

HTML5 (5.2) DESDE CERO 2021 Dorian

Vocabulario web

Ip

es el identificador numérico único que representa a una maquina en la red de internet, cada pagina o aplicación esta alojada en una maquina y mediante este ip podemos acceder a ellas

Dominio web

Un nombre de dominio (a menudo denominado simplemente dominio) es un nombre fácil de recordar asociado a una dirección IP física de Internet. Al utilizar un nombre de dominio en lugar de una dirección IP numérica para identificar una ubicación en Internet, es mucho más fácil recordar y escribir direcciones web.

Se trata de un nombre único que se muestra después del signo @ en las direcciones de correo electrónico y después de www. en las direcciones web. Por ejemplo, el nombre de dominio example.com se podría traducir a la dirección física 198.102.434.8. Otros ejemplos de nombres de dominio podrían ser google.com y wikipedia.org.

Cualquiera puede comprar un nombre de dominio. Solo tienes que ir a un registrador o un host de dominios, encontrar un nombre que nadie más utilice y abonar una pequeña cuota anual para ser su propietario.

- Los nombres de dominio se organizan en diferentes niveles
 - **Nivel 1:** Genericos y territoriales
 - Genericos: son dominios de propósito general : .com , .org , .gob, .net, .edu, .biz, .mobi, .xxx
 - El registro de nombre de domio de .com, .org y .net no esta sometido a ningún tipo de verificación previa, el primero que lo registre será el dueño del dominio
 - Territoriales: son aquellos que identifican el país .es para España, .co para Colombia, .ec para ecuador y asi, el nombre de registro de nombres de dominio territoriales lo administra según las normas de cada pais
 - **Nivel 2:** Son los nombres de dominio, lo que cualquier persona puede registrar en cualquier momento. El **dominio de segundo nivel** de esta página acopicesar.org
 - **sería acopicesar**, y el de primer nivel el .org
 - **Nivel 3:** Son extensiones de los dominios, o también llamados subdominios. Un ejemplo sería algo del estilo .tercernivel.midominiodesegundonivel.com
 - La jerarquía de los dominios va de derecha a izquierda

DNS

DNS corresponde a las siglas en inglés de "Domain Name System", es decir, "Sistema de nombres de dominio". Este sistema es básicamente la agenda telefónica de la Web que organiza e identifica dominios. Mientras que una agenda telefónica traduce un nombre como "Acme Pizza" en el número de teléfono correspondiente, el sistema DNS traduce una dirección web como "www.google.com" en la dirección IP física, como "74.125.19.147", del ordenador que aloja ese sitio (en este caso, la página principal de Google).

Sitio web

Es un conjunto de uno o varios recursos web alojados en el mismo dominio

Servidor Web

Es un ordenador destinado a servir recursos web, alojar paginas, aplicaciones web y almacenar los datos de las mismas

Hosting

Es el espacio de almacenamiento alquilado en el servidor web de una compañía de alojamientos. Pones los archivos de tu sitio web en este espacio, y el servidor web suministra el contenido a los usuarios que lo solicitan.

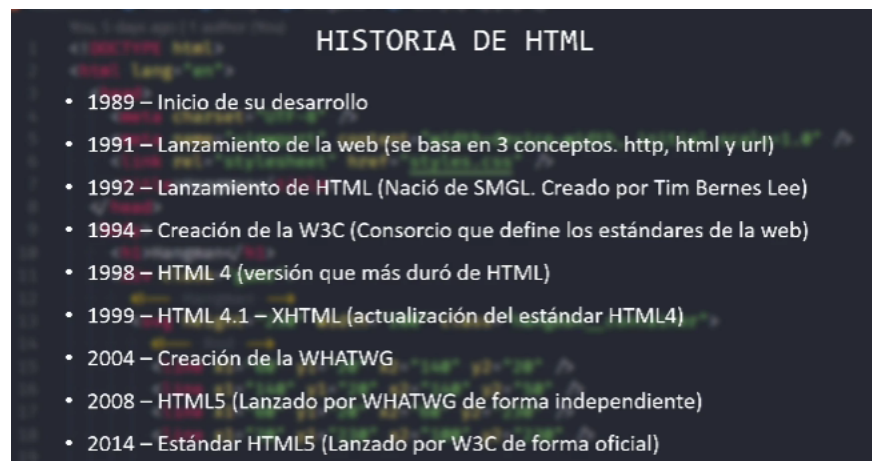
Peticion

es la accion de solicitar un recurso o informacion a un servidor web

HTML

- Es un lenguaje de marcado de Hipertexto (Hyper text markup language)
- HTML no es un lenguaje de programación es un lenguaje de estructura
- Es la base con la que estan creadas Todas las paginas web
- Su objetivo es decirle al navegador cual es la estructura del documento de la pagina web, que elementos tiene, como estan organizados
- HTML no debe ser utilizado para definir el aspecto visual

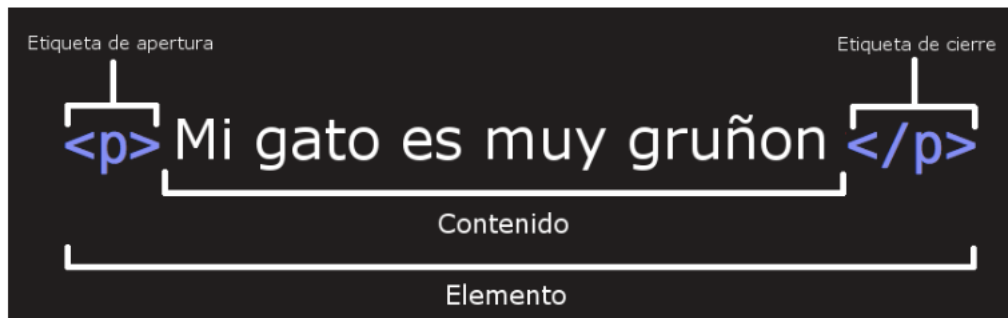
historia



Etiquetas de HTML5

Anatomía de un elemento HTML

Explora este párrafo en mayor profundidad.



Los elementos pueden también tener atributos, que se ven así:



```
<!DOCTYPE html> <!--Establece el estandar de HTML que va ser utilizado en el documento-->

<!--Establece el inicio del documento HTML-->
<html lang="es">
  <!--Head es utilizado para definir datos e informacion del sitio web al navegador-->
  <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>Mi primera pagina web</title>
  </head>
  <!--Representa todo el contenido de la pagina web-->
  <body>

  </body>

</html>
```

Headings y paragrahps

```
<!DOCTYPE html>
<html lang="en">
  <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>Titulos y parrafos</title>
  </head>

  <body>
```

```

<h1>harry potter</h1>
<h2>sinopsis</h2>
<p>
  Lorem, ipsum dolor sit amet consectetur adipisicing elit. Minima maiores fuga
  omnis odit minus eligendi incididunt officiiis possimus, voluptates ab nisi
  provident dolor non optio iusto modi? Impedit, ipsa adipisci!
</p>
<h2>Novelas</h2>
<p>
  Lorem ipsum dolor sit amet consectetur, adipisicing elit. Quis doloremque
  accusantium id natus pariatum necessitatibus, debitis porro? Laborum deserunt
  voluptatum minus dignissimos, unde dolore omnis reiciendis sunt optio accusantium
  amet.
</p>

<h3>Harry potter y la piedra filosofal</h3>
<p>
  Lorem ipsum dolor sit amet consectetur adipisicing elit. Minus voluptatem
  voluptatibus ea porro necessitatibus, quasi assumenda nesciunt unde provident est
  nisi, iure delectus, nobis optio praesentium consequuntur eveniet consectetur
  tempora.
</p>
</body>
</html>

```

etiquetas de contenido basicas

▼ header

Su contenido esta basado en el nombre de la pagina, el logo, redes, menu de navegacion

El elemento de HTML Header (<header>) representa un grupo de ayudas introductorias o de navegación. Puede contener algunos elementos de encabezado, así como también un logo, un formulario de búsqueda, un nombre de autor y otros componentes.

▼ main

Representa el contenido principal de la pagina

Nota: no debe haber más de un elemento <main> en un documento, y este no debe ser descendiente de un elemento

<article>, <aside>, <footer>, <header>, o <nav>.

▼ footer

Un pie de página típicamente contiene información acerca de el autor de la sección, datos de derechos de autor o enlaces a documentos relacionados.

Notas de uso:

- Encierra la información acerca del autor en un elemento <address> que puede ser incluido dentro del elemento <footer>.
- El elemento <footer> no es contenido de sección y en consecuencia no introduce una nueva sección en el esquema.

Section vs Article

cuando un conjunto de contenidos cada uno de ellos por individual no dependen el uno del otro se usa un article y si estan relacionados entre se implementa un section para contenerlos, como una seccion deportiva, una de actualidad, otra de chismes y asi

los section se usan para agrupar contenido relacionado

los article se usan para

section

El elemento de *HTML section* (`<section>`) representa una sección genérica de un documento. Sirve para determinar qué contenido corresponde a qué parte de un esquema. Piensa en el esquema como en el índice de contenido de un libro; un tema común y subsecciones relacionadas. Es, por lo tanto, una etiqueta semántica. Su funcionalidad principal es estructurar semánticamente un documento a la hora de ser representado por parte de un agente usuario. Por ejemplo, un agente de usuario que represente el documento en voz, podría exponer al usuario el índice de contenido por niveles para navegar rápidamente por las distintas partes.

Notas de uso:

- Si la intención es indicar el contenido de un elemento `<section>`, es mejor usar el elemento `<article>` en su lugar, a modo de artículos independientes como en las revistas. `<section>` está diseñado para contenidos dependientes, pero diferenciados.
- No se debe usar el elemento `<section>` como un mero contenedor genérico; para esto ya existe `<div>`, especialmente si el objetivo solamente es aplicar un estilo (CSS) a la sección. Como regla general, el título de una sección debería aparecer en el esquema del documento.

article

El *Elemento article de HTML* (`<article>`) representa una composición auto-contenida en un documento, página, una aplicación o en el sitio, que se destina a distribuir de forma independiente o reutilizable, por ejemplo, en la indicación. Podría ser un mensaje en un foro, un artículo de una revista o un periódico, una entrada de blog, un comentario de un usuario, un widget interactivo o gadget, o cualquier otro elemento independiente del contenido.

Notas de uso:

- Cuando los elementos `<article>` están anidados, los internos representan artículos relacionados con el exterior. Por ejemplo, los comentarios de un blog pueden ser elementos `<article>` anidados al que representa la entrada del blog.
- Los datos del autor de un elemento `<article>` pueden ser proporcionados a través del elemento `<address>`, pero no se aplica a los elementos `<article>` anidados.
- La fecha de publicación y el tiempo de un elemento `<article>` pueden ser descritos con el atributo `pubdate` atributo de un elemento `<time>`.

La etiqueta `<article>`:

La etiqueta del article especifica contenido independiente y autónomo.

Un artículo debe tener sentido por sí mismo y debería ser posible Distribuirlo independientemente del resto del sitio.

Fuentes potenciales para el elemento artículo:

- Posts de foro
- Blog post
- Noticias
- Comentarios

aside


El **elemento HTML** `<aside>` representa una sección de una página que consiste en contenido que está indirectamente relacionado con el contenido principal del documento. Estas secciones son a menudo representadas como barras laterales o como inserciones y contienen una explicación al margen como una definición de glosario, elementos relacionados indirectamente, como publicidad, la biografía del autor, o en aplicaciones web, la información de perfil o enlaces a blogs relacionados.

Notas de uso:

- No utilices el elemento `<aside>` para etiquetar texto entre paréntesis, ya que este tipo de texto se considera parte del flujo principal.

: The Aside element - HTML: HyperText Markup Language | MDN

The HTML element represents a portion of a document whose content is only indirectly related to the document's main content. Asides are frequently presented as sidebars or call-out boxes. Do not use the element to tag parenthesized text, as this kind of text is considered part of the main flow.

 <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/aside>



elementos de bloque y línea

Un elemento de bloque se caracteriza por ocupar todo el ancho disponible a si el contenido no lo ocupe por completo

```
<!DOCTYPE html>
<html lang="en">
  <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>Elementos de bloque y línea</title>
  </head>
  <body>
    <!--Un elemento de bloque se caracteriza por ocupar todo el ancho disponible a si el contenido no lo ocupe por completo-->
    <h1>Soy un elemento de bloque</h1>
    <p>soy otro el elemento de bloque</p>
    <span>Soy un elemento de línea </span>
    <a href="">soy otro elemento de línea</a>
    <span>Y por ultimo soy otro elemento de línea</span>
  </body>
</html>
```

Soy un elemento de bloque

Un elemento de bloque

`h1` 1350 × 37

Soy un elemento de línea [soy otro elemento de línea](#) Y por ultimo soy otro elemento de línea

Elementos de línea

```
<h1>Elementos de línea</h1>
  <!-- em -> enfasis
    strong -> importante
    small -> menos importante que el texto normal
    br -> Forzar un salto de línea
    wbr -> Salto de línea si hiciera falta
    time -> Se usa para representar un contenido de hora/fecha le da un sentido
           para navegador entender el valor de la fecha utilizando el atributo
           datetime=" "

    i -> Estilo italic
    b -> Estilo bold negrita
```

```

u      -> underline

-->
<p>
  <em>El dinero</em> es importante pero <strong>la salud</strong> es lo mas importante
  <br><small>Saludos</small>
</p>
<p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Molestias vel, vero voluptatum similique excepturi
nemo accusamus doloremque vitae sequi, modi sed praesentium temporibus,<br> assumenda totam placeat autem.
Totam, illum illo.
Earum ipsum ullam, laborum quas deserunt repellendus similique possimus iusto nobis soluta illum fuga! Esse
atque inventore saepe, ratione velit odio, dicta neque suscipit amet harum corrupti doloremque illum id.
</p>

<p>G:\Unidades
  compartidas\<br> MEGA DRIVE\<br>PROGRAMACION\<br>Aprende Laravel y desarrolla_una_app_de_pedidos_en_linea\<br>1.Introducción
</p>

<<time datetime="2001-05-15T19:00">15 de Mayo</time>

<p>
  <i>italica</i>
  <b>negrita o bold </b>
  <u>subrayado</u>
</p>
<p>4 elevado a la 3, es esto 4<sup>3</sup></p>
<p>formula del agua H<sub>2</sub></p>

```

Elementos de linea

El dinero es importante pero **la salud** es lo mas importante
Saludos

Lorem ipsum dolor sit amet consectetur adipisicing elit. Molestias vel, vero voluptatum similique excepturi nemo accusamus doloremque vitae sequi, modi sed praesentium temporibus, assumenda totam placeat autem. Totam, illum illo. Earum ipsum ullam, laborum quas deserunt repellendus similique possimus iusto nobis soluta illum fuga! Esse atque inventore saepe, ratione velit odio, dicta neque suscipit amet harum corrupti doloremque illum id.

G:\Unidades compartidas\ MEGA DRIVE\PROGRAMACION\Aprende Laravel y desarrolla_una_app_de_pedidos_en_linea\1.Introducción

2011-11-18 15 de Mayo de 2001

italica **negrita o bold** subrayado

4 elevado a la 3, es esto 4³

formula del agua H₂O

Atributos

Atributos

Los atributos son valores adicionales que configuran los elementos y/o ajustan su comportamiento.

En términos generales hay dos tipos de atributos

- **Comunes:**
Su sintaxis es -> atributo="valor"
- **Booleanos:**
Su sintaxis es -> atributo

Atributos globales importantes

class : Este atributo sirve para agrupar un conjunto de elementos definiendo un conjunto de características en común también para seleccionar ese conjunto de elementos que contengan una clase definida

un solo elemento puede tener varias clases separadas por espacio

```
<div class="contenedor recuadro block"></div>
<div class="contenedor parrafos free"></div>
<p class="free"></p>
```

id : Es un identificador unico de un elemento no debe repertirse

```
<p id="Parrafo_principal_1"> soy un parrafo con un id Parrafo_principal_1 </p>
```

title : este atributo muestra un mensaje mostrando la descripcion del elemento al que pertenece al pasar sobre el

```
<p title="soy un parrafo con una descripcion">MAS info</p>
```

data-* : nos permite guardar algun valor a nivel de la etiqueta html

```
<p id="parrafo-2" data-mensaje="Mira soy un mensaje de un data contenido en la etiqueta">
  Datos pasados en la etiqueta
</p>
```

ENLACES

también conocidos también por link popularmente su objetivo es conectar una página web con otra página web, con un recurso tanto interno como externo, o con otro sitio web.

Tienen el atributo `href` donde especificamos la ruta del recurso o sitio web que queremos obtener

También tiene el atributo target, que configura cómo queremos visualizar el recurso o sitio que solicitamos.

Rutas

Rutas absolutas :

Acceden a un recurso utilizando la dirección completa de la ubicación del archivo
externas :Tienen un protocolo http o https y son únicas en la red se suelen utilizar

para entender cómo aplicar las rutas

Rutas relativas

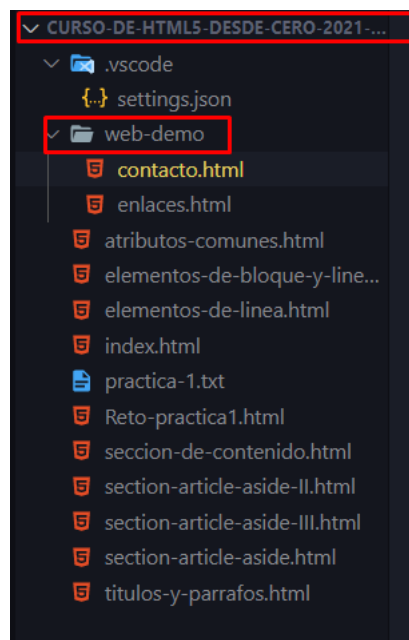
Las rutas relativas no utilizan la dirección completa, son relativas a la ubicación actual del archivo donde está siendo indicada la dirección del recurso, pueden ser relativas a la raíz del proyecto /

si el elemento al que queremos referenciar está en el mismo nivel podemos

- solo utilizar su nombre de archivo
- utilizar ./ y su nombre de archivo

si el elemento está a otro nivel podemos

- utilizar ../ para retroceder a un nivel anterior y desde ahí empezar a referenciar
- si se e



Atributos de los enlaces

target : Define donde se abra el recurso solicitado si es soportado por el user agent se abra en otra pestaña o en la misma de lo contrario el recurso será descargado , es recomendable el uso de `target="_blank"` , el valor por defecto es `"_self"`

Download : Podemos indicar mediante este atributo booleano la descarga de un recurso con la condición de que este tiene que estar ubicado dentro de nuestro servidor no puede ser externo

```
<h1>Home</h1>
<a href="/contacto.html" download target="_blank">Ir a Contacto</a>
<br>
<a href="https://www.google.com" download target="_blank">Ir a google</a>
<a href="/assets/images/husky.jpg" download>Descargar Husky</a>
<a href="/assets/images/husky.jpg" target="_blank">Ver Husky</a>
<a href="https://estaticos.muyinteresante.es/media/cache/1140x_thumb/uploads/images/gallery/59bbb29c5baf878503c9872/husky-siberiano-bosque" target="_blank">Ver Husky online
</a>
```

navegación con anclas

navegación a puntos específicos de otro documento

utilizamos la ruta del documento al cual queremos ir y con `#` y el `id` del elemento html al cual queremos referenciar

```
href="/paginas/blog.html#post-1"
```

```
< nav>
  <ul>
    <li><a href="/paginas/blog.html#post-1" target="_blank">Post 1</a></li>
    <li><a href="/paginas/blog.html#post-2" target="_blank">Post 2</a></li>
    <li><a href="/paginas/blog.html#post-3" target="_blank">Post 3</a></li>
    <li><a href="/paginas/blog.html#post-4" target="_blank">Post 4</a></li>
  </ul>
</nav>
```

Navegación a puntos específicos dentro del propio documento

usamos `#` y el `id` del elemento al cual queremos ir

```
<nav>
  <ul>
    <li><a href="#post-1">Post 1</a></li>
    <li><a href="#post-2">Post 2</a></li>
    <li><a href="#post-3">Post 3</a></li>
    <li><a href="#post-4">Post 4</a></li>
  </ul>
</nav>
```

Listas

las listas HTML tienen la función de listar contenido, En función del tipo de contenido podemos utilizar estos tipos de listas

▼ Lista desordenada

Se utilizan cuando el orden de los elementos dentro de la lista no tiene importancia

```
<h3>Lista de compra</h3>
<ul>
  <li>Pan</li>
  <li>Leche</li>
  <li>Huevo</li>
</ul>
```

Atributo type

En las listas desordenadas definine la manera en la cual sera representada la viñeta entre las cuales tenemos

- `type="disc"` es el disco relleno, es la representacio por defecto
- `type="square"` Es un cuadrado
- `type="circle"` Es un circulo vacio

▼ Lista ordenada

Se utilizan cuando el orden de los elementos dentro de la lista no tiene importancia o es una serie de pasos con un orden especificado, o una jerarquía

```
<h3>Orden de aprendizaje de tecnologias web</h3>
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ul>
```

Atributo type = ""

En las listas ordenadas definine la manera en la cual sera representado entre las cuales tenemos

- `type="A"` Letras Mayúsculas A, B, C, D...
- `type="a"` Letras Minúsculas a, ,b ,c ,d ...
- `type="I"` Números romanos en mayúscula I, II, III, IV, V ...
- `type="i"` Números romanos en minúscula i, ii, iii, iv, v ...
- `type="1"` Numeros naturales 1, 2, 3, 4, 5 es la numeracion establecida por defecto

Atributo start=""

El atributo start indica desde que punto de la numeracion se quiere empezar a contar el parametro a pasar en un numero entero sin importa el type que se esta implmentando en la lista es decir que si quiero que se empiece a contar desde 10 en numeros romanos seria

```
<ol type="I" start="5" reversed>
  <li>uno</li>
  <li>dos</li>
  <li>tres</li>
  <li>cuatro</li>
</ol>
```

Aprovecho para implementar el atributo **reversed** que se utiliza para que el orden sea una cuenta regresiva de mayor a menor

▼ <dl> </dl> Listas de definicion

Se utilizan para describir un conjunto de terminos como lo hacen los glosarios o los diccionarios

```
<h3>Lenguajes para el desarrollo web</h3>
<dl>
  <dt>HTML</dt>
  <dd>
    Es el lenguaje de marcado de hipertexto, es un lenguaje de marcado basado en etiquetas utilizado para
    definir la estructura basica
  </dd>
  <dt>CSS</dt>
```

```
<dd>Cascade Style Sheets: Es un lenguaje de diseño que se Utiliza para dar estilos al Html</dd>
<dt>JavaScript</dt>
<dd>Es un lenguaje de programacion para dar interactividad a un sitio</dd>
</dl>
```

▼ Listas anidadas

```
<h2>listas anidadas</h2>
<h3>lista principal</h3>
<ul>
  <li>
    <h3>lista anidada</h3>
    <ol>
      <li>elemento de la anidada</li>
      <li>elemento elemento de la anidada</li>
    </ol>
  </li>
  <li>elemento de la principal 1</li>
  <li>elemento de la principal 2</li>
</ul>
```

Tablas

Las tablas en html sirven para mostran contenido tabulado cuya finalidad es mostrar un contenido en una estructura de tabla y que no exista una manera mejor de ser representado

Estructura Básica

La estructura basica de una tabla esta compuesta por

▼ Table :

La etiqueta `<table>` encierra y representa todo el contenido de la tabla

▼ Tr :

La etiqueta table row `<tr>` encierra el contenido de una fila

▼ Td

La etiqueta table data `<td>` constituye una celda de un dao

cada fila `<tr>` esta constituida por un conjunto de elementos `<td>`, el numero de elementos `<tr>` dentro de la fila constituye el numero de columnas

▼ Caption

Para poder establecer titulos delas tablas se utiliza la etiqueta `<caption>` es una etiqueta opcional y se coloca justo despues de la etiqueta `<table>`

▼ Thead

contiene los titulos de las cabeceras de cada columna

▼ Th

la etiqueta table head `<th>` representan Las cabeceras de las columnas tablas se establecen con la etiqueta dentro de la etiqueta `<thead>`

▼ Tbody

Contiene el cuerpo del contenido de la tabla

```
<table>
  <caption>Titulo de la tabla</caption>
  <thead>
    <tr>
      <th></th>
      <th>L</th>
      <th>M</th>
      <th>M</th>
      <th>J</th>
      <th>V</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td></td>
      <td></td>
      <td></td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td></td>
      <td></td>
      <td></td>
    </tr>
  </tfoot>
</table>
```

Atributos de las tablas

rowspan : permite que una celda ocupe mas de una columna, es decir ocupe mas espacio horizontalmente

```
<td colspan="5">RECREO</td>
```

horario de Clases					
	L	M	M	J	V
8:30 - 9:30	Matemáticas	Matemáticas	Sociales	Matemáticas	Matemáticas
9:30 - 10:25	Lengua	Lengua	Matemáticas	Ciencias	Lengua
10:25 - 11:20	Edu. Física	Ciencias	Lengua	Lengua	Edu. Física
11:20 - 11:45	RECREO				
11:45 - 12:40	Inglés	Sociales	Inglés	Tutoría	Inglés
12:40 - 13:35	Musica	Edu. Física	Plástica	Religión	Musica
13:35 - 14:30	Tecnología	Francés	Tecnología	Francés	Tecnología
Total de asignaturas	12				

colspan : permite que una celda ocupe mas de una fila, es decir el espacio que ocupa verticalmente

```
<td rowspan="5">RECREO</td>
```

Tablas

horario de Clases						
	L	M	M	J	V	
8:30 - 9:30	Matemáticas	Matemáticas	Sociales	Matemáticas	Matemáticas	
9:30 - 10:25	Lengua	Lengua	Matemáticas	Ciencias	Lengua	
10:25 - 11:20	Edu. Física	Ciencias	Lengua	Lengua	Edu. Física	
11:20 - 11:45	RECREO	Inglés	Sociales	Inglés	Tutoría	Inglés
11:45 - 12:40		Musica	Edu. Física	Plástica	Religión	Musica
12:40 - 13:35		Tecnología	Francés	Tecnología	Francés	Tecnología
13:35 - 14:30						
Total de asignaturas	12					

Selección de columnas

Cuando necesitamos seleccionar una columna tenemos las etiquetas son: `<colgroup>` y `</colgroup>` (la de cierre es opcional). que nos permite seleccionar una columna en concreto.

Dentro pondremos tantas etiquetas `<col>` como columnas tengamos, cada etiqueta `<col>` equivaldrá a una columna siguiendo el mismo orden que tienen en la tabla

```
<colgroup style="background-color: rgb(205, 230, 67);">
  <col> <!-- Representa la seleccion de la primer columna -->
  <col> <!-- Representa la seleccion de la segunda columna -->
</colgroup>
```

Usando span en colgroup

```
<colgroup span="2" style="background-color: aqua;">
  <!-- Representa la seleccion de la tercera y cuarta columna por el atributo span="2" -->
</colgroup>
```

Si necesitamos que la etiqueta `<col>` agrupe mas de una columna, tenemos el atributo `span=" "`, que funciona exactamente igual que `rowspan` y `colspan`

```
<colgroup style="background-color: coral;">
  <col span="2">
</colgroup>
```

Resultados

- las primeras 2 columnas agrupadas con un colgroup y 2 etiquetas col son de color verde
- las 2 columnas siguientes azules es el resultado de un colgroup con atributo span=2
- las 2 ultimas de color coral es resultado de un colgroup con una col de atributo span=2

horario de Clases					
	L	M	M	J	V
8:30 - 9:30	Matemáticas	Matemáticas	Sociales	Matemáticas	Matemáticas
9:30 - 10:25	Lengua	Lengua	Matemáticas	Ciencias	Lengua
10:25 - 11:20	Edu. Física	Ciencias	Lengua	Lengua	Edu. Física
11:20 - 11:45	RECREO				
11:45 - 12:40	Inglés	Sociales	Inglés	Tutoría	Inglés
12:40 - 13:35	Musica	Edu. Física	Plástica	Religión	Musica
13:35 - 14:30	Tecnología	Francés	Tecnología	Francés	Tecnología
Total de asignaturas	12				

Otras etiquetas de importantes

Address

La etiqueta `<address>` proporciona información de contacto para una persona o personas, o para una organización.
`<address>`

Reset

HTML

CSS

```

1 <p>Contact the author of this page:</p>
2
3 <address>
4   <a href="mailto:jim@rock.com">jim@rock.com</a><br>
5   <a href="tel:+13115552368">(311) 555-2368</a>
6 </address>
7

```

Output

Contact the author of this page:

jim@rock.com
[\(311\) 555-2368](tel:+13115552368)

La información de contacto proporcionada por `<address>` el contenido de un elemento puede tomar cualquier forma que sea apropiada para el contexto y puede incluir cualquier tipo de información de contacto que sea necesaria, como una dirección física, URL, dirección de correo electrónico, número de teléfono, identificador de redes sociales, coordenadas geográficas, Etcétera. El elemento `<address>` debe incluir el nombre de la persona, personas u organización a la que se refiere la información de contacto.

- Solía ser el caso que `<address>` se suponía que un elemento solo se usaba para representar la información de contacto del autor del documento. Sin embargo, en las últimas versiones de especificaciones, su definición se ha actualizado, por lo que ahora se puede utilizar para marcar direcciones arbitrarias.
- Este elemento no debe contener más información que la información de contacto, como una fecha de publicación (que pertenece a un `<time>` elemento).
- Normalmente, un `<address>` `<footer>`

elemento se puede colocar dentro del elemento de la sección actual, si existe.

<blockquote>

Es un bloque que se utiliza para citar a otros autores o documentos se puede incluir el atributo cite para poner un enlace al documento original o fuente

```
<blockquote cite='http://html.conclase.net/w3c/html401...def-BLOCKQUOTE'>
  <p>
    <strong>Nota.</strong> Recomendamos que las implementaciones de hojas
    de estilo proporcionen un mecanismo para insertar signos de puntuación de citas
    antes y después de una cita delimitada por un BLOCKQUOTE de un modo apropiado según
    el contexto del idioma actual y el grado de anidamiento de las citas.
  </p>
</blockquote>
```

pre

la etiqueta pre se utiliza para escribir texto y se muestre tal cual como lo escribimos

div

se usa como una división del documento, semánticamente no significa nada, es un contenedor genérico, generalmente se utiliza para dar estilos a través de CSS o para usar algo denominado "Delegación de eventos" en JavaScript

hr

horizontal rule, es una línea horizontal que sirve para indicar un cambio de tema

Span

span -> Es un contenedor de línea, equivalente a div, se suele usar para encerrar palabras o pequeños textos y darles estilo a través de CSS o localizarlos desde javascript. Semánticamente no significa nada

```
<p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Corrupti, <span>dolorem</span>
asperiores quo mollitia tempora voluptatum quos necessitatibus a suscipit accusantium neque
cupiditate veniam laboriosam voluptatem animi ratione explicabo assumenda consequatur!
</p>
```

q

q -> Es el equivalente a blockquote, significa quote, por eso el de bloque se llama block - quote. Sirve para poner citas pero en línea.


```
<p>
  uno de los super heroes mas queridos por los fanaticos es iroman el es interpretado por <q>Robert downie
  jr</q>
</p>
```

code

code -> Sirve para encerrar código que queremos representar visualmente, suele ir unido con la etiqueta pre.

```
<pre>
  <code>
    span{
      color:brown;
      font-size:2rem;
    }
  </code>
</pre>
```

Caracteres especiales ascii

Códigos HTML - Tabla de caracteres y símbolos

Tabla de ASCII estándar, nombres de entidades HTML, ISO 10646, ISO 8879, ISO 8859-1 romano 1 Soporte para browsers: todos los browsers HTML 4.01, ISO 10646, ISO 8879, alfabeto romano extendido A y B, Soporte para browsers: Internet Explorer > 4, Netscape > 4

<https://ascii.cl/es/codigos-html.htm>

Formularios

Estructura básica de un formulario

La estructura básica de un formulario se compone de 4 elementos

form -> es la etiqueta que engloba nuestro formulario

label -> Sirve para escribir el nombre del campo a rellenar

input -> sirve para crear un campo que permitirá al usuario interactuar

button -> crea un botón que permitirá enviar el formulario

```
<form>
  <label>Nombre</label>
  <input type="text" >
  <button>Enviar formulario</button>
</form>
```

Asociar Inputs y Label

```
<form>
<!-- FORMA MAS USADA ATRIBUTO FOR EN EL LABEL RELACIONAR CON EL ID DEL INPUT -->
```

```

<label for="nombre">Nombre</label>
<input type="text" id="nombre" name="nombre">

<!-- FORMA 2 PONER EL INPUT EN EL CONTENIDO DEL LABEL-->
<label>
  Apellidos
  <input type="text">
</label>

<button>Enviar formulario</button>
</form>

```

Button vs input type=" button"

La principal diferencia entre ambos radica en que el input type button NO TIENE la capacidad de al ser clickeado de enviar el formulario y recargar la pagina, su uso puede estar dentro del formulario para ejecutar algun calculo JavaScript

Mientras que la etiqueta <button> tiene la capacidad de enviar el formulario y recargar la pagina

Si quieres que un input pueda enviar el formulario tiene que ser de tipo `type="submit"`

```

<form >
  <input type="text" placeholder="valor">
  <button> button</button>
  <input type="button" value="type button ">
  <input type="submit" value="enviar con submit">
</form>

```

Inputs de tipo fecha

```

month -> Se utiliza para introducir un mes
color: -> Se utiliza para especificar un color
email -> Se utiliza para introducir un email
number -> Se utiliza para valores numéricos
password -> Se utiliza para contraseñas
range -> Se utiliza para establecer un rango
reset -> Se utiliza para resetear el formulario
search -> Se utiliza para las barras de búsqueda
tel -> Se utiliza para introducir números telefónicos
text -> Valor por defecto
url -> Se utiliza para introducir URLs
week -> Se utiliza pra introducir el número de semana del año

```

Inputs de tipo fecha

▼ date

utilizado para capturar fechas día-mes-año

▼ datetime-local

Utilizado para capturar fechas y horas , pero tiene un problema no es soportado por todos los navegadores

▼ time

permite la entrada de horas

01:33 p. m. 🕒

▼ month

Permite la entrada de un mes y un año

febrero de 2021 📅

▼ week

Permite la entrada de una semana y un año

Semana 26, 2021 📅

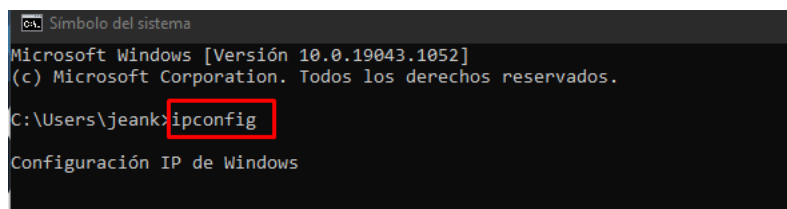
```
<form>
  <input type="date">
  <input type="datetime-local">
  <input type="date">
  <input type="time">
  <input type="month">
  <input type="week">
</form>
```

Inputs para móviles

Estas etiquetas tienen un comportamiento específico en el teclado en dispositivos móviles

```
<form>
  <div><label>Search <input type="search"></label></div>
  <div><label>telefono <input type="tel"></label></div>
  <div><label>correo <input type="email"></label></div>
  <div><label>contraseña <input type="password"></label></div>
  <div><label>url <input type="url"></label></div>
  <div><label>archivo <input type="file"></label></div>
</form>
```

lanzar en nuestro movil



```
C:\> Símbolo del sistema
Microsoft Windows [Versión 10.0.19043.1052]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\jeank> ipconfig

Configuración IP de Windows
```

```

Adaptador de LAN inalámbrica Wi-Fi:
Sufijo DNS específico para la conexión. . . : www.tendawifi.com
Vínculo: dirección IPv6 local. . . : fe80::ac64:5e73:a7d5:81af%18
Dirección IPv4. . . . . : 192.168.2.101
Máscara de subred . . . . . : 255.255.255.0
Puerta de enlace predeterminada . . . . . : 192.168.2.1

```

127.0.0.1:5500/web-demo/paginas/formularios/tipo-de-inputs.html

127.0.0.1 equivale a nuestra dirección ip en nuestra red local, luego tendríamos que ingresar el puerto y la url en el telefono luego de la ip

<http://192.168.2.101:5500/web-demo/paginas/formularios/tipo-de-inputs.html>

otros tipos de inputs

estos no pertenecen a un grupo o características que los puedan agrupar

```

color: -> Se utiliza para especificar un color
number -> Se utiliza para valores numéricos
range -> Se utiliza para establecer un rango
reset -> Se utiliza para resetear el formulario
text -> Valor por defecto

```

Mas tipos de inputs

color number range reset Restablecer texto texto

Inputs de Seleccion

▼ radio

- nos permiten seleccionar un valor dentro de un grupo de valores seleccionando un solo valor
- son relacionados entre si con el atributo `name` tomando el mismo valor
- si queremos que por defecto sea seleccionado uno se utiliza el atributo `checked`
- `value=""` permite al servidor donde es enviado los valores del formulario saber el valor del radio button seleccionado

```
<form>
  <h3>Genero</h3>
  <input type="radio" name="genero" id="generoM" value="Masculino">
  <label for="generoM">Masculino</label>
  <input type="radio" name="genero" id="generoF" value="Femenino">
  <label for="generoM">Femenino</label>
  <input type="radio" name="genero" id="generoD" value="Desconocido" checked>
  <label for="generoM">Desconocido</label>
</form>
```

Genero

☐ Masculino ☐ Femenino ☒ Desconocido

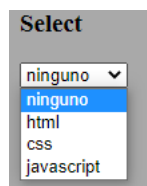
▼ Checkbox

- permite la selección múltiple de opciones
- Se relacionan entre si con el atributo `name`
- el atributo `value` determina el valor que sera enviado al servidor en caso de ser seleccionado
- si queremos que por defecto sea seleccionado uno se utiliza el atributo `checked`

```
<h3>checkbox</h3>
<form>
  <label>HTML<input type="checkbox" name="language" value="html"></label>
  <label>CSS<input type="checkbox" name="language" value="css"></label>
  <label>JavaScript<input type="checkbox" name="language" value="javascript"></label>
</form>
```

▼ Select

- permite la selección de un elemento dentro de una lista de opciones



- las opciones son declaradas con `option` y dentro el atributo `value`
- con el atributo `multiple` permite la selección de varias opciones de la lista utilizando la tecla ctrl



```

<h3>Select</h3>
<form>
  <Select name="lenguajes">
    <option value="">ninguno</option>
    <option value="html">html</option>
    <option value="css">css</option>
    <option value="javascript">javascript</option>
  </Select>
</form>

```

▼ Select Avanzado

Si tenemos muchas opciones podemos ordenarlas por categorías a través de la etiqueta `<optgroup>` con el atributo label para nombrar la categoría

```

<form>
  <h4>Select avanzado</h4>
  <label>Mascota
    <select name="mascotas">
      <optgroup label="Aves">
        <option value="loro">Loro</option>
        <option value="canario">Canario</option>
        <option value="guacamaya">Guacamaya</option>
      </optgroup>
      <optgroup label="4 patas">
        <option value="perro">Perro</option>
        <option value="ardilla">Ardilla</option>
        <option value="gato">Gato</option>
      </optgroup>
      <optgroup label="Reptil">
        <option value="toturga">Tortuga</option>
        <option value="lagarto">lagarto</option>
        <option value="serpiente">serpiente</option>
      </optgroup>
    </select>
  </label>Mascota
</form>

```

Otros inputs

▼ Datalist

- son muy similares a los select, pero tiene un filtro que permite buscar entre la lista de opciones
- se define un input con el atributo `list="ID_Datalist"` con el id del datalist esto permitira hacer el filtro

- el datalist es declarado y contiene un conjunto de `option` en el cual su atributo `value` será el visualizado en el datalist, si se tiene un contenido en el `option` puede ser usado como descripción del `value`

```
<h2>DataList</h2>
<form>
  <input list="mascotas">
  <datalist id="mascotas">
    <option value="loro"></option>
    <option value="canario"></option>
    <option value="guacamaya"></option>
    <option value="perro"></option>
    <option value="ardilla"></option>
    <option value="gato"></option>
    <option value="toturga"></option>
    <option value="lagarto"></option>
    <option value="serpiente"></option>
    <option value="sapo"></option>
  </datalist>
</form>
```

▼ Fieldset y legend

`fieldset` -> Se utiliza para agrupar elementos dentro de un formulario
`legend` -> Representa una etiqueta para el contenido del `fieldset`

```
<fieldset>
  <legend>Datos personales</legend>
  <label for="name"></label>
  <input type="text" id="name" spellcheck="true">
  <label for="surname"></label>
  <input type="text" id="surname">
</fieldset>
```

▼ File

Utilizado para seleccionar archivos del sistema

```
<label for="file"></label>
<input type="file" name="file" id="file">
```

▼ meter

representa un valor dentro de un rango conocido

```
<meter id="fuel" min="0" max="100" low="20" optimum="60" high="80" value="36">
</meter>
```

▼ Progress

representa el progreso de una tarea

establecemos un valor `max` y un `value` y la barra representara el % completado según el valor máximo y el valor definido



```
<label for="tarea">Tarea 1</label>
<progress id="tarea" value="75" max="100"></progress>
```

▼ Textarea

- es un campo de texto multilinea
- las `cols` definen el ancho del area y las `rows` el largo verticalmente
- a traves de css podemos modificar estos tamaños y bloquear que no se pueda modificar por el usuario



```
<label for="mensaje">Mensaje</label>
<textarea name="mensaje" id="mensaje" cols="30" rows="10"></textarea>
```

Atributos comunes de formularios

```
placeholder -> Da una pista de lo que el usuario tiene que
introducir
required -> Hace que un campo sea obligatorio
readonly -> Hace que un campo sea de solo lectura
min - max -> Establece el mínimo y máximo de un campo numérico
maxlength - minlength -> Establece el minimo y máximo de
caracteres de un campo de texto
selected -> Equivale a checked en los select, sirve para
establecer una opción por defecto
disabled -> Desactiva el campo, no se podrá escribir en el
autofocus
```

Contenido embebido

Es todo aquel contenido que nos traemos desde afuera

Imágenes

IMÁGENES

- Los formatos de imágenes para web los podemos clasificar en 2 tipos
 - Vectoriales
 - svg (recomendado siempre que se pueda)
 - Mapa de bits
 - jpg
 - png 8 y 24 (si necesitáis transparencias)
 - gif (si necesitáis una imagen animada)
 - webp (el formato que menos pesa)

- cada que sea posible en iconos, logos y otras imagenes usar svg que no pierde calidad y no agrega peso a la pagina
- tambien es de notar que no todas las imagenes pueden ser vectorizadas
- las imagenes de mapa de bits estan conformadas de pixels, a mayor es el tamaño de la imagen mayor es la cantidad de pixels, si acercamos en una imagen normal podremos ir notando la diferencia

jpg

- es el formato mas comprimido
- no soporta animaciones ni transparencias

Png

- Soportan transparencias

gif


- soportan ser una imagen animada

▼ Webp

- entre mayor resolución mayor es el peso y este formato permite reducir mucho el peso de las imagenes sin perder casi calidad
- soporta transparencias y animaciones

Convertir imágenes a WebP

WebP ha sido desarrollado por Google para habilitar la creación de imágenes más pequeñas y de mejor calidad que aceleren la navegación. Un fichero WebP consiste de datos de imagen VP8 y de un contenedor basado en RIFF. Es un nuevo formato de imagen que utiliza una compresión con pérdida

 <https://imagen.online-convert.com/es/convertir-a-webp>



```
<h2>Insertar imagenes</h2>



```

Device pixel Ratio


```
DPR (Device Pixel Ratio)
Es la relación que existe entre los píxeles reales que tiene el
dispositivo y los píxeles que tenemos disponibles para "pintar" contenido

DPR = píxeles reales / píxeles disponibles
```

viewport es el espacio disponible que tenemos para dibujar

What is my viewport

What Is My Viewport - A simple online tool for quickly finding the dimensions of your current device's viewport!

 <https://whatismyviewport.com/>

Para empezar, existen 2 tipos de píxeles:

- Píxeles físicos: son los píxeles reales que están en un monitor, smartphone, tv, etc. Por ejemplo, cuando se habla de 1920×1080 (Full HD), se refiere siempre a una pantalla de 1920 píxeles FÍSICOS de ancho y 1080 FÍSICOS de alto.
- Píxeles CSS (o del viewport): es la unidad de medida con la que trabaja CSS para pintar elementos en el viewport, y NO siempre son del mismo tamaño que un píxel físico.

Ahora bien, ¿Qué es el Device Pixel Ratio (DPR)? el DPR es el ratio entre los píxeles físicos y los píxeles CSS. Veamos algunos ejemplos para que sea más fácil de entender:

- Si mi DPR es de 1, el tamaño de 1 píxel CSS será de 1 píxel físico, por lo que ambos abarcarán el mismo espacio en pantalla.
- Si mi DPR es de 2, el tamaño de 1 píxel CSS será de 4 píxeles físicos (2 píxeles físicos de ancho x 2 píxeles físicos de alto).
- Si mi DPR es de 3, el tamaño de 1 píxel CSS será de 9 píxeles físicos (3 píxeles físicos de ancho x 3 píxeles físicos de alto).

El valor del DPR también puede ser decimal, como 2,5 o 2,75.

Sabiendo lo anterior, hay que tener en cuenta que cuando manejamos medidas `px` en HTML y CSS estamos hablando SIEMPRE de píxeles CSS (o del viewport) y NUNCA de píxeles físicos.

Entonces, si tenemos por ejemplo un DPR de 1 e insertamos una imagen de 300×300 píxeles CSS en el navegador (como podría ser con ``), esta imagen terminará ocupando 300×300 píxeles físicos. A su vez, si tenemos un DPR de 3 e insertamos una imagen de 300×300 píxeles CSS en el navegador, esta imagen terminará ocupando 900×900 píxeles físicos, o sea, 3 veces más píxeles que los que definimos por código.

Como dato a tener en cuenta, la mayoría de los smartphones modernos tienen un DPR de 2,75 o 3, ya que si tuviesen un DPR de 1 todos los elementos se verían muy pequeños en sus pantallas Full HD de sólo 5" o 5.5". Por otra parte, en pc el DPR por defecto es de 1, pero si hacemos zoom en el navegador el DPR aumentará a la par (con un zoom de 150% el DPR será de 1,5, con un zoom de 200% el DPR será de 2, y así). De hecho, pueden probar esto muy fácilmente en: <https://www.mydevice.io/>.

Para leer un poco más sobre el tema: <https://tomroth.com.au/dpr/>.

otra definicion interesante es

<https://binaria.com/blog/device-pixel-ratio-consejos-para-desarrolladores/#:~:text=En el mundo del desarrollo,no a la resolución física.>

En el mundo del desarrollo web, DPR (también llamado CSS Pixel Ratio) es lo que determina cómo la resolución de la pantalla de un dispositivo es interpretada por CSS. De esta forma, todas las propiedades CSS relacionadas con las dimensiones de los elementos responderá a la resolución lógica y no a la resolución física.

La razón por la que se creó el DPR o CSS pixel ratio fue el aumento de la resolución en los dispositivos móviles. Con el aumento de la densidad de pixels físicos, si todos los dispositivos tuvieran un DPR de 1 las páginas se renderizarían demasiado pequeñas.

Tomemos como ejemplo un Samsung Galaxy S4

- Resolución física: 1080 x 1920
- Device pixel ratio: 3
- Resolución lógica: $(1080/3) \times (1920/3) = 360 \times 640$

Cuando visualizamos una web con este dispositivo, el CSS interpretará que el dispositivo tiene una resolución de 360×640 y las `@media` queries responderán a dicha resolución. Pero los elementos renderizados en la pantalla tendrán el triple de definición que una pantalla con una resolución física de 360×640.



iPhone 3G
Pixels físicos 320×480
Device Pixel Ratio **1**
Pixels lógicos $(320/1) \times (480/1) = 320 \times 480$



iPhone 6
Pixels físicos 750×1334
Device Pixel Ratio **2**
Pixels lógicos $(750/2) \times (1334/2) = 375 \times 667$

Las Imágenes y DPR

Supongamos que nuestro site tiene un logo en formato JPEG de 60×60 pixels. En un dispositivo con un DPR de 1 (por ejemplo, un portátil) la imagen se mostrará con el ratio real 1:1, sin embargo en un dispositivo con un ratio mayor (un smartphone o tablet), necesariamente la imagen se escalará para ocupar el mismo espacio que en un dispositivo con device pixel ratio 1. En el caso de nuestro logo 60×60 en un dispositivo con DPR 2 la imagen se escalará x2 cubriendo un área de 120×120 pixels. En la mayoría de casos ese escalado afectará negativamente a la calidad de la imagen, por eso nos encontramos que al navegar con el móvil las imágenes de algunas webs no se ven bien.



Device pixel ratio: consejos para desarrolladores

27 de julio de 2015

¿Por qué cuando navego con mi smartphone de última generación el texto se ve tan bien y las imágenes tan mal? Si mi smartphone tiene más pixels que mi TV de 55", ¿por qué en el móvil veo la versión mobile simplificada de los sitios web? ¿Por qué en mi iPhone 6 de 700€ parece que cabe menos información que en un Android normalito?

Si alguna vez te has hecho estas preguntas o si como desarrollador web quieres estar preparado para cuando alguien te asalte con una de ellas, deja que te explique algo.

¿Qué es el Device pixel ratio (DPR)?

Es el ratio entre los pixels lógicos y los pixels físicos de un dispositivo.

Llamamos pixel físico a uno de los miles de puntos (hardware) que forman la pantalla de un dispositivo. Un pixel lógico, sin embargo, no tiene dimensiones específicas, sino que están establecidas por software por el fabricante. De esta manera en un dispositivo con device pixel ratio 1, un pixel lógico ocupará un pixel físico, en cambio en un dispositivo con device pixel ratio 2 un pixel lógico ocupará un área de la pantalla de 2x2 pixels físicos.

Podríamos comparar el valor del DPR con un lápiz, cuanto más alto sea el valor más fina será su punta. Pero eso no significa necesariamente que vayamos a hacer la letra más pequeña.

En el mundo del desarrollo web, DPR (también llamado CSS Pixel Ratio) es lo que determina cómo la resolución de la pantalla de un dispositivo es interpretada por CSS. De esta forma, todas las propiedades CSS relacionadas con las dimensiones de los elementos responderá a la resolución lógica y no a la resolución física.

La razón por la que se creó el DPR o CSS pixel ratio fue el aumento de la resolución en los dispositivos móviles. Con el aumento de la densidad de pixels físicos, si todos los dispositivos tuvieran un DPR de 1 las páginas se renderizarían demasiado pequeñas.

Tomemos como ejemplo un Samsung Galaxy S4

- Resolución física: 1080 x 1920
- Device pixel ratio: 3
- Resolución lógica: $(1080/3) \times (1920/3) = 360 \times 640$

Cuando visualizamos una web con este dispositivo, el CSS interpretará que el dispositivo tiene una resolución de 360x640 y las `@media` queries responderán a dicha resolución. Pero los elementos renderizados en la pantalla tendrán el triple de definición que una pantalla con una resolución física de 360x640.





Un tamaño standard de monitor de escritorio es de 24" con una resolución de 1920×1080. Imagina que encogemos ese monitor a un tamaño inferior a las 5" manteniendo la misma resolución. Los elementos mostrados se verían demasiado pequeños.

Pero en un dispositivo con una densidad de pixels muy alta no necesariamente significa que vayamos a ver nuestro contenido más pequeño, porque el renderizado de la página estará siempre sujeto al DPR del dispositivo.

Las Imágenes

Supongamos que nuestro site tiene un logo en formato JPEG de 60×60 pixels. En un dispositivo con un DPR de 1 (por ejemplo, un portátil) la imagen se mostrará con el ratio real 1:1, sin embargo en un dispositivo con un ratio mayor (un smartphone o tablet), necesariamente la imagen se escalará para ocupar el mismo espacio que en un dispositivo con device pixel ratio 1. En el caso de nuestro logo 60×60 en un dispositivo con DPR 2 la imagen se escalará x2 cubriendo un área de 120×120 pixels. En la mayoría de casos ese escalado afectará negativamente a la calidad de la imagen, por eso nos encontramos que al navegar con el móvil las imágenes de algunas webs no se ven bien.



Soluciones a la pérdida de calidad debido al DPR

Existen varias soluciones a este problema:

El atributo `srcset` para las imágenes o la etiqueta `<picture>` nos permiten, entre otras cosas, definir múltiples orígenes para un solo elemento de imagen y el DPR en el que se deberían mostrar, pero aún no disponen del soporte necesario para considerarse una alternativa.

```
<h2>insertar imagenes segun el DPR(Device Pixel Ratio)</h2>

```

picture

vale la pena mencionar que es una manera moderna que la anterior

debe usarse una etiqueta img dentro de la etiqueta picture la cual utiliza de seguridad para mostrar algo en caso que no se cumplan las condiciones anteriores o no lo soporte

se pueden declarar varias fuentes y condiciones con cada etiqueta source

```
<picture>
  <!-- <source srcset="
    ./assets/imagenes/abstract-3d-hd-wallpapers.webp ,
    ./assets/imagenes/wallpapers-mobile-400.webp 3x"> -->
  <source media="(max-width:700px )"
    srcset="./assets/imagenes/wallpapers-mobile-400.webp ,
    ./assets/imagenes/wallpapers-mobile-400.jpg ">
  
</picture>
```

La más popular, sin embargo, pero la menos eficiente es la de cargar imágenes mayores y forzar por código su altura y anchura. De esta manera nos aseguraremos una buena visualización en dispositivos con mayor densidad de pixels pero supondrá una carga extra de datos para los dispositivos con DPR 1.

Adapta tu web a pantallas Retina y otras de alta resolución

Los nuevos productos de Apple tienen una gran densidad de píxeles, o lo que es lo mismo una alta resolución por espacio físico. Le dieron el nombre comercial de Retina, ya que anunciaron que sus pantallas ya tenían el número

<https://www.tictacsoluciones.com/blog/adapta-tu-web-a-pantallas-retina-y-otras-de-alta-resolucion/>



- CSS | MDN

El tipo de dato CSS , usado en media queries, define la densidad de píxeles de un dispositivo de salida, su resolución. Es inmediatamente seguido por una unidad de resolución (dpi, dpcm, ...). Como para cualquier dimensión CSS, no hay espacio entre la unidad literal y el número.

<https://developer.mozilla.org/es/docs/Web/CSS/resolution>



CSS media queries - CSS | MDN

Las media queries (en español "consultas de medios") son útiles cuando deseas modificar tu página web o aplicación en función del tipo de dispositivo (como una impresora o una pantalla) o de características y parámetros específicos (como la resolución de la pantalla o el ancho del viewport del navegador).

https://developer.mozilla.org/es/docs/Web/CSS/Media_Queries/Using_media_queries

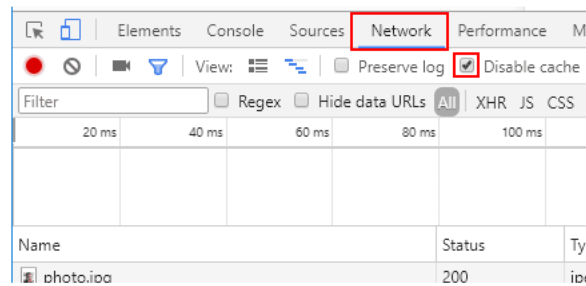


CSS

En el caso de las imágenes cargadas como `background` en nuestros estilos CSS disponemos de `@media` queries que nos permiten cargar imágenes de mayor o menor calidad según el DPR del dispositivo. A diferencia del atributo `srcset` o de la etiqueta `<picture>`, esta técnica ya consta del soporte necesario para considerarse una solución real.

¿Tienes dudas sobre cómo trabajar correctamente el device pixel ratio? Puedes comentarlas con nosotros en los comentarios de abajo.

Deshabilitar la cache



Audio

Es importante especificar atributos como controls , autoplay , muted y loop

tambien es importante remarcar los formatos mas utilizados en la web el mp3 y el ogg

```
<audio src="./assets/audio/audio.mp3" controls></audio>

<audio controls autoplay>
  <source src="./assets/wy.mp3" type="audio/mpeg" />
  <source src="./assets/wy.ogg" type="audio/ogg" />
  Tu navegador no soporta el formato de audio
</audio>
```

video

El `controls` atributo agrega controles de video, como reproducción, pausa y volumen.

Es una buena idea incluir siempre `width` y `height` atributos. Si no se configuran el alto y el ancho, la página puede parpadear mientras se carga el video.

El `<source>` elemento le permite especificar archivos de video alternativos entre los que el navegador puede elegir. El navegador utilizará el primer formato reconocido.

El texto entre el `<video>` y `</video>` etiquetas sólo se mostrará en los navegadores que no soportan el `<video>` elemento.

```
<video width="320" height="240" autoplay>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Favicon

<https://www.favicon-generator.org/>