

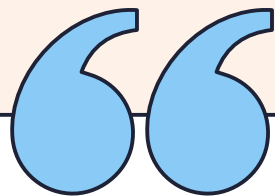


# Programación Web Básica.

U  
N  
Q



Prof. Jorge Nicolás Terradillos  
Prof. Fernando Blanco



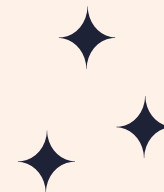
PARA EMPEZAR... ¿QUIÉN ES  
QUIÉN?

¡RONDA DE  
PRESENTACIÓN!



¿De dónde venimos? ¿Qué  
esperamos aprender?





# El curso

## Horario:

Miércoles de 16 a 18 hs. – Clases por videollamada

**Comisión A:** Viernes de 16 a 18 hs. – Clases presenciales en “La Bernalesa”

**Comisión B:** Martes de 16 a 18 hs. – Clases presenciales en “La Bernalesa”

Docentes: Prof. Jorge Nicolás Terradillos y Prof. Fernando Blanco

Contacto: [jorgenterradillos@gmail.com](mailto:jorgenterradillos@gmail.com)

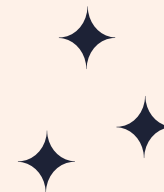
## Plataformas que usaremos:

Campus de UNQ: repositorio de material teórico y actividades, foros de consulta

Gmail: para comunicarnos respecto a las clases y facilitar material

Google Meet: para videollamadas





# ¿Cómo se acredita el curso?

## Prácticas.



A lo largo de las clases se les dejarán archivos con ejercicios prácticos opcionales

Estos mismos se trabajarán en clase y servirán para ensayar todo lo visto en clase

## Proyecto integrador

Desarrollar e implementar todo lo visto en un proyecto grupal (necesario para completar el curso)



# Temario del curso.

## 01 Conceptos básicos

Introducción a programación web, sitios básicos en HTML



Funcionamiento de la web, maquetado de sitios, lenguaje de marcado

## 02 CSS

Estructura, estilo y diseño de sitios web.



Elementos gráficos, flujo y orden de elementos

## 03 JavaScript

Programación de sitios web dinámicos



Manejo de objetos web, programación del front-end

## 04 GIT

Taller de gestión de proyectos web



Repositorios virtuales y control de versiones



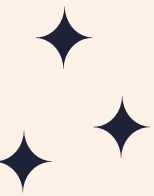


01

# Conceptos básicos de web

¿Qué es un sitio web? ¿Cómo funciona?





# ¿Que es un sitio web?

Es tan solo un conjunto de archivos alojados en una computadora (servidor) al que pueden acceder otras computadoras (clientes)

Combina elementos textuales y gráficos





# ¿Cómo funciona?

La tecnología fundamental es la arquitectura **cliente-servidor** que determina como interactúan las computadoras. Simplificado, esa arquitectura se ve así

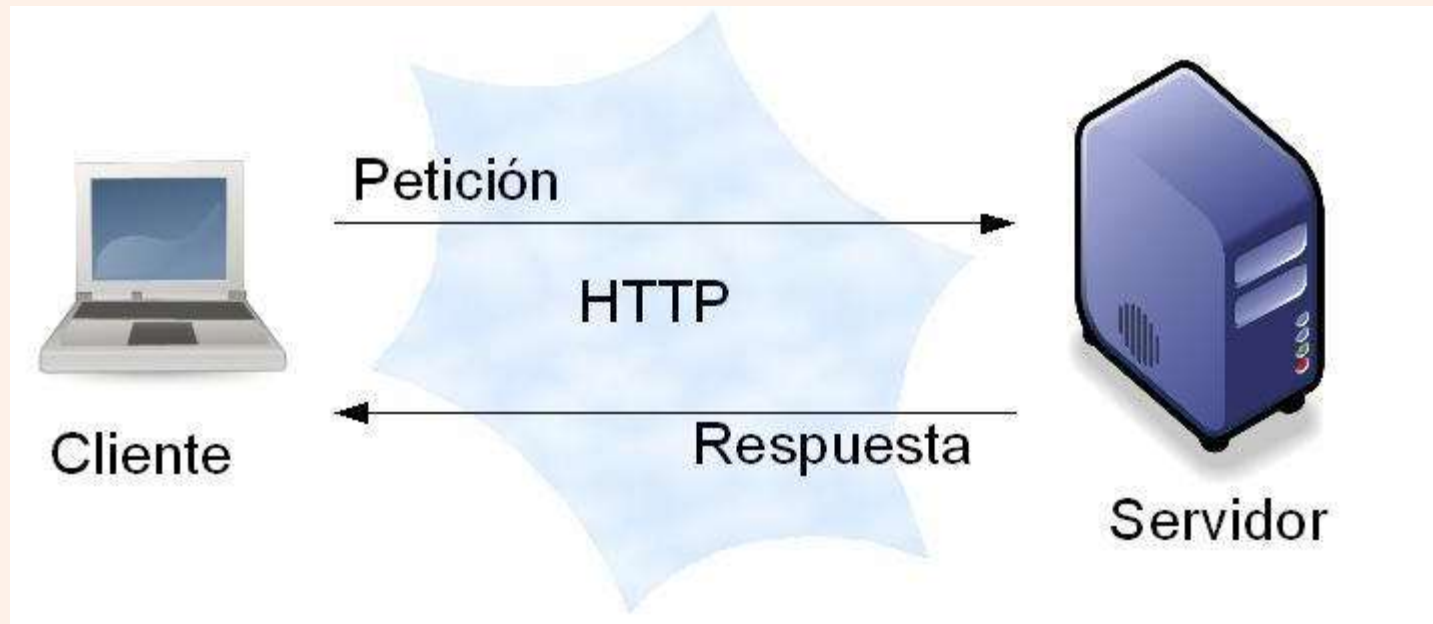


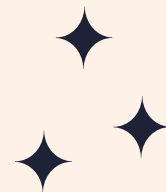




# El servidor

Los servidores son computadores que almacenan páginas web, sitios o aplicaciones

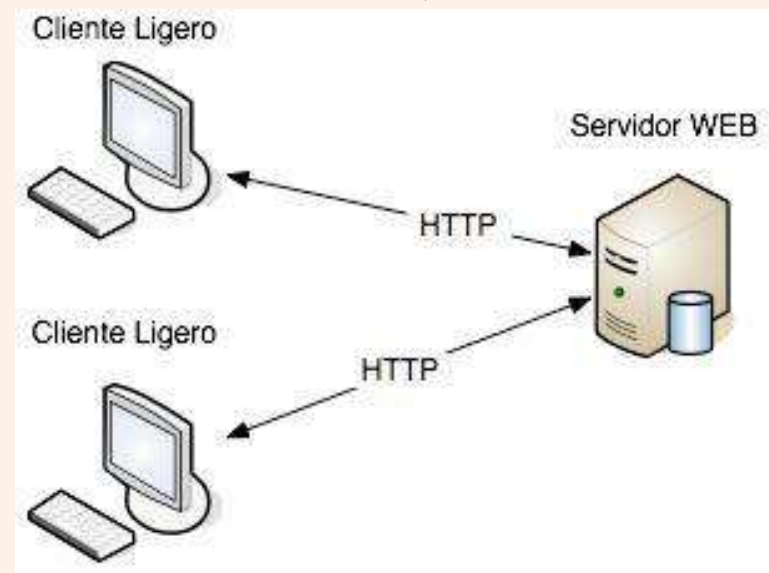




# El cliente

Los clientes son dispositivos de los usuarios conectados a Internet (por ejemplo, tu ordenador conectado a la red Wi-Fi o el teléfono conectado a la red de telefonía móvil)

Cuando un dispositivo cliente quiere acceder a una página web, una copia de la página web se descarga desde el servidor en el equipo cliente y se muestra en el navegador web del usuario





# Protocolos

Para interactuar en red, los dispositivos tienen ciertas reglas de comunicación o **protocolos**

Algunos de estos protocolos son:

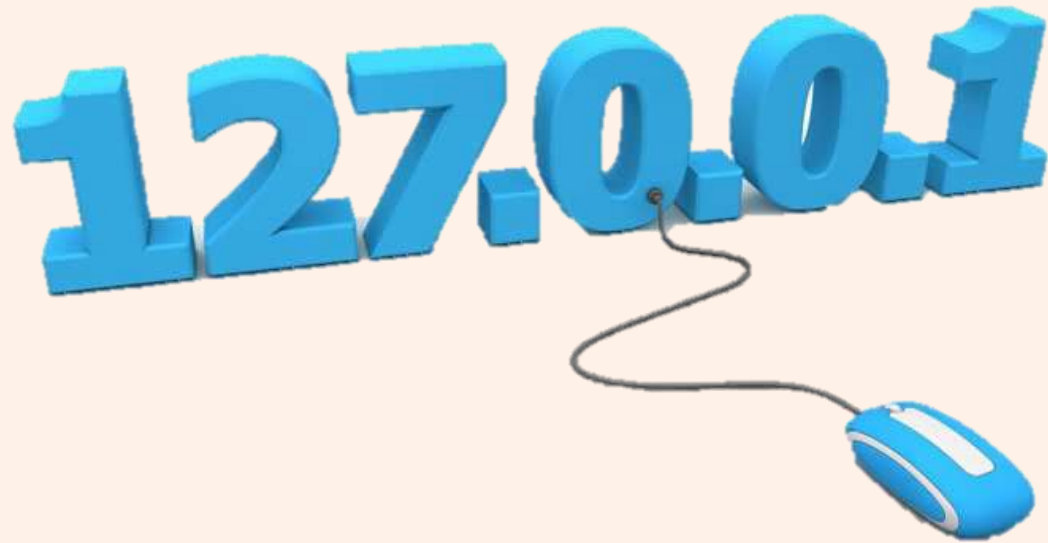
- **TCP/IP**
- **DNS**
- **HTTP**





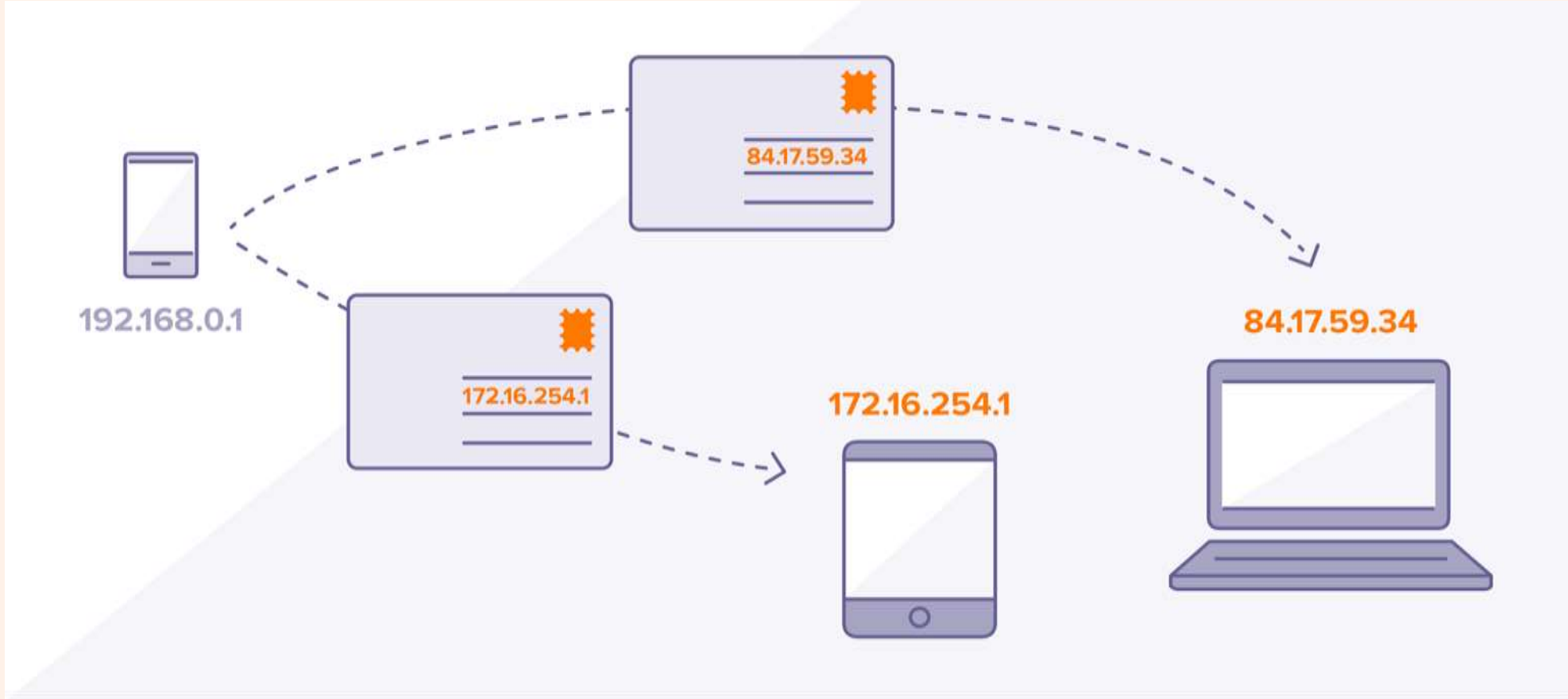
## ¿Qué es el IP?

Una IP (Internet Protocol) es una dirección única que identifica a un dispositivo en una red.



Esta se encuentra formada por cuatro números separados por un punto, comprendidos cada uno de ellos entre 0 y 255 (ejemplo: 192.168.10.3)



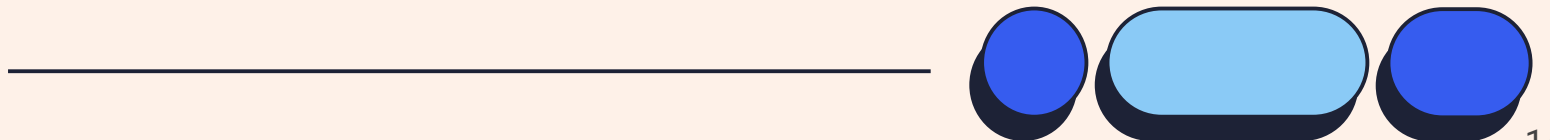


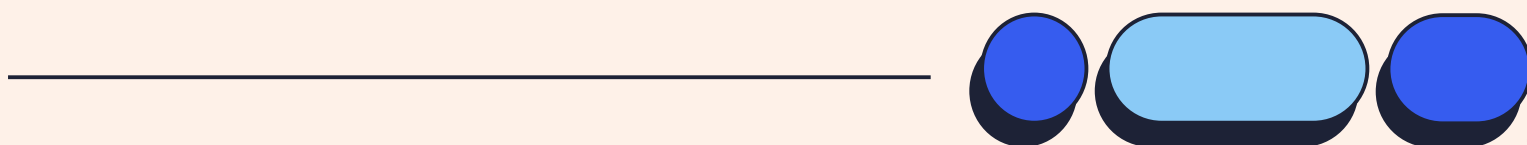
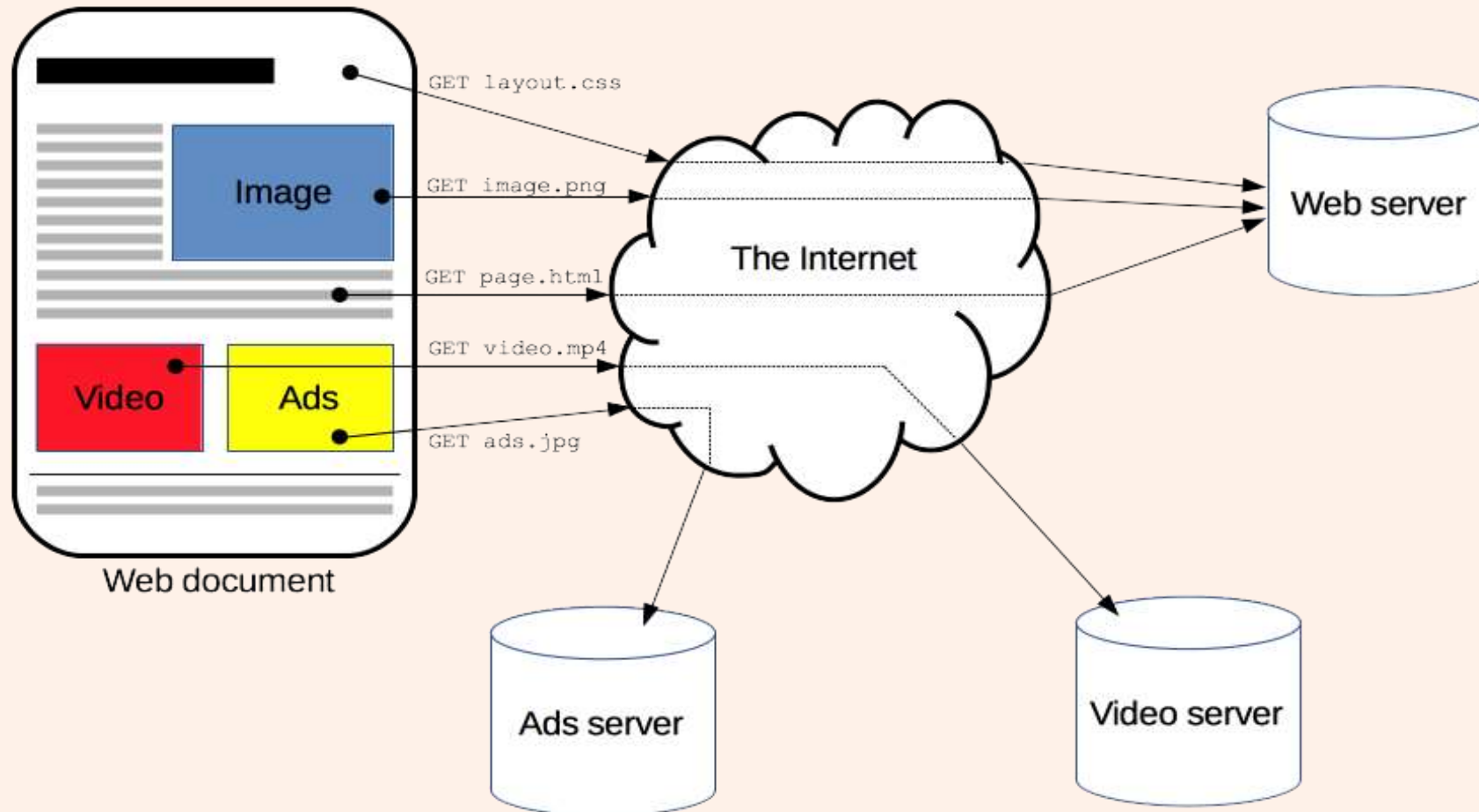
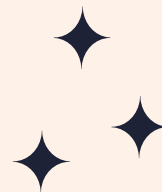


# Protocolo HTTP

Por su lado, el protocolo HTTP nos permite realizar una petición de datos y recursos, como pueden ser documentos HTML

Es la base de cualquier intercambio de datos en la Web, y un protocolo de estructura cliente-servidor.







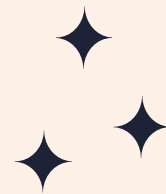
02

# Front end y back end

La separación entre las dos partes  
funcionales de un sitio web



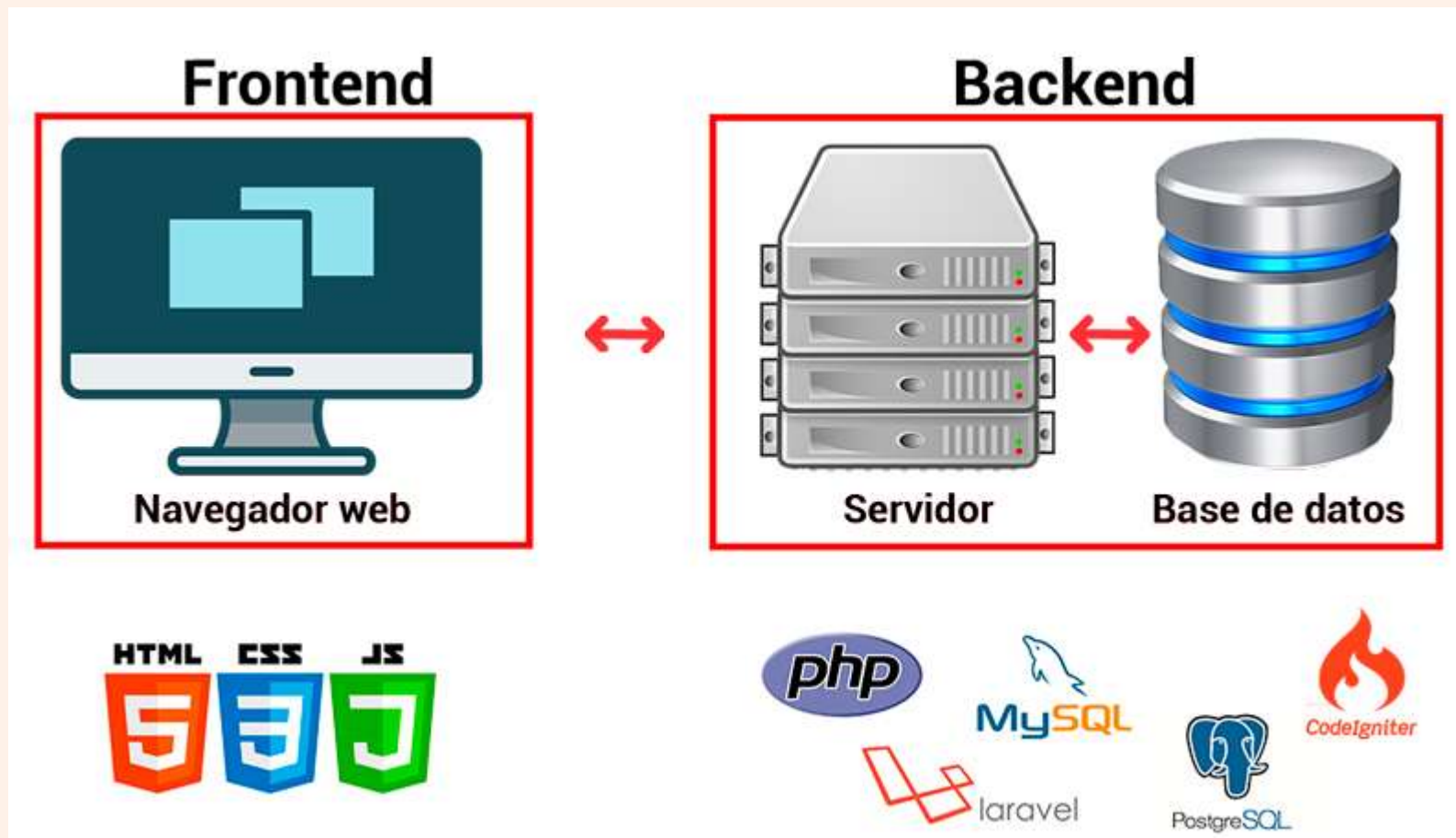
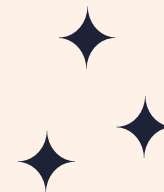




# Partes de un sitio

Por esta arquitectura, normalmente se separa todo sitio web en dos partes: una parte que es la que ve todo cliente (**front end**) y lo que es procesado en el servidor (**back end**)

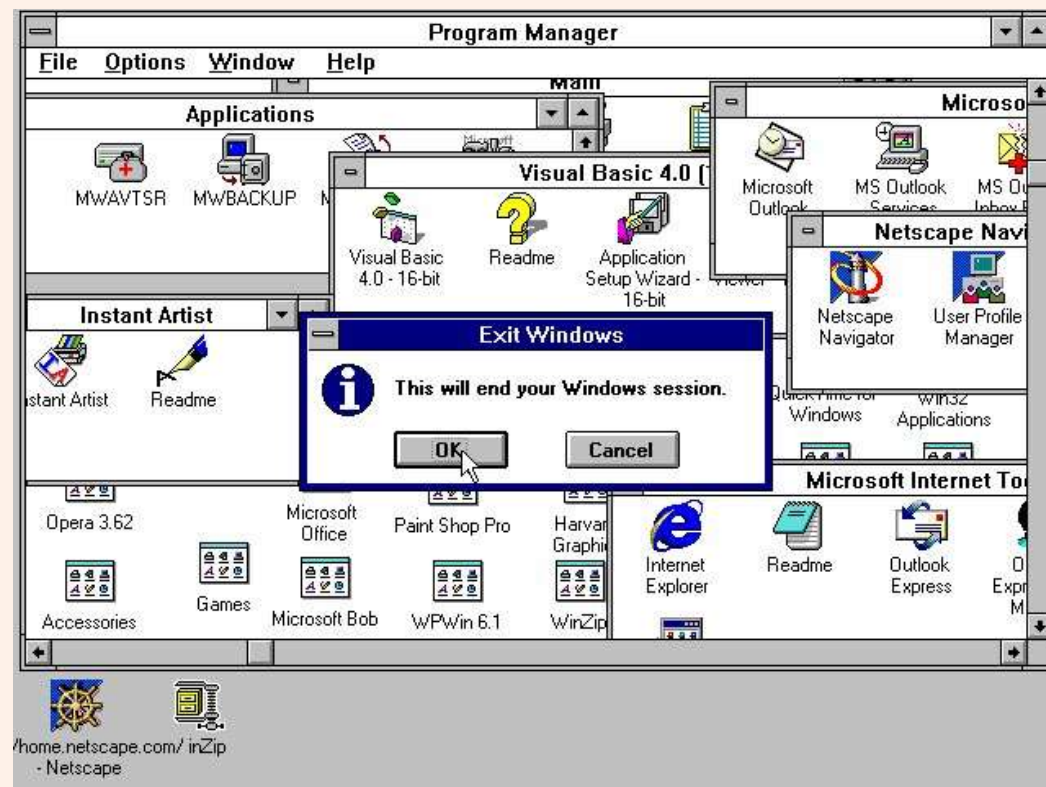




# Front end

El front end es la interfaz del sitio

Un software que muestra de forma visual todas las acciones posibles en una plataforma, así como la información disponible, para que los usuarios puedan interactuar con mayor facilidad



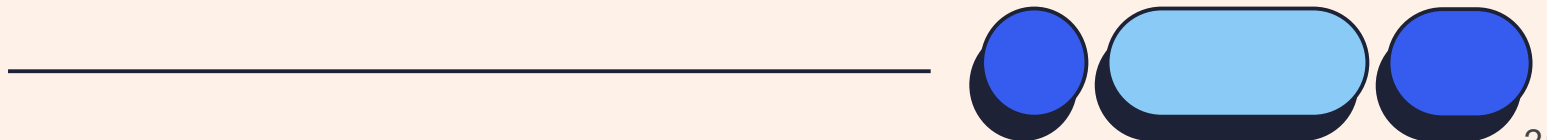


# Back end

El back end es el lado del servidor

Básicamente, cuando hablamos de "detrás de escena", es decir, el servidor y la base de datos que ayudan a entregar información del usuario desde una interfaz, hablamos del back-end.

Exige el dominio de otros términos de programación, lenguajes que requieren una lógica, ya que esta área es también la encargada de optimizar recursos, de la seguridad de un sitio y otros factores





## Back-end y bases

También está el manejo de la información de los usuarios, acumuladas en **bases de datos**

Cualquiera que prefiera especializarse como desarrollador de back-end actuará con lógica, funcionalidad del sitio, reglas, seguridad e integridad de la base de datos





# Sitios estáticos/dinámicos

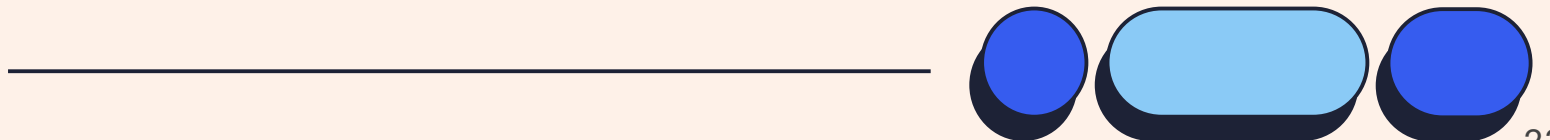
A partir de la interacción con el usuario y las respuestas del servidor, se pueden separar los sitios entre webs estáticas y webs dinámicas





## Web estática

- Tiene un número fijo de páginas y archivos HTML
- Siempre que se visita tiene el mismo contenido
- Es ligera y carga rápido

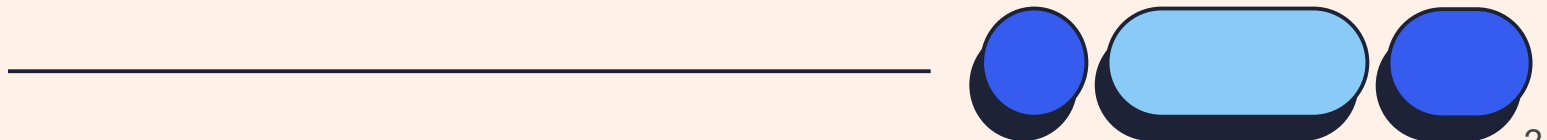




# Web dinámica

La palabra dinámica se refiere a elementos que cambian continuamente, son interactivos y funcionales.

En comparación con las páginas web estáticas, que son mayoritariamente informativas, una página web dinámica incluye aspectos que se caracterizan por la interactividad y la funcionalidad





## Estáticas



## Dinámicas

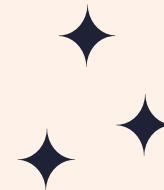




03

# Lenguajes de programación web



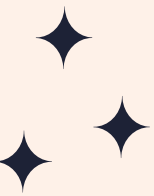


# Nuestros lenguajes

Como hemos dicho, los principales lenguajes que manejaremos son HTML, CSS y JavaScript

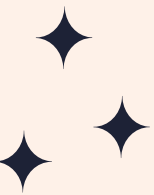
- HTML (lenguaje de marcado de hipertexto, se utilizan etiquetas que estructuran y organizan el contenido de la web).





- CSS (se encarga del formato y diseño visual de las páginas web escritas en html) para darle estructura y estilo al sitio.
- Javascript (un lenguaje de programación, rápido y seguro para programar centros de datos, consolas, teléfonos móviles o Internet) para complementar los anteriores y darle dinamismo a los sitios web.





# ¿Qué es HTML?

Es un lenguaje de **marcado**, no de programación

Esto quiere decir  
que trabaja  
diferenciado  
elementos a partir  
de ciertas marcas o  
etiquetas que va  
colocando

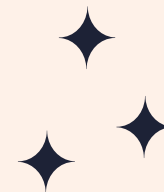
```
1 <html>
2   <head>
3     <meta charset="utf-8"/>
4     <title>Esto es un ejemplo</title>
5   </head>
6   <body>
7     <h1>Este es el título 1</h1>
8     <h2>Este es el título 2</h2>
9     <h3>Este es el título 3</h3>
10  </body>
11 </html>
```

**Este es el título 1**

**Este es el título 2**

**Este es el título 3**



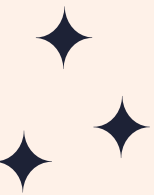


# Crear un archivo HTML

Para crear un archivo HTML no hace falta más que abrir el bloc de notas que tiene todo Windows, teclear lo siguiente y guardarlo como archivo HTML:

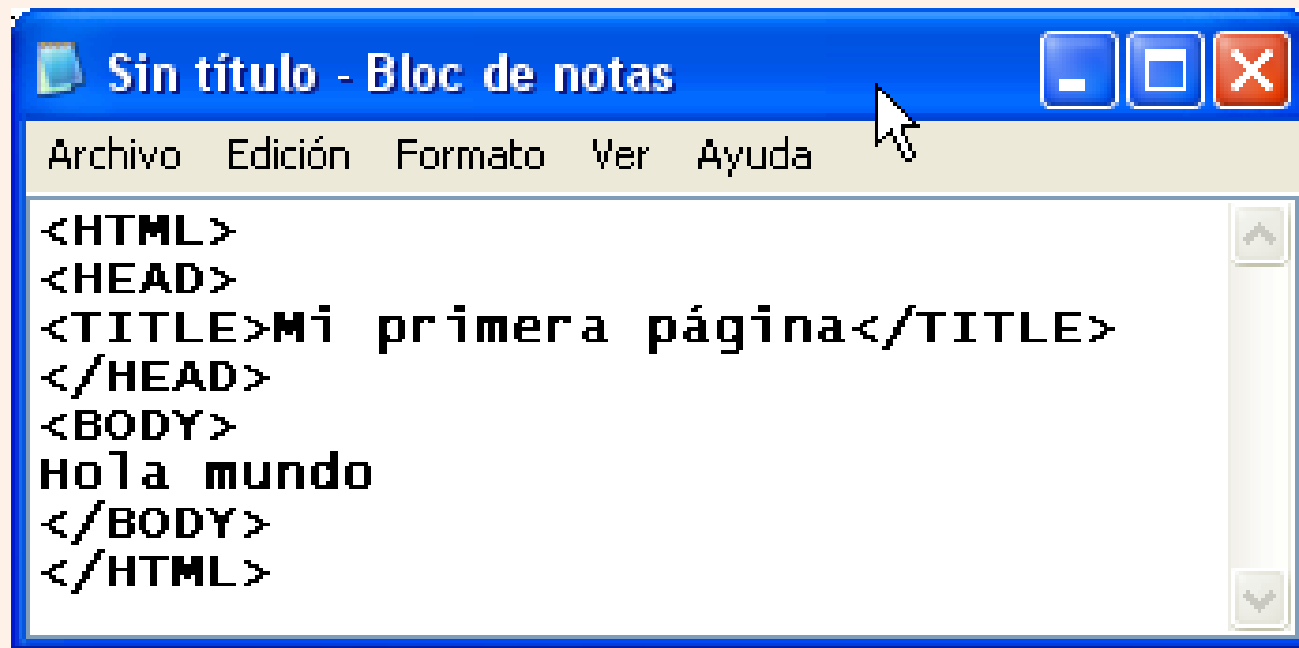
```
<doctype! html>
<html>
  <head>
  </head>
  <body>
    <p>HOLA MUNDO</p>
  </body>
</html>
```





# Actividad: primera pag estática

Prepararemos una página con el siguiente código y experimentemos un poco con ella



```
<HTML>
<HEAD>
<TITLE>Mi primera página</TITLE>
</HEAD>
<BODY>
Hola mundo
</BODY>
</HTML>
```





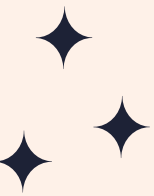
04

# El lenguaje HTML

Manejando el lenguaje básico de la  
programación web



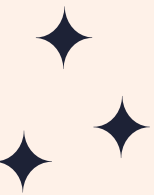




# Elementos HTML

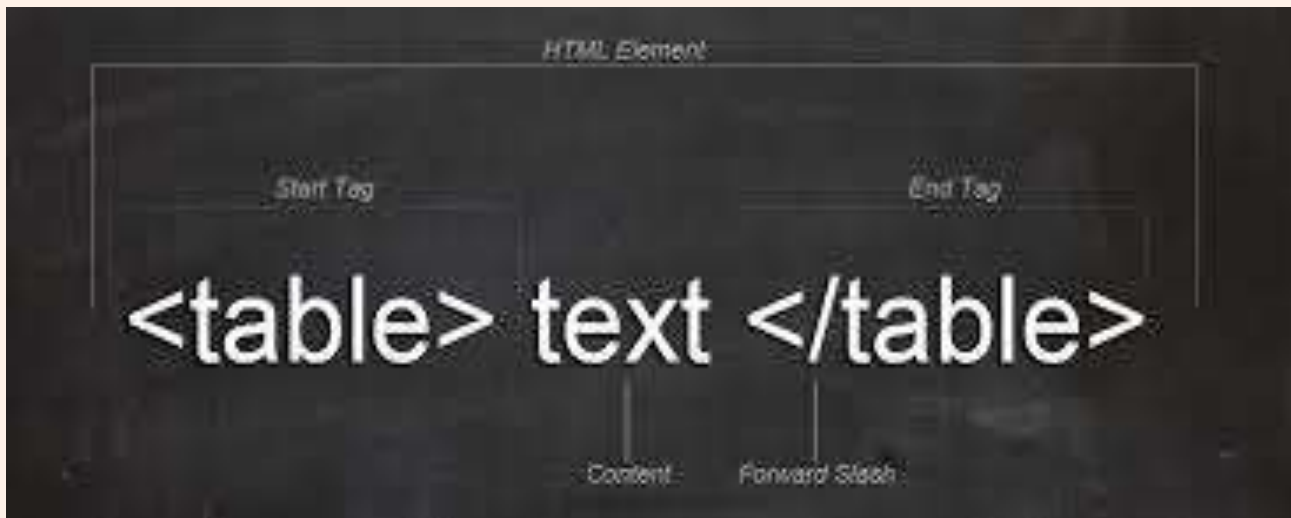
ELEMENTOS : Componente Básico de un archivo html.  
Los elementos los podemos pensar como ladrillos de una casa.





# Sintaxis en HTML

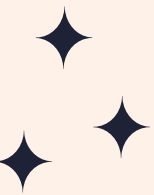
Como todo lenguaje, existen ciertas reglas de **sintaxis** en HTML



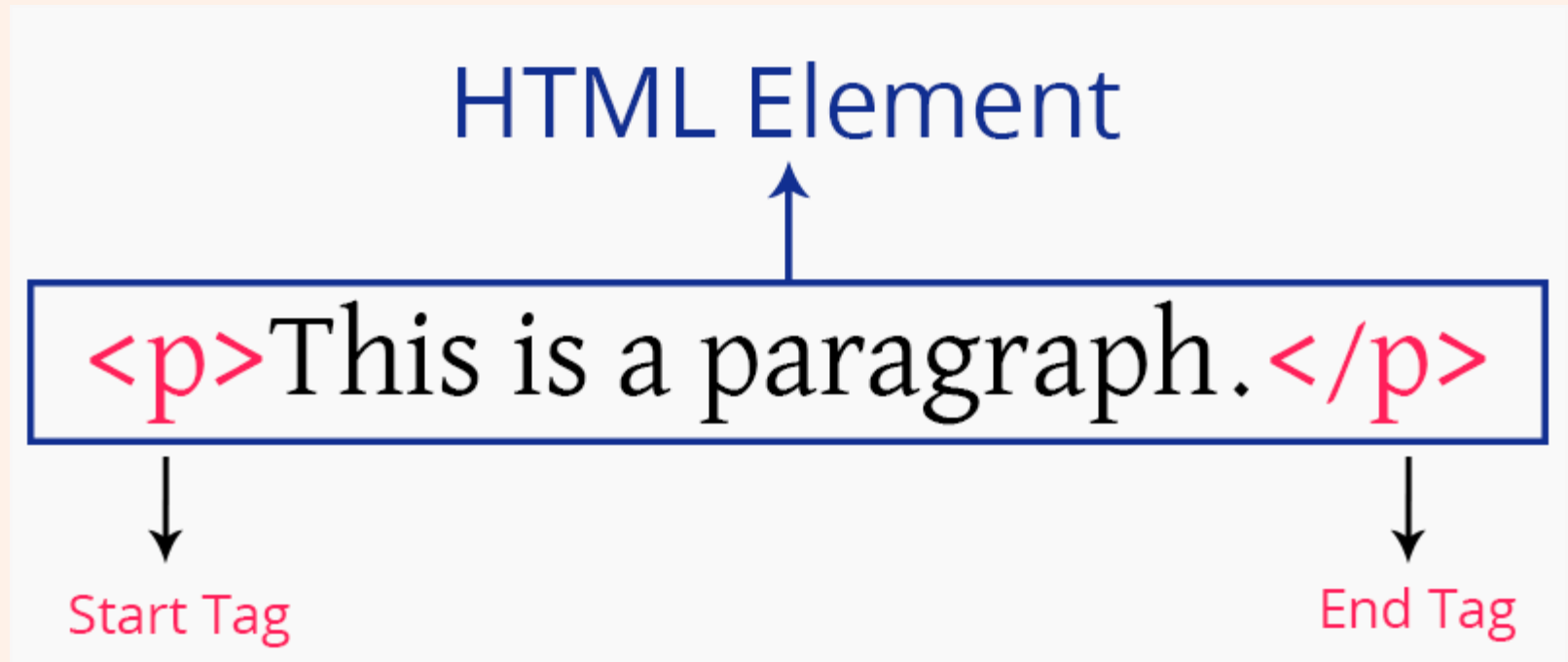
Todo elemento HTML es un contenido textual encerrado entre una etiqueta de apertura y otra de cierre

Lo que hay adentro se llamara contenido





Toda etiqueta tiene el nombre del tipo de elemento y toda etiqueta de cierre repite ese nombre con el símbolo «/»





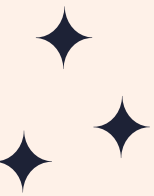
**<self closing tag />**



**This is an example of a self closing tag**

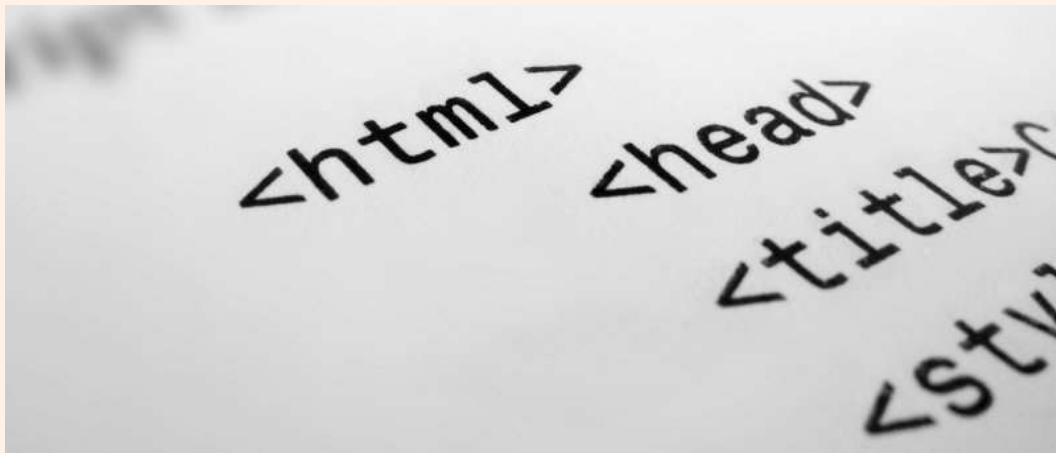
También existen  
ciertas etiquetas  
puntuales que no  
llevan etiqueta  
de cierre





# Indentacion

Algo importante a analizar cuando vemos archivos HTML es la indentación. Funcionalmente no habrá cambios, pero siempre se DEBE AGREGAR un nivel de indentación



¿Por qué es necesario e importante la indentación?  
FACILIDAD DE LECTURA DEL ARCHIVO  
El código se empieza a ver mas confuso cuando empieza a crecer





¿Cuanto usar de indentación? Se recomienda 2 espacios aunque es algo que lo puede decidir el desarrollador.

En HTML vamos a ir indentando los elementos que tenemos dentro de otros elementos , lo que arma una ESTRUCTURA BASTANTE PROFUNDA

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>
5              Sthefany Floriano
6          </title>
7      </head>
8      <body>
9          <p>
10              Indentando mi codigo
11          </p>
12      </body>
13 </html>
```

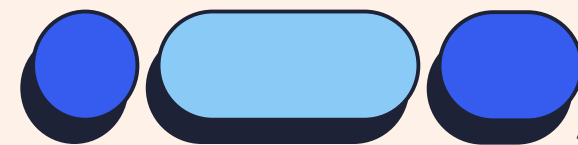
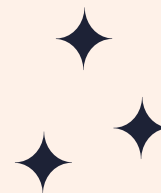


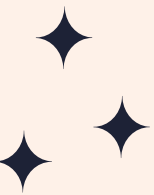


# Estructura HTML

Dentro del **elemento raíz** vamos a tener 2 elementos importantes : **HEAD**, **BODY**

## Estructura básica

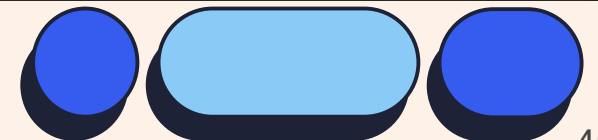




# Etiqueta HTML

Nuestro elemento primordial va a ser una declaración. Esta etiqueta es `<html>` y se cierra solo al final de todo el documento

```
HTML_h1-h6_Tag.html
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>HTML <h1>-<h6> tag</title>
5   </head>
6   <body>
7     <h1>This is the H1 heading</h1>
8     <h2>This is the H2 heading</h2>
9     <h3>This is the H3 heading</h3>
10    <h4>This is the H4 heading</h4>
11    <h5>This is the H5 heading</h5>
12    <h6>This is the H6 heading</h6>
13  </body>
14 </html>
```





# Head y metadatos

En la etiqueta head vamos a ver cosas importantes:  
archivos externos, tambien podemos personalizar el título de la página

Esta información se conoce como **metadatos** y es información que proporcionamos al navegador, no al usuario

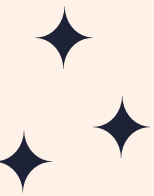




# Doctype

La primera línea de todo el documento va a ser la version que vamos a usar de HTML y se coloca así:



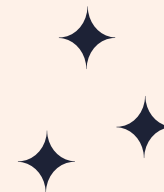


# Title

La etiqueta `<title>` proporciona al navegador el nombre de nuestra página

```
<!DOCTYPE html>
<html lang="es-ES">
<head>
  <meta charset="UTF-8">
  <title>Guía de estilo de HTML</title>
</head>
```



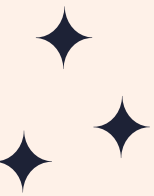


# Etiqueta de idioma

Dentro del tag `<html>` agregar el atributo `lang` y para el español darle el valor `"es"`

```
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4
5 </head>
6 <body>
7
8 </body>
9
10
11 </html>
```





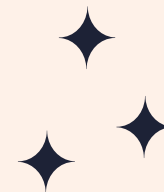
# ¿Qué es body?

La etiqueta que indica todo el «cuerpo» visible por los usuarios de nuestra página

```
<!DOCTYPE html>
<html>
  <head>
    <title>Estructura HTML</title>
  </head>
  <body>
    <h1>Mi página web</h1>
    <div>Hola Mundo</div>
  </body>
</html>
```

```
<html>
  <head></head>
  <body></body>
</html>
```





# Encabezados

Los encabezados o palabras tipo título dentro de nuestra página se agregan con las etiquetas header

Estas van del 1 al 6 en importancia y tendremos `<h1>` como la mas importante, `<h2>` la siguiente y así las demás

**Encabezado 1**

**Encabezado 2**

**Encabezado 3**

**Encabezado 4**

**Encabezado 5**

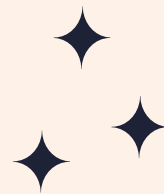
**Encabezado 6**





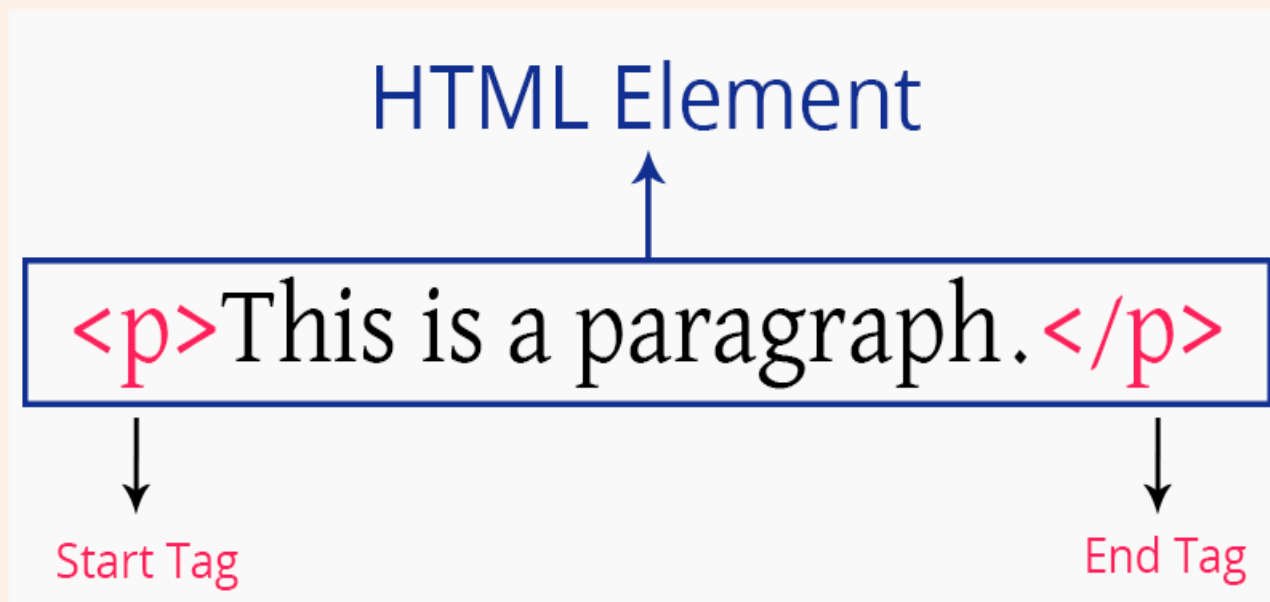
```
1  <!DOCTYPE html>
2  <html lang="es">
3      <head>
4      </head>
5      <body>
6          <h1>Encabezado 1</h1>
7      </body>
8  </html>
```



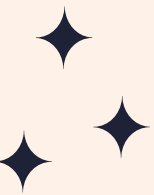


# Párrafos

Después de cada cierre de la etiqueta `<p>` habrá un salto de línea grande







# Salto de línea

Si queremos solo ir al renglón siguiente, usamos la etiqueta `<br>`. Es una etiqueta solo de apertura

```
<body>
  <p>
    Pablito clavó un clavito<br>
    en la calva de un calvito<br>
    en la calva de un calvito<br>
    Pablito clavó un clavito<br>
  </p>
</body>
```

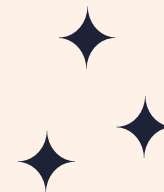
salto de línea

NAVEGADOR

Mi web

Pablito clavó un clavito  
en la calva de un calvito  
en la calva de un calvito  
Pablito clavó un clavito





# Comentarios

Videollamadas En programación, los comentarios son líneas que dejamos escritas y que el programa ignora pero que pueden quedar como mensajes para las personas que luego usen nuestro código

```
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4    </head>
5    <body>
6      <h1>App de gatos</h1>
7      <p>haz click aqui para ver mas gatos</p>
8      <!--esto es un comentario-->
9
10     </body>
11  </html>
```

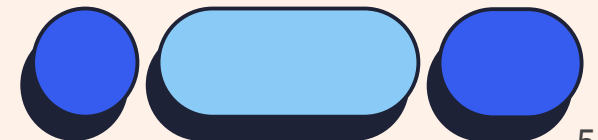




```
1  <!DOCTYPE html>
2  <html lang="es">
3      <head>
4      </head>
5      <body>
6          <h1>App de gatos</h1>
7          <p>haz click aqui para ver mas perros</p>
8          <!-- TODO: Agregar enlace a imagen de perros -->
9
10     </body>
11 </html>
```



Para comentar  
una línea: Ctrl  
+ /  
(usando Visual  
Studio)





# Letra negrita

Etiqueta `<b>` para las palabras en negrita

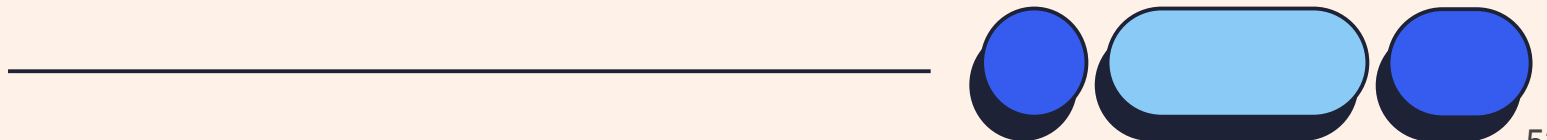




# Letra cursiva



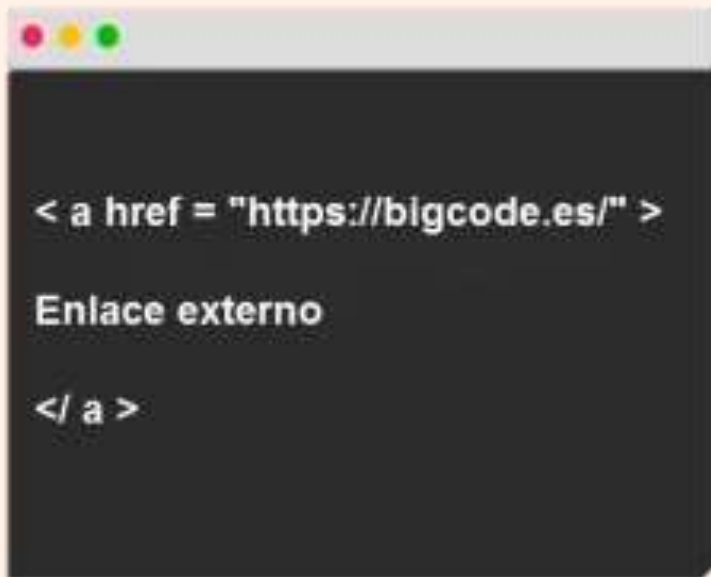
Etiqueta `<i>` para las palabras en cursiva





# Atributos

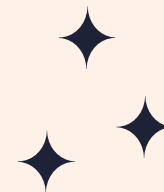
Las etiquetas tienen atributos, que son información adicional sobre la etiqueta y su contenido, como el idioma que pusimos en la etiqueta `<html>`



```
< a href = "https://bigcode.es/" >  
Enlace externo  
</ a >
```

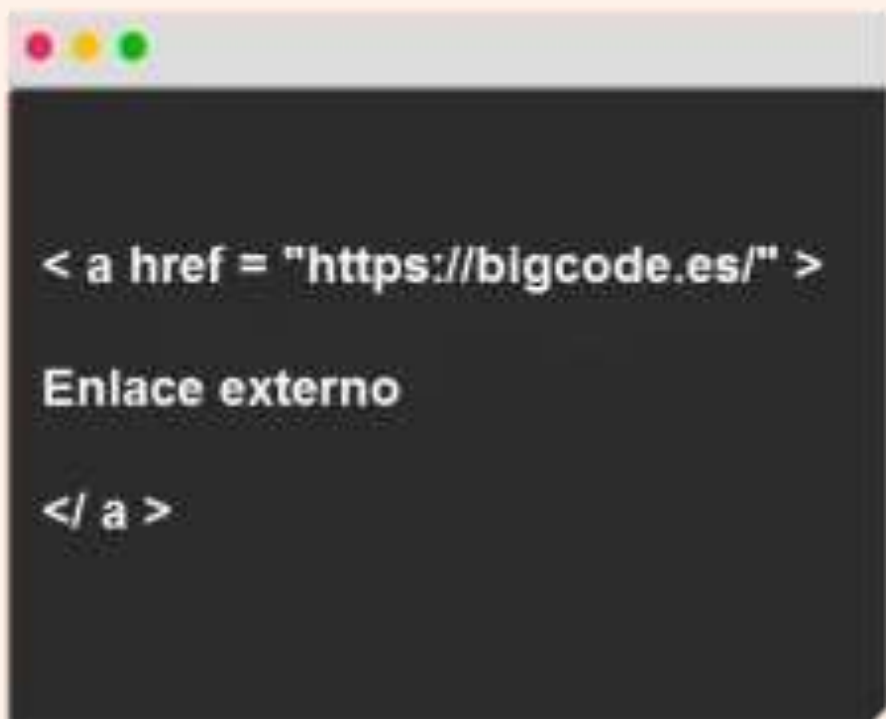
Se suelen escribir dentro de la etiqueta de apertura, llevar nombre y valor entre comillas





# Atributos y links

Un ejemplo es la etiqueta `<a>` utilizada para colocar hipervínculos o links



```
< a href = "https://bigcode.es/" >  
Enlace externo  
</ a >
```

En la etiqueta de apertura lleva el atributo **href** y su valor es la dirección a donde nos va a dirigir el link

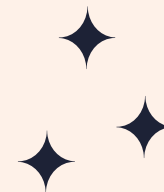




`<a href="https://www.trabajo.gba.gov.ar/ipf1/">`

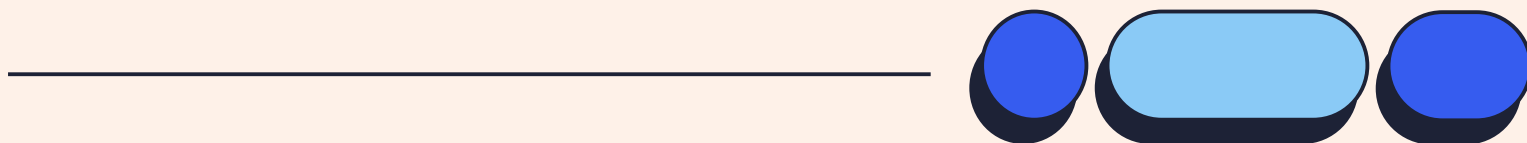


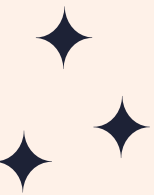




# Atributo target

Si dentro de `<a>` colocamos el atributo `target` y le damos el valor «`blank`», abrirá el link en pestaña nueva



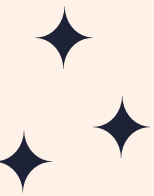


# Atributo rel

Atributo `rel="noopener noreferrer">` evita ataques llamados 'Tabnabbing'.

```
example.html > html > body > a
1  <!DOCTYPE html>
2  <html lang="es">
3    <head>
4    </head>
5    <body>
6      <h1>App de gatos</h1>
7      <p>haz click aqui para ver mas perros</p>
8      <!-- TODO: Agregar enlace a imagen de perros -->
9      
10     <a href="https://www.trabajo.gba.gov.ar/ipfl/" target="_blank" rel="noopener
11       norereferrer"> instituto provincial</a>
12   </body>
13 </html>
```





## Link a tus páginas

¿Y para mandarte a páginas que tengas en tu misma computadora?

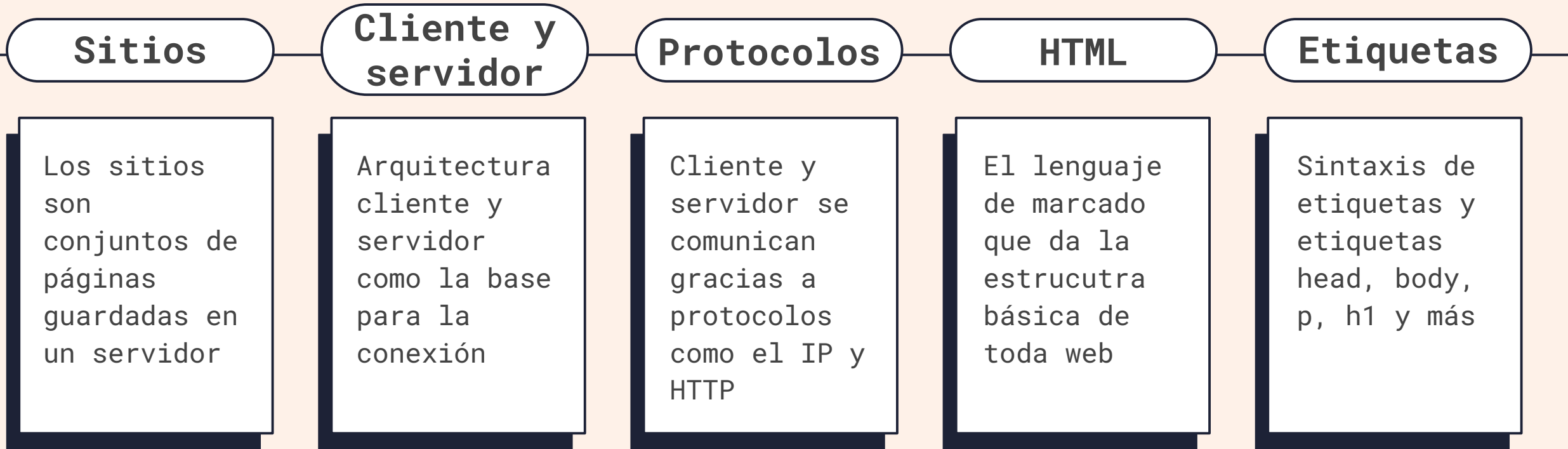
```
1 <a href="C:\Users\Usuario\Desktop\Carpeta\index.html"> Enlace al índice </a>
```

Usamos la ruta local del archivo(solo se verá en la misma computadora)





# Resumiendo





# Eso es todo

Se escuchan dudas, quejas y propuestas

F  
i  
n  
d  
e  
l  
a  
c  
l  
a  
s  
e

