencia, tener en cuenta que debe usar por lo menos 7 casos.
3. Establecer reglas CSS para elementos basados en cascada, tener en cuenta que debe usar por lo menos 5 casos.
4. Grabar el documento CSS.
5. Cargar el documento web en el navegador y visualizar los resultados.
6. Corregir los elementos que no tengan el aspecto visual deseado.
7. volver a cargar el documento web y luego guardar ambos archivos.

5. EXPERIENCIA DE PRÁCTICA N° 05: UTILIZACIÓN DE MEDIDAS

a) UTILIZACIÓN DE MEDIDAS ABSOLUTAS

1. Crear un nuevo documento CSS con el nombre de medidasabsolutas.CSS.
2. Establecer reglas CSS para elementos basados en medidas absolutas aplicadas a los elementos, tener en cuenta que debe usar por lo menos 3 casos para cada tipo de medida.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados de medidas y haga comparaciones entre los elementos configurados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. Volver a cargar el documento web y luego guardar ambos archivos.

b) UTILIZACIÓN DE MEDIDAS RELATIVAS

1. Crear un nuevo documento CSS con el nombre de medidasrelativas.CSS.
2. Establecer reglas CSS para elementos basados en medidas relativas aplicadas a los elementos, tener en cuenta que debe usar por lo menos 3 casos para cada tipo de medida.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados de medidas y haga comparaciones entre los elementos configurados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. Volver a cargar el documento web y luego guardar ambos archivos.

6. EXPERIENCIA DE PRÁCTICA N° 06: UTILIZACIÓN DE COLORES

a) CONFIGURACIÓN DE COLORES A TRAVÉS DEL NOMBRE

1. Crear un nuevo documento CSS con el nombre de colorespornombre.CSS.
2. Establecer reglas CSS para elementos basados en colores usando su nombre, tener en cuenta que debe usar por lo menos 3 casos.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados de medidas y haga comparaciones entre los elementos configurados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. Volver a cargar el documento web y luego guardar ambos archivos.

b) CONFIGURACIÓN DE COLORES A TRAVÉS DE LA REGLA RGB

1. Crear un nuevo documento CSS con el nombre de coloresrgb.CSS.
2. Establecer reglas CSS para elementos basados en colores rgb, tener en cuenta que debe usar por lo menos 3 casos, probar con valor entero y porcentaje.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados de medidas y haga comparaciones entre los elementos configurados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. Volver a cargar el documento web y luego guardar ambos archivos.

c) CONFIGURACIÓN DE COLORES A TRAVÉS DE CÓDIGO HEXADECIMAL

1. Crear un nuevo documento CSS con el nombre de coloreshexadecimal.CSS.
2. Establecer reglas CSS para elementos basados en colores con código hexadecimal aplicadas a los elementos, tener en cuenta que debe usar por lo menos 3 casos.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados de medidas y haga comparaciones entre los elementos configurados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. Volver a cargar el documento web y luego guardar ambos archivos.

d) CONFIGURACIÓN DE COLORES A TRAVÉS DE REGLA H LS

1. Crear un nuevo documento CSS con el nombre de coloreshls.CSS.
2. Establecer reglas CSS para elementos basados en las reglas hls aplicadas a los elementos, tener en cuenta que debe usar por lo menos 3 casos.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados de medidas y haga comparaciones entre los elementos configurados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. Volver a cargar el documento web y luego guardar ambos archivos.

e) CONFIGURACIÓN DE COLORES CON OPACIDAD

1. Crear un nuevo documento CSS con el nombre de medidasrelativas.CSS.
2. Establecer reglas CSS para elementos basados en medidas relativas aplicadas a los elementos, tener en cuenta que debe usar por lo menos 3 casos para cada tipo de medida.
3. Grabar el documento CSS.
4. Cargar el documento web en el navegador y visualizar los resultados de medidas y haga comparaciones entre los elementos configurados.
5. Corregir los elementos que no tengan el aspecto visual deseado.
6. Volver a cargar el documento web y luego guardar ambos archivos.

V

EJERCICIOS PROPUESTOS

1. Hacer la maquetación de una página web, se debe planificar la inclusión de cada uno de los tipos de selectores para reglas CSS que sean tratado en la presente práctica, separar el sitio web en por lo menos 3 documentos o páginas individuales.
2. Especificar claramente y documentar la forma de aplicación de los selectores para elementos HTML desde CSS.
3. Agregar lista tablas y formularios a su sitio web y adicionalmente validar y verificar las reglas CSS para color y medida..

CUESTIONARIO

1. ¿Qué es CSS?
2. ¿Por qué separamos HTML y CSS?
3. ¿Cómo se codifican páginas web con CSS?
4. ¿Cómo se ejecutan páginas web codificadas con CSS?
5. ¿Qué son los selectores?
6. ¿Cuántas formas de especificar reglas CSS existen para un documento web?
7. ¿Cómo se especifica una regla CSS en línea?
8. ¿Cómo se especifica una regla CSS en las etiquetas style?
9. ¿Cómo se especifica una regla CSS en un documento CSS vinculado?
10. ¿Cuáles son los principales tipos de selectores CSS?
11. ¿Qué son pseudo clases y cómo se utilizan?
12. ¿Qué son pseudo elementos u cómo se utilizan?
13. ¿Qué son los selectores combinados y múltiples y cómo se utilizan?
14. ¿Qué son los selectores de atributo y los selectores regex?
15. ¿Cómo se maneja la herencia dentro de las reglas CSS?
16. ¿Cómo se maneja la cascada dentro de las reglas CSS?
17. ¿Qué son las unidades de medida absolutas y cómo se utilizan dentro de las reglas CSS?
18. ¿Qué son las unidades de medida relativas y cómo se utilizan dentro de las reglas CSS?
19. ¿Qué es el color dentro de las reglas CSS?
20. ¿Cuántas formas de especificar el color en una regla CSS existen?

BIBLIOGRAFIA Y REFERENCIAS

- [1] «CSS Tutorial,» [En línea]. Available: <https://www.w3schools.com/css/>.
- [2] Mozilla, «MDN,» 24 06 2022. [En línea]. Available: <https://developer.mozilla.org/en-US/docs/Learn/HTML>.
- [3] M. Rubiales Gomez, Curso de Desarrollo Web HTML, CSS Y JavaScript, Madrid: ANAYA MULTIMEDIA, 2018.