

1. Más información sobre la página

1.1. Metadatos

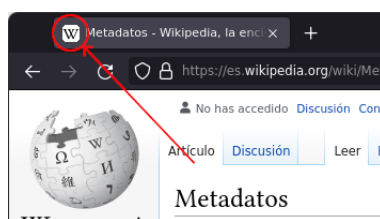
Dentro del elemento **head**, podemos agregar datos útiles, tanto para el navegador como para otras tipos de aplicaciones que podrían llegar a consultar la página. Es muy usual encontrar *metadatos*¹ con el uso del elemento vacío **meta**.

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
    scale=1.0">
  <link rel="stylesheet" href="css/style.css">
  <title>Mi galería</title>
</head>
```

Como podemos observar, aquí hay 2 metadatos, el primero indica al navegador la codificación² de los caracteres del archivo. Por lo general se adopta ‘UTF-8’ que es la codificación más habitual de la Web. El segundo, indica al navegador las dimensiones que deben adoptar los elementos de la página. Lo más habitual es que se le pida al navegador que el ancho se ajuste para ocupar todo el espacio disponible. Esto es de especial importancia para tener el control de cómo se verá la página en diferentes dispositivos con diferentes resoluciones de pantalla.

1.2. Favicon (*favorites icon*)

A modo de sumar a la *identidad de marca*, los sitios suelen mostrar las marcas y logotipos propios, incluso como un pequeño ícono en la barra o pestaña del navegador. A este se lo conoce como *favicon*, porque es mostrado y almacenado en listas de páginas favoritas o barras de accesos directos.



¹“Datos que describen otros datos. Metadatos se refiere a un grupo de datos que describen el contenido informativo de un objeto...” <https://es.wikipedia.org/wiki/Metadatos>

²Este tema es estudiado en Estructura y Funcionamiento de los Sistemas Informáticos I (4^{to} año).

Para poder agregar uno debemos utilizar el elemento **link** con el atributo **rel** igualado a “icon” o “shortcut icon” seguido de los atributos que indiquen el tipo y la referencia a la imagen deseada.

```
<link rel="icon" type="image/x-icon" href="img/favicon.ico">
```

Esta imagen tiene usualmente la dimensión de 16x16 pixeles, puede ser animada y la mayoría de los navegadores soporta los siguientes formatos:

Extensión	type
.ico	"image/x-icon"
.png	"image/png"
.gif	"image/gif"
.jpeg/.jpg	"image/jpeg"
.svg	"image/svg+xml"

2. Elementos de listado

Podemos agrupar ítems a través de elementos de listado. En **HTML** podemos optar por listas sin orden (*unordered list*) con la etiqueta **** o con orden (*ordered list*) con ****. En ambos casos los elementos ítem de lista (*list item*) utilizan la etiqueta ****.

Los elementos **li** siempre deben estar anidados dentro de un **ol** o **ul**. El navegador renderizará por defecto las listas de la siguiente manera:

```
<ul>
  <li>Item A</li>
  <li>Item B</li>
  <li>Item C</li>
</ul>

<ol>
  <li>Primero</li>
  <li>Segundo</li>
  <li>Tercero</li>
</ol>
```

- Item A
- Item B
- Item C

1. Primero
2. Segundo
3. Tercero

Además, es posible anidar elementos **ul** y **ol** dentro de otros para tener diferentes niveles de listado o numeración, incluso mezclando unos con otros.

```
<ul>
  <li>Item A</li>
  <li>Item B</li>
  <li>Item C</li>
  <ul>
    <li>Item subA</li>
    <li>Item subB</li>
    <ol>
      <li>subSubB Primero</li>
      <li>subSubB Segundo</li>
      <li>subSubB Tercero</li>
    </ol>
  </ul>
</ul>

<ol>
  <li>Primero</li>
  <li>Segundo</li>
  <ol>
    <li>Primer sub</li>
    <li>Segundo sub</li>
    <ul>
      <li>Item subSegundoA</li>
      <li>Item subSegundoB</li>
    </ul>
    <li>Tercer sub</li>
  </ol>
  <li>Tercero</li>
</ol>
```

- Item A
 - Item B
 - Item C
 - Item subA
 - Item subB
 - 1. subSubB Primero
 - 2. subSubB Segundo
 - 3. subSubB Tercero
-
1. Primero
 2. Segundo
 - 1. Primer sub
 - 2. Segundo sub
 - Item subSegundoA
 - Item subSegundoB
 - 3. Tercer sub
 3. Tercero

2.1. Decoración de listas

Para los componentes de lista hay algunas propiedades específicas de CSS para poder decorarlas y establecer sus reglas. Por ejemplo tenemos:

list-style-type: Establece el tipo de viñetas para la lista (viñetas cuadradas o circulares para una lista no ordenada; números, letras, o números romanos para una lista ordenada.)

list-style-position: Establece si las viñetas aparecen dentro de los elementos de la lista o fuera de ellos, antes del inicio de cada elemento.

list-style-image: Permite usar una imagen personalizada para la viñeta.

```
<ul>
  <li>
    
  </li>
  <li>
    
  </li>
  <li>
    
  </li>
</ul>

<ol>
  <li>
    
  </li>
  <li>
    
  </li>
  <li>
    
  </li>
</ol>
```

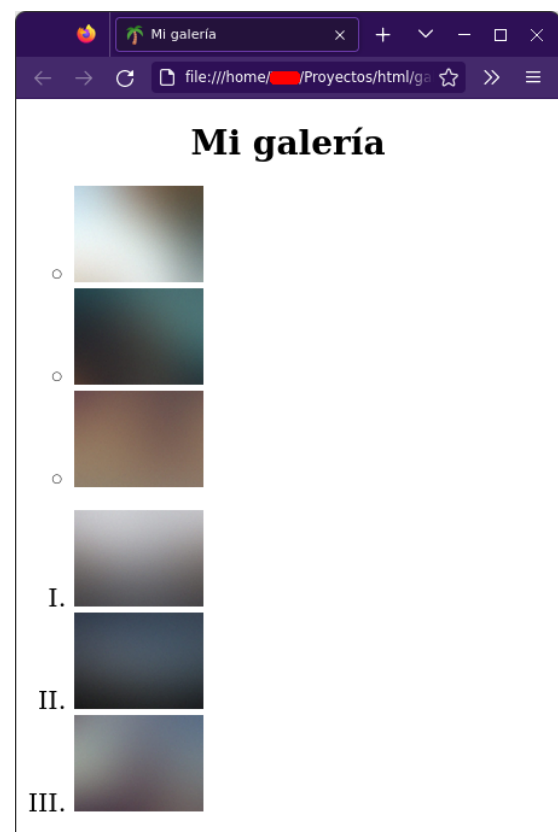
style.css

```
h1 {
  text-align: center;
}

ul {
  list-style-type: circle;
}

ol {
  list-style: upper-roman;
}

li {
  font-size: x-large;
}
```



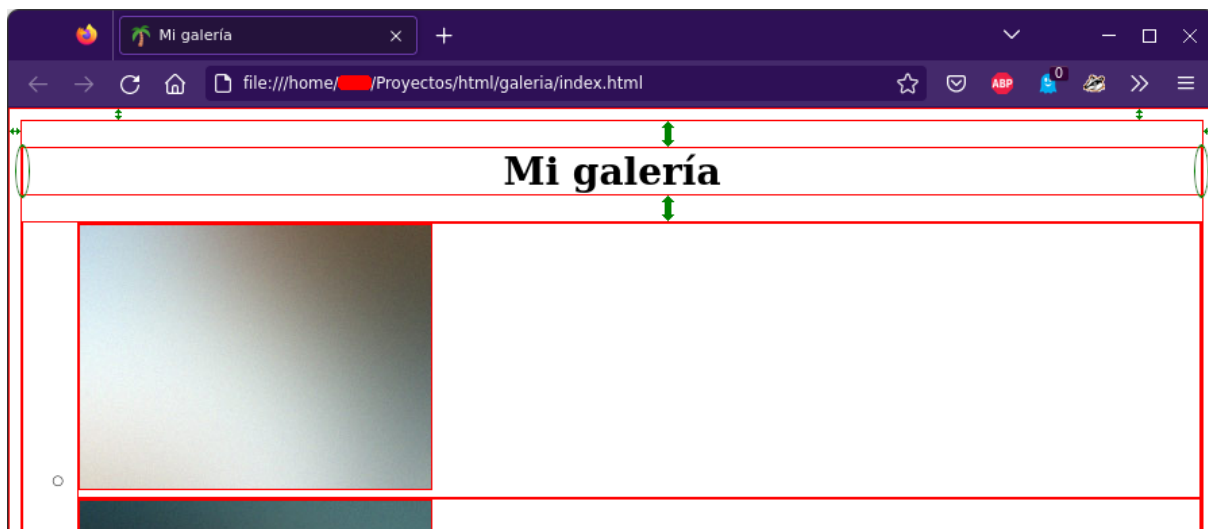
3. Margenes y relleno

3.1. Clave *margin*

Como exploramos en la clase anterior aquí estuvimos utilizando elementos de bloque, estos se caracterizan por ajustarse u ocupar una línea completa, haciendo que los elementos anteriores y siguientes queden arriba o abajo respectivamente. Entre ellos existirá una separación denominada **margen** (*margin*) y además los elementos ocuparán un determinado espacio llamado **relleno** (*padding*). Para entender mejor este concepto utilizaremos una decoración especial que nos ayudará a entender los cambios que ocurren al modificar uno y otro.

```
* {  
  border: 1px solid red;  
}
```

El carácter ‘*’ (asterisco) es lo que se conoce como un comodín (*wildcard*) que indica “todos los elementos”. Y a todos ellos les estamos agregando un borde de **1px**, con tramado solido (línea constante) de color rojo. Esto creará una caja al rededor de cada elemento de nuestro **HTML**.



El elemento **html** es el que está contra el borde del contenido de la ventana. Separado por un margen, está el **body** que contiene todo lo demás. Se observa que el **h1** tiene un margen superior e inferior, pero no posee márgenes a los costados, por ello se ve “pegado” contra el **body**.

Para modificar estos podemos utilizar las siguientes claves en las reglas de **CSS**:

```
margin-top  
margin-right  
margin-bottom  
margin-left
```

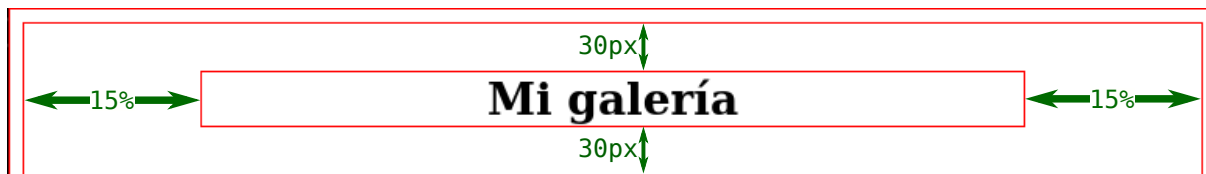
Pero además existe una manera abreviada de utilizar estas reglas utilizando la clave **margin**, la cual se aplica de la siguiente manera:

```
/* Aplica a todos los cuatro lados */  
margin: 10px;  
  
/* Vertical | Horizontal */  
margin: 5px 15px;  
  
/* Arriba | Horizontal | Abajo */  
margin: 5px 5% 7px;  
  
/* Arriba | Derecha | Abajo | Izquierda */  
margin: 2px 5px 7px 4px;
```

Si se especifica una sola unidad, entonces esta se aplica a todos los lados; si hay dos unidades se interpreta que la primera es por la vertical y la segunda el margen horizontal; de haber 3 valores, el primero será el margen superior, luego el margen horizontal y por último el inferior; y si hay cuatro valores, estos indicarán las cuatro esquinas empezando por la superior en sentido de las agujas del reloj.

Por ejemplo si aplicáramos la siguiente regla al **h1**:

```
h1 {  
  text-align: center;  
  margin: 30px 15%;  
}
```



3.2. Clave *padding*

La sintaxis del **relleno** (*padding*) de elementos es similar a la de **margin**. Existiendo las claves para cada uno de sus lados, pero además uno 'flexible' denominado simplemente

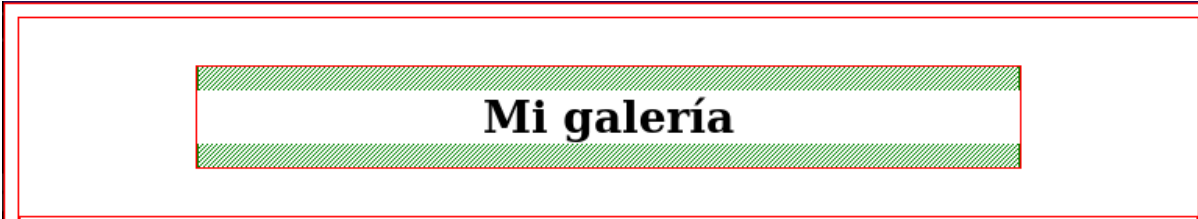
padding.

```
padding-top  
padding-right  
padding-bottom  
padding-left
```

```
/* Aplica a todos los cuatro lados */  
padding: 10px;  
  
/* Vertical | Horizontal */  
padding: 5px 15px;  
  
/* Arriba | Horizontal | Abajo */  
padding: 5px 5% 7px;  
  
/* Arriba | Derecha | Abajo | Izquierda */  
padding: 2px 5px 7px 4px;
```

Por ello si agregáramos la siguiente combinación de *clave-valor* se obtendría el resultado que se muestra en la siguiente imagen.

```
h1 {  
  text-align: center;  
  margin: 30px 15%;  
  padding: 15px 0;  
}
```

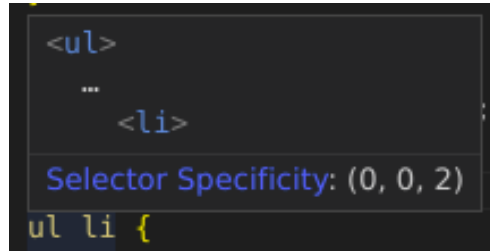


Mi galería

4. Reglas en elementos anidados

Podemos especificar reglas que se apliquen a elementos dentro de otros, es decir que se decoren únicamente los elementos que se encuentran dentro de otros. Por ejemplo, es posible decorar los elementos `li`, pero únicamente los que estén dentro de elementos `ul`:

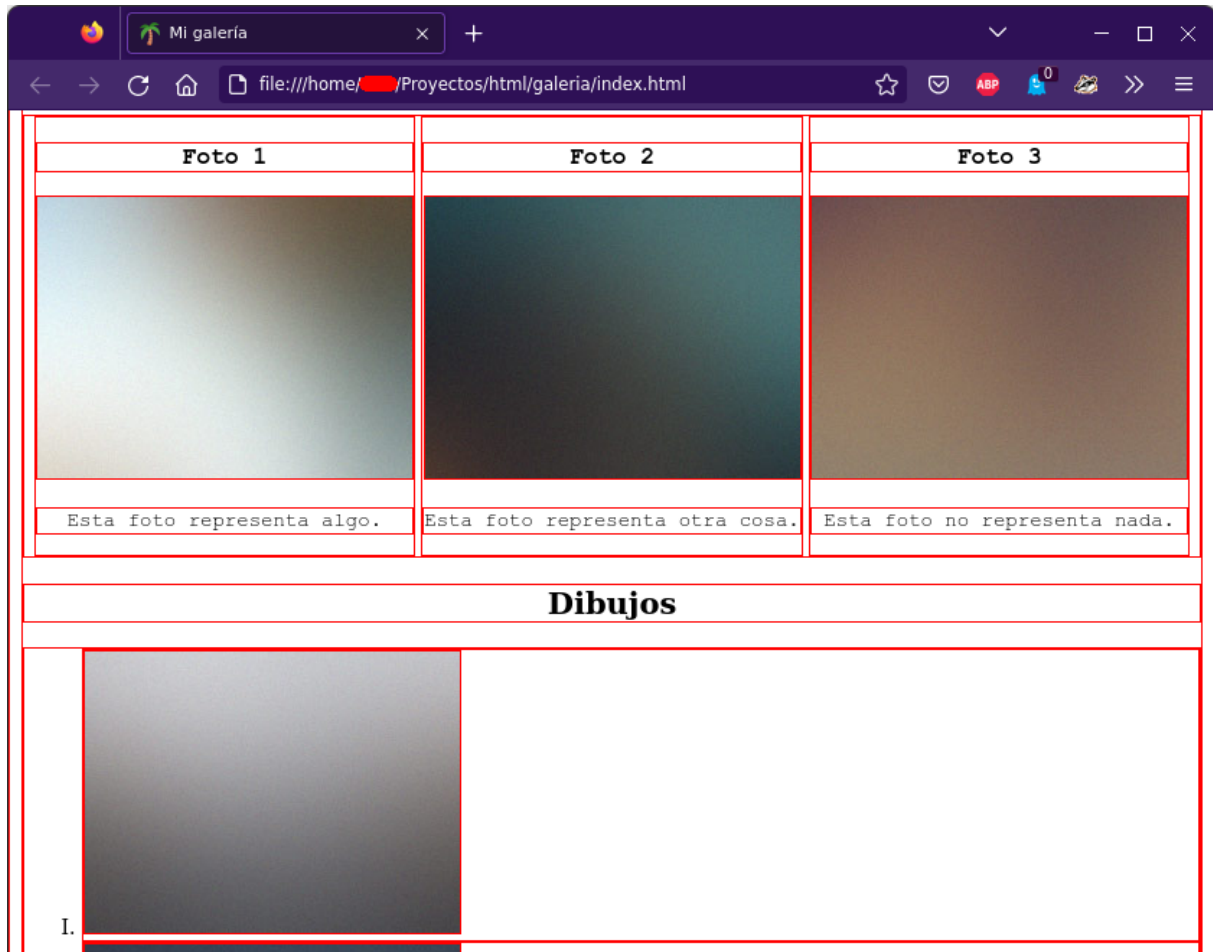
```
ul li {  
  display: inline-block;  
  font-family: 'Courier New',  
    Courier, monospace;  
}
```



Utilizando **CSS** podemos forzar a un elemento de bloque a ser decorado como un elemento de línea utilizando la calve **display**.

Continuemos con el ejemplos, y veamos como afecta esto a los elementos **li** dentro del elementos **ul**:

```
<body>  
<h1>Mi galería</h1>  
<h2>Fotos</h2>  
<ul>  
  <li>  
    <h3>Foto 1</h3>  
      
    <p>Esta foto representa algo.</p>  
  </li>  
  <li>  
    <h3>Foto 2</h3>  
      
    <p>Esta foto representa otra cosa.</p>  
  </li>  
  <li>  
    <h3>Foto 3</h3>  
      
    <p>Esta foto no representa nada.</p>  
  </li>  
</ul>  
<h2>Dibujos</h2>  
<ol>  
  <li>  
      
  </li>  
  <li>  
      
  </li>  
  <li>  
      
  </li>  
</ol>  
</body>
```



Para poder compactar un poco más los elementos presentes se pueden reducir los márgenes y rellenos. Apliquemos las siguientes reglas:

```
h1,h2 {
    text-align: center;
}

h1 {
    margin: 30px 15%;
    padding: 15px 0;
}

ul {
    list-style-type: none;
    text-align: center;
    margin: 0;
    padding: 0;
}

ul li {
    display: inline-block;
    font-family: 'Courier New', Courier, monospace;
}
```

```
/*Todos los elementos dentro de un li que esté dentro de un ul*/  
ul li * {  
    margin: 0;  
    padding: 0;  
}  
/* Todos lo p dentro de un li dentro de ul */  
ul li p {  
    text-align: left;  
}  
/* Todos los img dentro de un li dentro de un ul */  
ul li img {  
    display: block;  
}  
  
ol {  
    list-style: upper-roman;  
}
```

Como podemos observar, aquí hay reglas "solapadas", es decir, primero se establecen reglas generales para los elementos dentro de un **li** aplicando el comodín '*', y luego se aplican reglas para elementos específicos como **p** e **img**. **Siempre se aplicarán las reglas más específicas**, sobrescribiendo a las más generales. Es por ello, que los párrafos en los **li** dentro de los **ul** se escriben alineados a la izquierda y no al centro como se especificó en las reglas para **ul**.



5. Recursos

5.1. Referencias a los elementos HTML utilizados:

- Elemento `<meta>`:

- https://www.w3schools.com/tags/tag_meta.asp
- <https://developer.mozilla.org/es/docs/Web/HTML/Element/meta>

- Elemento `<link>`:

- https://www.w3schools.com/tags/tag_link.asp
- <https://developer.mozilla.org/es/docs/Web/HTML/Element/link>

- Elemento `<ul>`:

- https://www.w3schools.com/tags/tag_ul.asp
- <https://developer.mozilla.org/es/docs/Web/HTML/Element/ul>

- Elemento `<ol>`:

- https://www.w3schools.com/tags/tag_ol.asp
- <https://developer.mozilla.org/es/docs/Web/HTML/Element/ol>

- Elemento `<li>`:

- https://www.w3schools.com/tags/tag_li.asp
- <https://developer.mozilla.org/es/docs/Web/HTML/Element/li>

5.2. Propiedades de CSS utilizadas:

- `display`:

- https://www.w3schools.com/cssref/pr_class_display.asp
- <https://developer.mozilla.org/es/docs/Web/CSS/display>

- **border:**

- https://www.w3schools.com/cssref/pr_border.asp
- <https://developer.mozilla.org/es/docs/Web/CSS/border>

- **margin:**

- https://www.w3schools.com/css/css_margin.asp
- <https://developer.mozilla.org/es/docs/Web/CSS/margin>

- **padding:**

- https://www.w3schools.com/css/css_padding.asp
- <https://developer.mozilla.org/es/docs/Web/CSS/padding>

- **list-style-type:**

- https://www.w3schools.com/cssref/pr_list-style-type.php
- <https://developer.mozilla.org/en-US/docs/Web/CSS/list-style-type>

5.3. Otros recursos

- **Sobre el favicon**

- https://www.w3schools.com/html/html_favicon.asp
- <https://developer.mozilla.org/en-US/docs/Glossary/Favicon>
- <https://www.favicon.cc/>
- <https://favicon.io/>

- **Definición de *Media-Types***

- <https://www.iana.org/assignments/media-types/media-types.xhtml>

- ***Lorem Ipsum***

- https://es.wikipedia.org/wiki/Lorem_ipsum
- <https://www.lipsum.com/>