

Desarrollo de aplicaciones web full stack

Hojas de estilo CSS

Hojas de estilo CSS

Para empezar a ver las reglas de CSS y como se aplican; primero se aborda que cada navegador garantiza unos estilos mínimos, por defecto, para cada elemento HTML plasmado en la página.

Estos estilos por defecto no son los que el cliente necesita, ni muestran la página como se quiere ver; por eso, los estilos se deben modificar y personalizar según las necesidades y requerimientos del cliente y esto lo hace el desarrollador de la página, a través de la aplicación de CSS.

De acuerdo con lo anterior, el navegador tiene una organización predeterminada de los elementos HTML en su forma de interpretarlos y esta es por bloques o por líneas.

Por bloques son la configuración por defecto del navegador y esto indica que cada etiqueta o elemento HTML lo ordena como una caja que ocupa todo el ancho de la página y, por ende, los pone uno debajo del otro.

Figura 1

Representación por defecto del navegador



La otra forma de organizar es por líneas y los organizaría de forma horizontal uno al lado del otro, hasta llenar el ancho de la página.

Para iniciar a personalizar los estilos por defecto del navegador, se van a ver los Modelos de caja; esto quiere decir que un navegador es un grupo de cajas ordenadas, organizadas, siguiendo las reglas CSS. El modelo por trabajar es el de cajas, pero antes de empezar se verán los códigos CSS y las formas de aplicar las reglas.

Siempre se trabajarán los estilos desde un archivo externo y por eso se deben ver las formas de referencias, los estilos o reglas, desde ese archivo a la página web que lo implemente.

Existen tres formas de hacer referencia a los elementos HTML que se les quiere aplicar las reglas y son:

- Referencia por la palabra clave del elemento.
- Referencia por el atributo id.
- Referencia por el atributo class.

Referencia por clave

Esta referencia indica que las reglas de CSS se aplican utilizando la palabra de cada etiqueta; por ejemplo, body para aplicarle estilos a la etiqueta body y así con todas las etiquetas.

```
P {
    Color: red;
    Font-size: 20px;
}
```

Como se puede ver en el código anterior, se está aplicando un estilo por clave, es decir, por el nombre de la etiqueta y se aplicará a todas las etiquetas que existan en la página.

En este caso, se le está aplicando un estilo que le cambia el color y el tamaño de la letra, al texto que está dentro de **p**.

Lo anterior funciona, pero qué pasa si no se necesita aplicar el estilo a todas las etiquetas **p** sino solamente a una o algunas, en específico. Para eso es necesario usar la referencia por el atributo **id** o **class**.

Referencia por id

El id es una forma única de etiquetar un elemento HTML, es decir, que tendrá un nombre único que no se puede repetir en toda la página.

CSS para referenciar un elemento por su id se hace a través del símbolo # y frente al nombre con el que se va a identificar.

Figura 2

Referencia id

```
<!DOCTYPE html>
<html>
<head>
  <title>Mi página</title>
  <link rel="stylesheet" type="text/css" href="CSS/estilos.css">
</head>
<body>

  <p id="parrafo1">
    La Revolución 4.0 trae consigo nuevas herramientas y tecnologías
    como son Realidad Aumentada, Realidad Virtual, Internet de las cosas,
    etc.
  </p>
</body>
</html>
```

En la imagen anterior, se observa cómo se agrega el atributo **id** a la etiqueta **p**; ahora, con ese **id** se hace referencia desde el archivo externo CSS para aplicar las reglas necesarias.

Figura 3

Estilo CSS por id

```
#parrafo1{
    font-size: 15px;
    color: blue;
}
```

La aplicación de las reglas CSS con **id** indican que solo a la etiqueta que tenga ese **id** llamado **parrafo1** se le aplican las reglas de letra tamaño 15px y de color azul.

La otra forma de referenciar un estilo CSS es a través del atributo class.

Referencia por class

En las páginas, es muy inusual necesitar aplicar estilos específicos a cada etiqueta para usar siempre el id, mientras que es más común requerir aplicar unas reglas a un conjunto de elementos en particular; en ese sentido, a esos elementos se les agrega el atributo class y se les asigna el mismo nombre, es decir, serían elementos de la misma clase y cuando se apliquen reglas a esa clase, se aplica a todos los elementos que la componen.

Figura 4

Atributo class

```
<!-- Fin página -->
<link rel="stylesheet" type="text/css" href="CSS/estilos.css">
</head>
<body>

  <p class="texto1">
    La Revolución 4.0 trae consigo nuevas herramientas y tecnologías
    como son Realidad Aumentada, Realidad Virtual, Internet de las cosas,
    etc.
  </p>
  <p class="texto1">
    Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
    tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
    veniam,
    quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
    consequat. Duis aute irure dolor in reprehenderit in voluptate velit
    esse
    cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat
    cupidatat non
    proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
  </p>
</body>
```

Como se puede ver en la imagen anterior, se decía en el párrafo, el atributo class es un nombre que identifica un grupo de elementos que pertenecen a la misma clase y, por ende, tendrán el mismo estilo.

A continuación, se observa el código de reglas CSS que se aplica a la clase texto1 y, por ende, cambiaría el texto escrito en las dos etiquetas p que pertenecen a esa misma clase.

Figura 5*Estilos por atributo class*

```
.parrafo1{
  font-size: 15px;
  color: blue;
}
```

Aparte de los selectores anteriores, CSS 3 agregó nuevos selectores que determinan espacialmente ciertas reglas o formas de aplicar las reglas; en estos casos los selectores son $>$, $+$ y \sim .

Estos selectores se basan en la relación entre elementos y así seleccionan cuales van a ser impactados por las reglas CSS codificadas.

```
div > p.parrafo1{
  color: blue;
}
```

El selector $>$ aplicado en el código anterior indica que el elemento a ser afectado es el de la derecha del símbolo, pero siempre y cuando su elemento padre sea el de la izquierda. En otras palabras, se afectará al elemento `p` con el atributo `class`, llamado `parrafo1`, pero siempre y cuando sea hijo de un elemento `div`.

El otro selector es el $+$ y referencia que el elemento de la derecha será el impactado por las reglas de estilos siempre y cuando sea el inmediatamente precedido por el elemento de la izquierda.

```
p.parrafo2 + p {
  font-size: 14px;
}
```

Según el párrafo anterior, el código expresado indica que se impactará un elemento `p` siempre que sea el inmediatamente precedido por un elemento `p` con el atributo `class` llamado `parrafo2`.

Después de ver los conceptos básicos de cómo aplicar los estilos CSS a las páginas a través de los diferentes selectores, se va a proceder a crear una plantilla HTML básica y hacer una maquetación con estilos CSS externos.

Para esta aplicación se revisará lo que se llama modelo de caja tradicional, el cual ha sido implementado desde la primera versión de CSS y es soportado por todos los navegadores del mercado.

Plantilla

La plantilla que se va a crear tendrá todos los elementos HTML que componen la estructura y permitirá conocer en un paso a paso, como se hace un proceso de maquetación siguiendo el modelo de caja tradicional.

Como se seguirá el modelo de caja tradicional, este modelo necesita agrupar todas las cajas en una, para poder centrarla y darle un tamaño específico; por eso se agregará una etiqueta <div> que agrupará todos los elementos de la página.

Con base en lo anterior, la plantilla queda de la siguiente manera:

Figura 6

Plantilla HTML

```

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="utf-8">
  <meta name="description" content="Plantilla HTML5">
  <meta name="keywords" content="HTML5, CSS3, JavaScript">
  <title>Mi primera maquetación</title>
  <link rel="stylesheet" type="text/css" href="CSS/estilos.css">
</head>
<body>
<div id="contenedor">
  <header id="cabecera">
    <h1>Desarrollo web con HTML5</h1>
  </header>
  <nav id="menu">
    <ul>
      <li>Inicio</li>
      <li>Servicios</li>
      <li>Proyectos</li>
      <li>Contacto</li>
    </ul>
  </nav>
  <section id="seccion">
    <article>
      <header>
        <hgroup>
          <h1>Revolución tecnológica 4.0</h1>
          <h2>Internet de las cosas</h2>
        </hgroup>
        <time datetimes="2020-09-15" pubdate>publicado
          15-09-2020</time>
      </header>
      <p>Este parrafo es de prueba para el articulo de internet de
        las cosas, por eso se tomara un parrafo ejemplo.
        Lorem ipsum dolor sit amet, consectetur adipisicing elit,
        sed do eiusmod
        tempor incididunt ut labore et dolore magna aliqua. Ut enim
        ad minim veniam,
        quis nostrud exercitation ullamco laboris nisi ut aliquip ex
        ea commodo
        consequat. Duis aute irure dolor in reprehenderit in
        voluptate velit esse
        cillum dolore eu fugiat nulla pariatur. Excepteur sint
        occaecat cupidatat non
        proident, sunt in culpa qui officia deserunt mollit anim id
        est laborum.
      </p>
      <figure>
        
      </figure>
      <figure>
        
        <figcaption>
          Imagen de IOT
        </figcaption>
      </figure>
    </article>
    <footer>
      <p>comentarios (0)</p>
    </footer>
  </div>

```

```

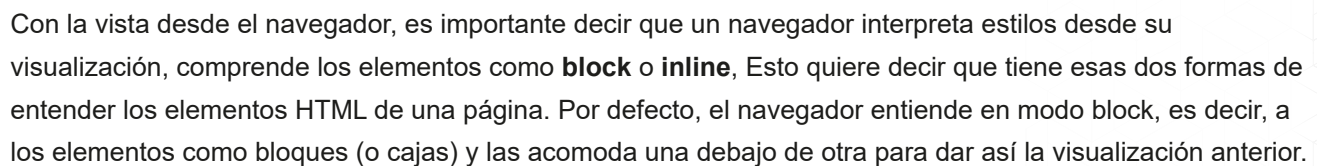
<article>
  <header>
    <hgroup>
      <h1>Revolución tecnológica 4.0</h1>
      <h2>Ejemplo de otro articulo</h2>
    </hgroup>
    <time datetime="2020-09-15" pubdate>publicado
    10-09-2020
    </time>
  </header>
  Este parrafo es de prueba para el articulo de internet de
  las cosas, por eso se tomara un parrafo ejemplo.
  Lorem ipsum dolor sit amet, consectetur adipisicing elit,
  sed do eiusmod
  tempor incididunt ut labore et dolore magna aliqua. Ut enim
  ad minim veniam,
  quis nostrud exercitation ullamco laboris nisi ut aliquip ex
  ea commodo
  consequat. Duis aute irure dolor in reprehenderit in
  voluptate velit esse
  consequat. Duis aute irure dolor in reprehenderit in
  voluptate velit esse
  cillum dolore eu fugiat nulla pariatur. Excepteur sint
  occaecat cupidatat non
  proident, sunt in culpa qui officia deserunt mollit anim id
  est laborum.
  <figure>
    
    <figcaption>
      Imagen de IOT para el otro articulo
    </figcaption>
  </figure>
  <footer>
    <p>comentarios (0)</p>
  </footer>
</article>
</section>
</section>
<aside id="columna">
  <blockquote>Texto Mensaje uno</blockquote>
  <blockquote>Texto Mensaje dos</blockquote>
</aside>
<footer id="pie">
  Derechos Reservados &copy; Jose Aguilar - 2020
</footer>
</div>
</body>
</html>

```

Como se puede observar, la plantilla de la imagen 6 contiene todos los elementos de HTML5 (section, header, aside, article, footer) que permiten darle un esqueleto y sentido a la página, para el entendimiento del navegador.

Así como está la plantilla, si se revisa en un navegador, se ve de la siguiente manera:

Desarrollo web con HTML5



The diagram illustrates the structure of an HTML document using five distinct sections, each represented by a colored rectangular block. The sections are arranged vertically and separated by thin white lines. The sections are: a light blue block for the header, a light orange block for the navigation bar, a light green block for the main content section, a light purple block for the sidebar/aside, and a light red block for the footer. Each block contains its corresponding HTML tags in a monospaced font.

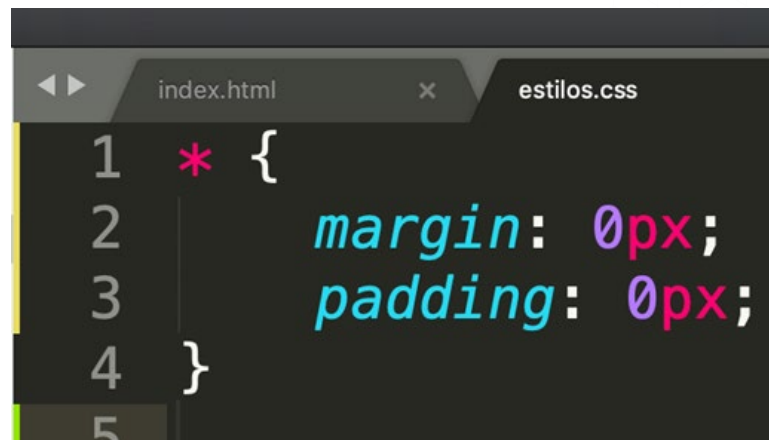
Section	HTML Tags
Header	<code><header></code> <code></header></code>
Navigation Bar	<code><nav></code> <code></nav></code>
Main Content Section	<code><section></code> <code></section></code>
Sidebar/Aside	<code><aside></code> <code></aside></code>
Footer	<code><footer></code> <code></footer></code>

Se va ir aplicando y mostrando por elementos, es decir, primero el encabezado de la página, después el menú y así sucesivamente.

Para iniciar, se van a ajustar o resetear todos los elementos de la plantilla con un selector universal que es el asterisco (*). Los elementos traen un margen y un padding por defecto, los cuales se van a poner en cero para que no obtengan ningún valor y poder después manipularlos libremente.

Figura 9

*Selector universal,
aplicando a todos los elementos*



```

1  * {
2      margin: 0px;
3      padding: 0px;
4  }
5

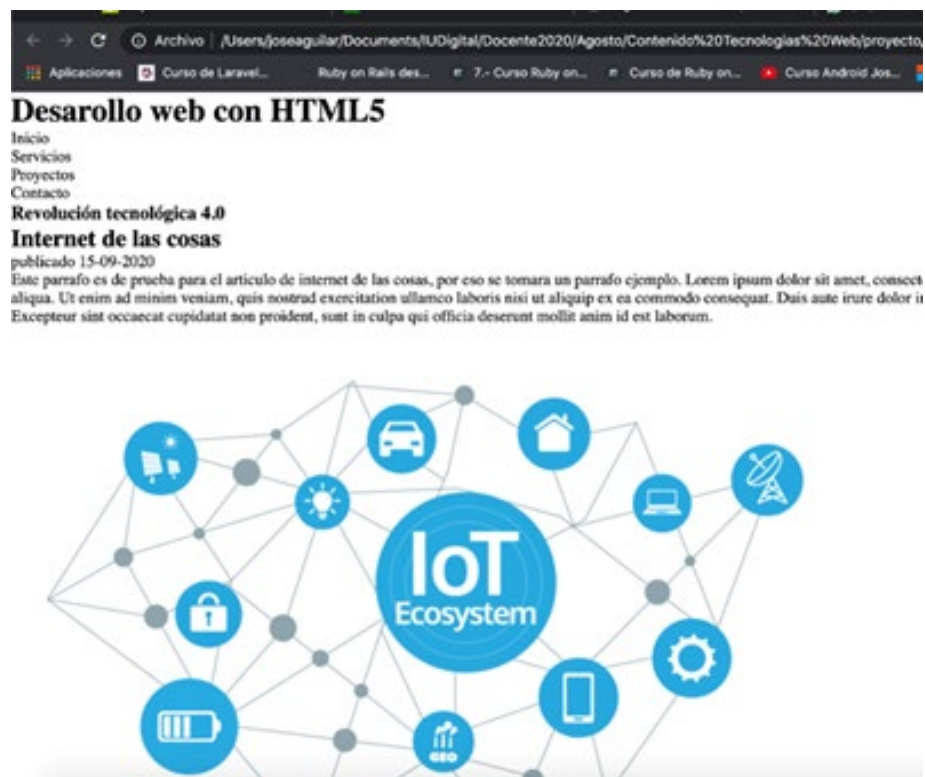
```

Con el código anterior, se ponen los márgenes y el padding de todos los elementos en cero, para poder personalizarlos de acuerdo con las necesidades de la página.

Con esta regla en el navegador, se observa de la siguiente manera.

Figura 10

*Plantilla con márgenes
y padding en cero*



La imagen anterior muestra que, en el navegador, los elementos ya no tienen ninguna margen y se pegan totalmente a los bordes de la página; de esta manera ya están listos para aplicarles las reglas necesarias sin que sus valores que tenían por defecto, influyan alterando el resultado.

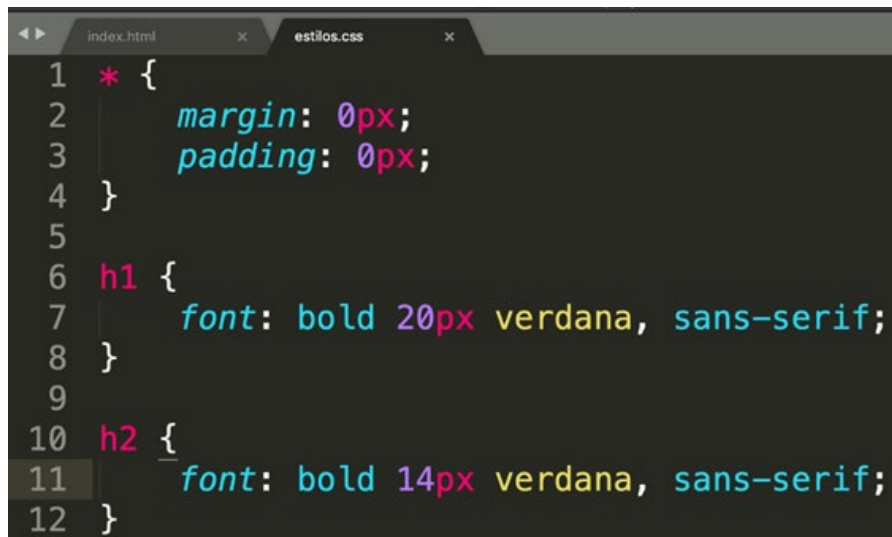
Con el margin y el padding en cero, se asegura que todos los elementos tengan sus márgenes interno y externo en cero pixeles.

Para continuar con las reglas se van a personalizar, los títulos y subtítulos que se utilizan en la página, es decir, se va a cambiar el tamaño, tipo de letra y la fuerza o el grueso de la letra.

Para que estas reglas se apliquen a todos los títulos y subtítulos de **h1** y **h2** se debe utilizar la aplicación de estilo por referencia, es decir, referenciando directamente la etiqueta.

Figura 11

Estilos en etiquetas h1 y h2



```
1 * {
2     margin: 0px;
3     padding: 0px;
4 }
5
6 h1 {
7     font: bold 20px verdana, sans-serif;
8 }
9
10 h2 {
11     font: bold 14px verdana, sans-serif;
12 }
```

La imagen anterior muestra la propiedad font, asignada a los elementos <h1> y <h2> la cual permite aplicar todos los estilos para los títulos y subtítulos en una sola línea.

Las propiedades que se aplican usando la etiqueta font son: el primer valor es el de font-style donde se está diciendo que es estilo bold (negrita), el siguiente valor es font-size y se asigna el tamaño de la fuente, para terminar con la font-family que sería la familia o tipo de fuente que se utilizará para la letra.

El paso siguiente es darles una regla básica a los elementos nuevos de HTML5, porque algunos navegadores aún no los identifican y los tratan como inline. Estos elementos se necesitan de tipo block, para poder aplicar el estilo de cajas tradicional.

Figura 12

*Estilo block a
elementos HTML5*

```
header, section, footer, aside, nav, article, figure, figcaption,
hgroup{
  display: block;
}
```

La imagen anterior muestra cómo se aplica una regla display; esta regla indica cómo entiende el navegador los elementos y el tipo block es colocar como si fueran cajas cada elemento y los pondría uno debajo del otro.

Continuando con el modelo de caja tradicional, el primer elemento a personalizar o impactar con reglas CSS es el body. A esta etiqueta se le debe aplicar un tamaño máximo para que se acomode a las diferentes pantallas y se debe posicionar horizontalmente.

Figura 13

Centrando el contenido del cuerpo

```
body {
  text-align: center;
}
```

Figura 14

*Página con el texto
del body centrado*



La imagen anterior muestra cómo el navegador comienza a visualizar el contenido, acorde con los estilos que se van aplicando.

Siguiendo con la maquetación, ahora se le debe dar un tamaño máximo al cuerpo de la página y esto se hace a través del elemento div que contiene todos los elementos del cuerpo de la página.

Este div será la caja principal del modelo de caja que se está construyendo, de esta forma se modificará el tamaño y alineación del elemento.

Figura 15*Estilo al div contenedor*

```
body {
    text-align: center;
}

#contenedor {
    width: 960px;
    margin: 15px auto;
    text-align: left;
}
```

Como se observa en el código anterior, se está haciendo referencia por el id del elemento que, en este caso, se llama contenedor.

Al contenedor que es el div que encierra todos los elementos de la página, se le aplican reglas de tamaño de ancho, alineación de contenido o texto a la izquierda, y el margin indica que da 15px al margen superior e inferior del div y se declaran como automáticos los márgenes de izquierda y derecha.

A continuación, se muestra cómo se ven estos estilos aplicados en el navegador.

Figura 16*Estilo al div contenedor*

En la imagen anterior, se observa cómo el contenedor ajusta el tamaño del contenido a un ancho específico y queda centrado en la página porque el body tiene el estilo text-align que centra su contenido.

Continuando con la maquetación, se debe seguir con la etiqueta <header>; recuerde que a esta etiqueta se le asignó un id llamado cabecera.

Como se mencionó al inicio cada elemento de HTML5 es interpretado como block y también el CSS inicial asegura ese bosquejo; un elemento block es una caja que, verticalmente, ocupa el tamaño según su contenido y, horizontalmente, ocupa el 100 % de la página o de su elemento padre.

Teniendo esto presente, se aplican los siguientes estilos a la cabecera.

Figura 17*Estilos de la cabecera*

```
#contenedor {
width: 960px;
margin: 15px auto;
text-align: left;
}

#cabecera {
background: #FFFB9;
border: 1px solid #999999;
padding: 20px;
}
```

El código que se aplica para personalizar la cabecera, indica un código hexadecimal que asigna un color amarillo a través de la regla background; también en la regla border, se crea un borde con una línea sólida de 1px de grosor y con color gris, como indica el código hexadecimal.

Los colores hexadecimales los puede revisar y tomar de la siguiente página:

<https://htmlcolorcodes.com/es/>

El padding aplica un margen interno de 20px en el elemento que contiene la cabecera. Lo anterior se visualiza en el navegador de la siguiente manera.

Figura 18*Cabecera personalizada con CSS*

El navegador muestra la cabecera ajustada con los colores y el borde aplicados en los códigos CSS.

Ahora se va ajustar la barra de navegación que será el menú de la página, para ponerle un estilo horizontal y adecuado, según las necesidades.

Figura 19

Estilos para la barra de navegación

```
#menu {
  background: #CCCCCC;
  padding: 5px 15px;
}

#menu li {
  display: inline-block;
  list-style: none;
  padding: 5px;
  font: bold 14px verdana, sans-serif;
}
```

El código anterior se aplica al id llamado menú, que sería el que se puso a la etiqueta HTML <nav>; se da un estilo de color de fondo gris y unos márgenes internos de 5px y 15px.

La propiedad padding funciona lo mismo que margin, tiene cuatro valores los cuales son: superior, derecho, inferior, izquierdo, en este orden. Si se declara solo un valor, este será asignado para todos los espacios alrededor del contenido del elemento. En cambio, si se especifican dos valores, el primero será asignado como margen interno de la parte superior e inferior del contenido y el segundo valor, será asignado al margen interno de los lados, izquierdo y derecho.

El siguiente código que tiene la referencia #menu li eso indica que se van aplicar los estilos al elemento li que contenga como elemento padre uno con id llamado menú.

La etiqueta li en la plantilla diseñada, es la que tiene los ítems del menú y por eso se le aplican las propiedades de display y list-style. Display indica cómo se van a visualizar los elementos y será en inline-block diciendo que serán bloques, pero en línea horizontal.

La propiedad list-style tiene un valor de none; esto indica que no tenga ningún estilo de lista y así se quitan las viñetas. El padding y font son propiedades que ya se han aplicado y se conoce su funcionamiento.

Figura 20

Visualización barra de navegación



Como se observa, el menú ya tiene forma y se observa como las barras de navegación encontradas en las diferentes páginas. Una vez configurado el menú a través de estilos, se procede con los siguientes elementos en la estructura que son el section y aside.

Siguiendo con el modelo de caja tradicional al section y al aside, se le aplica la propiedad float donde el bloque se vuelve flotante dentro del espacio de la página donde esta y se puede ubicar a la izquierda o derecha, según las necesidades del diseño.

Figura 21

Propiedades aplicadas a etiquetas section y aside

```
#seccion {
  float: left;
  width: 660px;
  margin: 20px;
}

#columna {
  float: left;
  width: 220px;
  margin: 20px 0px;
  padding: 20px;
  background: #CCCCCC;
}
```

El Código muestra cómo al id sección que corresponde a la etiqueta section, se aplica el float para que ese bloque flote a la izquierda del espacio de pantalla, en donde está y ocupando un ancho (width) de 660px con márgenes de 20px en todos sus lados.

Al ocupar solo 660px queda el resto de campo para que floten los demás bloques necesarios; en este caso, el siguiente bloque es el aside que tiene el id llamado columna. Esta etiqueta es personalizada con un float de valor izquierdo, un ancho de 220px con unos márgenes internos y externos específicos, así como un color de fondo gris.

Figura 22

Estilos en section y aside



En el navegador se puede observar cómo el section se sitúa en la parte izquierda de la pantalla y como es flotante, permite acoplar la caja del aside también flotante, del lado izquierdo del espacio que queda.

Estas secciones tienen estilo de ancho definido, fuente de letra y el aside un color de fondo.

Para terminar la estructura principal, se modifica el elemento estructural faltante y es el pie de página (footer).

Al footer se le aplican los siguientes estilos.

Figura 23

Estilos del footer

```
#columna {
  float: left;
  width: 220px;
  margin: 20px 0px;
  padding: 20px;
  background: #CCCCCC;
}

#pie {
  clear: both;
  text-align: center;
  padding: 20px;
  border-top: 2px solid #999999;
}
```

La propiedad clear restaura el flujo normal del documento, es decir, prohíbe que el elemento en cuestión, se posicione al lado izquierdo o derecho de alguna caja flotante.

El valor both significa que ambos lados del elemento se restaurarán al flujo normal de bloques, uno debajo de otro.

Clear tiene también como función, empujar los bloques verticalmente, haciendo que los bloques flotantes ocupen un área real en la pantalla. Sin esta propiedad, el navegador presenta el documento en pantalla como si los elementos flotantes no existieran y las cajas se superponen.

Figura 24

Estilos del footer



Imagen de IoT para el otro artículo
comentarios (0)

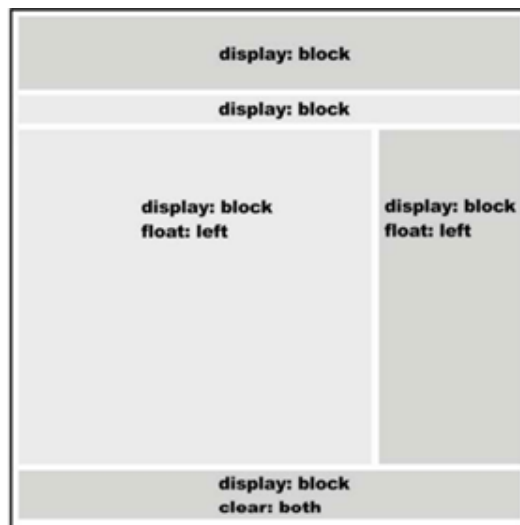
Derechos Reservados © Jose Aguilar - 2020

El navegador muestra cómo queda el footer debajo de los elementos section y aside, maquetando la forma de la página, para terminar la estructura base.

A continuación, se muestra cómo es el modelo tradicional de cajas, que acabamos de aplicar.

Figura 25

Representación del modelo de cajas tradicional



Ahora, para terminar la maquetación, se va a dar estilos a los artículos o los elementos article y las imágenes.

Primero, se personalizarán las imágenes en su tamaño y su título para que se ajusten a lo necesitado a través del siguiente código.

Figura 26

Estilos en las imágenes de la página

```
img{
    width: 70%;
    height: 70%;
}

figcaption {
    font: italic 14px verdana, sans-serif;
}
```

Con esas propiedades aplicadas, se le está diciendo a la imagen que su tamaño en ancho y alto, será el 70 % del original y acorde con el tamaño de la página y el espacio en donde esté contenida.

Las otras propiedades cambian la fuente del título de la imagen.

Ahora, se termina de maquetar esta plantilla dándole un estilo a los artículos.

Figura 27*Estilos etiqueta <article>*

```

article {
  background: #FFFBCC;
  border: 1px solid #999999;
  padding: 20px;
  margin-bottom: 15px;
}

article footer {
  text-align: right;
}

time {
  color: #999999;
}

```

Los estilos que aparecen en la imagen anterior, ya son conocidos y son los que se han estado utilizando a lo largo de esta maquetación, color de fondo, borde, márgenes internos y externos, alineación del pie de página a la derecha y color gris del texto de publicación del artículo.

Con estos estilos, se termina la maquetación y la página se ve de la siguiente manera.

Figura 28*Visualización plantilla maquetada*

Así, con este modelo de cajas tradicional, se puede hacer cualquier organización o diseño que se desee; en este caso, se hizo este ejemplo como práctica dentro del proceso de adquisición del conocimiento.