

# INTRODUCCION AL DESARROLLO WEB



# COMPONENTES DE UNA APLICACIÓN WEB



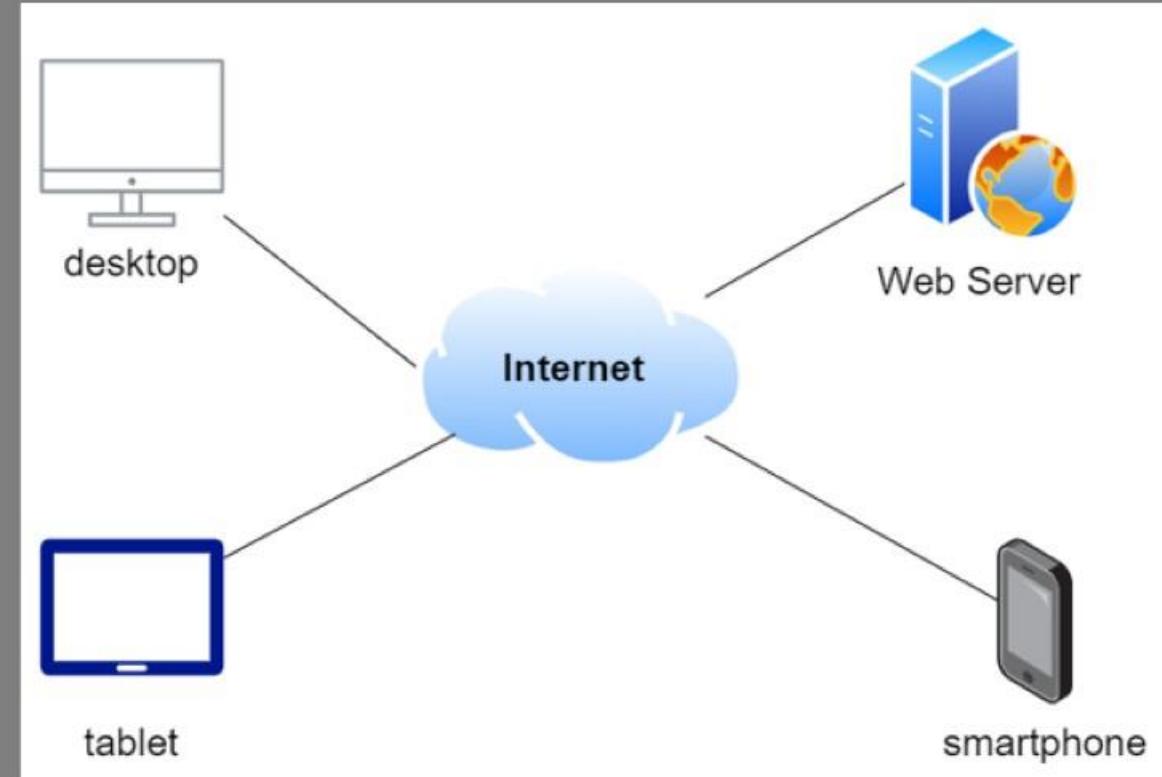
Cliente

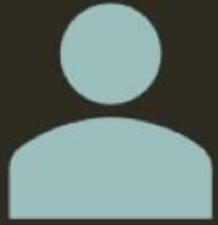


Servidor Web



Red

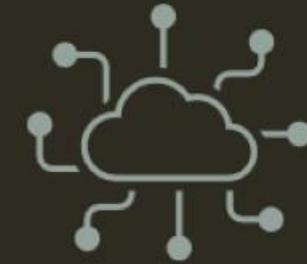




Los clientes usan programas conocidos como navegadores

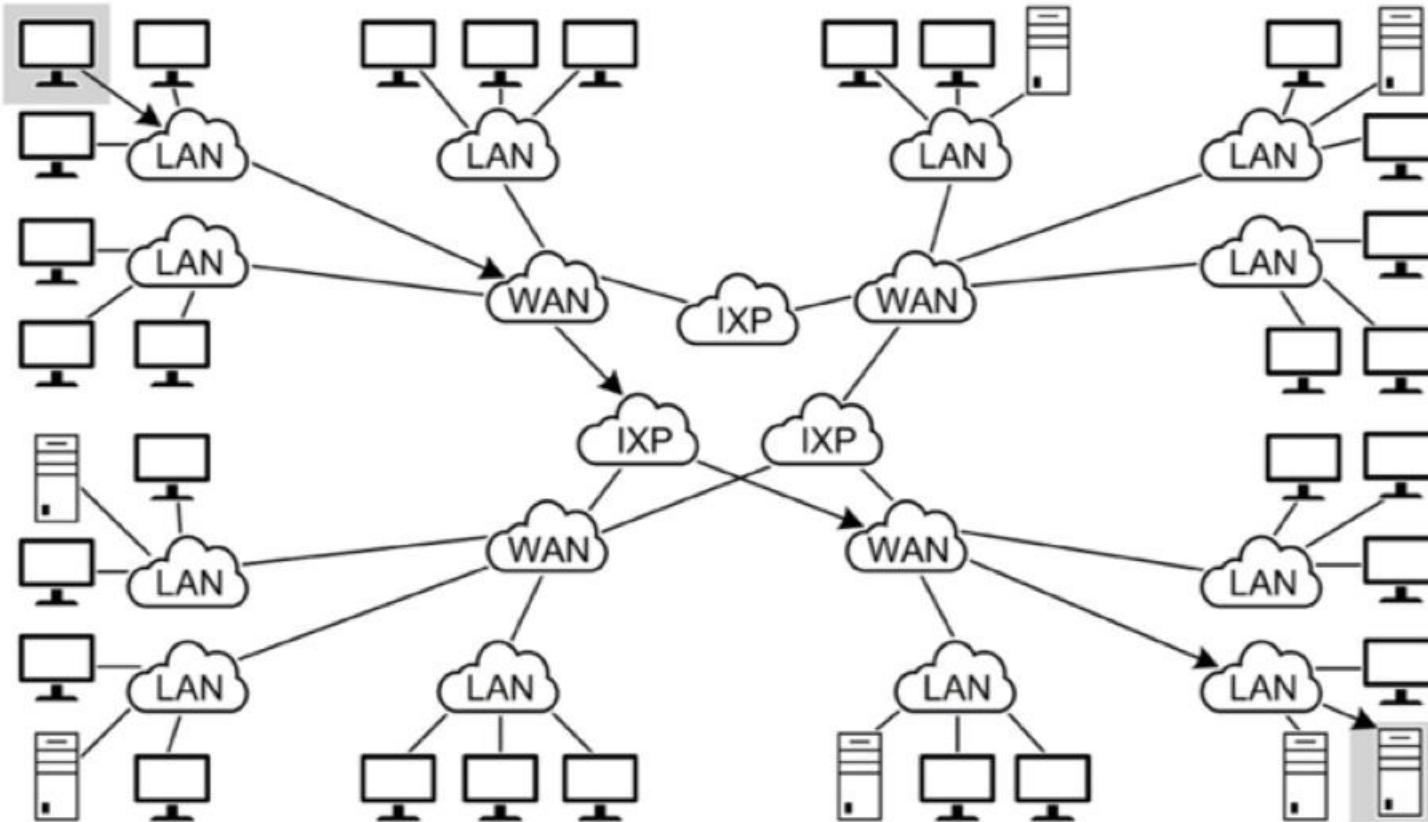


El servidor web retorna las páginas solicitadas por el navegador



La red permite comunicar a los clientes y servidores

# COMPONENTES APLICACIÓN WEB (MODELO CLIENTE/SERVIDOR)



# ARQUITECTURA DE INTERNET

LAN. Red de área local

WAN. Red de área amplia

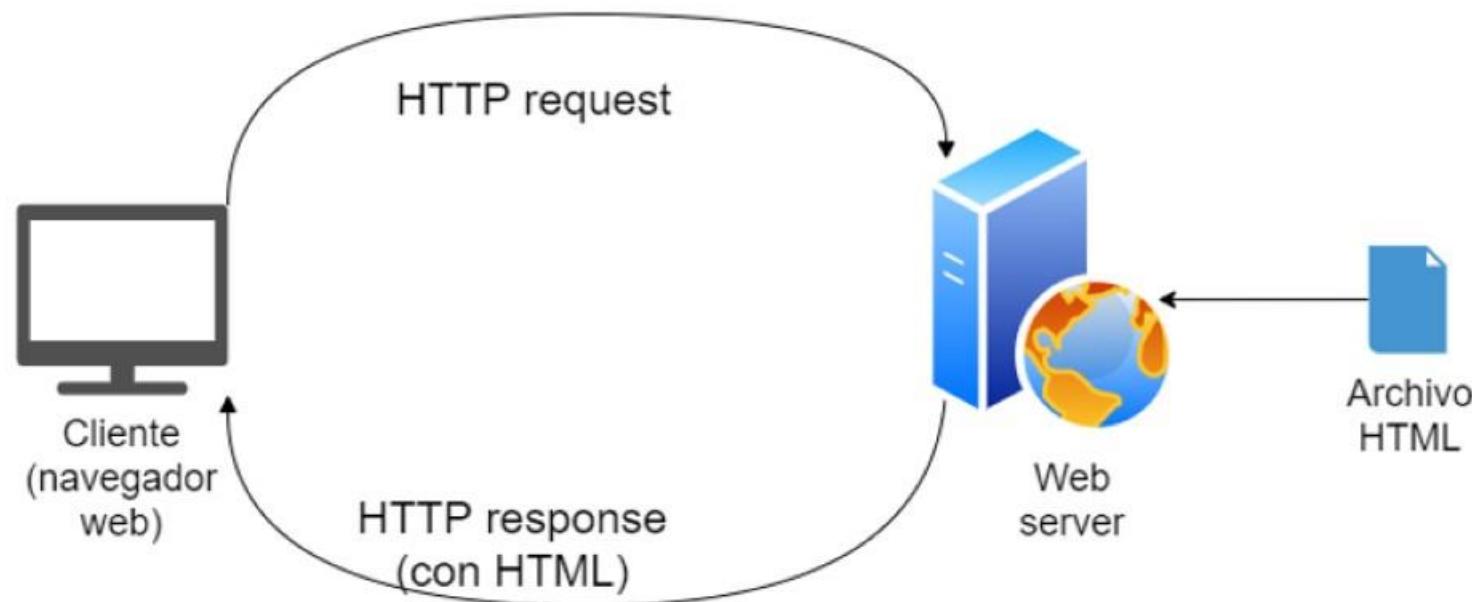
IXP. Punto de Intercambio de Tráfico de Internet

ISP. Proveedor de servicio de internet. ISP posee una WAN y renta el acceso a su red.

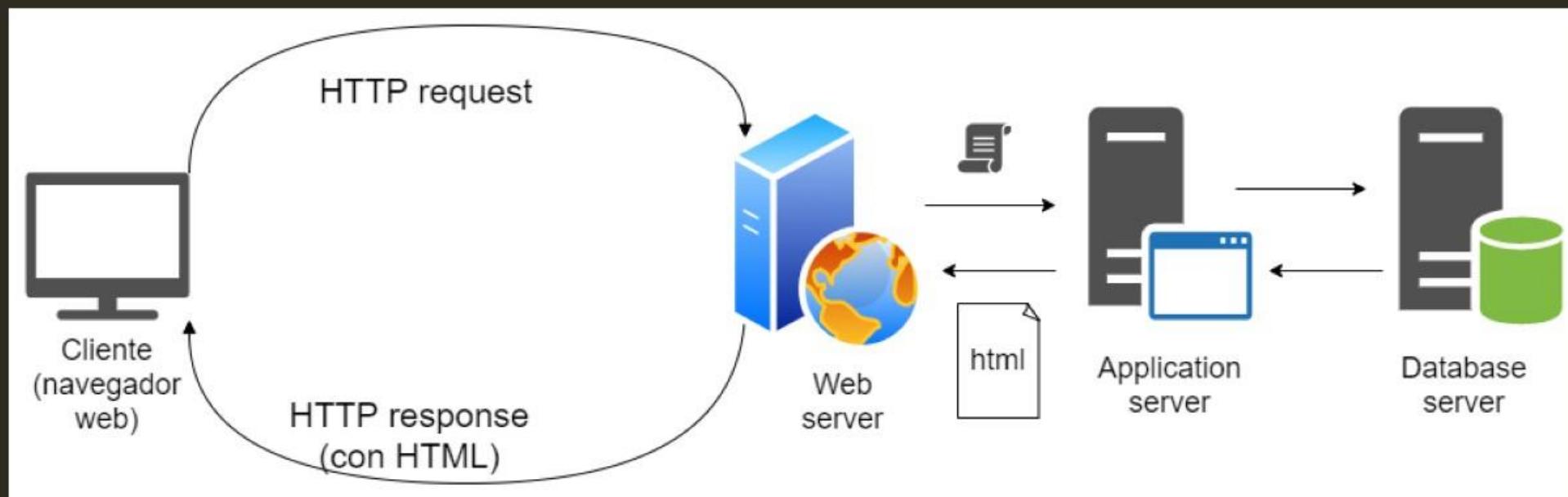
Internet. Consiste de muchas WANs que han sido conectadas con IXP

# COMO PROCESA UN SERVIDOR WEB UNA PAGINA WEB ESTÁTICA

- HTTP. Protocolo de transferencia de hipertexto
- HTML. Lenguaje de marcas de hipertexto



# COMO PROCESA UN SERVIDOR WEB UNA PAGINA WEB DINAMICA



# Protocolos de internet

- ❖ FTP (File Transfer Protocol). Protocolo para el intercambio de archivos
- ❖ SMTP (Simple Mail Transfer Protocol). Protocolo para envío de correos
- ❖ POP3 (Post Office Protocol) e IMAP (Internet Message Access Protocol). Protocolos para recibir correos.
- ❖ HTTP (Hypertext Transfer Protocol). Protocolo para intercambiar archivos tales como texto, imágenes, audio, video, u otro contenido multimedia en la web.
- ❖ TCP/IP. Protocolo oficial de internet.
- ❖ TCP (Transmission Control Protocol). Asegura la integridad de la comunicación de red.
- ❖ IP (Internet Protocol). Son un conjunto de reglas que controla como los datos son enviados entre computadoras sobre internet.



# HTML

HTML es un lenguaje que describe el contenido y estructura de un documento marcando o etiquetando diferentes elementos del documento.

Elementos de un documento: título, encabezado principal, párrafos, listas, tablas, etc.

# Versiones HTML

Año	Versión
1991	Tim Berners-Lee invento HTML
1993	Dave Raggett elaboró HTML+
1995	HTML 2.0 primer standard oficial
1997	Recomendación W3C: HTML 3.2
1999	Recomendación W3C: HTML 4.01
2000	Recomendación W3C: XHTML 1.0
2008	WHATWG HTML5 primer borrador oficial
2012	WHATWG HTML5 Living Standard
2014	Recomendación W3C: HTML 5
2017	Recomendación W3C: HTML5.1 segunda versión
2017	Recomendación W3C: HTML 5.2

Web Hypertext Application Technology Working Group (WHATWG)



The World Wide Web Consortium (W3C)





Visual Studio Code



- ❖ Visual Studio Code. <https://code.visualstudio.com/>
- ❖ Agregar la extensión Live Server y Prettier

## HERRAMIENTAS PARA TRABAJAR CON HTML

# Atajos de teclado VS CODE

Ctrl+X Cortar (sin seleccionar)

Ctrl+C Copiar (sin seleccionar)

Ctrl+Shift+P, F1 Mostrar Paleta de comandos

Alt+Fecha arriba/abajo Mueve línea

Shift+Alt+Fecha arriba/abajo copia línea

Ctrl+Shift-K borra la línea

Shift+Alt+a Agrega/elimina comentarios

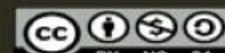
Ctrl+D agregar a la selección

Alt+click inserta un nuevo cursor

Ctrl+L Seleccionar una línea

Ctrl+Shift+L Seleccionar todas las ocurrencias seleccionadas

Shift+Alt+arrastrar mouse seleccionar en modo columna



# Introducción a las etiquetas (elementos html)

- ❖ Los tags son los elementos de construcción elementales en cada documento HTML
- ❖ Existen dos tipos de elementos: Elementos con contenido y Elementos vacíos.

```
<elemento>contenido</elemento>
```

```
<elemento>
```

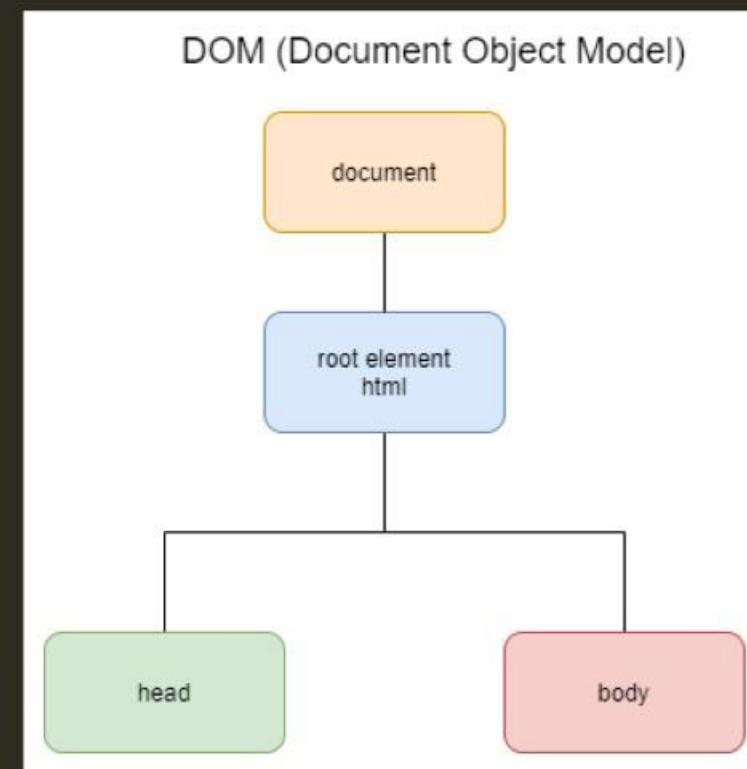
```
<elemento/>
```



# Estructura de un documento HTML

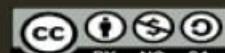
- ❖ Un documento HTML tiene dos secciones principales
  - ❖ head. Esta sección contiene información acerca del documento.
  - ❖ body. Esta sección contiene información desplegada por el browser.

```
<!DOCTYPE html>
<html>
<head>
    contenido del head
</head>
<body>
    contenido del body
</body>
</html>
```



# Subir una página a GitHub Pages

- ❖ La carpeta que contenga nuestro proyecto web debemos hacer que sea controlado por git.
- ❖ Crear un repositorio en GitHub donde subiremos el proyecto
- ❖ En la terminal de git escribir el código para enlazar el repositorio local con el remote:
  - ❖ `git remote add origin urlrepositorio`
- ❖ Confirmar los cambios que se tengan en local
- ❖ Subir el repositorio a GitHub:
  - ❖ `git push -u origin master`
- ❖ Una vez terminada la operación de subida. Ingresar a la configuración del repositorio en GitHub.



# Subir una página a GitHub Pages

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

✓ Your site is published at <https://empresatest.github.io/app/>

**Source**  
Your GitHub Pages site is currently being built from the `master` branch. [Learn more.](#)

Branch: master / (root) Save

Select branch

- master
- None

# Agregando atributos a los elementos(tags) html

- ❖ Los elementos HTML pueden tener uno o mas atributos.

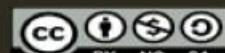
```
<elemento atributo1="valor1" atributo1="valor2">contenido</elemento>
```

```
<elemento atributo1="valor1" atributo1="valor2"/>
```

```
<h1 id="encabezado-h1">Ejemplo</h1>
```

# Atributos principales de HTML

Atributo	Descripción
class = “texto”	Define una clasificación(clase) general del elemento
hidden	Indica que el elemento debe estar oculto.
id = “text”	Proporciona un identificador único para el elemento
lang=“text”	Especifica el lenguaje del contenido del elemento
style=“definición”	Define el estilo o apariencia del elemento
tabindex=“entero”	Especifica el orden del elemento al presionar tab.



# Manejo de espacios en blanco

- ❖ El espacio, tabulador o cambio de línea es considerado un espacio en blanco.
- ❖ Cuando el navegador lee un archivo HTML, ignora la presencia de espacios en blanco entre los elementos HTML. Lo siguiente representa lo mismo:

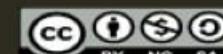
```
<p>Hola HTML</p>
```

```
<p>Hola  
</p>
```

HTML

Hola HTML

Hola HTML



# Creando el encabezado del documento (head)

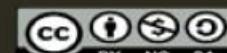
- ❖ En la sección head del documento se colocan los metadatos que describen el documento.

Elemento	Descripción
title	Título del documento
meta	Proporciona una lista de valores de metadatos tales como: keywords, descripción, autor, viewport. También se puede especificar la codificación de caracteres.
link	Especifica recursos externos del documento
script	Se utiliza para especificar código de programación
style	Define estilos que serán aplicados al documento

# Agregar comentarios a HTML

- ❖ Para agregar comentarios se utiliza <!-- comentario -->
- ❖ En VS CODE presionar Shift+Alt+a para agregar o quitar comentarios.

```
<!-- Este es un comentario que no es tomado en cuenta  
     por el navegador -->
```



# Estructura de una página web

## *Header*

HOME      OUR TEAM      PROJECTS      CONTACT            Go!

### *Article heading*

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Donec a diam lectus. Set sit amet ipsum mauris. Maecenas congue ligula as quam viverra nec consectetur ant hendrerit. Donec et mollis dolor. Praesent et diam eget libero egestas mattis sit amet vitae augue. Nam tincidunt congue enim, ut porta lorem lacinia consectetur.

#### *subsection*

Donec ut libero sed accu vehicula ultricies a non tortor. Lorem ipsum dolor sit amet, consectetur adipisicing elit. Aenean ut gravida lorem. Ut turpis felis, pulvinar a semper sed, adipiscing id dolor.

Pelientesque auctor nisi id magna consequat sagittis. Curabitur dapibus, enim sit amet elit pharetra tincidunt feugiat nist imperdiet. Ut convallis libero in urna ultrices accumsan. Donec sed odio eros.

#### *Another subsection*

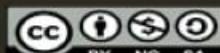
Donec viverra mi quis quam pulvinar at malesuada arcu rhoncus. Cum sociis natoque penatibus et manis dis parturient montes, nascetur ridiculus mus. In rutrum accumsan ultricies. Mauris vitae nisi at sem facilisis semper ac in est.

Vivamus fermentum semper porta. Nunc diam velit, adipiscing ut tristique vitae sagittis vel odio. Maecenas convallis ullamcorper ultriced. Curabitur ornare, ligula semper consectetur sagittis, nisi diam iaculis velit, is fringille sem nunc vet mi.

©Copyright 2050 by nobody. All rights reversed.

### *Related*

- [Oh I do like to be beside the seaside](#)
- [Oh I do like to be beside the sea](#)
- [Although in the North of England](#)
- [It never stops raining](#)
- [Oh well...](#)

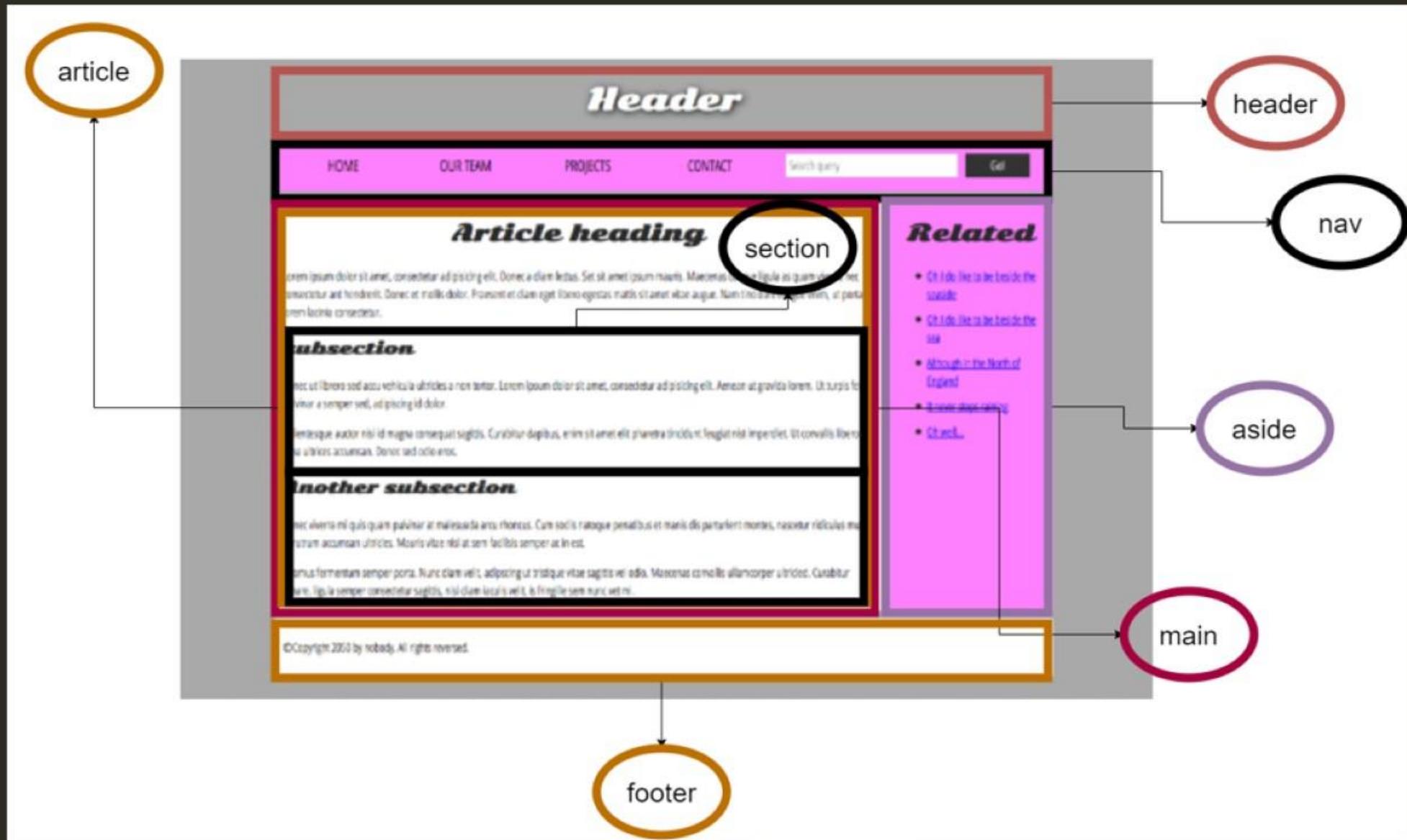


# Escribiendo el contenido de la página web (body)

## ❖ Elementos para marcar las secciones del documento.

Elemento	Descripción
address	Marca información de contacto
article	Marca un bloque de contenido que tiene sentido por sí mismo aparte del resto de la página (por ejemplo una entrada en un blog). Un artículo de revista, noticia, entrada de un blog. <b>HTML5</b>
aside	Representa una sección de una página que consiste en contenido que está indirectamente relacionado con el contenido principal del documento.
body	Cuerpo del documento
footer	Píe para una sección, artículo o página. <b>HTML5</b>
h1, h2, h3, h4, h5, h6	Encabezados se utilizan para definir la estructura del documento.
header	Define el encabezado del documento, artículo o sección. <b>HTML5</b>
nav	Se utiliza para colocar los links de navegación. <b>HTML5</b>
section	Representa una sección genérica de un documento. Sirve para determinar qué contenido corresponde a qué parte de un esquema. <b>HTML5</b>

# Semántica HTML





## Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI

Validate by File Upload

Validate by Direct Input

### Validate by URI

Validate a document online:

Address:

► More Options

Check

# VALIDAR UN DOCUMENTO HTML

<https://validator.w3.org/>

# ELEMENTOS DE AGRUPACIÓN

Cada elemento de agrupación organiza contenido similar en un distinto grupo.

# Elementos de agrupación

Elemento	Descripción
blockquote	Se utiliza para citar contenido que se obtiene de una fuente externa
div	Elemento de agrupación genérico dentro del documento
dl	Se utiliza para crear listas de descripciones. Las cuales pueden contener diversos términos para los cuales se tiene una o mas definiciones.
dt	Contiene la descripción de un termino o concepto que formará parte de una lista de definiciones
dd	Contiene la descripción o definición asociado a un termino en una lista de descripciones
figure	Define una imagen, ilustración, diagrama, bloque de código, el cual es referenciado en el documento. <b>HTML5</b>
figcaption	Contiene una descripción asociada a una figura. <b>HTML5</b>
hr	Regla horizontal que marca una transición a otro tema
main	Marca el contenido principal del documento. <b>HTML5</b>
ol	Se utiliza para definir una lista ordenada
ul	Se utiliza para definir una lista desordenada
li	Se utiliza para definir un simple elemento que forma parte de una lista ordenada o desordenada
p	Marca un párrafo de texto
pre	Contiene un bloque de texto preformatado



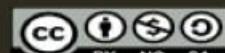
# Elementos a nivel de texto

Elemento	Descripción
a	Se utiliza para crear un hiperlink
abbr	Marca una abreviación
b, i	b se utiliza para negrita y el elemento i cursiva
strong	Se utiliza para marcar texto importante
em	Marca texto enfatizado
br	Se utiliza para aplicar un cambio de línea
cite	Se utiliza para referenciar un trabajo creativo. Puede ser un libro, película o programa.
mark	Se utiliza para marcar el texto
small	Marca comentarios, derechos de autor, texto legal
span	Contenedor genérico en línea. Se utiliza para dar estilo a un texto.
sub	Subíndice
sup	Superíndice
time	Marca un valor de tiempo. <b>HTML5</b>



# Caracteres Especiales

- ❖ Para insertar caracteres especiales en html se puede hacer uso de entidades. Las entidades son un texto que comienza con el símbolo & y termina con ;. Ejemplos:
  - ❖ &copy;
  - ❖ &nnbsp;
  - ❖ &hearts;
- ❖ Se pueden agregar caracteres UNICODE. Para esto se escribe el símbolo & seguido de # así como el código del carácter y termina con ;
  - ❖ &#128047;



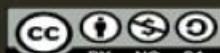
# Etiquetas para elementos incrustados

Elemento	Descripción
audio	Se utiliza para agregar contenido de audio al documento. <b>HTML5</b>
embed	Permite incrustar contenido externo de otras aplicaciones.
iframe	Permite incrustar contenido externo de otra pagina o recurso de internet.
img	Permite incrustar una imagen en el documento
Video	Se utiliza para agregar un video al documento. <b>HTML5</b>

# TABLAS HTML

```
<table>
  <tr>
    <td>celda</td>
    <td>celda</td>
    <td>celda</td>
    <td>celda</td>
  </tr>
  <tr>
    <td>celda</td>
    <td>celda</td>
    <td>celda</td>
    <td>celda</td>
  </tr>
</table>
```

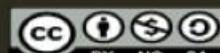
La etiqueta **table** marca la tabla, la etiqueta **tr** define los renglones y las etiquetas **td** marcan las celdas.



# TABLAS HTML ENCABEZADO

```
<table>
  <tr>
    <th>celda</th>
    <th>celda</th>
    <th>celda</th>
    <th>celda</th>
  </tr>
  <tr>
    <td>celda</td>
    <td>celda</td>
    <td>celda</td>
    <td>celda</td>
  </tr>
</table>
```

La etiqueta th se utiliza para definir el encabezado de la tabla



# Unir celdas vertical (rowspan) y horizontalmente (colspan)

```
<table>
  <tr>
    <th>col1</th>
    <th>col2</th>
    <th>col3</th>
    <th>col4</th>
    <th>col5</th>
  </tr>
  <tr>
    <td>celda1</td>
    <td>celda2</td>
    <td rowspan="3">celda3</td>
    <td>celda4</td>
    <td>celda5</td>
  </tr>
  <tr>
    <td>celda1</td>
    <td>celda2</td>
    <td colspan="2">celda4</td>
  </tr>
  <tr>
    <td>celda1</td>
    <td>celda2</td>
    <td colspan="2" rowspan="2">celda4</td>
  </tr>
</table>
```

col1	col2	col3	col4	col5
celda1	celda2	celda3	celda4	celda5
celda1	celda2		celda4	
celda1	celda2	celda3	celda4	

```
<table>
  <colgroup>
    <col class="primera" />
    <col class="segunda" />
    <col class="tercera" />
  </colgroup>
  <thead>
    <tr>
      <th>uno</th>
      <th>uno</th>
      <th>uno</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>uno</td>
      <td>uno</td>
      <td>uno</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td colspan="3">pie</td>
    </tr>
    <tr>
      <td>&copy;</td>
      <td>Derechos reservados</td>
      <td></td>
    </tr>
  </tfoot>
</table>
```

# Tbody, tfoot, thead, colgroup

uno	uno	uno
uno	uno	uno
pie		
©	Derechos reservados	

# Formularios

url

week  Semana --, ----

Button  Button

submit  Enviar

reset  Restablecer

select  LTI

LTI  
LA  
CP  
LNI

select multiple

select optgroup  LTI

textarea

datalist  lenguaje

Input text  escriba su nombre aquí

checkbox

color

date  dd / mm / aaaa

datetime

datetimelocal  dd / mm / aaaa -- : -- -----

email

file  Seleccionar archivo Ningún archivo seleccionado

hidden



submit button image

month  ----- de -----

number

password

masculino  femenino

range

search

submit  Enviar

tel

time  -- : -- -----



CSS

Hojas de estilo en cascada

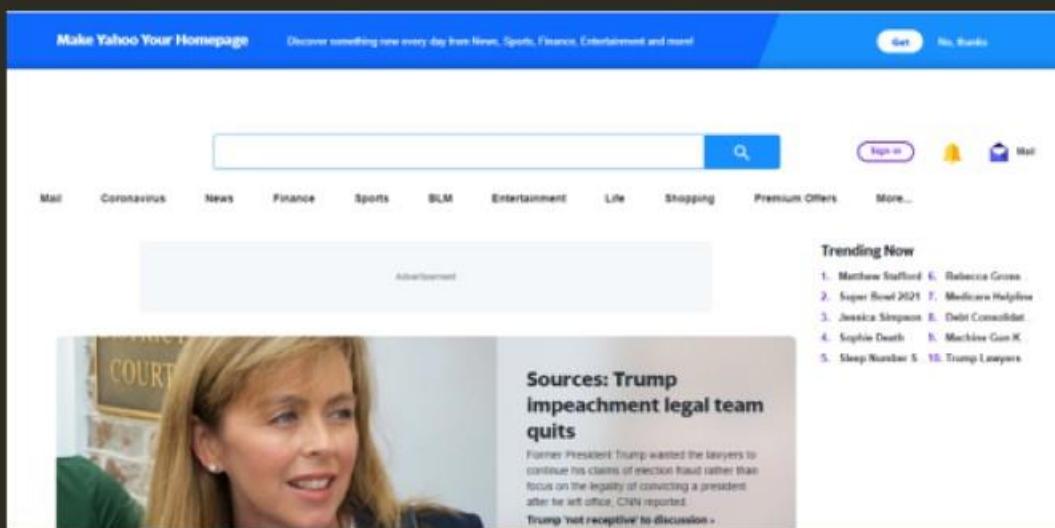


# CSS

Global  
HD

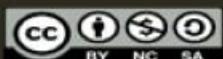
# ¿QUÉ ES CSS?

CSS es un lenguaje para definir la apariencia o estilo de las páginas web, escritas en HTML.



# ¿CÓMO SE USA CSS?

- ❖ Estilos en línea
- ❖ Bloques de estilo
- ❖ Hojas de estilo externas



# Estilos en línea y bloques de estilo

```
* HTML 24 unsaved changes X
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Facpya</title>
7   <style>
8     h1 {
9       color: blue;
10    }
11  </style>
12 </head>
13 <body>
14   <h1>Encabezado</h1>
15   <p style="color:red; border:1px solid lime;">Lorem ipsum dolor sit
16   amet.</p>
17 </body>
18 </html>
```

## Encabezado

Lorem ipsum dolor sit amet.



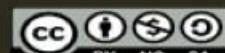
# Hojas de estilo externas

```
index.html < index.html > html > body
1   <!DOCTYPE html>
2   <html lang="es">
3     <head>
4       <meta charset="UTF-8" />
5       <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6       <title>Aprendiendo CSS</title>
7       <link rel="stylesheet" href="styles.css" />
8     </head>
9     <body>
10    <h1>Hola mundo</h1>
11  </body>
12 </html>

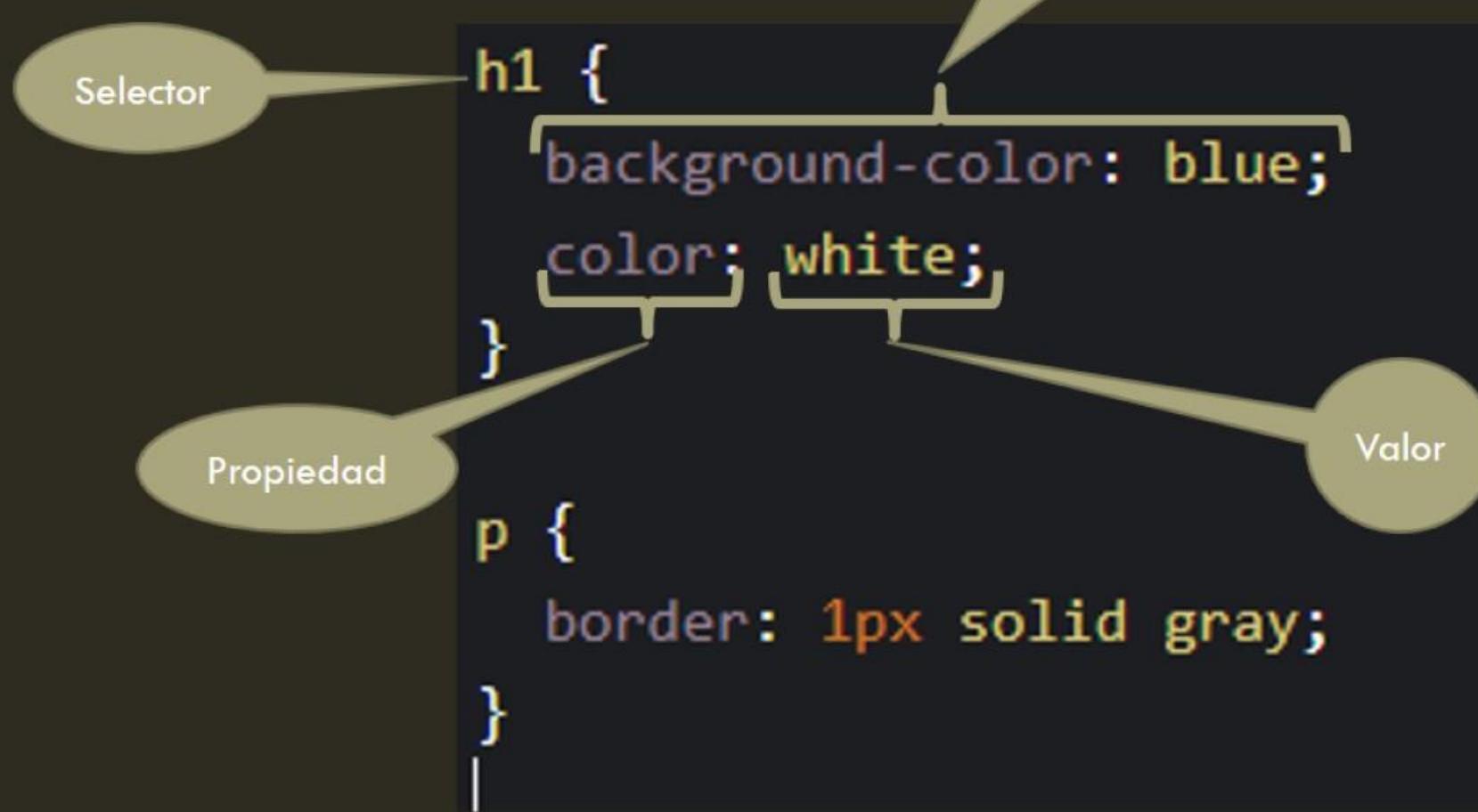
# styles.css < index.html > styles.css > ...
# styles.css > ...
1  h1 {
2    background-color: aquamarine;
3  }
4
```

# REGLAS CSS

```
h1 {  
    background-color: blue;  
    color: white;  
}  
  
p {  
    border: 1px solid gray;  
}
```

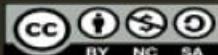


# CSS



# SELECTORES BASICOS

- ❖ Selectores Universal
- ❖ Selectores de tipo
- ❖ Selectores por ID
- ❖ Selectores de Clase
- ❖ Selector de atributo



# Selector Universal (\*)

## HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6 <title>CSS Selector Universal</title>
7 </head>
8 <body>
9 <h1>Aprendiendo CSS</h1>
10 <p>Css se utiliza para dar estilo a una p gina web</p>
11 </body>
12 </html>
```

## CSS

```
1 * {
2   color : red;
3 }
```

## Aprendiendo CSS

Css se utiliza para dar estilo a una p gina web



# Selectores de tipo

The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. The 'HTML' tab contains the following code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>CSS</h1>
6   <p>Lorem, ipsum dolor.</p>
7   <p>Lorem ipsum dolor sit.</p>
8 </body>
9 </html>
```

The 'CSS' tab contains the following code:

```
1 * p {
2   color: blue;
3 }
```

A red exclamation mark icon is located at the bottom center of the editor window.

## CSS

Lorem, ipsum dolor.

Lorem ipsum dolor sit.



# Selectores por id

HTML 8 unsaved changes X

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>CSS</h1>
6   <p id="parrafo-principal">Lorem, ipsum dolor.</p>
7   <p>Lorem ipsum dolor sit.</p>
8 </body>
9 </html>
```

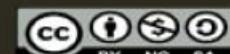
CSS

```
1 #parrafo-principal {
2   color:green;
3 }
```

## CSS

Lorem, ipsum dolor.

Lorem ipsum dolor sit.



# Selectores de clase

The screenshot shows a code editor interface with two tabs: "HTML" and "CSS".

**HTML Tab:**

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head></head>
4   <body>
5     <h1 class="outline">Css</h1>
6     <p>Lorem ipsum dolor sit amet.</p>
7     <p class="outline">Lorem ipsum dolor sit amet consectetur.</p>
8   </body>
9 </html>
```

**CSS Tab:**

```
1 .outline {
2   outline: 1px solid red;
3 }
```

The output pane on the right displays the rendered HTML. It shows an 

# element with a red outline and a element with a red outline, both containing placeholder text.

# Selectores de clase de un tag en particular

The screenshot shows a code editor interface with two tabs: "HTML" and "CSS".

**HTML Tab:**

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1 class="outline">Css</h1>
6   <p>Lorem ipsum dolor sit amet.</p>
7   <p class="outline">Lorem ipsum dolor sit amet consectetur.</p>
8 </body>
9 </html>
```

**CSS Tab:**

```
1 p.outline {
2   outline: 1px solid red;
3 }
```

The output pane on the right displays the rendered HTML. It shows an 

# element with the text "Css" and a element with the text "Lorem ipsum dolor sit amet.". The second element is highlighted with a red border, indicating it is styled by the ".outline" CSS selector.

# Selectores por atributo

- ❖ [nombre]. Todos los elementos que tengan el atributo sin importar su valor
- ❖ [nombre="valor"]. Selecciona a elementos que tengan el atributo con el valor especificado.
- ❖ [nombre ~="valor"]. Selecciona a elementos que tengan el atributo y cuyo valor sea una lista de palabras separadas por espacio, una de las cuales es valor.
- ❖ [nombre \*= "valor"]. Selecciona a elementos que contengan la cadena valor
- ❖ [nombre^="valor"]. Selecciona a elementos cuyo atributo comience con valor
- ❖ [nombre\$="valor"]. Selecciona a elementos cuyo atributo finalice con valor.



# Selectores por atributo

HTML      26 unsaved changes X

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>Css</h1>
6   <a href="#pie-pagina">Pie de p  gina</a>
7   <a href="https://www.youtube.com">Youtube</a>
8   <a href="https://miyoutube.com">Mi Youtube</a>
9   <footer id="pie-pagina"></footer>
10 </body>
11 </html>
```

CSS

```
1 a[href*="youtube"] {
2   background-color: tomato;
3 }
4
5 a[href^="#" ] {
6   background-color: pink;
7 }
```

Css

Pie de p  gina Youtube Mi Youtube

# Selectores de lista

## HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>CSS</h1>
6   <p>Lorem ipsum dolor sit.</p>
7   <p>Lorem ipsum dolor sit amet.</p>
8   <a href="">Link</a>
9 </body>
10 </html>
```

## CSS

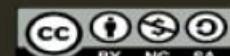
```
1 a, p {
2   border: 10px solid green;
3 }
```

## CSS

Lorem ipsum dolor sit.

Lorem ipsum dolor sit amet.

[Link](#)



# Selectores compuestos

## HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1 class="importante">CSS</h1>
6   <p>Lorem ipsum dolor sit.</p>
7   <p class="importante">Lorem ipsum dolor sit amet.</p>
8   <a href="">Link</a>
9 </body>
10 </html>
```

## CSS

Lorem ipsum dolor sit.

Lorem ipsum dolor sit amet.

[Link](#)

## CSS

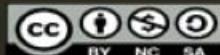
9 unsaved changes X

```
1 h1.importante {
2   background-color:red;
3 }
```



# Combinadores CSS

- ❖ Selector de descendientes
- ❖ Selector hijo
- ❖ Selector hermano adyacente
- ❖ Selector hermano general



# Selector de descendientes

## HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
8 <body>
9 <div>
10 <h1 class="importante">CSS</h1>
11 <p>Lorem ipsum dolor sit.</p>
12 <section>
13 <p class="importante">Lorem ipsum dolor sit amet.</p>
14 <a href="">Link</a>
15 </section>
16 </div>
17 </body>
18 </html>
```

## CSS

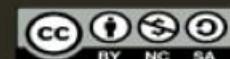
```
1 div p {
2   background-color:tomato;
3 }
```

## CSS

Lorem ipsum dolor sit.

Lorem ipsum dolor sit amet.

[Link](#)



# Selector de descendientes hijos directos

## HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <div>
6     <h1 class="importante">CSS</h1>
7     <p>Lorem ipsum dolor sit.</p>
8     <section>
9       <p class="importante">Lorem ipsum dolor sit amet.</p>
10      <a href="">Link</a>
11    </section>
12  </div>
13 </body>
14 </html>
```

## CSS

```
1 div > p {
2   background-color:tomato;
3 }
```

## CSS

Lorem ipsum dolor sit.

Lorem ipsum dolor sit amet.

[Link](#)



# Selector hermano general

The screenshot shows a code editor interface with two tabs: "HTML" and "CSS".

**HTML Tab:**

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <div>
6     <div class="header">header</div>
7     <div class="body">body</div>
8     <div class="footer">footer</div>
9   </div>
10 </body>
11 </html>
```

**CSS Tab:**

```
1 div.header ~ div {
2   background-color: purple;
3 }
```

A floating context menu is open over the "header" element in the HTML tree, showing options: "header", "body", and "footer". The "header" option is highlighted.

# Selector hermano adyacente

The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. The 'HTML' tab displays the following code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <div>
6     <div class="header">header</div>
7     <div class="body">body</div>
8     <div class="footer">footer</div>
9   </div>
10 </body>
11 </html>
```

The 'body' element is highlighted with a green background. The 'CSS' tab displays the following code:

```
1 .header + div {
2   background-color: green;
3 }
```

A browser preview window is visible on the right side of the interface, showing a dark-themed page with a green header and a white body containing the word 'body'.

# PSEUDO-CLASES

- ❖ Las pseudo-clases permiten seleccionar elementos basados en su estado.
- ❖ Comienzan con dos puntos.
- ❖ Algunos ejemplos son:
  - ❖ :active
  - ❖ :checked
  - ❖ :focus
  - ❖ :hover
  - ❖ :valid
  - ❖ :invalid
  - ❖ :visited
  - ❖ :first-child, etc.



# Pseudo-clases

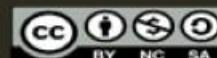
## HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>Pasa el cursor sobre mi</h1>
6 </body>
7 </html>
```

Pasa el cursor sobre mi

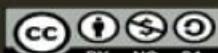
## CSS

```
1 h1:hover {
2   outline: 1px solid blue;
3 }
```



# PSEUDO-ELEMENTOS

- ❖ Las pseudo-elementos permiten seleccionar parte de un elemento.
- ❖ Comienzan con “::”.
- ❖ Algunos ejemplos son:
  - ❖ ::first-line      bloque
  - ❖ ::first-letter      bloque
  - ❖ ::before, ::after



# Pseudo-elementos

## HTML

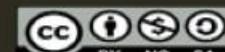
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>Encabezado</h1>
6   <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Nisi delectus quibusdam, sit adipisci, porro, quas totam voluptatum ut repellendus eum aperiam deleniti. Aspernatur, fuga corporis ipsam quaerat labore voluptates ratione. Nostrum recusandae provident, voluptatum fugit quibusdam ex illum iste suscipit repudiandae repellat labore.
7   </p>
8 </body>
9 </html>
```

## CSS

```
1 p::first-line {
2   background-color: yellow;
3 }
```

## Encabezado

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Nisi delectus quibusdam, sit adipisci, porro, quas totam voluptatum ut repellendus eum aperiam deleniti. Aspernatur, fuga corporis ipsam quaerat labore voluptates ratione. Nostrum recusandae provident, voluptatum fugit quibusdam ex illum iste suscipit repudiandae repellat labore.



# Especificidad

- ❖ Pueden existir conflictos al momento de aplicar los estilos, debido que pueden existir varias reglas que se pueden aplicar a un elemento.

HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>Encabezado</h1>
6   <p class='parrafo'>Lorem ipsum dolor sit.</p>
7 </body>
8 </html>
```

CSS

```
1 .parrafo {
2   color: blue;
3 }
4
5 p {
6   color: orange;
7 }
```

Encabezado

Lore ipsum dolor sit.

# Especificidad

Estilos en línea



Selectores por id



Selectores de Clases, pseudo-clases, atributos



Selectores de Elemento, pseudo-elementos

# Calculando la especificidad

0

0

0

0

Estilos en  
línea

Selectores  
por id

Selectores  
de Clases,  
pseudo-  
clases,  
atributos

Selectores  
de  
Elemento,  
pseudo-  
elementos

# Especificidad

```
HTML
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>Encabezado</h1>
6   <p class="parrafo">Lorem ipsum dolor sit.</p>
7 </body>
8 </html>
```

```
CSS
1 /* especificidad 0010 */
2 .parrafo {
3   color: blue;
4 }
5
6 /* especificidad 0001 */
7 p {
8   color: orange;
9 }
```

## Encabezado

Lorem ipsum dolor sit.

# !important

## HTML

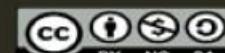
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head></head>
4 <body>
5   <h1>Encabezado</h1>
6   <p class="parrafo">Lorem ipsum dolor sit.</p>
7 </body>
8 </html>
```

## CSS

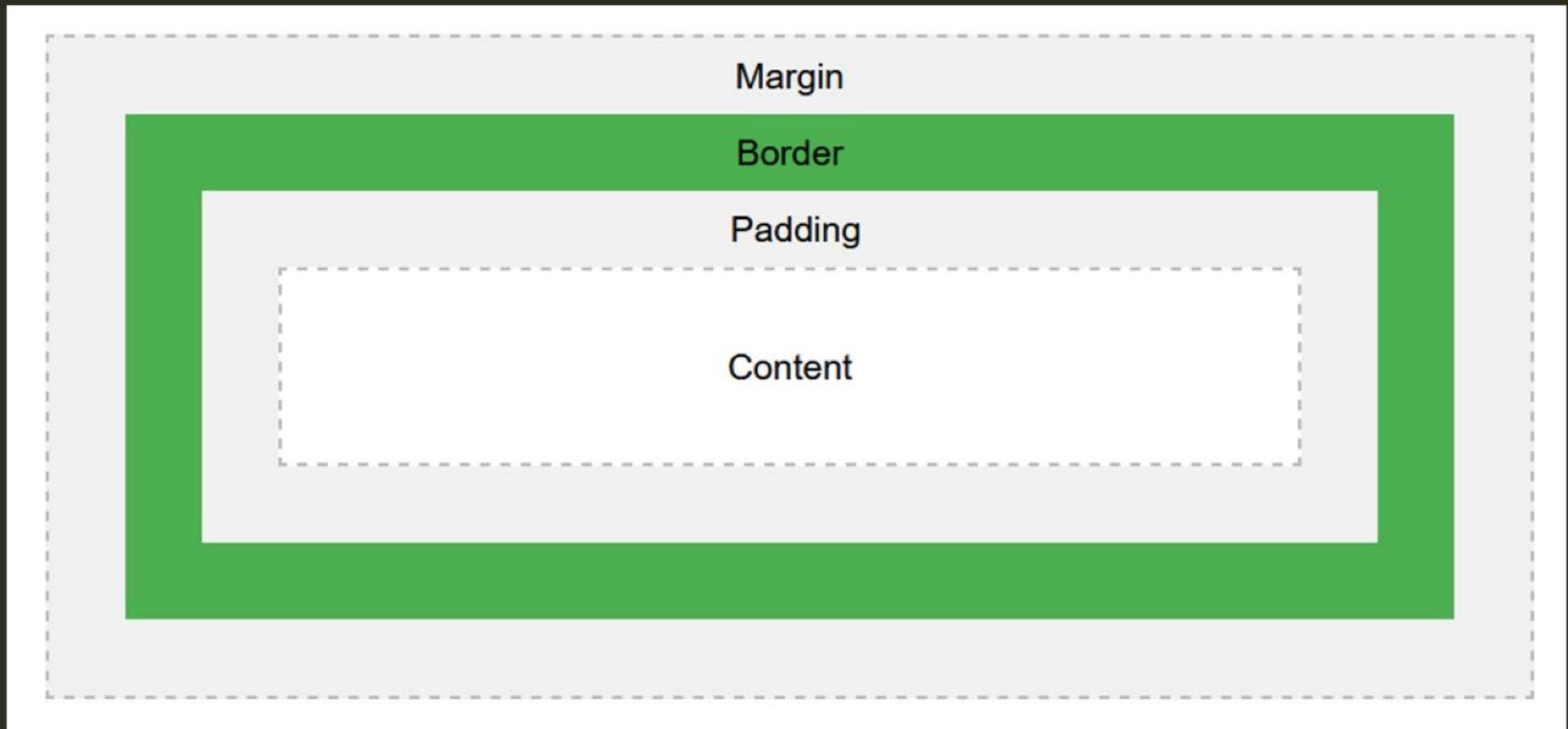
```
1 /* especificidad 0010 */
2 .parrafo {
3   color: blue;
4 }
5
6 /* especificidad 0001 */
7 p {
8   color: orange !important;
9 }
```

## Encabezado

Lorem ipsum dolor sit.



# Modelo de caja



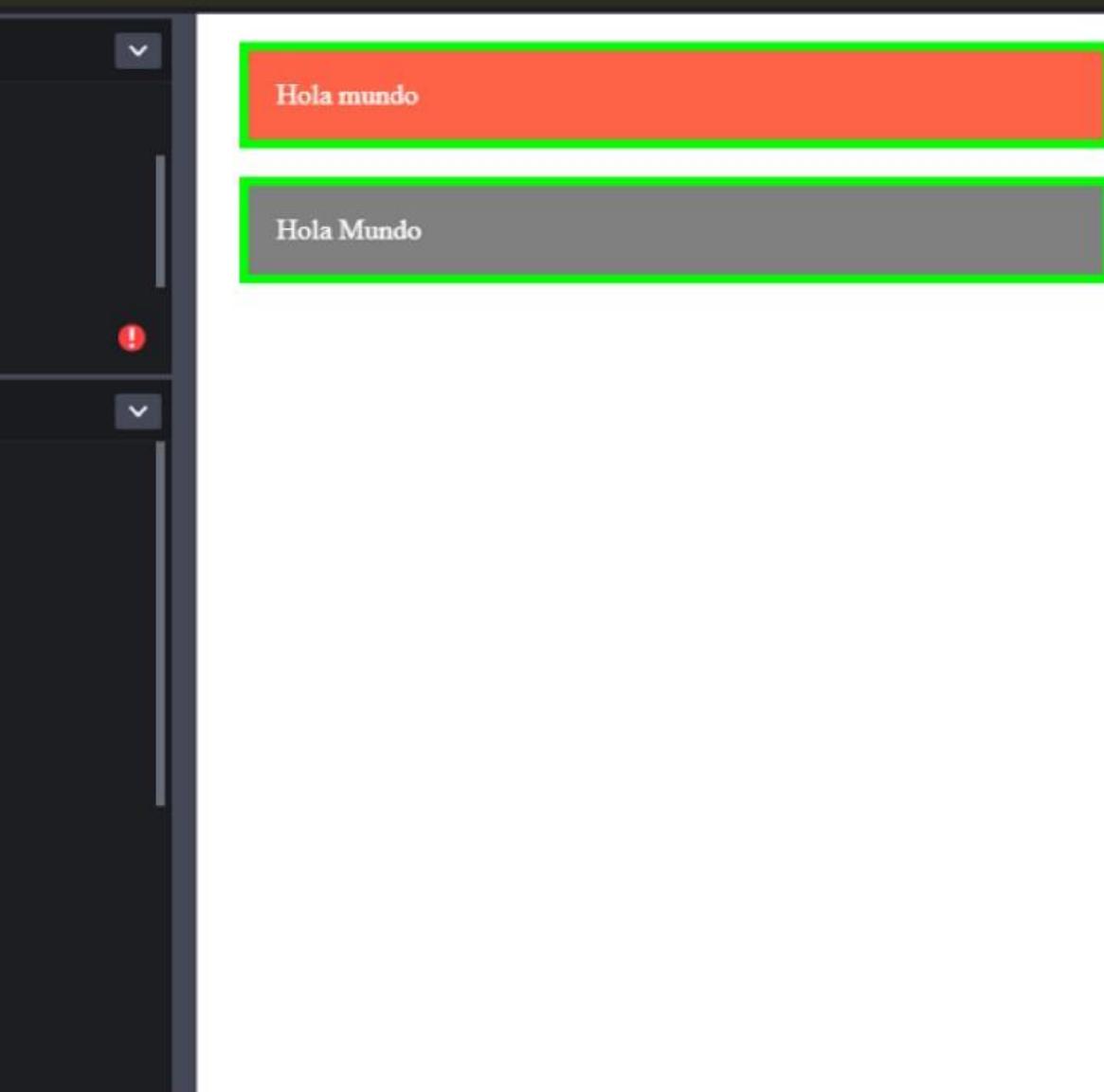
# Modelos de caja

## HTML

```
8 <body>
9   <div class="caja1">Hola mundo</div>
10  <div class="caja2">Hola Mundo</div>
11 </body>
12 </html>
```

## CSS

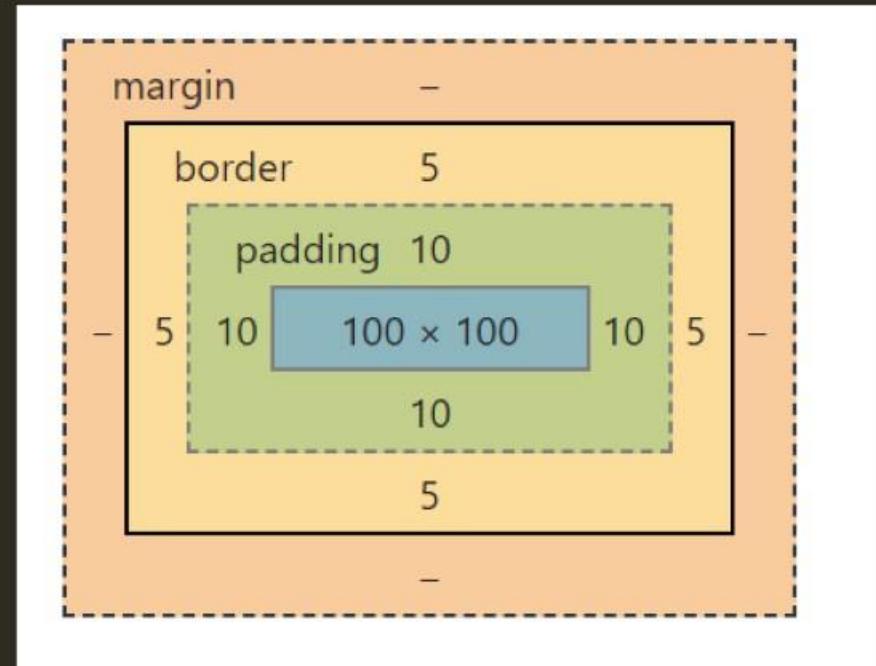
```
1 .caja1 {
2   background-color: tomato;
3 }
4 .caja2 {
5   background-color: grey;
6 }
7 div {
8   color: white;
9   padding: 1rem;
10  margin: 1rem;
11  border: 5px solid lime;
12 }
```



# Box sizing (content-box)

- ❖ El tamaño de la caja se especifica con `width` y `height`. `box-sizing` es una propiedad que tiene dos valores: `content-box` y `border-box`.
- ❖ `content-box` es por default
- ❖ Con `content-box` las propiedades `width` y `height` son tratadas como el alto y ancho del área de contenido. El tamaño real se obtiene sumando el padding y el borde.

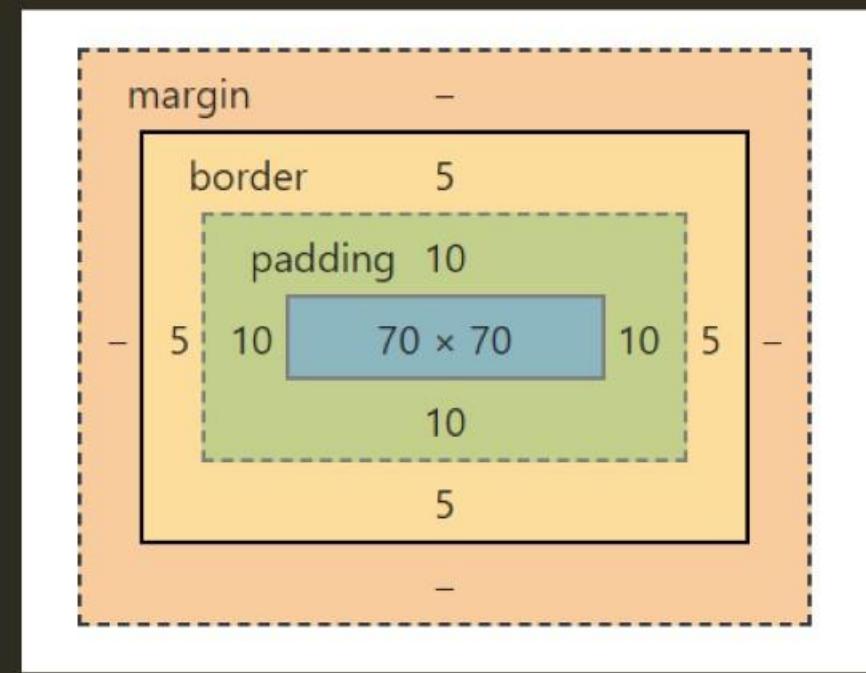
```
div {  
    padding: 10px;  
    height: 100px;  
    width: 100px;  
    background-color: aquamarine;  
    border: 5px solid tomato;  
}
```



# Box sizing (border-box)

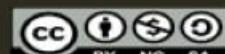
- ❖ Con border-box las propiedades width y height son tratadas como el alto y ancho del área de contenido más el tamaño del padding y el borde.

```
div {  
  padding: 10px;  
  height: 100px;  
  width: 100px;  
  background-color: aquamarine;  
  border: 5px solid tomato;  
  box-sizing: border-box;  
}
```



# Elementos HTML en bloque y en línea

- ❖ Existen elementos en línea, en bloque y combinación de ambos.
- ❖ Elementos en bloque: Siempre toman una línea completa y ocupa todo el ancho de su contenedor. La altura se ajusta a su contenido, pero se puede especificar.
- ❖ Elementos en línea: siguen el flujo normal del texto. Toman el espacio suficiente para su contenido. Especificar alto y ancho no provoca ningún efecto.
- ❖ Elementos inline-block: es una combinación de los primeros dos. Este tipo de elementos siguen el flujo natural del texto, pero se les puede especificar ancho y alto.



## Block, inline, inline-block

```
* HTML
 9 <div>soy block</div>
10 <p>
11   Lorem ipsum, <i>dolor</i> <a href="#">inline</a>sit amet
12   consectetur adipisicing elit. Ab, <a class="link-inline-block"
13   href="#">sunt?</a> ipsum elit.
14 </p>
15 <p>otro texto</p>
```

soy block

# Unidades de medida

- ❖ Pixeles, cm, mm, in, pt.
- ❖ em. Medida relativa. Es relativa al tamaño de fuente del elemento.

The screenshot displays the Microsoft Edge DevTools interface. On the left, the **Elements** panel shows the DOM structure:

```
<body>
  <h1>Encabezado</h1>
</body>
</html>
```

The **CSS** panel contains the following styles:

```
h1 {
    font-size: 20px;
    padding: 3em;
}
```

The **Computed** tab in the bottom right pane shows the computed styles for the **Encabezado** element, which includes:

Style	Value
margin-inline-start	0px
padding-bottom	60px
padding-left	60px
padding-right	60px

At the bottom right of the interface, the text "briel López Solórzano" is visible.

em

The screenshot illustrates a workflow for styling an H1 header. In the top-left, the HTML code defines a single `<article>` element containing an `<h1>Encabezado</h1>`. The top-right shows the corresponding CSS rules: `article { font-size: 10px; }` and `h1 { font-size: 5em; }`. Below the code editor, the browser's main content area displays the word "Encabezado" in a large, bold, dark gray font. At the bottom, the browser's developer tools are open, specifically the Elements tab. The DOM tree shows the `<article>` node selected. The Styles tab of the developer tools is active, showing the computed font size for the `<h1>` element is `50px`, which is the result of the CSS rule `font-size: 5em;` being applied to the `10px` base font size defined in the `article` selector.

HTML 21 unsaved changes

```
8 <body>
9 <article>
10 <h1>Encabezado</h1>
11 </article>
12
13 </body>
14 </html>
```

CSS

```
1 * article {
2   font-size: 10px;
3 }
4 * h1 {
5   font-size: 5em;
6 }
```

Encabezado

Console Assets Shortcuts

Elements Console Sources Network Performance Memory > ⚙️ ⋮ X

▼ <article> Styles Computed Layout Event Listeners DOM Breakpoints >

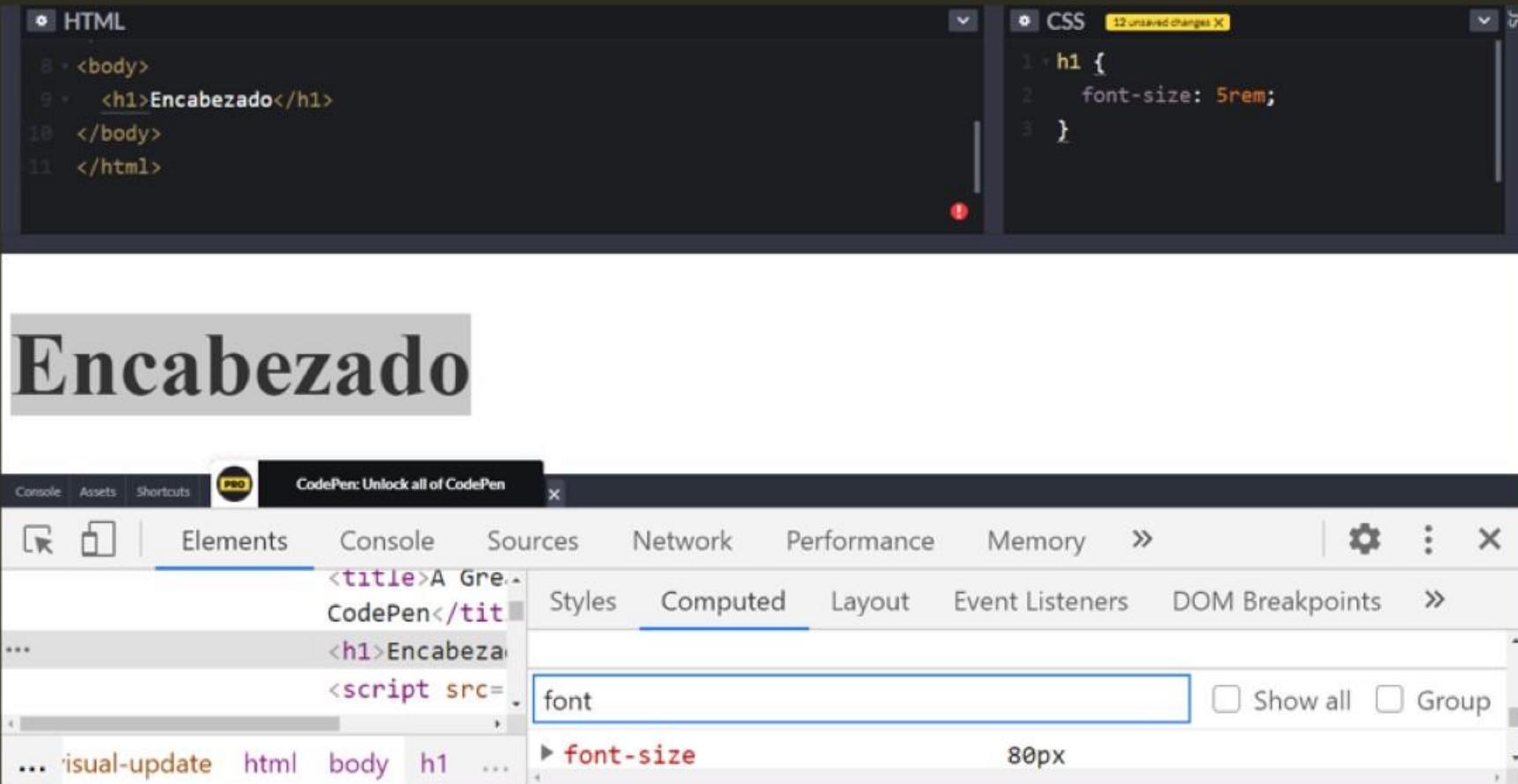
... <h1>Encabe: font-size  X  Show all  Group

... </article>

▶ font-size 50px

rem

- ❖ rem. Medida relativa. Es relativa al tamaño de fuente la página. El tamaño por default en la mayoría de los navegadores es de 16px.



The screenshot displays a browser's developer tools with the following components:

- HTML Panel:** Shows the code for a simple page:

```
8 <body>
9   <h1>Encabezado</h1>
10 </body>
11 </html>
```
- CSS Panel:** Shows the CSS rule for the h1 element:

```
1 h1 {
2   font-size: 5rem;
3 }
```
- Browser Preview:** Shows the word "Encabezado" in a large, bold, dark font.
- Developer Tools Header:** Shows "CodePen: Unlock all of CodePen".
- Elements Tab:** Shows the DOM tree with nodes for title, h1, and script.
- Computed Tab:** Shows the computed styles for the selected element, with "font" typed into the search bar. It lists "font-size: 80px".

# Unidades Viewport: vw y vh

- ❖ Viewport es el área actualmente visible en un navegador web.
- ❖ Estas medidas se utilizan para técnicas de diseño responsive.

```
HTML
1 </head>
2 <body>
3   <div>
4     Hola Viewport
5   </div>
6 </body>
7 </html>

CSS
1 div {
2   background-color: tomato;
3   width: 80vw;
4   height: 50vh;
5 }
```



# Porcentajes %

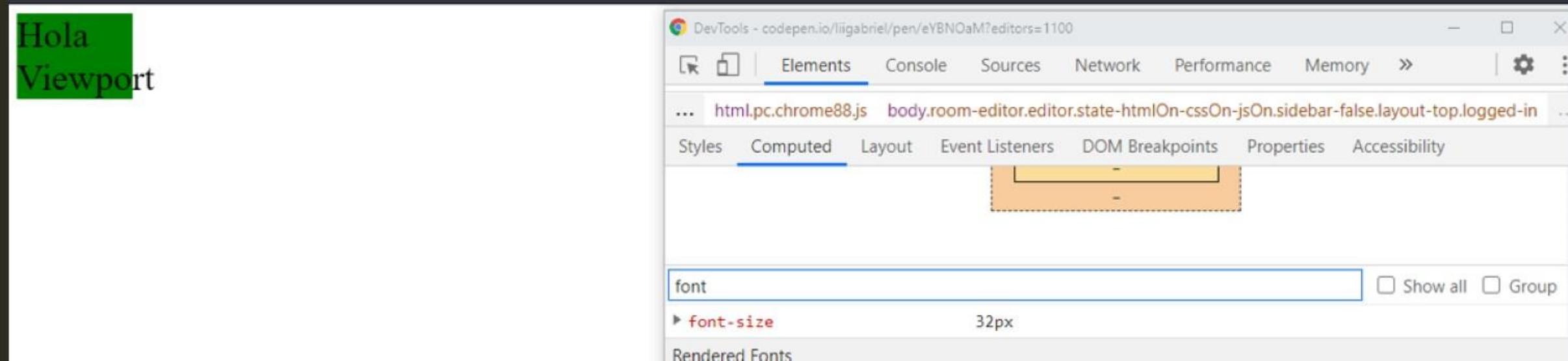
- ❖ Es una unidad relativa al tamaño de otro valor. Por ejemplo: en font-size el porcentaje es relativo al Font-size del elemento padre.

The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. The 'HTML' tab displays the following code:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title> Hola Viewport </title>
5   </head>
6   <body>
7     <div>
8       Hola Viewport
9     </div>
10    </body>
11  </html>
```

The 'CSS' tab contains the following styles:

```
1 * div {
2   font-size: 200%;
3   background-color: green;
4   width : 100px;
5 }
```



# COLORES

- ❖ Predefinidos
- ❖ Colores RGB
- ❖ Colores HSL

	floralwhite
	forestgreen
	gainsboro
	ghostwhite
	gold
	goldenrod
	greenyellow
	grey
	honeydew
	hotpink
	indianred
	indigo

# COLORES PREDEFINIDOS

---

[https://developer.mozilla.org/es/docs/Web/CSS/color\\_value](https://developer.mozilla.org/es/docs/Web/CSS/color_value)

# Colores

## HTML

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head></head>
4 <body>
5   <div class="pred">Predeterminado</div>
6   <div class="rgb">Valor RGB</div>
7   <div class="frgba">función rgba</div>
8   <div class="hsl">función hsl</div>
9   <div class="hsla">función hsla</div>
10
11 </body>
12 </html>
```

## CSS

```
1 .pred {
2   background-color: lawngreen;
3 }
4 .rgb {
5   background-color:#4d375d;
6 }
7 .frgba{
8   background-color: rgba(77,55,93,0.5);
9 }
10 .hsl{
11   background-color: hsl(45,90%,50%)
12 }
13 .hsla{
14   background-color: hsl(45,90%,50%,0.3)
15 }
```



Predeterminado

Valor RGB

función rgba

función hsl

función hsla

# Overflow

- ❖ Cada elemento HTML es una caja rectangular, por lo general se ajusta a su contenido.
- ❖ Cuando se especifica el alto y ancho puede ser que el contenido se desborde

The screenshot shows a split-screen interface for a code editor. On the left, the 'HTML' tab is active, displaying the following code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width,
6       initial-scale=1">
6   <title>A Great Demo on CodePen</title>
7 </head>
8 <body>
9   <p>Lorem ipsum dolor sit amet consectetur
```

On the right, the 'CSS' tab is active, displaying the following styles:

```
1 * p {
2   height: 200px;
3   width: 200px;
4   background-color: green;
5   border: 1px solid black;
6 }
```

Below the code editor, a preview window shows a green square with white text. The text is a long paragraph of placeholder text (Lorem ipsum) that exceeds the boundaries of the green box, demonstrating content overflow.

# overflow

- ❖ Para overflow se tienen las siguientes opciones:

hidden

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Incidunt qui quibusdam nemo doloribus quae quam architecto fugit, maiores quis eius tenetur doloremque officia, blanditiis eveniet asperiores repellat deleniti perspiciatis rerum aliquam, omnis facilis. Odio, fuga ex illo nulla nam laborum modi

scroll

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Incidunt qui quibusdam nemo doloribus quae quam architecto fugit, maiores quis eius tenetur doloremque officia, blanditiis eveniet asperiores repellat deleniti perspiciatis rerum aliquam, omnis facilis. Odio, fuga ex illo nulla nam laborum modi

auto

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Incidunt qui quibusdam nemo doloribus quae quam architecto fugit, maiores quis eius tenetur doloremque officia, blanditiis eveniet asperiores repellat deleniti perspiciatis rerum aliquam, omnis facilis. Odio, fuga ex illo nulla nam laborum modi

FLOTAR ELEMENTOS HTML

# FLOAT ELEMENTOS HTML

- Con css se puede flotar un elemento a la izquierda o derecha del borde de su contenedor.
- La propiedad que se utiliza es float, sus valores son left, right y none.



```
article > img {  
    width: 300px;  
    float: right;  
    margin-left: 1em;  
}
```

# Resolver el problema de colapso del contenedor

. Cuando se utiliza el atributo float se puede generar el siguiente problema: no se muestra el fondo en rojo debido a que el alto es cero.

HTML

```
1 <nav>
2   <ul>
3     <li><a href="#">Home</a></li>
4     <li><a href="#">About</a></li>
5     <li><a href="#">Services</a></li>
6     <li><a href="#">Clientes</a></li>
7   </ul>
8 </nav>
9 <footer>Pie de página</footer>
```

CSS

```
48 unsaved changes X
6 nav ul {
7   background-color: red;
8 }
9
10 nav ul li {
11   float:left;
12   width: 25%;
13 }
```

The screenshot shows a web page with a dark header. On the left, there is a navigation bar with four items: "Home", "About", "Services", and "Clientes". Below the navigation bar, the text "Pie de página" is displayed. On the right, there is some faint, illegible text. The browser interface includes tabs for "HTML" and "CSS", and a status bar indicating "48 unsaved changes X".



# Resolver el problema de colapso del contenedor

The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. The 'HTML' tab contains the following code:

```
1 <nav>
2   <ul class="clearfix">
3     <li><a href="#">Home</a></li>
4     <li><a href="#">About</a></li>
5     <li><a href="#">Services</a></li>
6     <li><a href="#">Clientes</a></li>
7   </ul>
8 </nav>
9 <footer>Pie de página</footer>
```

The 'CSS' tab contains the following code:

```
6
7 .clearfix::after {
8   clear:both;
9   content:"";
10  display:table;
11 }
12 nav ul {
13   background-color: red;
14 }
15
16 nav ul li {
17   float:left;
18   width: 25%;
19 }
```

Home

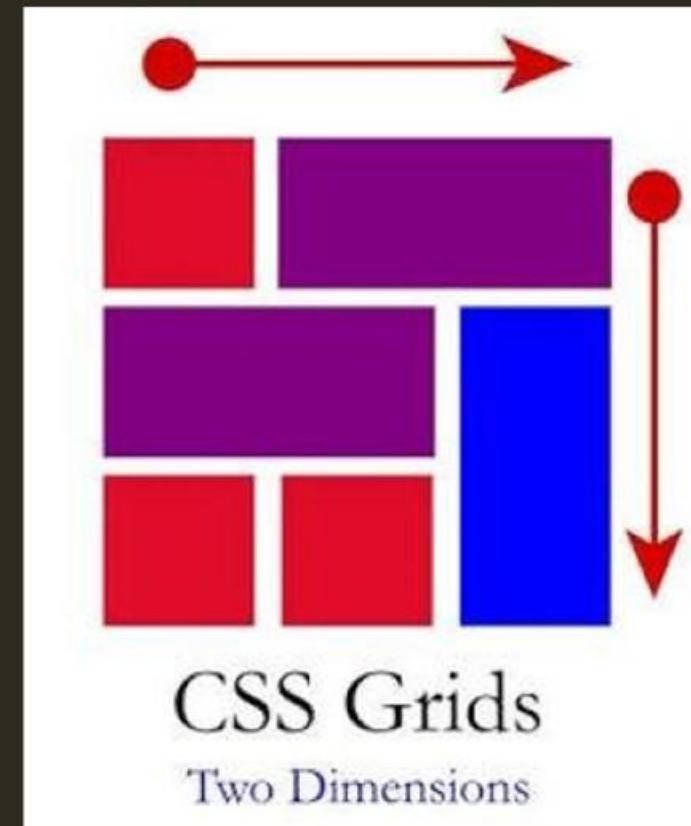
About

Services

Clientes

Pie de página

# FLEXBOX Y GRID



[https://www.w3schools.com/css/css3\\_flexbox.asp](https://www.w3schools.com/css/css3_flexbox.asp)

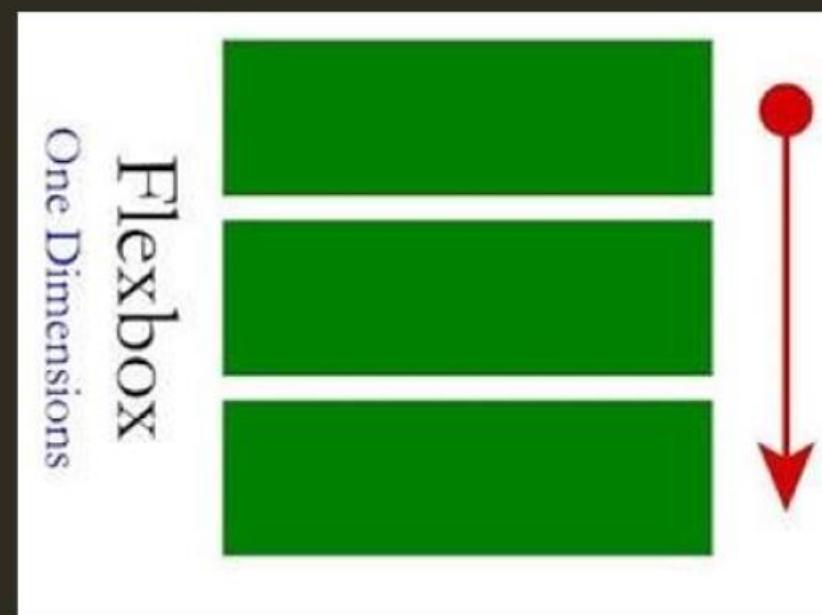
[https://www.w3schools.com/css/css\\_grid.asp](https://www.w3schools.com/css/css_grid.asp)



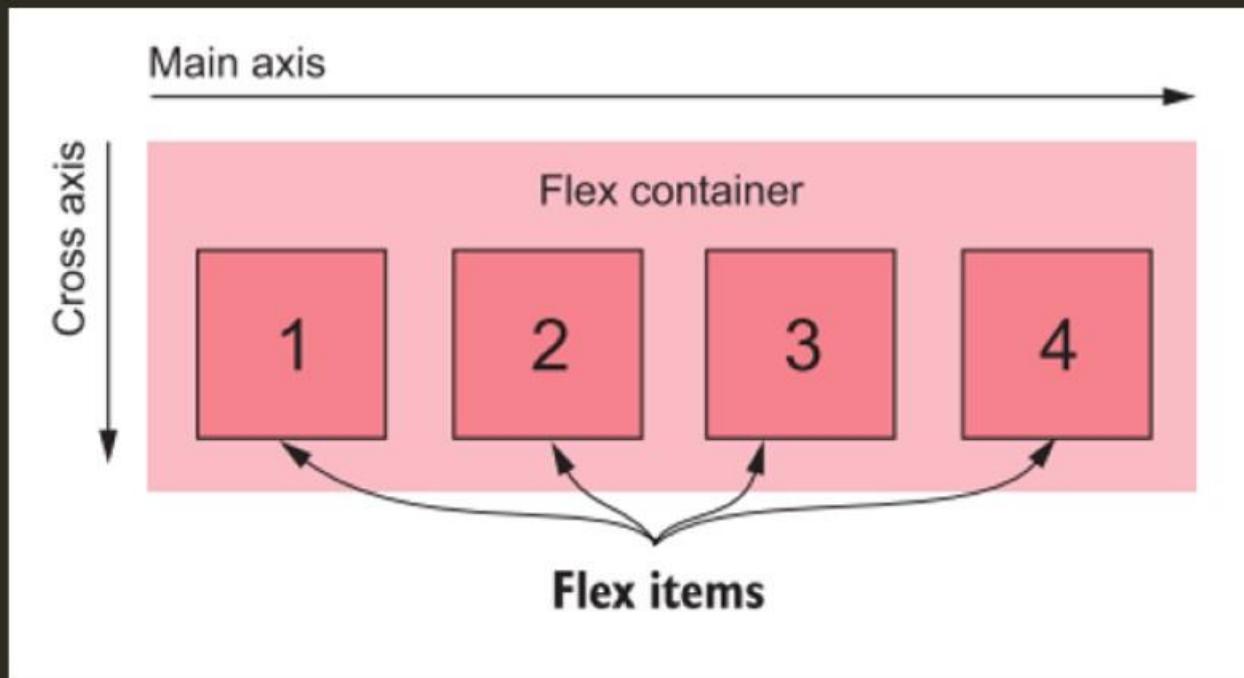
Dra. Laura Alicia Hernández Moreno, Dr. Juan Gabriel López Solórzano

# FLEXBOX

El Módulo de Caja Flexible, comúnmente llamado flexbox, fue diseñado como un modelo unidimensional de layout, y como un método que pueda ayudar a distribuir el espacio entre los ítems de una interfaz y mejorar las capacidades de alineación

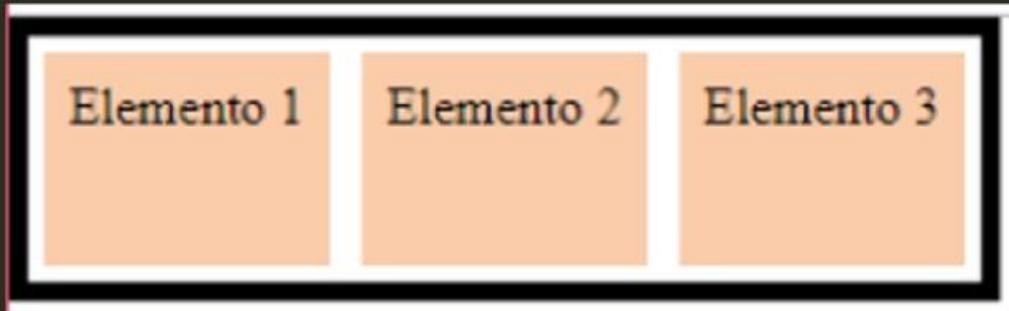


# EJES PRINCIPALES



# TERMINOLOGIA

Contenedor flexible y elementos flexibles



CSS

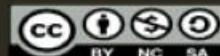
```
.contenedor {  
    display: flex;  
}
```

HTML

```
<div class="contenedor">  
    <div>Elemento 1</div>  
    <div>Elemento 2</div>  
    <div>Elemento 3</div>  
</div>
```

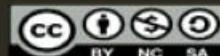
# PROPIEDADES DEL CONTENEDOR

- `flex-direction`
- `flex-wrap`
- `flex-flow`
- `justify-content`
- `align-items`
- `align-content`



# PROPIEDADES DEL CONTENEDOR

- flex-direction
- flex-wrap
- flex-flow
- justify-content
- align-items
- align-content



# Flex-direction

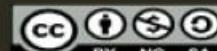
Especifica como colocar los elementos flexibles. Los valores posibles son: row, column, row-reverse, column-reverse

```
#site-nav {  
    display: flex;  
    background-color: #aeba12;  
    flex-direction: row;  
}
```

[Link1](#)[Link2](#)[Link3](#)[Link4](#)[Link5](#)[Link6](#)[Link7](#)[Link8](#)

```
#site-nav {  
    display: flex;  
    background-color: #aeba12;  
    flex-direction: column;  
}
```

[Link1](#)  
[Link2](#)  
[Link3](#)  
[Link4](#)  
[Link5](#)  
[Link6](#)  
[Link7](#)  
[Link8](#)



# Flex-wrap

Esta propiedad especifica si los elementos flexibles deben ajustarse o no.

```
#site-nav {  
    display: flex;  
    background-color: #aeба12;  
    flex-direction: row;  
    flex-wrap : no-wrap;  
}  
  
#site-nav li:nth-child(1) {  
    width: 90%;  
}
```

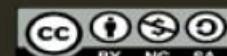
[Link1](#)

[Link2](#)[Link3](#)[Link4](#)[Link5](#)[Link6](#)[Link7](#)[Link8](#)

```
#site-nav {  
    display: flex;  
    background-color: #aeба12;  
    flex-direction: row;  
    flex-wrap : wrap;  
}  
  
#site-nav li:nth-child(1) {  
    width: 90%;  
}
```

[Link1](#)  
[Link3](#)[Link4](#)[Link5](#)[Link6](#)[Link7](#)[Link8](#)

[Link2](#)



Dra. Laura Alicia Hernández Moreno, Dr. Juan Gabriel López Solórzano

# Flex-flow

Es una propiedad corta para especificar las propiedades flex-direction y flex-wrap properties.

```
#site-nav {  
    display: flex;  
    background-color: #aeba12;  
    flex-flow: row wrap;  
}  
  
#site-nav li:nth-child(1) {  
    width: 90%;  
}
```



# Justify-content

Esta propiedad alinea los elementos flexibles sobre el eje principal

flex-start

[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#) [Link6](#) [Link7](#) [Link8](#)

space-around

[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#) [Link6](#) [Link7](#) [Link8](#)

flex-end

[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#) [Link6](#) [Link7](#) [Link8](#)

space-between

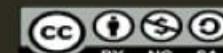
[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#) [Link6](#) [Link7](#) [Link8](#)

center

[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#) [Link6](#) [Link7](#) [Link8](#)

space-evenly

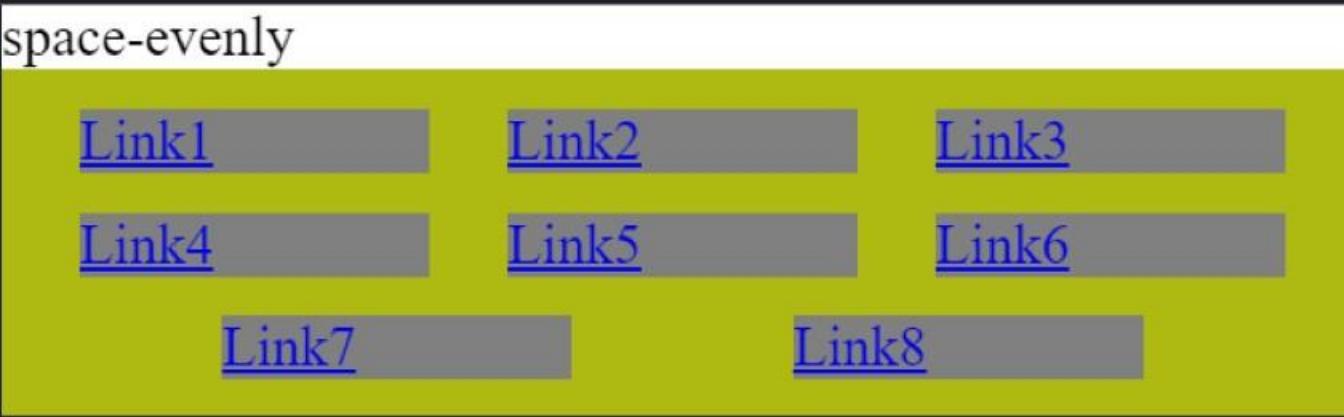
[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#) [Link6](#) [Link7](#) [Link8](#)



# Align-content

Esta propiedad alinea las líneas de contenido flexible (esta propiedad aplica para cuando el contenedor tiene varias líneas). Los posibles valores son: flex-start, flex-end, space-around, space-between, space-evenly y stretch (este es el default)

space-evenly

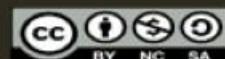


Link1      Link2      Link3  
Link4      Link5      Link6  
Link7      Link8



# Propiedades de los elementos flexibles

- **order.** Propiedad que se utiliza para cambiar el orden de los elementos flexibles
- **flex-grow**
- **flex-shrink**
- **flex-basis**
- **flex**
- **align-self**



# Flex-grow

Especifica que tanto crecerá un elemento flexible relativo al resto de los elementos. Valor inicial 0

```
1 <nav>
2   <ul id="site-nav">
3     <li id="l1"><a href="#">Link1</a></li>
4     <li id="l2"><a href="#">Link2</a></li>
5     <li id="l3"><a href="#">Link3</a></li>
6
7   </ul>
8 </nav>
```

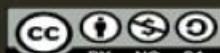
  

```
25 #l1 {
26   flex-grow: 1;
27 }
28
29 #l2 {
30   flex-grow: 5;
31 }
32
33 #l3 {
34   flex-grow: 1;
35 }
```

Link1

Link2

Link3



# Flex-shrink

Especifica que tanto se contrae un elemento flexible relativo al resto de los elementos. Valor por default 1.

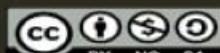
```
1 <nav>
2   <ul id="site-nav">
3     <li><a href="#">Link1</a></li>
4     <li><a href="#">Link2</a></li>
5     <li><a href="#">Link3</a></li>
6     <li><a href="#">Link4</a></li>
7     <li><a href="#">Link5</a></li>
8     <li><a href="#">Link6</a></li>
9     <li><a href="#">Link7</a></li>
10    <li><a href="#">Link8</a></li>
11  </ul>
12 </nav>
```

```
10 #site-nav {
11   display: flex;
12   background-color: #aeба12;
13   justify-content: center;
14 }
15
16 #site-nav li:nth-child(1) {
17   width: 800px;
18 }
19
20 #site-nav li:nth-child(2) {
21   flex-shrink: 0;
22   width: 800px;
23 }
```

Link1

Link2

Link3 Link4 Link5 Link6 Link7 Link8



Dra. Laura Alicia Hernández Moreno, Dr. Juan Gabriel López Solórzano

# Flex-basis

Especifica la longitud inicial del elemento flexible

```
1 <nav>
2   <ul id="site-nav">
3     <li><a href="#">Link1</a></li>
4     <li><a href="#">Link2</a></li>
5     <li><a href="#">Link3</a></li>
6     <li><a href="#">Link4</a></li>
7     <li><a href="#">Link5</a></li>
8     <li><a href="#">Link6</a></li>
9     <li><a href="#">Link7</a></li>
10    <li><a href="#">Link8</a></li>
11  </ul>
12 </nav>
```

```
10 #site-nav {
11   display: flex;
12   background-color: #aeба12;
13   justify-content: center;
14 }
15
16 #site-nav li:nth-child(1) {
17   flex-basis: 800px;
18 }
19
20 #site-nav li:nth-child(2) {
21   flex-shrink: 0;
22   flex-basis: 800px;
23 }
```

Link1

Link2

Link3 Link4 Link5 Link6 Link7 Link8



Dra. Laura Alicia Hernández Moreno, Dr. Juan Gabriel López Solórzano

# Flex

- Esta propiedad es una versión corta de las propiedades flex-grow, flex-shrink, y flex-basis.

## HTML

```
1 <nav>
2   <ul id="site-nav">
3     <li><a href="#">Link1</a></li>
4     <li><a href="#">Link2</a></li>
5     <li><a href="#">Link3</a></li>
6     <li><a href="#">Link4</a></li>
7     <li><a href="#">Link5</a></li>
8     <li><a href="#">Link6</a></li>
9   </ul>
10 </nav>
```

## CSS

```
11 #site-nav {
12   display: flex;
13   background-color: #aeба12;
14   justify-content: center;
15 }
16
17 #site-nav li:nth-child(3) {
18   flex: 0 0 20%;
19 }
```

Link1Link2

Link3

Link4Link5Link6



# Align-self

- Permite alinear independientemente cada elemento.

The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. The 'HTML' tab contains the following code:

```
1 <nav>
2   <ul id="site-nav">
3     <li><a href="#">Link1</a></li>
4     <li><a href="#">Link2</a></li>
5     <li><a href="#">Link3</a></li>
6     <li><a href="#">Link4</a></li>
7     <li><a href="#">Link5</a></li>
8     <li><a href="#">Link6</a></li>
9   </ul>
10 </nav>
```

The 'CSS' tab contains the following code:

```
11 #site-nav {
12   height: 40vh;
13   display: flex;
14   background-color: #aeба12;
15   justify-content: space-around;
16   align-items: center;
17 }
18
19 #site-nav li:nth-child(3) {
20   align-self: flex-end;
21 }
22
23
```

[Link1](#)

[Link2](#)

[Link4](#)

[Link5](#)

[Link6](#)

[Link3](#)



**GRID**

# Grid css

- Grid nos ofrece un sistema de rejilla compuesto de filas y columnas, que nos facilitan el diseño de páginas web.

**Header**

HOME OUR TEAM PROJECTS CONTACT

Search query

**Article heading**

**subsection**

Donec ut libero sed accu vehicula ultricies a non tortor. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean ut gravida lorem. Ut turpis felis, pulvinar a semper sed, adipiscing id dolor.

Pelentesque auctor nisi id magna consequat sagittis. Curabitur dapibus, enim sit amet elit pharetra tincidunt feugiat nisl imperdiet. Ut convallis libero in urna ultrices accumsan. Donec sed odio eros.

**Another subsection**

Donec viverra mi quis quam pulvinar at malesuada arcu rhoncus. Cum sociis natoque penatibus et manis dis parturient montes, nascetur ridiculus mus. In rutrum accumsan ultricies. Mauris vitae nisi at sem facilisis semper ac in est.

Vivamus fermentum semper porta. Nunc diam velit, adipiscing ut tristique vitae sagittis vel odio. Maecenas convallis ullamcorper ultricies. Curabitur ornare, ligula semper consectetur sagittis, nisi diam iaculis velit, is fringilla sem nunc vel mi.

©Copyright 2050 by nobody. All rights reversed.

**Related**

- [Oh I do like to be beside the seaside](#)
- [Oh I do like to be beside the sea](#)
- [Although in the North of England](#)
- [It never stops raining](#)
- [Oh well...](#)



## Contenedor grid

- Para que un contenedor se comporte como tipo grid se debe especificar su propiedad display a: grid o inline-grid.



# Especificando las columnas para el grid

- Para indicar las columnas se utiliza la propiedad `grid-template-columns`, en esta se especifica el ancho que tendrá cada una de ellas.

The screenshot shows a code editor with two tabs: 'HTML' and 'CSS'. The 'HTML' tab contains the following code:

```
1 <body>
2   <header></header>
14  <main></main>
19  <aside></aside>
22  <footer></footer>
23 </body>
```

The 'CSS' tab contains the following code:

```
16
17 body {
18   display: grid;
19   grid-template-columns: 1fr 1fr;
20 }
```

Below the code editor, a preview of the web page is shown. The page has a header, a main content area, and an aside section. The aside section contains the text "Empresa Acme" and a list of five links. The main content area contains the text "Articulo Interesante". At the bottom, there is a green footer bar with the text "Aside" on the left and "Pie de pagina" on the right. The entire page has a light orange background.

# Especificando renglones para el grid

- Para especificar renglones se puede utilizar la propiedad `grid-template-rows`

The screenshot shows a code editor with two tabs: HTML and CSS. The HTML tab contains the following code:

```
1 <body>
2   <header></header>
14  <main></main>
19  <aside></aside>
22  <footer></footer>
23 </body>
```

The CSS tab contains the following code:

```
16
17 body {
18   display: grid;
19   grid-template-columns: 1fr 1fr;
20   grid-template-rows: auto 100px;
21   grid-gap: 1em;
22 }
```

Below the code editor, the browser's layout is displayed. It consists of four main sections arranged in a grid:

- Header:** A light blue header bar.
- Main Content:** A large orange area containing the text "Articulo Interesante".
- Aside:** A green sidebar containing the text "Aside".
- Footer:** An orange footer bar containing the text "Pie de pagina".

The layout is defined by the CSS rules in the code editor, specifically using the `grid-template-rows` property to create two rows: one for the main content and one for the footer.

# Controlando espacios entre columnas y renglones

- Para especificar el espacio entre columnas se puede utilizar la propiedad grid-gap. Si tiene un solo valor se aplicar el espacio tanto para columnas y filas. Pero si se indica dos valores el primero aplica a filas y el segundo a columnas.

The screenshot shows a code editor interface with two tabs: 'HTML' and 'CSS'. The 'HTML' tab contains the following code:

```
1 <body>
2   <header></header>
3   <main></main>
4   <aside></aside>
5   <footer></footer>
6 
7 </body>
```

The 'CSS' tab contains the following code:

```
16
17 body {
18   display: grid;
19   grid-template-columns: 1fr 1fr;
20   grid-gap: 1em;
21 }
```

Below the code editor, a preview of the web page is shown. It consists of two columns. The left column is labeled 'Empresa Acme' and contains a list of five links: link1, link2, link3, link4, and link5. The right column is labeled 'Articulo Interesante'. At the bottom of the page, there is a green bar labeled 'Aside' and an orange bar labeled 'Pie de pagina'.

# Abarcar mas de una celda (filas y columnas)

- Para este fin se utilizan las propiedades grid-column y grid-row

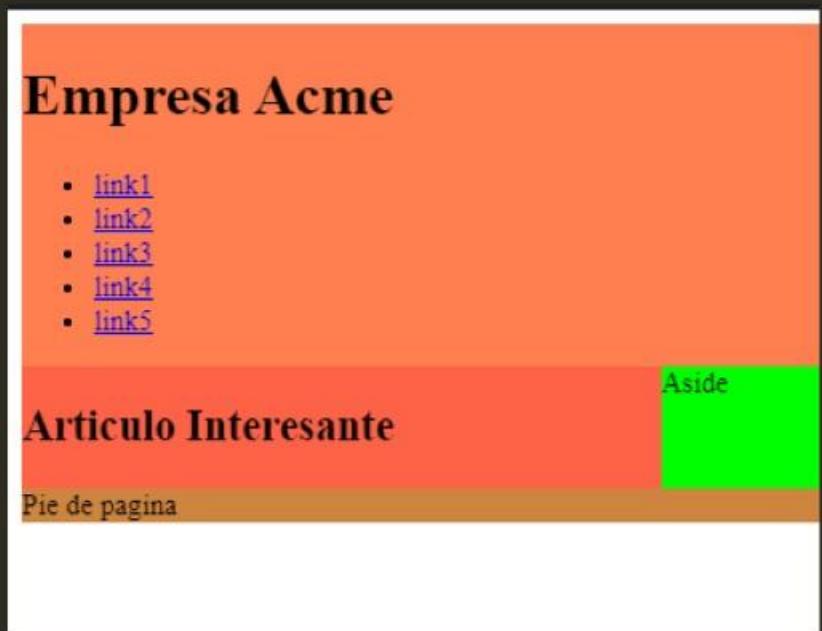
```
HTML
<header>
<h1>Empresa Acme</h1>
<ul>
<li>link1</li>
<li>link2</li>
<li>link3</li>
<li>link4</li>
<li>link5</li>
</ul>
<article>
<h2>Articulo Interesante</h2>
</article>
<aside>
<p>Aside</p>
</aside>
<footer>
<p>Pie de pagina</p>
</footer>
```

```
CSS
17 * body {
18   display: grid;
19   grid-template-columns: 4fr 1fr;
20   grid-template-rows: auto 100px;
21   grid-gap: 1em;
22   justify-content: space-around;
23 }
24
25 * header {
26   grid-column: 1 / -1;
27 }
28
29 * footer {
30   grid-column: 1 / -1;
31 }
```

olórzano

# Grid areas

- Para definir el espacio que ocupa un elemento dentro del grid se puede utilizar grid-template-areas



```
body {  
    display: grid;  
    grid-template-columns: 4fr 1fr;  
    grid-template-areas : "header header"  
                         "main aside"  
                         "footer footer"  
}  
  
header {  
    grid-area : header;  
}  
main {  
    grid-area: main;  
}  
aside {  
    grid-area: aside;  
}  
footer {  
    grid-area: footer;  
}
```

# JAVASCRIPT



# JAVASCRIPT

---

Javascript es el lenguaje para la web

---

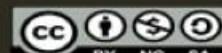
La mayoría de los sitios web utilizan javascript

---

Todos los navegadores modernos incluyen un interprete de javascript

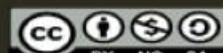
---

Javascript es un lenguaje de alto nivel, dinámico, interpretado, soporta la programación orientada a objetos y la programación funcional.

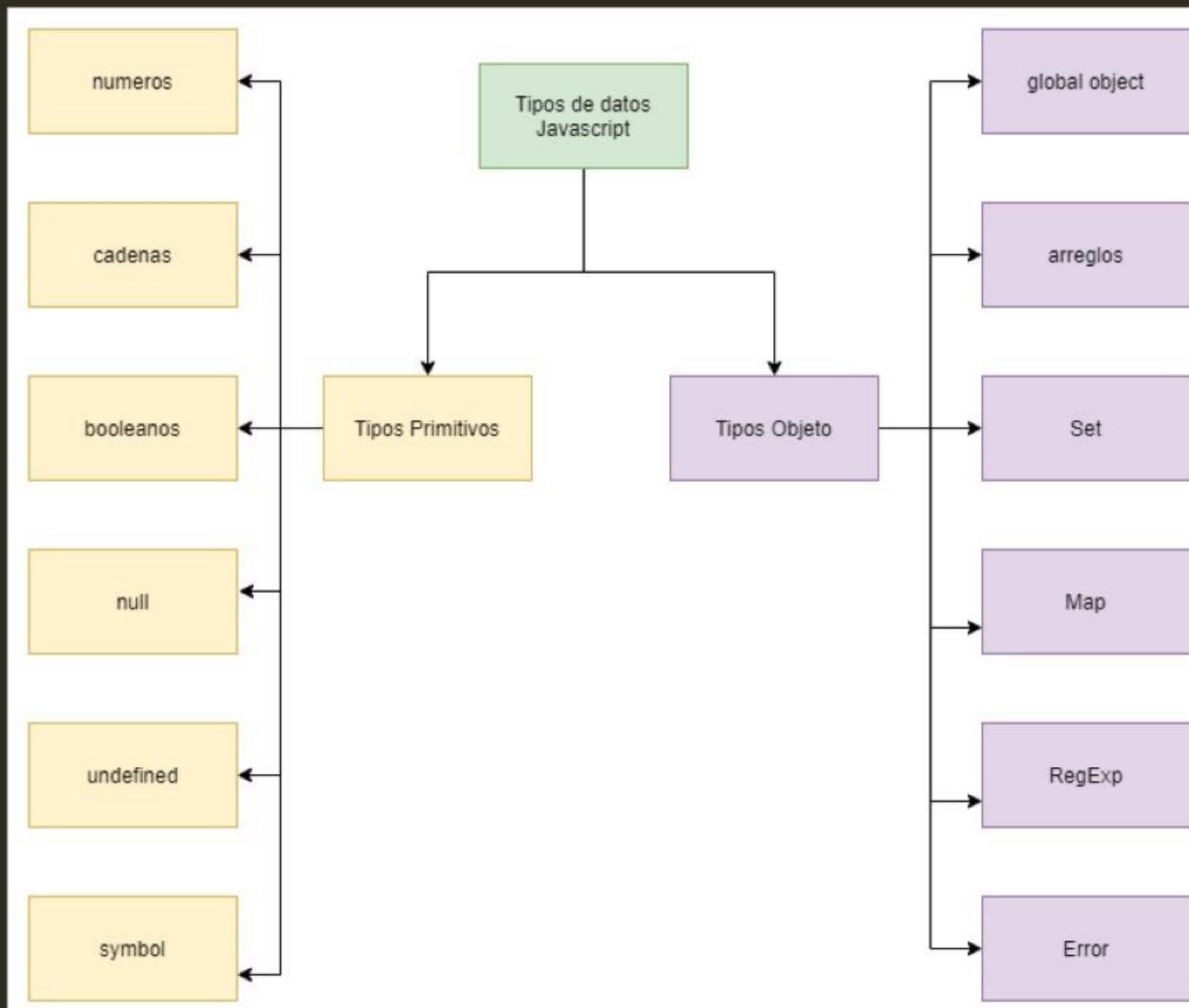


# COMENTARIOS

```
// Esto es un comentario de una línea  
  
/* Esto es un comentario  
multilinea.  
*/
```



# Tipos de datos javascript



# Operaciones Aritméticas y números especiales

```
> 8 + 2
```

```
< 10
```

```
> 8 - 2
```

```
< 6
```

```
> 8 * 2
```

```
< 16
```

```
> 8 / 2
```

```
< 4
```

```
> 8 % 2
```

```
< 0
```

```
> 7 / "hola"
```

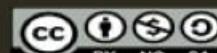
```
< NaN
```

```
> 1 / 0
```

```
< Infinity
```

```
> -1 / 0
```

```
< -Infinity
```



# Cadenas

```
//cadenas  
"hola"  
'nueva'  
'otra cadena'
```

```
//cadenas con secuencias de escape  
"primera línea\nsegunda línea"
```

```
//operaciones con cadenas  
"facpya " + " programa "
```

```
//las plantillas literales pueden incluir expresiones  
'El sueldo total es: ${1200 + 300}'
```



# Operador typeof

```
5
6 console.log(typeof 3);
7 console.log(typeof "prueba");
```

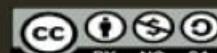
The screenshot shows the Google Chrome DevTools interface with the 'Console' tab selected. The console output displays two lines of text: 'number' and 'string'. Each line is preceded by the file name 'app.js:6' and 'app.js:7' respectively, indicating the source of the logged values.

Output	Source
number	app.js:6
string	app.js:7

# Valores booleanos, operadores de comparación y operadores lógicos

```
> 4 == 3  
< false  
  
> 4 > 3  
< true  
  
> 4 >= 3  
< true  
  
> 4 < 3  
< false  
  
> 4 <= 3  
< false  
  
> 4 != 3  
< true  
  
>
```

```
> 4 == 3 || 4 > 2  
< true  
  
> 4 == 3 && 4 > 2  
< false  
  
> !(4 == 3) && 4 > 2  
< true  
  
>
```



# Operador ternario

```
> 5 > 2 ? 1 : 0
```

```
< 1
```

```
> -5 > 2 ? 1 : 0
```

```
< 0
```

# DECLARACIÓN DE VARIABLES Y CONSTANTES

```
//declaración de variables
let i;
let nombre;
let edad=33, ciudad="monterrey";

//declaración de constantes
const FACTOR = 1.45;
const VEL = 1500;
```

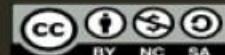
# DECLARACIÓN DE VARIABLES Y CONSTANTES

```
let apps = ['windows 10', 'vscode', 'oracle']

let computadora = {
    "marca": "hp",
    "discoDuro": "1tb",
    "precio": 12000,
    "softwareInstalado": apps
}
```

# Funciones útiles

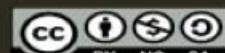
```
// para imprimir en la consola  
console.log("este texto será mostrado en la consola");  
  
// para solicitar datos de entrada  
let nombre = prompt("Ingresa tu nombre:")  
  
// para convertir cadena a numero  
let edad = Number(prompt("Ingresa tu edad:"))  
  
// para escribir en el documento  
document.write("<h1>Esto es un título</h1>")
```



# Sentencias condicionales

```
// if simple  
if( condicion ) {  
    sentencias  
}  
  
// if compuesto  
if( condicion ) {  
    sentencias  
}  
else {  
    sentencias  
}
```

```
//if anidado  
if( condicion ) {  
    sentencias  
}  
else if (condicion) {  
    sentencias  
}  
else if (condicion) {  
    sentencias  
}  
else {  
    sentencias  
}
```



# Sentencia condicional múltiple switch

```
let estadoCivil = "casado";

switch (estadoCivil) {
    case "soltero":
        //sentencias
        break;
    case "casado":
        //sentencias
        break;
    case "viudo":
        //sentencias
        break;
    default:
        //sentencias
        break;
}
```



# Ciclos

```
// while
let numero = 1
while (numero < 11) {
    console.log(numero);
    numero++;
}

// do while
numero = 1
do {
    console.log(numero);
    numero++;
}while (numero < 11);

//for
for (let index = 1; index < 11; index++) {
    console.log(index);
}
```

# Funciones

- Las funciones en general es asociar un bloque de código a un nombre.

```
//declaración de una función similar a una variable
const cuadrado = function(x){
    return x * x;
};

//declaración de una función usando function
function cuadrado2(x){
    return x * x;
}

// declaración de funciones flecha
const cuadrado3 = (x) => {
    return x * x;
};

const cuadrado4 = x => x * x
```

# Objeto document

- Referenciar a una colección de elementos basados en su tag name

```
document.getElementsByTagName("h2");
```

- Referenciar una colección de elementos basados en el nombre de una clase

```
document.getElementsByClassName("title")
```

- Referenciar a un elemento basado en su ID

```
document.getElementById("titulo-principal")
```

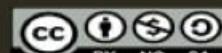
# Propiedades para cambiar contenido

- innerHTML
- outerHTML
- textContent

```
let p = document.getElementById("parrafo");

console.log(p.textContent);
console.log(p.innerHTML);
console.log(p.outerHTML);

p.textContent= "nuevo";
```



# DJANGO FRAMEWORK WEB



# DJANGO

Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como modelo–vista–controlador (MVC).

# CREAR Y ACTIVAR UN AMBIENTE VIRTUAL

```
pc@jgls MINGW64 ~/Desktop/django
$ python -m venv env
```

```
pc@jgls MINGW64 ~/Desktop/django
$ source env/Scripts/activate
```

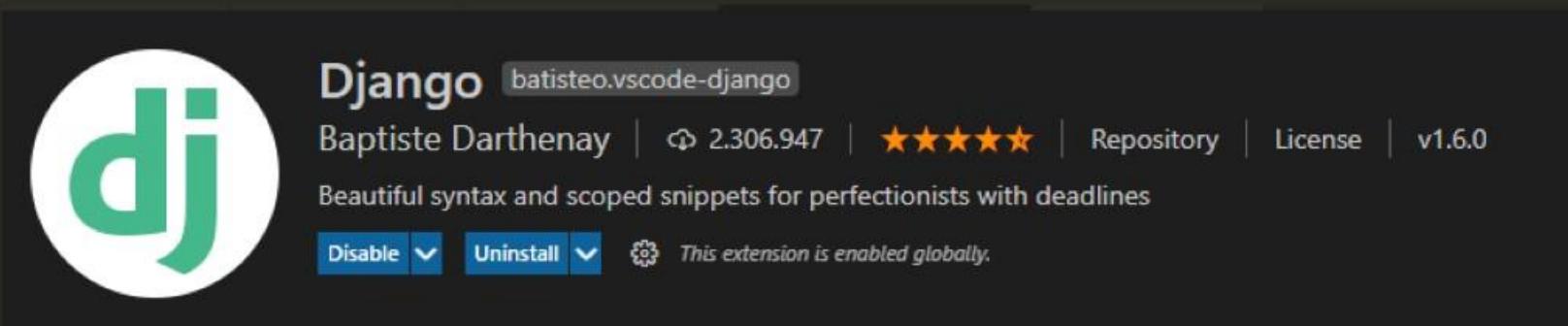
# INSTALAR Y VERIFICAR VERSION DJANGO

```
(env)
pc@jgls MINGW64 ~/Desktop/django
$ pip install django
```

---

```
(env)
pc@jgls MINGW64 ~/Desktop/django
$ python -m django --version
3.2.1
```

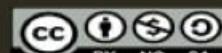
# INSTALAR Y CONFIGURAR EXTENSIÓN DJANGO



```
"files.associations": {  
    "**/*.html": "html",  
    "**/templates/**/*.*": "django-html",  
    "**/templates/**/*": "django-txt",  
    "**/requirements{/**,*}.{txt,in)": "pip-requirements"  
},  
"emmet.includeLanguages": {"django-html": "html"}
```

# INICIAR UN PROYECTO

```
(env)
pc@jgls MINGW64 ~/Desktop/django
$ django-admin startproject website
```



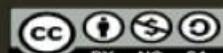
# CREANDO UNA BASE DE DATOS

- Django incluye varias aplicaciones por default (autenticación y administración de usuarios y administración del sitio). Para ello se necesita una base de datos.

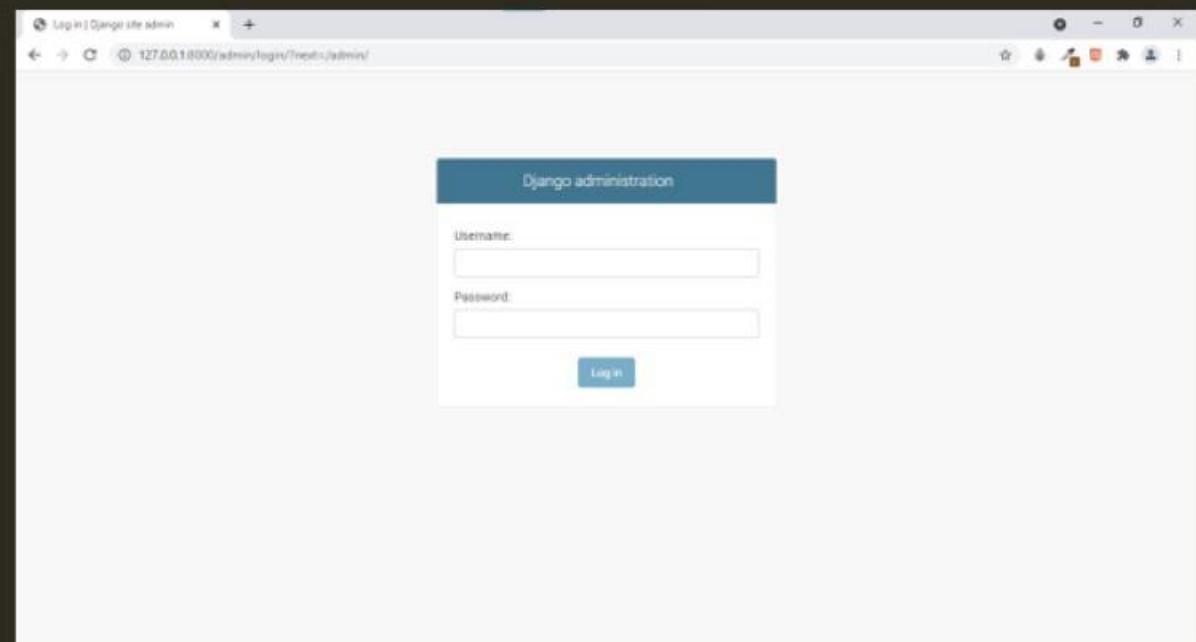
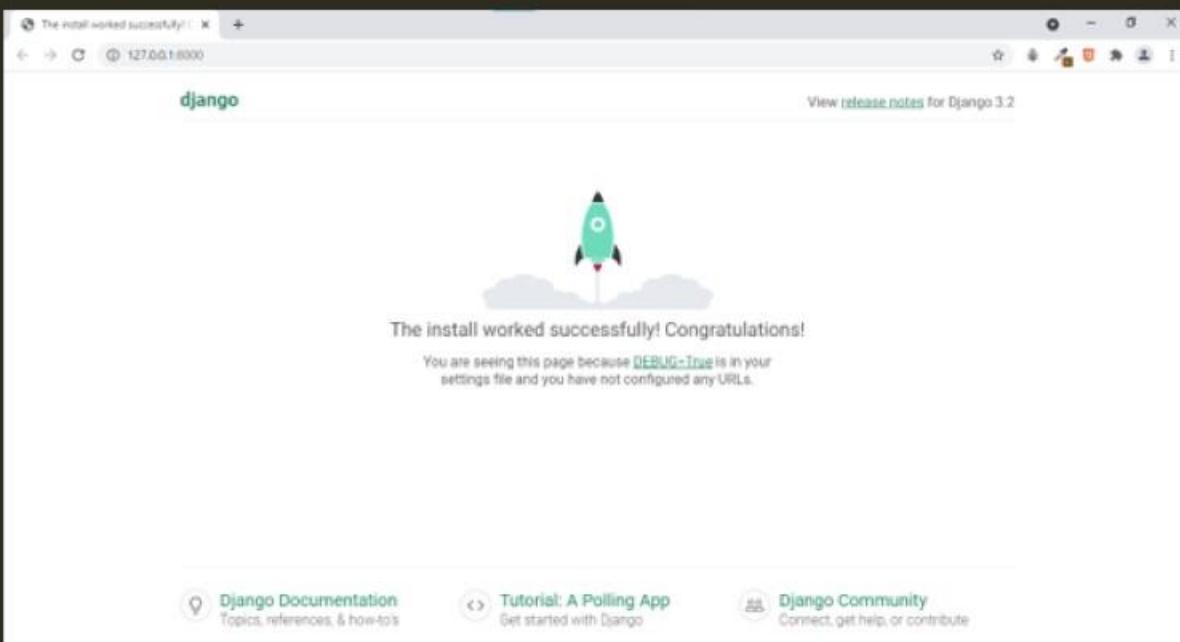
```
(env)
pc@jgls MINGW64 ~/Desktop/django/website
$ python manage.py migrate
```

# EJECUTAR EL SERVIDOR

```
(env)
pc@jgls MINGW64 ~/Desktop/django/website
$ python manage.py runserver
```

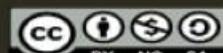


# PROYECTO DJANGO FUNCIONANDO



# USANDO EL SITIO DE ADMINISTRACIÓN

```
(env)
pc@jgls MINGW64 ~/Desktop/django/website
$ python manage.py createsuperuser
```



# ACCEDIENDO A LA BASE DE DATOS SQLITE3

Descargar las herramientas Sqlite3 para Windows.

<https://www.sqlite.org/download.html>



Abrir la base de datos

A screenshot of a terminal window titled "MINGW64:/c/gpo51/django/website". The user is running the command "winpty ./sqlite3 db.sqlite3". The output shows the SQLite version (3.36.0), the date and time (2021-06-18 18:36:39), and the prompt "Enter ".help" for usage hints.". The user then runs ".tables" which lists tables: auth\_group, auth\_group\_permissions, auth\_permission, auth\_user, auth\_user\_groups, auth\_user\_permissions, django\_admin\_log, django\_content\_type, django\_migrations, django\_session. The user then runs "select \* from auth\_user;" and "sqlite> .schema auth\_user" to view the schema of the auth\_user table, which includes columns like id, password, last\_login, first\_name, last\_name, email, is\_staff, is\_active, and is\_superuser. Finally, the user runs ".quit" to exit the SQLite shell.

# BD BROWSER SQLITE3

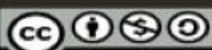
<https://sqlitebrowser.org/dl/>

The screenshot shows the DB Browser for SQLite interface with the following details:

- Database Structure:** Shows the schema of the database.
- Tables (12):**
  - auth\_group
  - auth\_group\_permissions
  - auth\_permission
  - auth\_user
  - auth\_user\_groups
  - auth\_user\_user\_permissions
  - django\_admin\_log
  - django\_content\_type
  - django\_migrations
  - django\_session
  - productos\_producto
  - sqlite\_sequence
- Indices (15):**
  - auth\_group\_permissions\_group\_id\_b1...
  - auth\_group\_permissions\_group\_id\_pe...
  - auth\_group\_permissions\_permission\_id...
  - auth\_permission\_content\_type\_id\_2f...
  - auth\_permission\_content\_type\_id\_c0...
  - auth\_user\_groups\_group\_id\_973954h
  - auth\_user\_groups\_user\_id\_dae12ed0b
  - auth\_user\_groups\_user\_id\_group\_id\_d...
  - auth\_user\_user\_permissions\_permission\_id...
  - auth\_user\_user\_permissions\_user\_id\_d...
  - auth\_user\_user\_permissions\_user\_id\_d...
  - django\_admin\_log\_content\_type\_id\_e...
  - django\_admin\_log\_user\_id\_c564ebef
  - django\_content\_type\_app\_label\_mode...
  - django\_session\_expire\_date\_a5c5d663
- Views (0)**
- Triggers (0)**

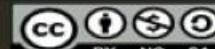
The schema code is as follows:

```
CREATE TABLE "auth_group" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "name" varchar(150) NOT NULL UNIQUE);
CREATE TABLE "auth_group_permissions" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "group_id" integer NOT NULL REFERENCES "auth_group" ("id"), "permission_id" integer NOT NULL REFERENCES "auth_permission" ("id"));
CREATE TABLE "auth_permission" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "content_type_id" integer NOT NULL REFERENCES "django_content_type" ("id"), "codename" varchar(100) NOT NULL, "name" varchar(50) NOT NULL, "model" varchar(100));
CREATE TABLE "auth_user" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "password" varchar(128) NOT NULL, "last_login" datetime NULL, "is_superuser" integer NOT NULL, "username" varchar(150) NOT NULL, "first_name" varchar(30), "last_name" varchar(30), "email" varchar(254));
CREATE TABLE "auth_user_groups" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "user_id" integer NOT NULL REFERENCES "auth_user" ("id"), "group_id" integer NOT NULL REFERENCES "auth_group" ("id"));
CREATE TABLE "auth_user_user_permissions" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "user_id" integer NOT NULL REFERENCES "auth_user" ("id"), "permission_id" integer NOT NULL REFERENCES "auth_permission" ("id"));
CREATE TABLE "django_admin_log" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "action_time" datetime NOT NULL, "object_id" text NULL, "object_repr" varchar(255), "change_message" longtext, "content_type_id" integer NOT NULL REFERENCES "django_content_type" ("id"), "user_id" integer NOT NULL REFERENCES "auth_user" ("id"));
CREATE TABLE "django_content_type" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "app_label" varchar(100) NOT NULL, "model" varchar(100));
CREATE TABLE "django_migrations" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "app" varchar(255) NOT NULL, "name" varchar(255) NOT NULL, "applied" datetime NOT NULL);
CREATE TABLE "django_session" ("session_key" varchar(40) NOT NULL PRIMARY KEY, "session_data" text NOT NULL, "expire_date" datetime NOT NULL);
CREATE TABLE "productos_producto" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "descripcion" varchar(30) NOT NULL, "presentacion" varchar(30) NOT NULL, "precio" float NOT NULL, "existencias" integer NOT NULL, "ultimo_compra" date NOT NULL, "ultimo_venta" date NOT NULL);
CREATE TABLE "sqlite_sequence(name,seq)"
```



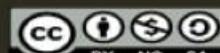
# ACCEDIENDO A LA BASE DE DATOS DESDE PYTHON

```
(env)
jg@hp MINGW64 /c/repos/django-proyectos/django-proyecto1/website_root
$ python manage.py shell
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from productos.models import Producto
>>> Producto.objects.values()
<QuerySet []>
>>> p = Producto(descripcion="producto 1", presentacion="caja", precio=20)
>>> p.save()
>>> Producto.objects.values()
<QuerySet [{"id": 7, "descripcion": "producto 1", "presentacion": "caja", "precio": 20.0}]>
>>> p = Producto(descripcion="producto 2", presentacion="botella", precio=22)
>>> p.save()
>>> Producto.objects.values()
<QuerySet [{"id": 7, "descripcion": "producto 1", "presentacion": "caja", "precio": 20.0}, {"id": 8, "descr
>>> Producto.objects.filter(precio=20).values()
<QuerySet [{"id": 7, "descripcion": "producto 1", "presentacion": "caja", "precio": 20.0}]>
>>> Producto.objects.filter(precio=22).delete()
(1, {'productos.Producto': 1})
>>> Producto.objects.values()
<QuerySet [{"id": 7, "descripcion": "producto 1", "presentacion": "caja", "precio": 20.0}]>
>>> Producto.objects.filter(id=7).update(presentacion="cambio descripcion")
1
>>> Producto.objects.values()
<QuerySet [{"id": 7, "descripcion": "producto 1", "presentacion": "cambio descripcion", "precio": 20.0}]>
>>> █
```



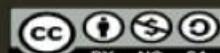
# PRIMERA APLICACIÓN DJANGO

```
(env)
pc@jgls MINGW64 ~/Desktop/django/website_root
$ python manage.py startapp productos
```



# AGREGANDO LA APP AL PROYECTO (SETTINGS.PY)

```
INSTALLED_APPS = [
    'productos.apps.ProductosConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]
```



# ESTRUCTURA DE UNA APLICACIÓN

```
    ✓ productos
        > __pycache__
        > migrations
        ⚡ __init__.py
        ⚡ admin.py
        ⚡ apps.py
        ⚡ models.py
        ⚡ tests.py
        ⚡ views.py
```

# PRIMERA VISTA

```
1 from django.shortcuts import render
2 from django.http import HttpResponseRedirect
3
4 # Create your views here.
5 def index(request):
6     return HttpResponseRedirect("<h1>Mi primera vista</h1>")
```

# CONFIGURANDO URLs PARA ACCEDER A NUESTRA VISTA

- Crear un archivo llamado urls.py dentro de nuestra app

```
from django.urls import path
from . import views

urlpatterns = [
    path('',views.index,name="index"),
]
```

# AGREGAR AL SITIO LAS URL DE NUESTRA APP

```
from django.contrib import admin
from django.urls import include, path

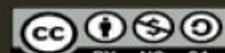
urlpatterns = [
    path('',include('productos.urls')),
    path('admin/ ', admin.site.urls),
]
```

# VISTA DESPLEGADA



# CREANDO UN MODELO PARA PRODUCTO

```
# Create your models here.  
class Producto(models.Model):  
    descripcion = models.CharField(max_length=60)  
    presentacion = models.CharField(max_length=50)  
    precio = models.FloatField()
```



# REGISTRAR EL MODELO PARA EL SITIO ADMIN

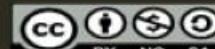
```
from django.contrib import admin
from .models import Producto
# Register your models here.
admin.site.register(Producto)
```



# CORRER LAS MIGRACIONES PARA PRODUCTO

```
(env)
pc@jgls MINGW64 ~/Desktop/django/website_root
$ python manage.py makemigrations
←[36;1mMigrations for 'productos':←[0m
←[1mproductos\migrations\0001_initial.py←[0m
  - Create model Producto
```

```
(env)
pc@jgls MINGW64 ~/Desktop/django/website_root
$ python manage.py migrate
←[36;1mOperations to perform:←[0m
←[1m  Apply all migrations: ←[0madmin, auth, contenttypes, productos, sessions
←[36;1mRunning migrations:←[0m
  Applying productos.0001_initial...←[32;1m OK←[0m
```



# PERSONALIZANDO ADMIN PRODUCTO

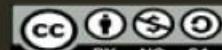
```
1  from django.contrib import admin
2  from .models import Producto
3  # Register your models here.
4  class ProductoAdmin(admin.ModelAdmin):
5      list_display = ('descripcion','presentacion')
6      ordering = ('presentacion',)
7      search_fields = ('descripcion',)
8
9  admin.site.register(Producto,ProductoAdmin)
```

# plantillas y contenido static

- Dentro del proyecto crear las carpetas templates y static.
- Configurar django (settings.py) para que reconozca las carpetas

```
from pathlib import Path
import os
TEMPLATES = [
{
    'BACKEND': 'django.template.backends.django.
DjangoTemplates',
    'DIRS': [
        os.path.join(BASE_DIR, 'website/templates')
    ,
],
    # Static files (CSS, JavaScript, Images)
    # https://docs.djangoproject.com/en/3.2/howto/static-files/
    STATIC_URL = '/static/'

    STATICFILES_DIRS = [ os.path.join(BASE_DIR, 'website/static'), ]
```



# Ejemplo de plantilla

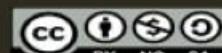
```
1  {% load static %}  
2  <!DOCTYPE html>  
3  <html lang="en">  
4  <head>  
5      <meta charset="UTF-8">  
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">  
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">  
8      <title>Esta pagina es la mejor</title>  
9      <link rel="stylesheet" href="{% static 'css/estilos.css' %}">  
10 </head>  
11 <body>  
12     <h1>dldl</h1>  
13       
14     <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Harum repellat vel molestiae dolore  
15       magni. Facere velit libero qui, incidunt earum dicta ducimus optio ut tenetur quasi consectetur  
16       dolorum, veritatis corporis?</p>
```



# ACTUALIZANDO LA VISTA

```
from django.shortcuts import render
from django.http import HttpResponseRedirect

# Create your views here.
def index(request):
    return render(request, 'main.html')
```



# PREPARANDO PROYECTO DJANGO PARA GITHUB

```
❖ .gitignore X  
❖ .gitignore  
1   __pycache__/  
2   *.sqlite3
```

