



El futuro digital  
es de todos

MinTIC

```
te( "name" );  
"type" );
```

```
if ( type == "sprite" )  
  
std::string item_name = item->Attribute( "name" );  
std::string spritename = item->Attribute( "spritename" );  
float x = boost::lexical_cast<float>( item->Attribute( "x" ) );  
float y = boost::lexical_cast<float>( item->Attribute( "y" ) );  
float offset = boost::lexical_cast<float>( item->Attribute( "offset" ) );  
  
SpriteDescList::iterator sp = sprite_descs.begin();  
for( ; sp != sprite_descs.end(); ++sp )  
    if ( sp->name_ == spritename )  
        break;
```

## Ciclo 3:

Desarrollo de Software



**Misión  
TIC2022**

VERSIÓN 1.0

Unidad de educación  
continua y permanente  
Facultad de Ingeniería



Unidad Camilo Torres  
Calle 44 # 45-67  
Bloque 85 piso 1



(57) + 314 5000  
uec\_ibog@unaleduco

# Componente de Presentación

## (HTML y CSS)

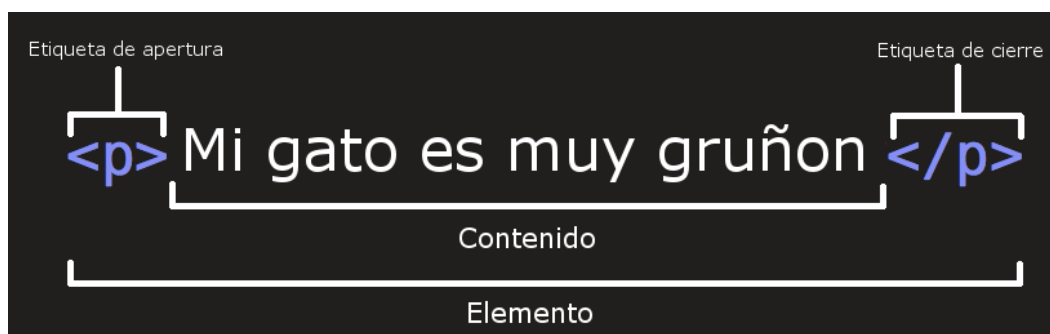
### Actividad Práctica

Para la construcción de la capa de presentación del caso de estudio, es necesario conocer los elementos que se utilizan para la construcción de páginas web. De esta forma, en esta sesión se explicarán dos lenguajes: **HTML** y **CSS**. HTML es un lenguaje de marcado, y CSS es un lenguaje de hojas de estilo. A continuación, se mostrará cómo funciona cada uno, y las diferencias entre ambos lenguajes.

### Explicación y uso de HTML

#### Elemento HTML

En general, un *documento HTML* consiste en una serie de *elementos HTML* que se utilizan para seccionar el contenido de la página web, y para que este se vea o comporte de cierta manera. La estructura básica de un *elemento HTML* es la siguiente:



La gran mayoría de *elementos HTML* consta de una etiqueta de apertura, de un contenido, y de una etiqueta de cierre. Como se puede apreciar en la imagen, la sintaxis de una etiqueta de apertura es el nombre del elemento (En este caso, *p*), encerrado por paréntesis angulares (*< >*), mientras que la sintaxis de una etiqueta de cierre es la misma, pero incluye una barra de cierre (*/*) antes del nombre de la etiqueta.

En HTML existen múltiples etiquetas, cada una con su propia función y utilización. En la imagen, se utiliza la etiqueta *<p>*, esta indica al navegador web que el contenido del *elemento HTML* es un párrafo, y que se debe presentar como tal. Así, la etiqueta de un *elemento HTML* se utiliza para indicar al navegador web el tipo de *elemento HTML* y, por ende, cómo se debe presentar su contenido en la página web.

#### Atributos de un elemento HTML

En algunas ocasiones, es necesario añadir información adicional a un elemento sin cambiar su contenido, que

sirva para definir su comportamiento. Para esto, es posible asignar atributos a un elemento HTML en su etiqueta de apertura, la sintaxis para ello se muestra en la siguiente imagen:

Atributo

```
<p class="editor-note">Mi gato es muy gruñon</p>
```

En la imagen, `class` es el nombre del atributo y `"editor-note"` es su valor. En general, un atributo siempre debe tener: al menos un espacio de separación con la etiqueta o con los demás atributos, el nombre del atributo seguido de un signo igual (=), y el valor del atributo encerrado entre comillas dobles (" ").

En este caso, el atributo `class` se utiliza para asignar `"editor-note"` como la clase del elemento. Se profundizará la utilidad de este atributo más adelante, con la explicación de CSS.

## Elementos Anidados

En HTML es posible ubicar elementos dentro de otros elementos, esto es útil para organizar un documento HTML, o para presentar el contenido de cierta forma. Por ejemplo, para mostrar el texto *"Mi gato es muy gruñón"*, se utilizaría la etiqueta:

```
<p>Mi gato es <strong>muy</strong> gruñon</p>
```

Como se puede suponer, la etiqueta `<strong>` muestra el contenido del elemento en negrilla. Cuando se anidan elementos es necesario asegurarse que se hace adecuadamente, es decir, que la etiqueta de cierre del elemento interno se encuentra antes de la etiqueta de cierre del elemento externo.

## Elementos Vacíos

Anteriormente se explicó la estructura básica de los elementos HTML y se indicó que dicha estructura es la que siguen la mayoría de ellos. Los elementos que no siguen esta estructura se conocen como *elementos vacíos*, pues no requieren de contenido y, por ende, de etiqueta de cierre. Por esta razón, estos elementos constan únicamente de una etiqueta de apertura:

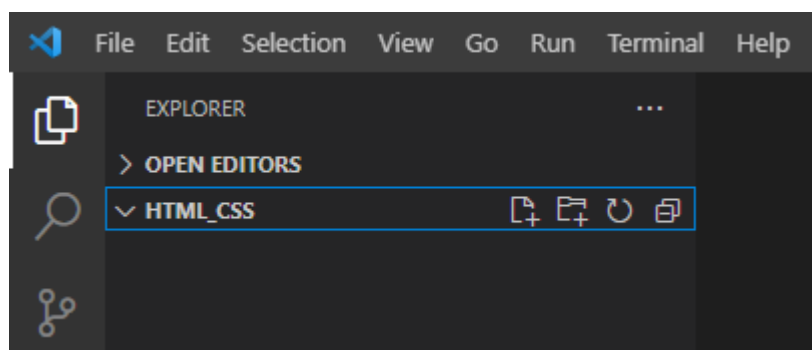
```

```

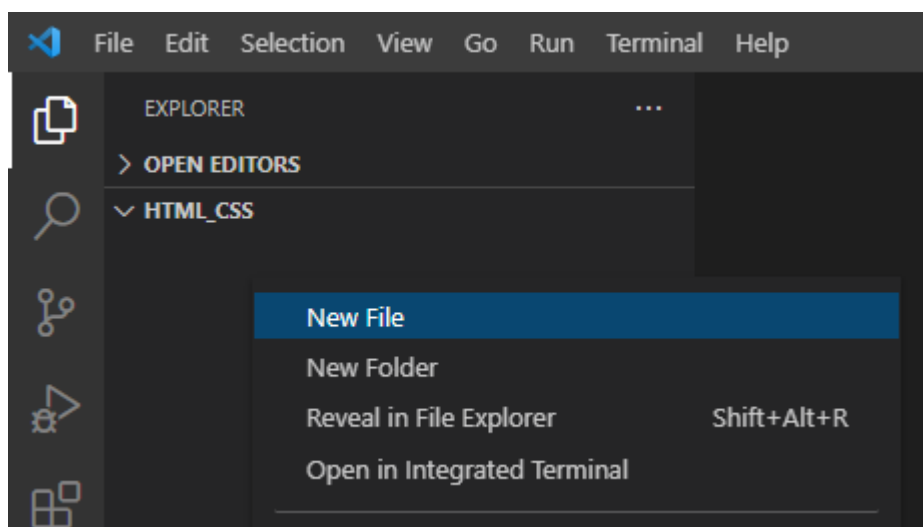
En general, la información que requiere la etiqueta de un *elemento vacío* se indica por medio de atributos. Sin embargo, existen etiquetas que no requieren de atributos pues son de propósito único. Un elemento HTML se considera vacío de acuerdo con la etiqueta que posee. Son realmente pocas las etiquetas que no requieren de contenido, las que más se usan son `<img>`, para presentar una imagen en la página web, y `<br>` para realizar un salto de línea. Esta última, es una de las etiquetas que no requiere atributos, pues su único propósito es realizar un salto de línea.

## Ejercicio HTML

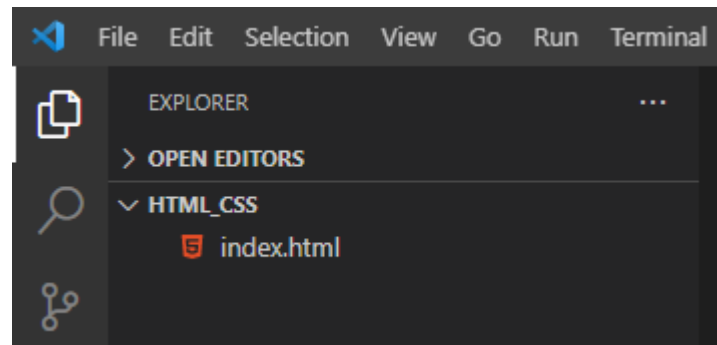
Una vez se han comprendido los conceptos básicos de HTML, se va a realizar un ejercicio para poner en práctica lo aprendido. De esta forma, lo primero que se debe hacer es crear un archivo HTML. Para ello, se crea una carpeta en cualquier ubicación del computador y se abre con Visual Studio Code, de la misma forma que se ha hecho en sesiones anteriores. En esta carpeta se ubicarán los archivos de la página web:



En la carpeta, se pulsa clic derecho del mouse y se selecciona la opción *New File*:



Allí se crea el [archivo HTML](#) donde se realizará el ejercicio. En este caso, se llamará [index.html](#):



Una vez se ha creado el archivo, se debe abrir y copiar la siguiente estructura en él:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Mi Página de Prueba</title>
  </head>

  <body>

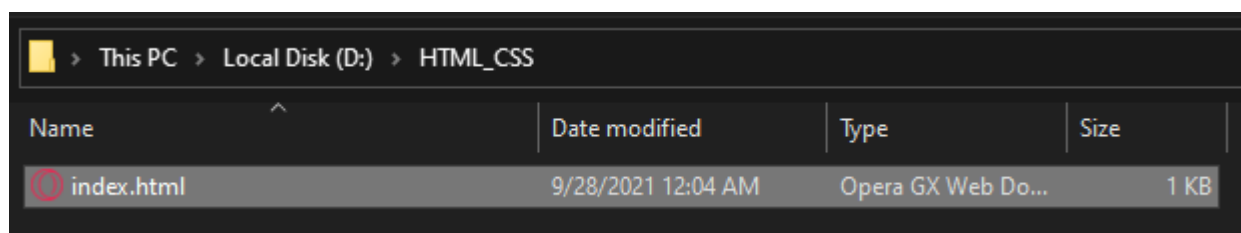
  </body>
</html>
```

Hasta ahora se han visto los conceptos básicos de los elementos HTML individuales. Sin embargo, para integrarlos dentro de una página web, es necesario seguir la estructura anterior. Esta estructura permite a los navegadores web reconocer el archivo como una página web, y configurar lo necesario para su correcta visualización. Para ello se utilizan las etiquetas:

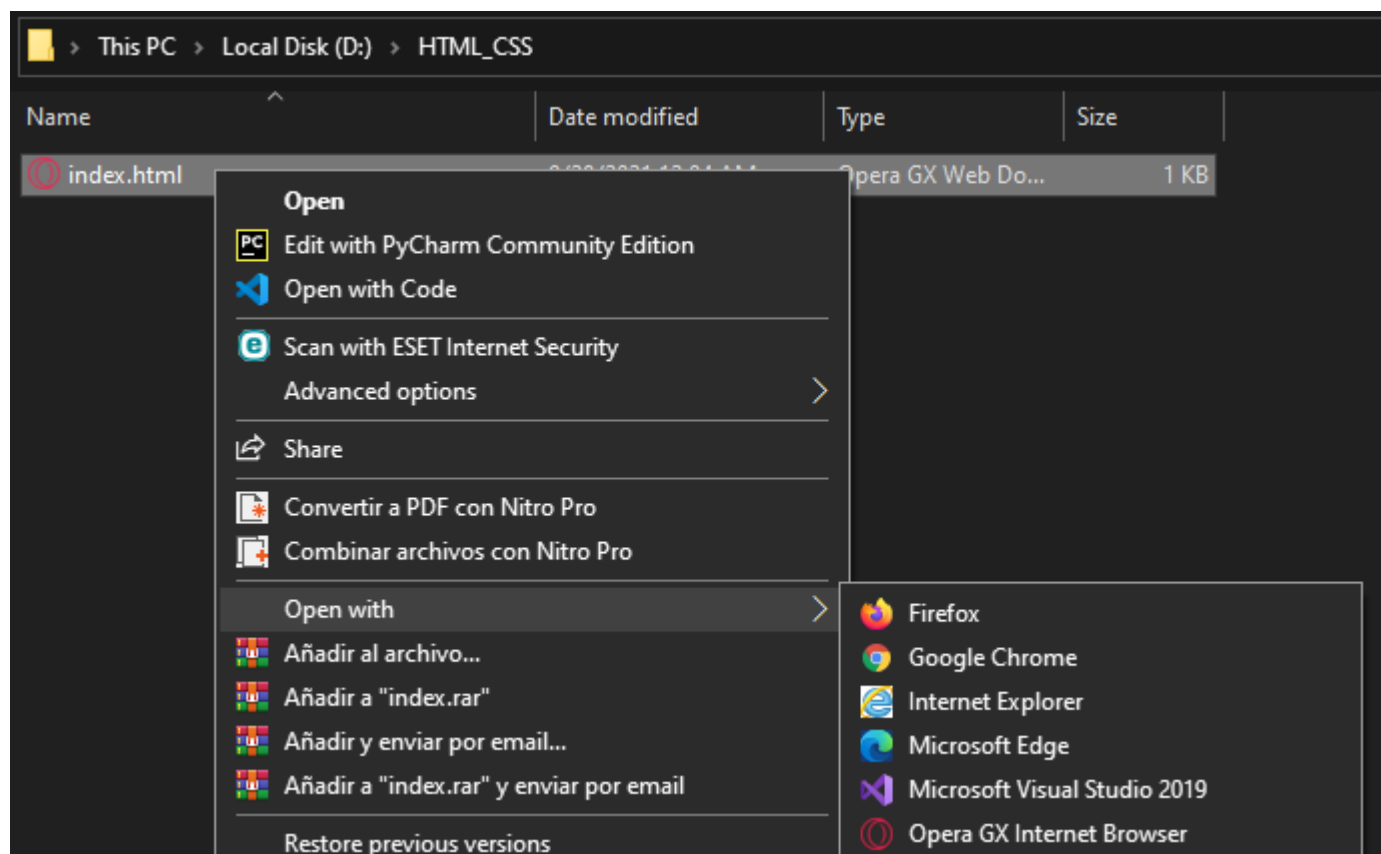
- [<!DOCTYPE html>](#): se utiliza para indicarle al navegador web el tipo de documento que representa el archivo.
- [<html>](#): se utiliza para encerrar todo el contenido de la página web.
- [<head>](#): se utiliza para añadir información adicional a la página web, sin que esta sea visible en el navegador web. El contenido de un elemento [<head>](#) se puede considerar como los atributos de la página web.
- [<meta charset="utf-8" />](#): se utiliza para indicarle al navegador web el conjunto de caracteres que se utilizarán en la página web. [Utf-8](#) incluye casi todos los caracteres de todos los idiomas humanos.

- `<title>`: se utiliza para indicar al navegador web el nombre de la página web y, por ende, el texto que se mostrará en la pestaña del navegador al abrir la página web.
- `<body>`: este elemento encierra todo el contenido que se desea mostrar al usuario que visite la página web. Por esta razón, los elementos HTML que se muestren a continuación se deben ubicar en el contenido de esta etiqueta.

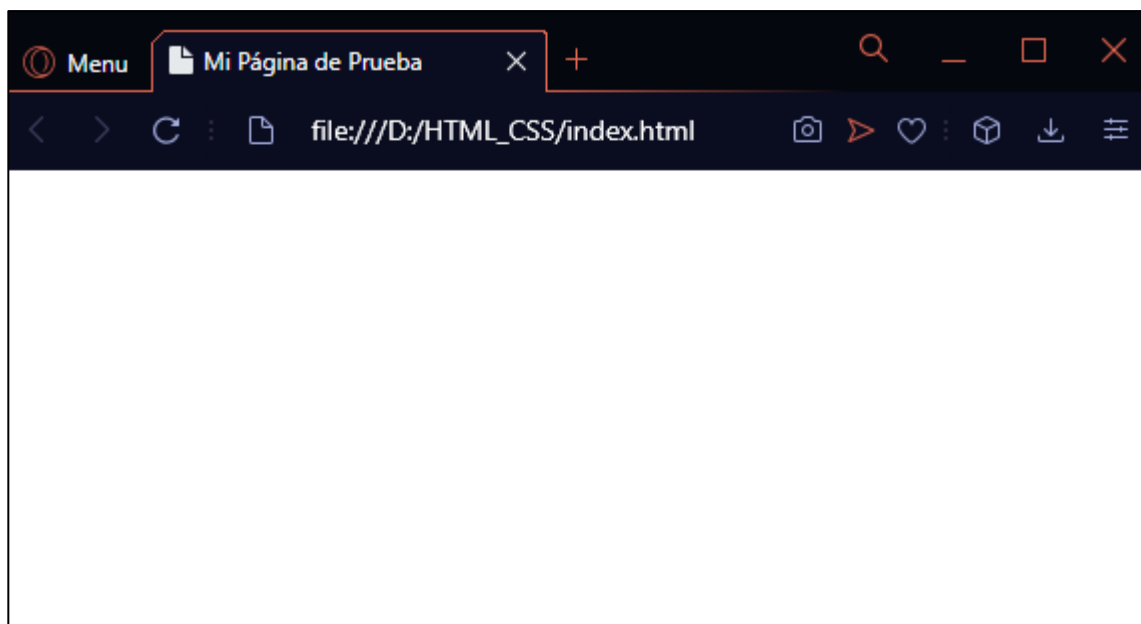
Una vez se guardan los cambios del archivo HTML, para visualizar y comprobar que un navegador web puede abrirlo, es necesario dirigirse a la carpeta contenedora de la página web:



Allí, se debe pulsar clic derecho sobre el archivo HTML creado anteriormente, y se debe elegir el navegador web con el cual se desea abrir el archivo:



En este caso se utilizará *Opera*, sin embargo, todos los navegadores web procesan en documento HTML de la misma forma, por lo cual no hay problema si se utiliza cualquier otro navegador. Una vez se selecciona el navegador web, este se abre y muestra el contenido de la página web. Hasta el momento esta no tiene contenido, por lo cual se muestra una página en blanco:



Se puede notar que el nombre la pestaña es el nombre de la página web indicado anteriormente, por lo tanto, se sabe que los cambios se realizaron satisfactoriamente. A continuación, se mostrarán algunas etiquetas que permitirán ingresar información a la página web.

Muchas veces cuando se abre una página web se encuentra un título que describe su contenido, y algunos subtítulos que dividen el contenido. Para mostrar títulos y subtítulos se utilizan las etiquetas `<h1>`, `<h2>`, `<h3>` y `<h4>`. En esencia, las 4 etiquetas se utilizan de la misma forma, lo único que varía es el tamaño de la letra del texto que se muestra en la página web, siendo `<h1>` la etiqueta cuyo texto es el más grande, y `<h4>` la etiqueta cuyo texto es el más pequeño. Para poner un título y un subtítulo en la página, se ubican los siguientes elementos dentro del *body*:

```
<h1>Mi Página Web</h1>

<h2>Contenido</h2>
```

Posteriormente, si se necesita agregar un párrafo de información acerca del contenido de la página web se

utiliza la etiqueta `<p>`, la misma que se explicó anteriormente. En el archivo, se agregará el siguiente elemento bajo el subtítulo “*Contenido*”:

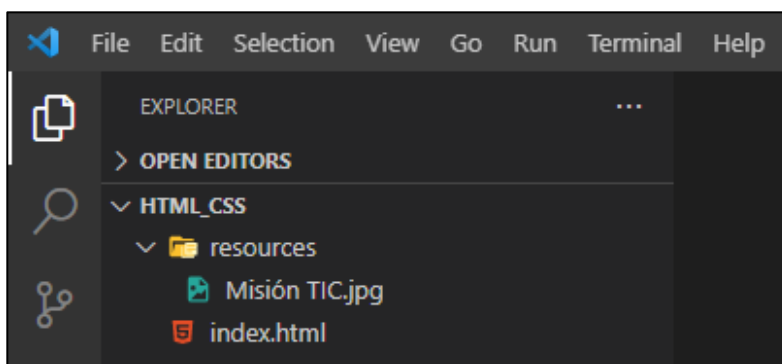
```
<p>
  Esta es un página web de ejemplo cuyo contenido requerirá
  del uso de algunas de las etiquetas más comunes.
  <br>
  <br>
  Dichas etiquetas permitirán añadir a la página web:
</p>
```

Se puede notar el uso de la etiqueta `<br>`, que como se indicó anteriormente, sirve para insertar saltos de línea en el texto. Por el texto ingresado en el párrafo anterior se puede inferir que se va a listar un conjunto de elementos. Para hacerlo, se utiliza la etiqueta `<ul>` para indicar que se van a listar elementos, junto con la etiqueta `<li>`, que contiene a cada elemento de la lista. Estas etiquetas se utilizan de la siguiente forma:

```
<ul>
  <li>Títulos y subtítulos</li>
  <li>Párrafos</li>
  <li>Saltos de línea</li>
  <li>Listas</li>
  <li>Imágenes</li>
  <li>Enlaces</li>
</ul>
```

Otra de las etiquetas útiles en la construcción de páginas web es la etiqueta vacía `<img>`, la cual muestra la imagen referenciada en su atributo `src`. Para utilizarla se debe ubicar una imagen (JPEG, JPG, PNG, etc.) en la carpeta contenedora de la página web. Esta imagen se puede ubicar junto al documento HTML, sin embargo, para mantener el orden de las carpetas, se va a crear la carpeta *resources* y allí se ubicará la imagen que se va a utilizar en la página web:





Una vez se ha ubicado la imagen, se añade la siguiente etiqueta en el documento HTML:

```

```

El valor del atributo `src` es la dirección de la imagen que se desea mostrar. Se debe tener en cuenta que dicha dirección es relativa al documento HTML, es decir, que el navegador web buscará esa dirección partiendo de la carpeta que contiene al documento HTML.

La próxima etiqueta que se utilizará será la etiqueta `<a>`, la cual se utiliza para insertar un enlace que redirige a la página indicada en su atributo `href`, y resalta el texto del contenido para mostrar la existencia de un enlace. En la página web se utilizará esta etiqueta dentro de un párrafo de la siguiente forma:

```
<p>
Si se desea conocer más acerca de las etiquetas y sus usos se puede
dirigir a la <a href="https://www.w3schools.com/">página de W3Schools</a>,
allí se encuentra la documentación de HTML, CSS, JavaScript y de otros
lenguajes. Muy útil para aprender más acerca de la construcción de
páginas web.
</p>
```

Cabe añadir que el atributo `href` también se puede utilizar para redirigir a otras secciones de la página web. Una vez se han añadido todos los elementos anteriores a la página web, el `body` se debe ver así:

```
<body>
<h1>Mi Página Web</h1>
```



```
<h2>Contenido</h2>

<p>
  Esta es un página web de ejemplo cuyo contenido requerirá
  del uso de algunas de las etiquetas más comunes.
  <br>
  <br>
  Dichas etiquetas permitirán añadir a la página web:
</p>

<ul>
  <li>Títulos y subtítulos</li>
  <li>Párrafos</li>
  <li>Saltos de línea</li>
  <li>Listas</li>
  <li>Imágenes</li>
  <li>Enlaces</li>
</ul>

<h3>Imagen</h3>



<h3>Conclusión</h3>

<p>
  Si se desea conocer más acerca de las etiquetas y sus usos se puede
  dirigir a la <a href="https://www.w3schools.com/">página de W3Schools</a>,
  allí se encuentra la documentación de HTML, CSS, JavaScript y de otros
  lenguajes. Muy útil para aprender más acerca de la construcción de
  páginas web.
</p>
</body>
```

Si se abre el documento HTML con el navegador web, se puede ver lo siguiente:



Para concluir con la explicación de HTML, se va a introducir una de las etiquetas más utilizadas para agrupar elementos: la etiqueta `<div>`. Si bien esta etiqueta solo tiene esa función (Agrupar elementos HTML), será muy útil para la integración de un documento HTML con CSS y JavaScript, como se verá más adelante. Para utilizar esta etiqueta basta únicamente con encerrar uno o más elementos HTML dentro de la etiqueta. Por ejemplo, los elementos del documento HTML creados anteriormente se pueden dividir de la siguiente forma:

```
<body>
  <h1>Mi Página Web</h1>

  <div class="Contenido">
    <h2>Contenido</h2>
```

```

<p>
    Esta es un página web de ejemplo cuyo contenido requerirá
    del uso de algunas de las etiquetas más comunes.
    <br>
    <br>
    Dichas etiquetas permitirán añadir a la página web:
</p>

<ul>
    <li>Títulos y subtítulos</li>
    <li>Párrafos</li>
    <li>Saltos de línea</li>
    <li>Listas</li>
    <li>Imágenes</li>
    <li>Enlaces</li>
</ul>
</div>

<div class="Imagen">
    <h3>Imagen</h3>

    
</div>

<div class="Conclusión">
    <h3>Conclusión</h3>

    <p>
        Si se desea conocer más acerca de las etiquetas y sus usos se puede
        dirigir a la <a href="https://www.w3schools.com/">página de W3Schools</a>,
        allí se encuentra la documentación de HTML, CSS, JavaScript y de otros
        lenguajes. Muy útil para aprender más acerca de la construcción de
        páginas web.
    </p>
</div>
</body>

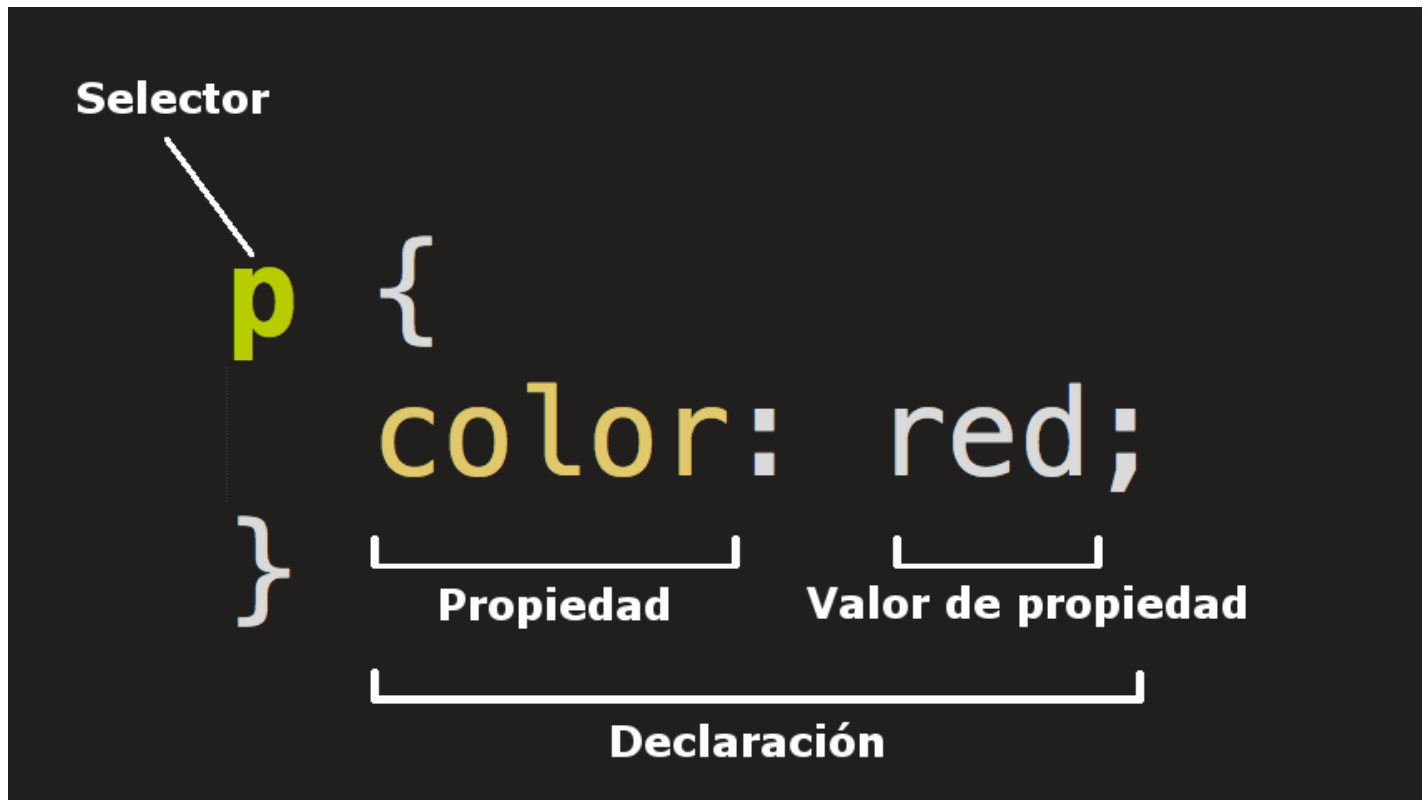
```

Estas divisiones se utilizarán en el ejercicio de CSS que se encuentra a continuación.

## Explicación y uso de CSS

### Reglas de CSS

CSS es un lenguaje que se utiliza para dar estilo a cada elemento de HTML individualmente. Para lograrlo, se debe asignar una *regla predeterminada* a cada uno de estos elementos. Todas las *reglas* tienen la siguiente estructura:



De esta forma, una *regla* consta de:

- Selector: es una cadena de texto que indica el elemento HTML al cual se aplicará la *regla*.
- Declaración: es el conjunto de propiedad y valor, que especifica una *regla de estilo* que debe seguir el elemento HTML.
- Propiedad: es el nombre de la propiedad cuyo valor se desea cambiar en el elemento.
- Valor de propiedad: es el valor que se asigna a la propiedad.

Además, cuando se declara una *regla* se debe tener en cuenta que:

- Aparte del selector, todas las *reglas* deben estar encerrada entre llaves (`{}`).

- Dentro de toda declaración se debe utilizar dos puntos (:) para separar la propiedad de su valor
- Toda declaración debe tener un punto y coma (;) al final, que la separe de las demás declaraciones.

## Seleccionar varios elementos

Es posible aplicar una regla a múltiples elementos HTML, para ello, se separa cada selector por una coma (,). Un ejemplo podría ser el siguiente:

```
p,li,h1 {  
    color: red;  
}
```

## Tipos de selectores

Existen varias formas de asignar una regla a un elemento HTML, pero las más utilizadas son las siguientes:

- Selectores de etiquetas (*tag*): son aquellos que asignan la regla a todos los elementos HTML que utilizan la etiqueta especificada. Por ejemplo, para aplicar una regla sobre todos los elementos que tengan la etiqueta `<p>` se declara la regla de la siguiente forma:

```
p {  
    color: red;  
}
```

- Selectores de identificación (*#id*): son aquellos que asignan la regla al elemento HTML que tiene cierto *id* especificado. Por ejemplo, para aplicar una regla sobre el elemento cuyo *id* es *"my-id"* se declara la regla de la siguiente forma:

- 

```
#my-id {  
    color: red;  
}
```

- Selectores de clase (*.class*): son aquellos que asignan la regla a los elementos HTML que tienen cierta clase (*class*) especificada. Por ejemplo, para aplicar una regla sobre todos los elementos que tienen la clase (*class*) *"my-class"* se declara la regla de la siguiente forma:

```
.my-class {  
  color: red;  
}
```

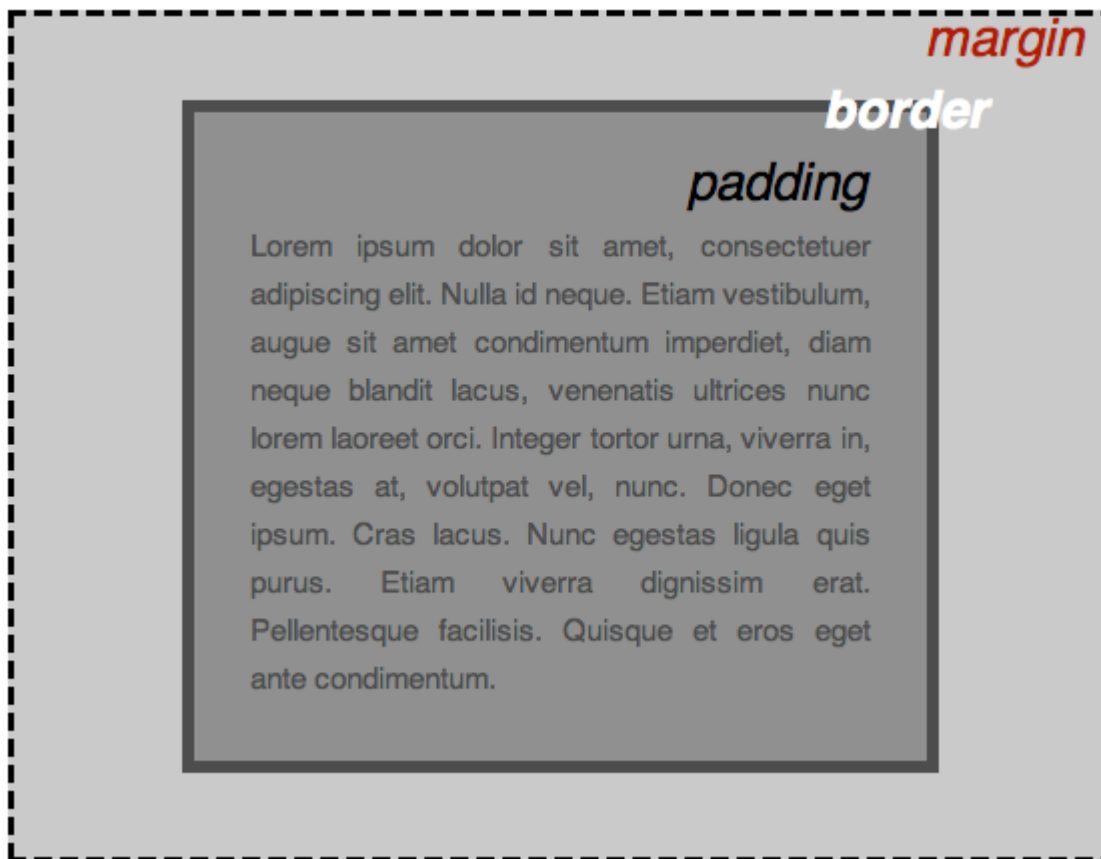
## Modelo de cajas

En general, la mayoría de los elementos HTML de una página web se pueden pensar como cajas apiladas una sobre la otra. Por esta razón, el diseño de CSS se basa en el modelo de cajas, en el cual, todos los elementos que ocupan un espacio en la página web tienen una caja asociada. De esta forma, diseñar una página web con CSS, se trata en su mayoría de organizar y ajustar las propiedades de las cajas y sus contenidos.

Existen varias propiedades que tienen las cajas, sin embargo, sus tres propiedades principales son:

- *Padding* (relleno): es el espacio interno del elemento que separa su contenido del marco.
- *Border* (marco): es la línea que delimita lo que pertenece y lo que no pertenece al elemento.
- *Margin* (margen): es el espacio externo al elemento, lo que lo separa de los demás elementos.

El espacio que le corresponde a cada propiedad se puede apreciar mejor en la siguiente imagen:



Estas propiedades se utilizarán más adelante para organizar el contenido de la página web.

## Ejercicio CSS

Existen 3 formas diferentes de añadir código CSS a un documento HTML:

1. Asignando el atributo style a un elemento HTML, cuyo valor es el código CSS asignado:

```
<p style="color: red; font-family: Verdana;">
  Mi gato es muy gruñón
</p>
```

2. Ubicando todo el código CSS del documento web en la etiqueta *style* que se encuentra en el *header* del documento HTML:

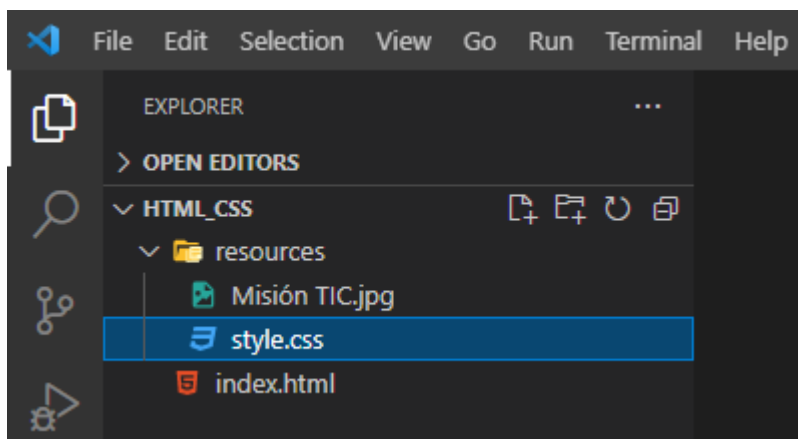


```
<head>
  <meta charset="utf-8" />
  <title>Mi Página de Prueba</title>
  <style type="text/css">
    p {
      color: red;
      font-family: Verdana;
    }
  </style>
</head>
```

3. Ubicando todo el código CSS en un documento independiente, e importándolo en el *header* del documento HTML:

```
<head>
  <meta charset="utf-8" />
  <title>Mi Página de Prueba</title>
  <link href="resources/style.css" rel="stylesheet" type="text/css"/>
</head>
```

Cualquiera de los 3 métodos es completamente válido. Sin embargo, para este ejercicio se creará un archivo CSS dentro de la carpeta *resources* de la página web creada anteriormente y se realizará su debida importación en el documento HTML:



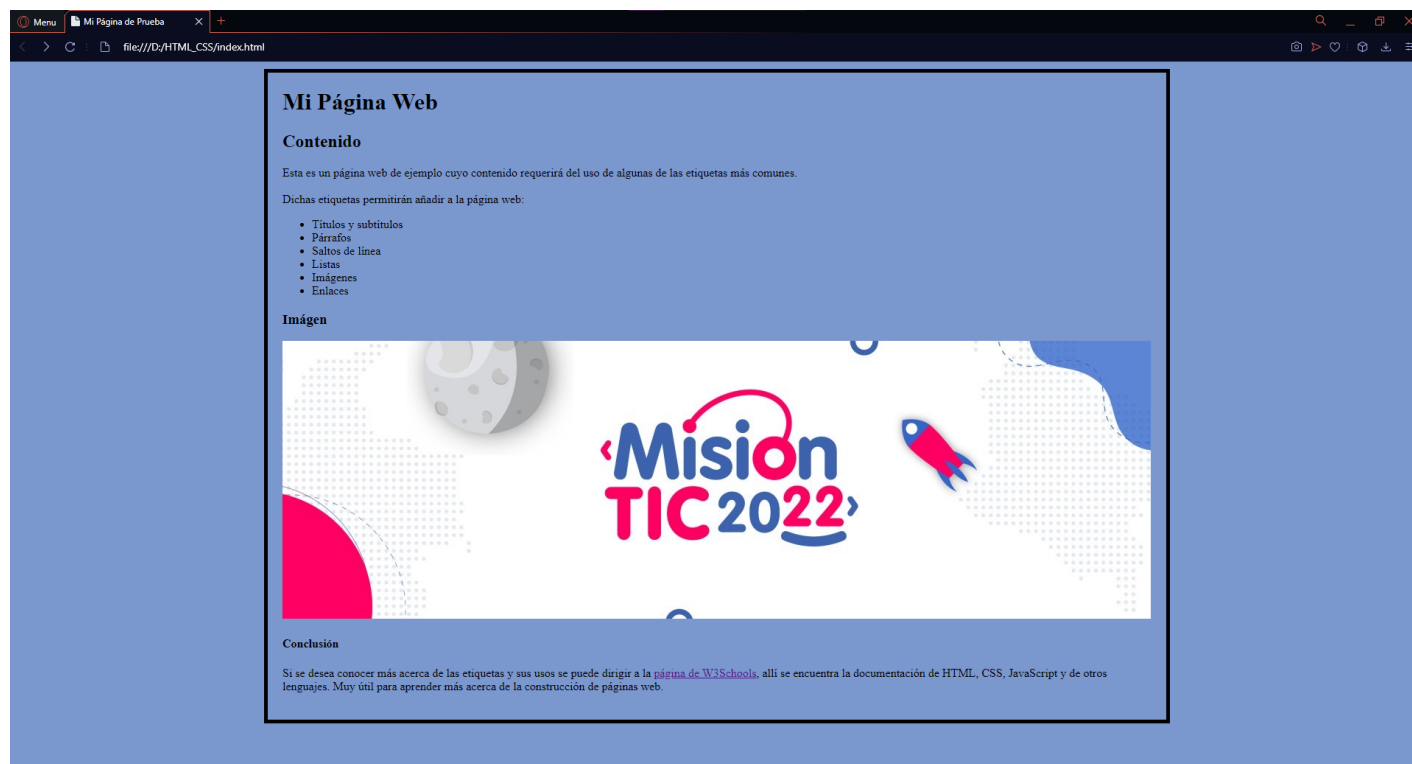
Para su importación en el archivo HTML, se añade el siguiente elemento en el contenido de la etiqueta *head*:

```
<link href="resources/style.css" rel="stylesheet" type="text/css"/>
```

Este le indica al navegador que el código CSS de la página web se encuentra en el archivo indicado. Una vez se ha importado el archivo, se pueden hacer modificaciones sobre el código CSS y se podrán notar los cambios en la página web. Por ejemplo, si se añade la regla:

```
body {
  width: 1180px;
  margin: 10px auto;
  background-color: #7a97ce;
  padding: 0px 20px 20px 20px;
  border: 5px solid black;
}
```

La página se verá así:



En esta regla se cambia el estilo de la etiqueta `body` del documento HTML. Cambiando elementos como el ancho del elemento, la distancia con otros elementos (*margin*), el fondo de la página, la distancia de su contenido con los bordes (*padding*), y el ancho, estilo y color del marco (*border*) del elemento. En esta regla se deben notar tres aspectos:

1. Las distancias se miden en *px*, es decir, en píxeles. Existen otras unidades de medición, pero esta es la más utilizada.
2. A atributos como *margin*, *padding* y *border* se les puede ingresar más de un valor. Cuantos valores y que valores se pueden poner a un atributo depende completamente de cada uno de ellos. En el caso de *margin* y *padding* se pueden ingresar hasta 4 valores, donde cada valor representa en orden la distancia que toma el elemento externa (*margin*) o internamente (*padding*) en su borde superior, izquierdo, inferior y derecho. El valor *auto* busca que todas las márgenes tengan la misma distancia y, por ende, que el elemento se encuentre centrado.
3. El valor de un color se puede describir de dos formas: Con el nombre del color en inglés (como *black*), o con el código de color, ya sea con su código rgb o con su código hexadecimal (El que se usa en el atributo *background-color*).

La siguiente regla que se creará en el archivo CSS es la siguiente:

```
h1 {
  color: white;
  text-align: center;
  font-family: cursive;
}
```

En esta regla se indica que toda etiqueta de título `<h1>` debe tener texto de color blanco, centrado, y que su tipografía debe ser cursiva.

Anteriormente se crearon etiquetas `<div>` para agrupar los elementos de la página web, y en cada una de ellas se indicó el atributo *class*. Este atributo se utilizará a continuación para otorgarle una regla CSS a dos de estas secciones:

```
.Contenido {
  color: #5E1C36;
  border-radius: 10px;
  margin: 0px 40px 0px 40px;
  padding: 10px 40px 10px 40px;
```



```
background-color: #E34481;
}

.Conclusión {
  color: #e1e3ec;
  font-size: 20px;
  text-align: center;
  border-radius: 10px;
  margin: 0px 40px 0px 40px;
  padding: 5px 40px 5px 40px;
  background-color: #5C85D5;
}
```

La gran mayoría de atributos utilizados en estas reglas ya se vieron anteriormente. Los dos atributos nuevos que se utilizan son *border-radius*, que se encarga de redondear la caja del elemento de acuerdo con la distancia ingresada en píxeles, y *font-size*, que como su nombre indica se utiliza para cambiar el tamaño del texto.

Los atributos de CSS son muy poderosos, con ellos es posible incluso ocultar un elemento HTML de la página web. Para mostrar esto se utilizará la siguiente regla:

```
.Imagen h3 {
  display: none;
}
```

Esta regla tendrá efecto únicamente en las etiquetas `<h3>` que se encuentren dentro de un elemento con la clase *Imagen*. Por lo cual, solo el subtítulo “Imagen” de la página web debe desaparecer (el subtítulo “conclusión” debe permanecer visible). Este tipo de selectores se utiliza para asegurar que la regla aplica únicamente a los elementos deseados.

Finalmente, la última regla que se va a añadir al CSS de la página web es el siguiente:

```
.Imagen img {
  height: 300px;
  display: block;
  border-radius: 100px;
  border: 3px solid white;
  margin: 20px auto 20px auto;
```

```
}
```

De la misma forma que la regla anterior, los atributos definidos en esta regla tendrán efecto únicamente en las etiquetas `<img>` que se encuentren en un elemento con la clase `Imagen`. En esta regla se cambian las dimensiones de la imagen, se redondean las esquinas de la imagen, y se centra la imagen.

Una vez se han realizado todas las reglas, el documento CSS debe verse así:

```
body {
  width: 1180px;
  margin: 10px auto;
  background-color: #7a97ce;
  padding: 0px 20px 20px 20px;
  border: 5px solid black;
}

h1 {
  color: white;
  text-align: center;
  font-family: cursive;
}

.Contenido {
  color: #5E1C36;
  border-radius: 10px;
  margin: 0px 40px 0px 40px;
  padding: 10px 40px 10px 40px;
  background-color: #E34481;
}

.Conclusión {
  color: #e1e3ec;
  font-size: 20px;
  text-align: center;
  border-radius: 10px;
  margin: 0px 40px 0px 40px;
  padding: 5px 40px 5px 40px;
  background-color: #5C85D5;
}
```

```

.Imagen h3 {
  display: none;
}

.Imagen img {
  height: 300px;
  display: block;
  border-radius: 100px;
  border: 3px solid white;
  margin: 20px auto 20px auto;
}

```

Si se abre el documento HTML con un navegador web, se podrán ver los cambios realizados:



**Nota:** de ser necesario, en el material de la clase se encuentra una archivo **C3.AP.12. html-css.rar**, con el desarrollo del ejercicio realizado en esta guía.