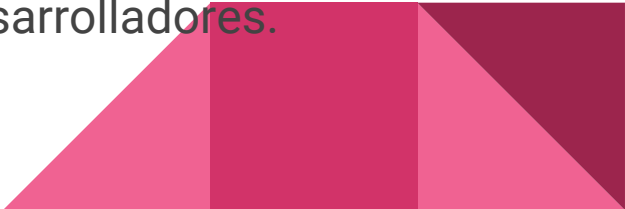


Diseño de páginas web

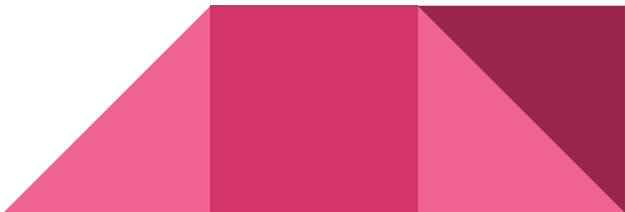
¿Quién soy?

- Emanuel Matulich
- Casi Ingeniero en Sistemas de Información.
- 23 años.
- Web developer en redSupernova.
- contacto:
 - matulichema@gmail.com
 - Asunto: curso diseño de páginas web 2017.

Temario

1. Tecnología y estándares web. Introducción al lenguaje HTML. Marcado estructurado semánticamente con HTML. Profundización sobre el lenguaje HTML, etiquetas, links, atributos, etc.
 2. Novedades de HTML5. Maquetación y diseño para web.
 3. Elementos estructurales y semántica. Rutas absolutas y relativas. Trabajo con estilos.
 4. Separación de contenido (HTML) y presentación (CSS). Trabajo con formularios. Multimedia.
 5. Responsive design. Utilizando herramientas para desarrolladores.
- 

Unidades

1. ¿Donde empiezo?
 2. Cómo funciona la web.
 3. Creamos una página simple - introducción a la estructura básica de HTML.
 4. Marcando texto (Marking up text).
 5. Agregando links.
 6. Agregando Imagenes.
 7. Tablas.
 8. Formularios.
 9. What's up HTML5?
- 

Unidades

11. Introducción a CSS.
12. Formatear texto.
13. Colores y fondos.
14. Pensando en la caja (Box model Margin, Padding, Border).
15. Posicionamiento.
16. Transiciones, transformaciones y animaciones.
17. Javascript una introducción.
18. Responsive Web Design

*Idea práctico integrador



Recursos del curso

Repositorio en GitHub:

<https://github.com/EmaMatulich/curso-web-victor-mercante>



Unidad N° 1: ¿Donde empiezo?

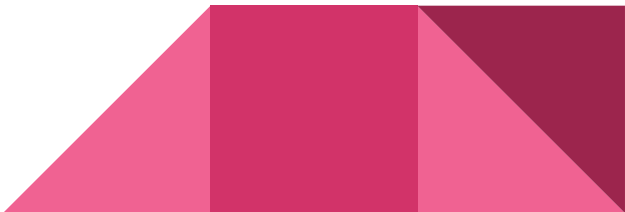
- Que hace un web designer?
- Que lenguajes necesito aprender?
- Que software y equipo necesito?



Unidad N° 1: ¿Donde empiezo?

Que hace un web designer?

Un web designer puede abarcar una o varias de las siguientes disciplinas:

- Diseño gráfico o visual.
 - Diseño de interfaz de usuario o experiencia de usuario.
 - Escribir HTML y CSS.
 - Scripting.
 - Content strategy.
 - Multimedia.
- 

Unidad N° 1: ¿Donde empiezo?

Diseño gráfico.	Consiste en crear la imagen del sitio, es decir, logos, imágenes, colores, fuentes, layout, para asegurar que el sitio es consistente con el mensaje que quiere representar. (uso de photoshop, corel y similares).
Diseño de interfaz y experiencia de usuario.	Identificar el objetivo del sitio, como va a ser usado, como se van a mover los usuarios en el, entre otros. Consiste en asegurar que la experiencia completa en el sitio sea agradable, favorable, cómoda para el usuario.
HTML y CSS, Scripting	Crear los documentos, hojas de estilos, scripts y demás archivos que hacen al sitio.
Content strategy.	Crear el contenido de información que se visualiza en el sitio y su mantención, asegurando que está de acuerdo con la identidad de marca y objetivos de marketing de entidad que necesite el sitio (empresa, ONG, etc).
Multimedia.	Crear videos, sonidos, animaciones, juegos interactivos, etc.
Multi rol	Desempeñar varias de las disciplinas anteriores.

Unidad N° 1: ¿Donde empiezo?

Que lenguajes necesito aprender?

- Hypertext Markup Language (HTML)
- Cascading Style Sheets (CSS)
- JavaScript and DOM scripting.

La W3c es una organización que observa el desarrollo de estas tecnologías: <http://www.w3c.es/>

Además ofrecen tutoriales: <https://www.w3schools.com/>



Unidad N° 1: ¿Donde empiezo?

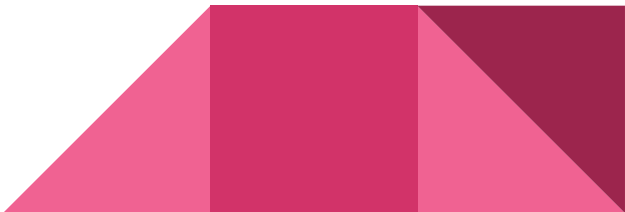
Hypertext Markup Language (HTML)

- Es el lenguaje usado para crear archivos de páginas web. La última versión es HTML5.
- NO ES UN LENGUAJE DE PROGRAMACIÓN, es un **LENGUAJE DE MARCADO**.
- Permite identificar y describir los componentes variados del contenido de una página como títulos, párrafos, listas, secciones, artículos, etc.
- HTML indica la ESTRUCTURA del contenido de una página.



Unidad N° 1: ¿Donde empiezo?

Cascading Style Sheets (CSS)

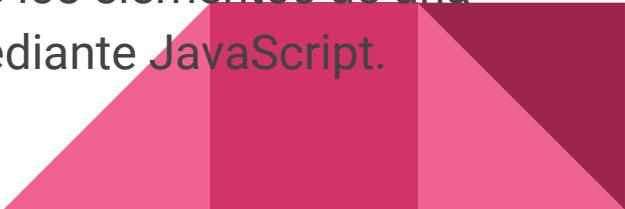
- Describe cómo debería lucir o mostrarse el contenido. Es decir define la **PRESENTACIÓN VISUAL** de una página.
 - Permite especificar tipos de fuentes, colores, imágenes de fondo, tamaño de fuente, efectos, etc.
 - Permit especificar cómo se presentará el sitio en una PC de escritorio, tablet, celular, Smart TV, o cualquier otro dispositivo con acceso a internet y un browser.
 - La versión actual es CSS3.
- 

Unidad N° 1: ¿Donde empiezo?

JavaScript/DOM scripting

- Es un lenguaje de scripting (automatiza la ejecución de tareas) que permite agregar interacción y comportamiento a los sitios web:
 - Validar entradas de formularios.
 - Agregar y remover estilos.
 - Guardar información en los browser.
 - Crear widgets como menús expandibles, juegos, etc.

DOM és el Document Object Model, se refiere a la lista de los elementos de una página web que pueden ser accedidos y manipulados mediante JavaScript.




Unidad N° 1: ¿Donde empiezo?

Server-side programming (No forma parte del curso)


Algunos sitios son puramente estáticos o con alguna funcionalidad de JavaScript pero toda esta funcionalidad viene cuando se solicita la página.

Para añadir funcionalidad más compleja como carrito de compras, inicio de sesión, acceso a base de datos, se necesitan aplicaciones corriendo en un servidor, algunos de los lenguajes usados para escribir web applications son PHP, Python, Java, NodeJs, Ruby ASP.NET, etc.



Unidad N° 1: ¿Donde empiezo?

Que software y equipo necesito?

- Una computadora con cualquier sistema operativo. (Opción virtualización.)
 - Un navegador a elección propia y varios navegadores alternativos.
(Problemas de soporte de estándares / considerar navegadores por cada sistema operativo.)
 - Otros dispositivos donde probar tales como celulares, tablets, etc.
(Virtualización o opciones de desarrollador de navegadores.)
 - Software de edición de imágenes. (Muchas veces no es necesario.)
 - Editor de texto. (Atom, Sublime Text, Visual studio code, etc.)
- 

Unidad N°2 Cómo funciona la web

- Una explicación de la web.
- El rol del servidor.
- El rol del navegador.
- URLs y sus componentes.
- Anatomía de una página web.



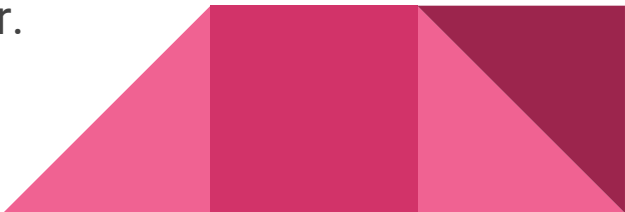
Unidad N°2 Cómo funciona la web

Una explicación de la web. (Internet vs Web)

- Internet es una red de computadoras conectadas con el propósito de compartir información. Existen varias formas de pasar información entre las computadoras, tales como emails, transferencia de archivos (FTP), entre otros, Estos métodos estandarizados para transferir datos o documentos sobre una red son llamados protocolos.
- La Web (World Wide Web o WWW), es una de las formas en que la información puede ser compartida sobre internet. Permite que los documentos sean linkeados desde otro a través de links.
- La Web usa el protocolo HTTP (Hyper Text Transfer Protocol).

Unidad N°2 Cómo funciona la web

El rol del servidor

- Es un Software que permite a una computadora comunicarse con otras. Su rol es esperar solicitudes (Request) de información, una vez recibida, recupera y envía (retrieve and send) la información solicitada.
 - El servidor permite recuperar los archivos HTML, imágenes, archivos CSS que forman una página web.
 - Un servidor se ejecuta en un computadora conectada a internet con una dirección IP y un puerto en el que escucha el servidor.
- 

Unidad N°2 Cómo funciona la web

El rol del servidor

- Es asignar un “nombre de dominio” a esta dirección, tal como google.com, en lugar de usar la IP nos podemos referir al servidor a través de el nombre de dominio.
- Es posible configurar el servidor para que una dirección IP mapee a más de nombre de dominio usando subdominios, ejemplo: dev.google.com, sub.google.com, google.com, etc.



Unidad N°2 Cómo funciona la web


El rol del servidor

- El puerto por defecto para los servidores web es el 80 para el protocolo HTTP, que es el encargado de responder con los archivos HTML, imágenes, archivos CSS, etc.



Unidad N°2 Cómo funciona la web

El rol del navegador

- Solicita a servidores peticiones.
 - Es usado para acceder a los recursos en la web. Los servidores retornan documentos a los navegadores para que los visualice.
 - Puede recibir por parte de los servidores imágenes, videos, datos, archivos HTML, css, entre otros que constituyen una página.
 - Es el encargado de visualizar una página web a partir de las definiciones que brindan los archivos que la componen.
- 

Unidad N°2 Cómo funciona la web

URLs y sus componentes.

- Cada página o recurso en la web tiene su propia dirección URL.
- URL significa Uniform Resource Locator, traducido a algo como Localizador Uniforme de Recursos.
- La URL identifica de manera unívoca a un recurso en la web.



Unidad N°2 Cómo funciona la web

Una URL tiene 3 componentes:

- Protocolo.
- Nombre del sitio.
- Ruta absoluta o dirección del archivo del documento o recurso al que se quiere acceder.

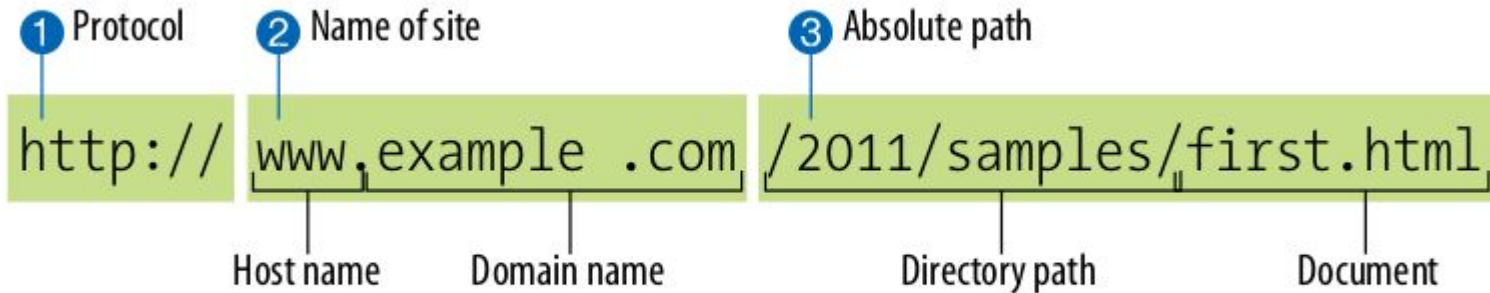


Figure 2-1. The parts of a URL.

Unidad N°2 Cómo funciona la web

- El protocolo `http://` indica que se va a usar HTTP para la transferencia de los archivos.
- La ruta absoluta por ejemplo `/2012/samples/first.html` le indica al servidor que busque en sus archivos en la “carpeta” 2012 una carpeta llamada samples y dentro de esta el archivo `index.html`
- La ruta inicia con el símbolo `/` indicando el comienzo del directorio o estructura de archivos del servidor. Se puede ver como la carpeta que contiene a todas las carpetas y archivos.



Unidad N°2 Cómo funciona la web

Archivos por defecto

Cuando una URL no contiene una ruta dirigida a un archivo en particular, al realizar la solicitud el servidor va a devolver el llamado archivo por defecto.

Este archivo es usualmente nombrado index.html pero este se puede cambiar.

Algunos ejemplos:

<http://www.google.com.ar/index.html>

<http://www.tn.com.ar/deportes/futbol/index.html>



Unidad N°2 Cómo funciona la web

Anatomía de una página web

Una página web es una unión de varios archivos:

- Documento HTML.
- Archivos CSS.
- Imágenes, videos, entre otros.



Unidad N°2 Cómo funciona la web

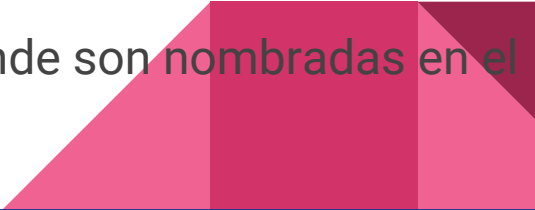
Anatomía de una página web

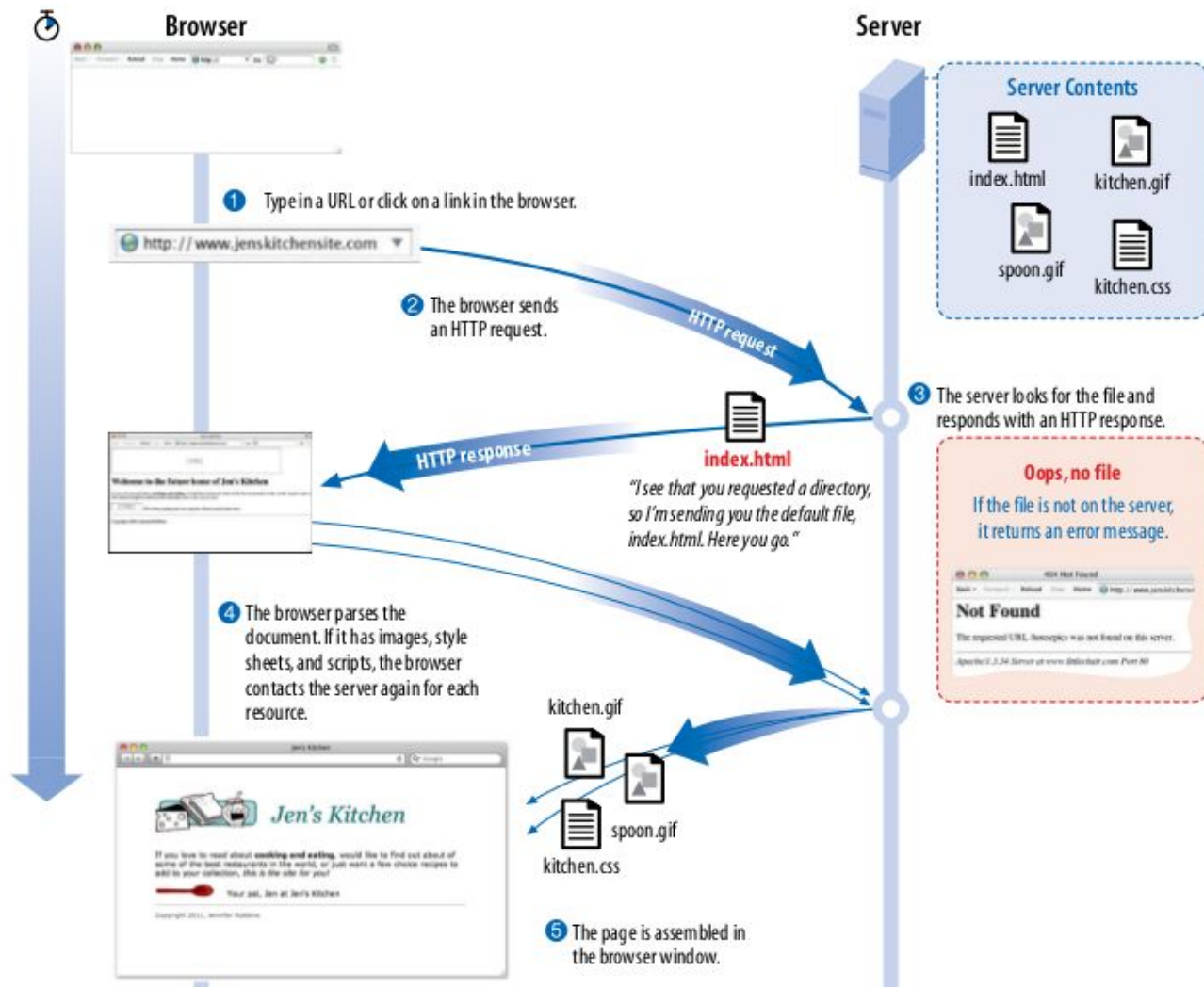
- El documento HTML a través de etiquetas de marcado le indica al navegador cómo estructurar la página.
- Los archivos CSS indican el estilo o como se ve la página.
- Las imágenes, videos, audios y otros recursos son agregados para la página.



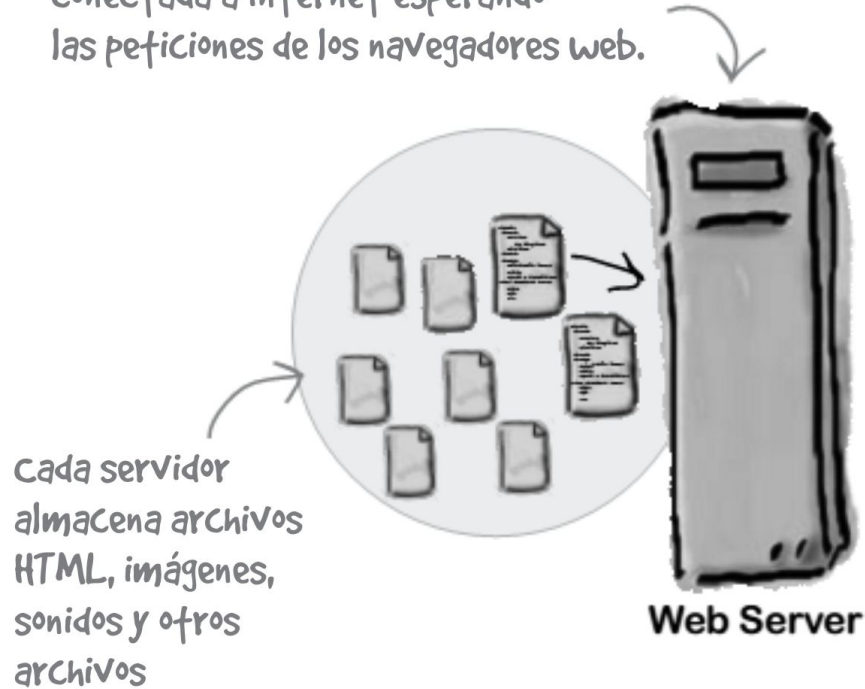
Unidad N°2 Cómo funciona la web

En resumen:

- Se solicita una página web a través de la URL.
 - EL navegador envía una petición HTTP al servidor.
 - El servidor busca por el archivo asociado a la URL. Si lo encuentra lo retorna y sino da un mensaje de error “404 archivo no encontrado”.
 - El navegador analiza el documento HTML, si la página contiene imágenes u otros links a otros archivos, el navegador vuelve a contactarse con el servidor para solicitar dichos archivos.
 - El navegador pone las imagenes (u otro recurso) donde son nombradas en el documento.
 - Muestra la página.
- 

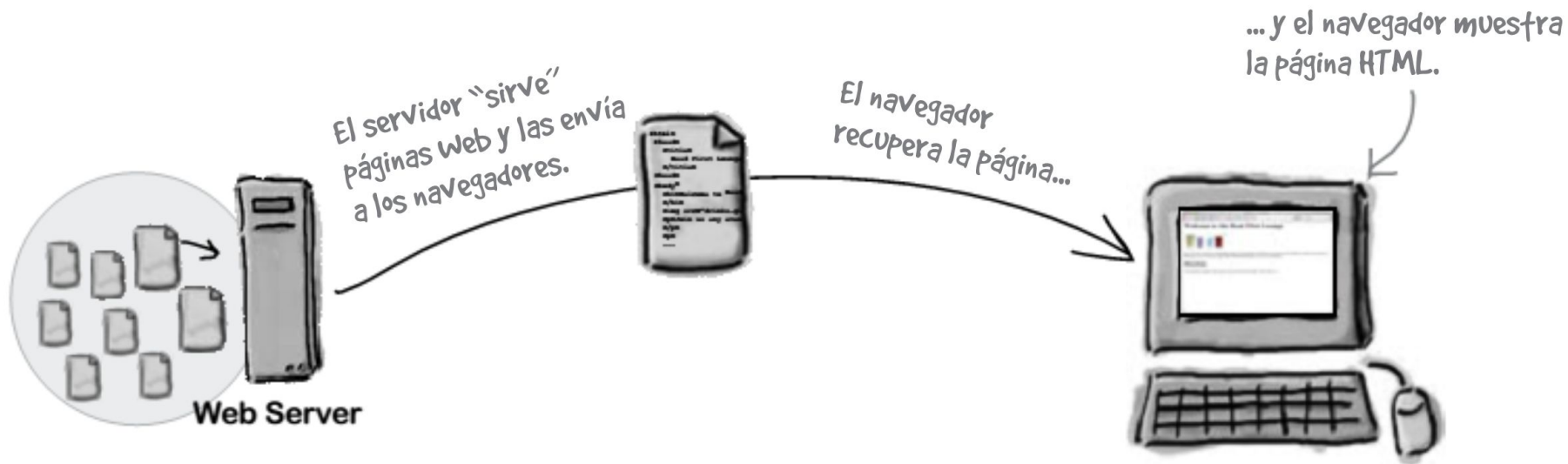


El servidor es una computadora
conectada a internet esperando
las peticiones de los navegadores web.



Los navegadores hacen
peticiones de páginas HTML
u otros recursos por ej: imágenes





Unidad N°3 Creamos una página simple

Configuración del entorno de desarrollo:

- Instalar editor de texto Atom.
- Instalar los siguientes paquetes:
 - atom-beautify
 - linter
 - pigments
 - linter-htmlhint
 - linter-csslint
 - linter-jshint
 - linter-jsonlint



Unidad N°3 Creamos una página simple

Como instalar paquetes en atom:

- En la barra de menú superior hacer click izquierdo en Edit/Edición.
- Seleccionar la opción preferences/preferencias.
- Seleccionar la opción Install/instalar.
- Hacer click en el botón packages/paquetes.
- Introducir el nombre del paquete y enter.



Unidad N°3 Creamos una página simple

Ejercicio N° 1: Crear una página web simple

Objetivos:

- Entender cómo funciona el marcado de HTML.
- Ver como los navegadores interpretan los documentos HTML.
- Aprender la estructura básica de todo documento HTML.

Material:

Carpeta clase 1 - Ejercicio 1



Unidad N°3 Creamos una página simple

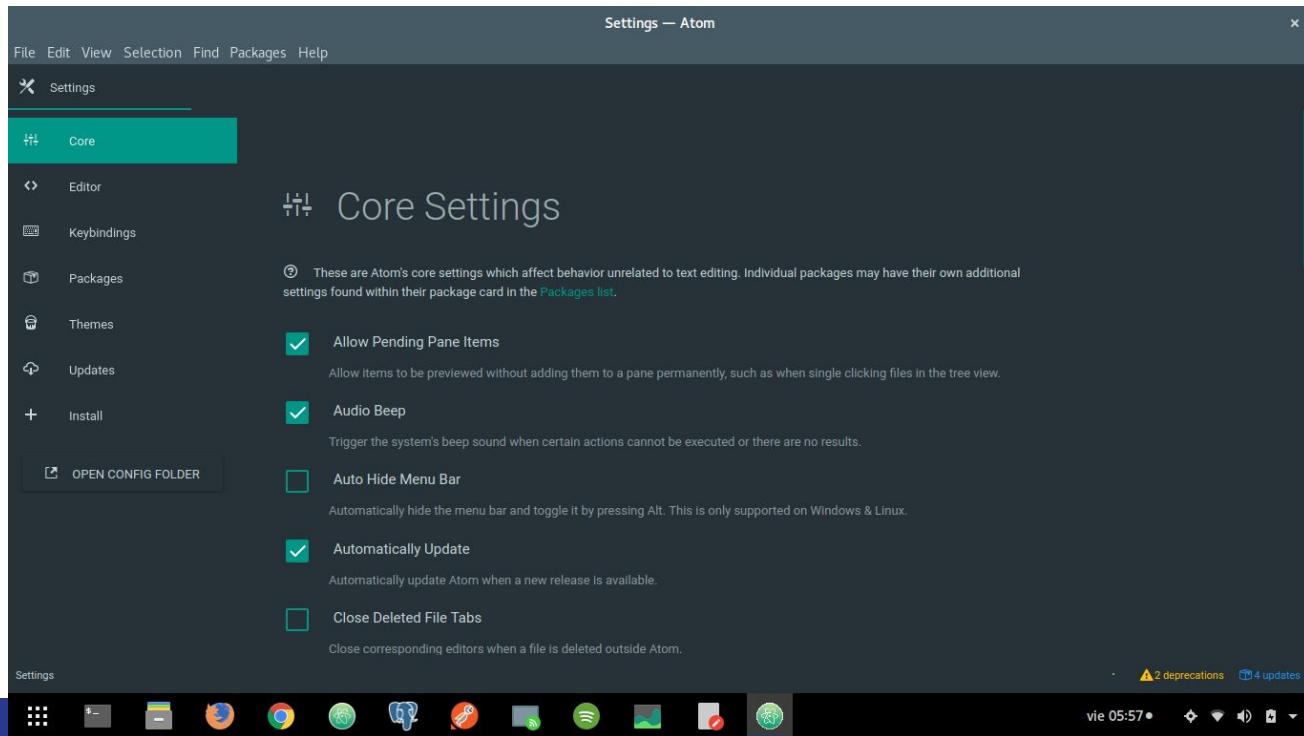
Ejercicio 1, pasos:

1. Definir el contenido.
2. Dar estructura al documento.
3. Identificar elementos de texto.
4. Agregar imagen
5. Cambiar un poco el look de la página.



Unidad N°3 Creamos una página simple

Manos a la obra, abrimos el editor de texto atom



Unidad N°3 Creamos una página simple

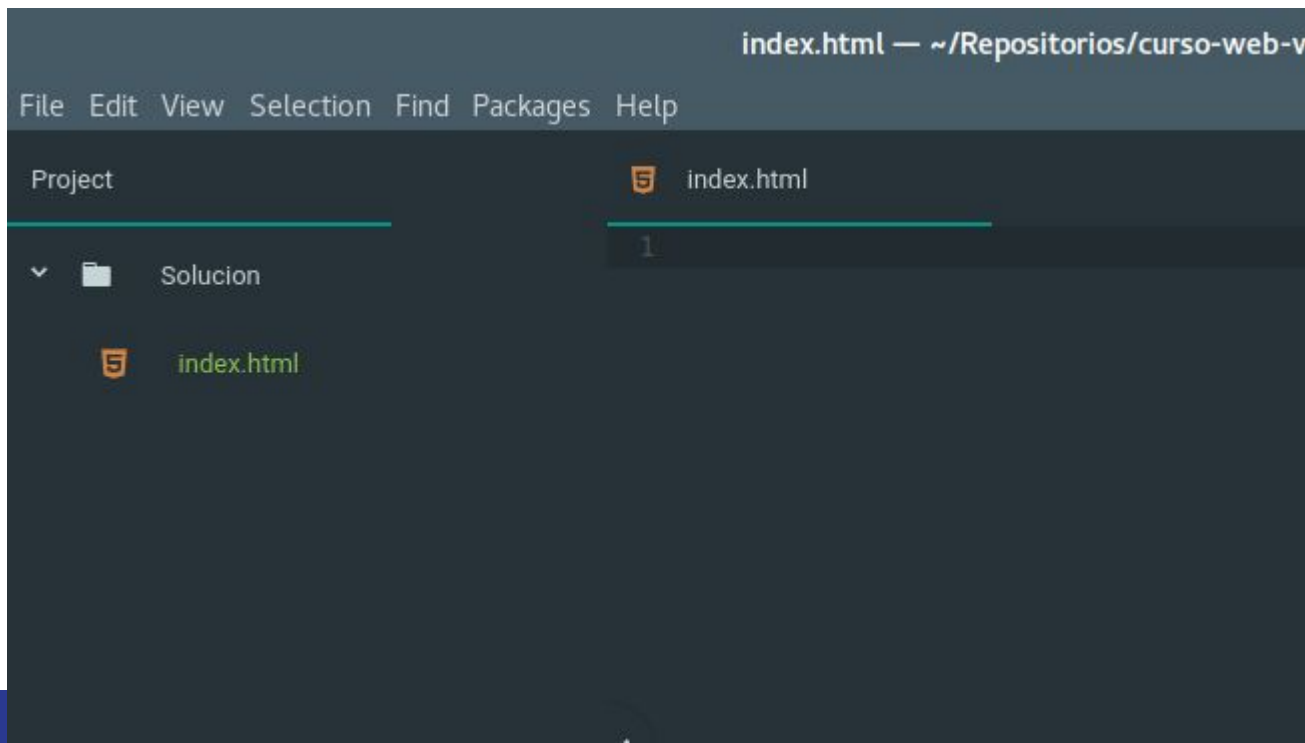
Como vimos antes para definir una página web es necesario un archivo HTML, para esto, vamos a crear uno:

1. Click en archivo.
2. Nuevo Archivo.
3. Click en archivo y luego guardar como.
4. Seleccionamos una ubicación y lo guardamos como index.html



Unidad N°3 Creamos una página simple

Una vez creado el archivo tendría que aparecer algo así:



Unidad N°3 Creamos una página simple

Algunas consideraciones al nombrar archivos:

- Usar el tipo de archivo adecuado en la extensión (.html, .png, .gif)
- Tratar de evitar los espacios (usar - o _).
- Evitar caracteres especiales como %, #, /, :, ;, etc
- entre otras cosas.



Unidad N°3 Creamos una página simple

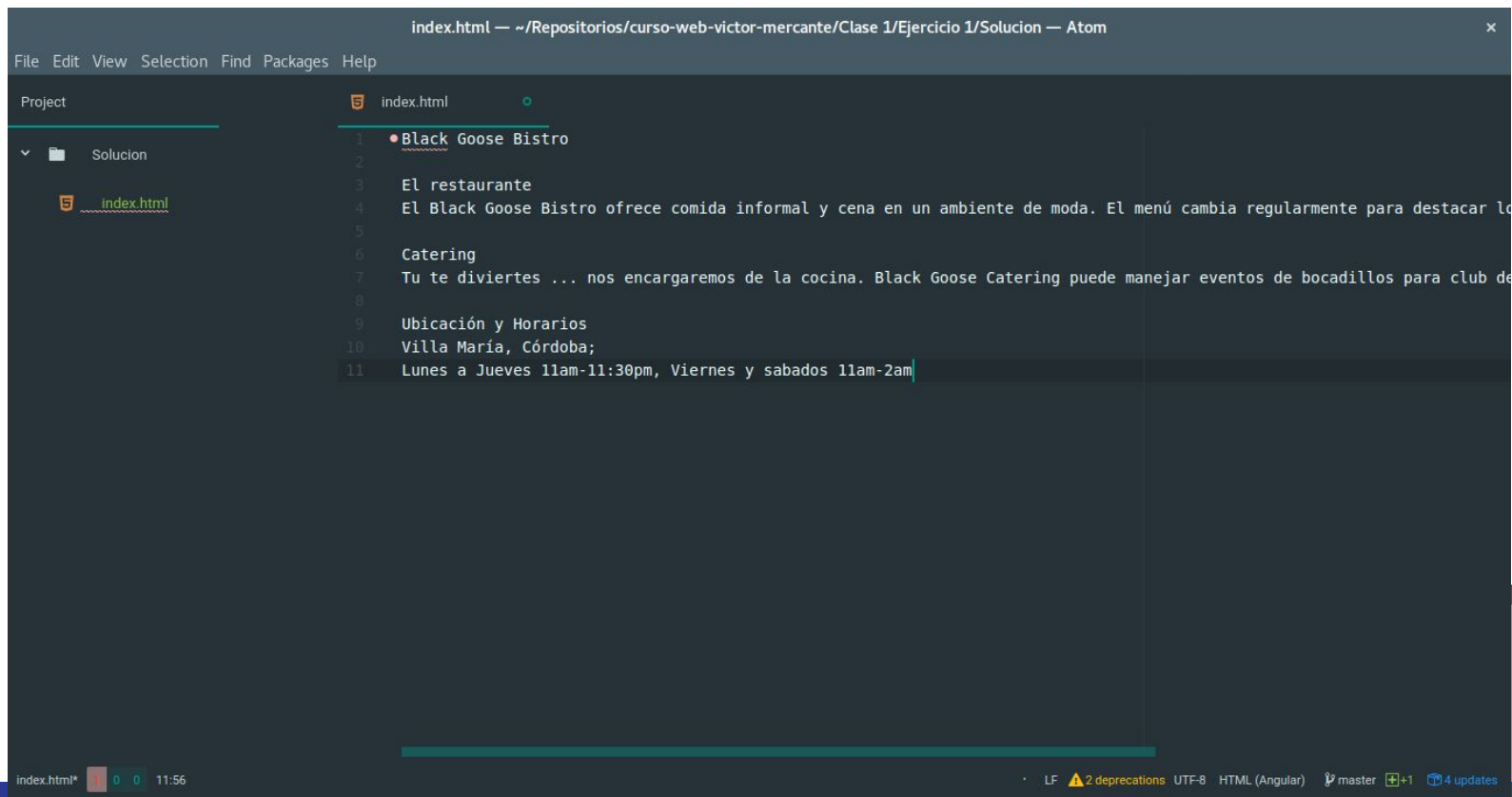
Paso N° 1 definir el contenido

Toda página web tiene un contenido, en nuestro caso lo tenemos definido en el archivo `blackgoose.txt`

Este paso consiste en copiar o transcribir este contenido a nuestro archivo HTML.



Unidad N°3 Creamos una página simple



The screenshot shows the Atom code editor with a file named `index.html` open. The editor's interface includes a menu bar (File, Edit, View, Selection, Find, Packages, Help), a project sidebar on the left showing a folder named 'Solucion' containing 'index.html', and a main text area. The text area contains the following HTML content:

```
1 •Black Goose Bistro
2
3 El restaurante
4 El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar lo
5
6 Catering
7 Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para club de
8
9 Ubicación y Horarios
10 Villa María, Córdoba;
11 Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am
```

The status bar at the bottom of the editor displays 'index.html*' on the left, a central indicator with '0' and '0' in red and green boxes, and the time '11:56'. On the right side of the status bar, it shows 'LF', '2 deprecations' with a warning icon, 'UTF-8', 'HTML (Angular)', 'master', '+1', and '4 updates'.

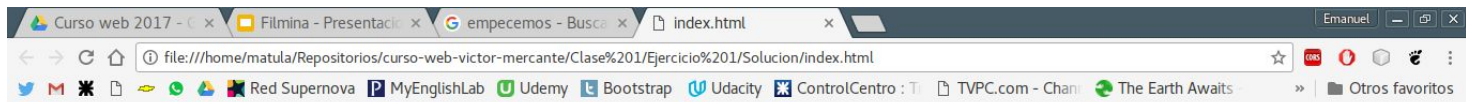
Unidad N°3 Creamos una página simple

Continuamos paso 1:

Guardamos el archivo y le damos doble click.



Unidad N°3 Creamos una página simple




Black Goose Bistro El restaurante El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar los ingredientes más frescos. Catering Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para club de bridge a recaudadores de fondos corporativos elegantes. Ubicación y Horarios Villa María, Córdoba; Lunes a Jueves 11am-11:30pm, Viernes y sábados 11am-2am



Unidad N°3 Creamos una página simple

Resultado paso 1:

No parece una página el texto está todo pegado, el navegador no muestra los quiebres de línea, la apariencia es fea. Esto nos da pauta para ver que ignoran los navegadores en un documento HTML inicialmente:

- Múltiples espacios.
 - Quiebres de líneas (enter entre una oración y otra).
 - Tabs.
 - Texto en comentarios.
 - Marcado no reconocido.
- 

Unidad N°3 Creamos una página simple

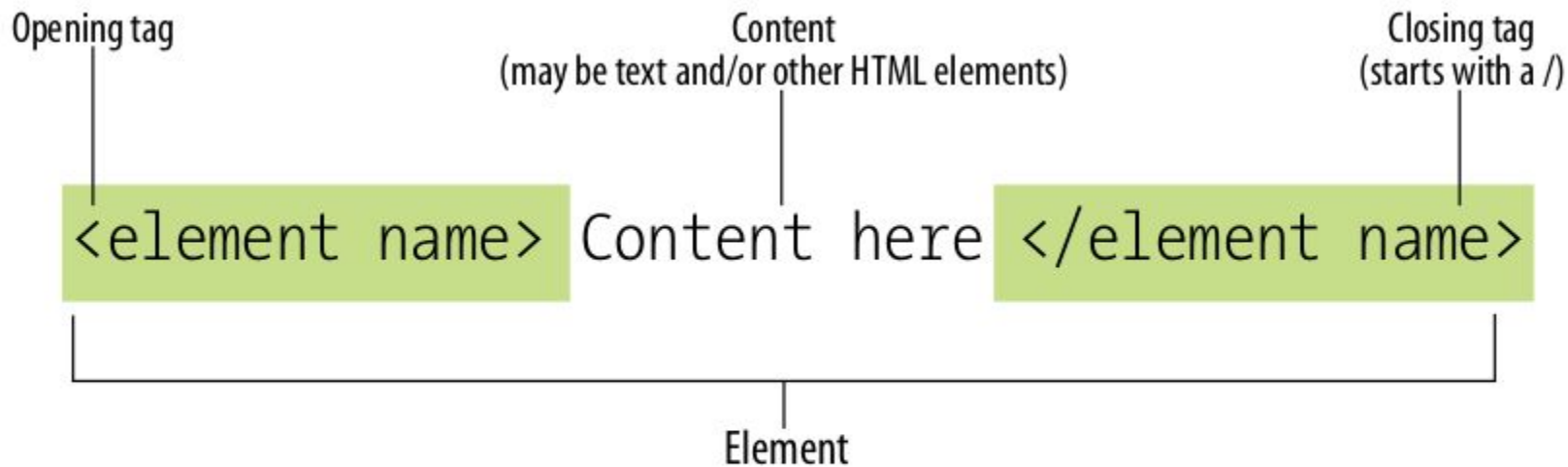
Paso 2, dar estructura al contenido:

Para dar estructura a un contenido HTML usa una estructura llamada elemento HTML o etiqueta HTML.

Veamos un ejemplo genérico.



Unidad N°3 Creamos una página simple



```
<h1>Black Goose Bistro</h1>
```

Unidad N°3 Creamos una página simple

Como vemos un elemento HTML tiene dos partes:

- **Tag:**
 - es el identificador de un elemento html y define su nombre.
 - Los navegadores no muestran el tag en la presentación de la página, solo su contenido.
 - Tiene un start tag o tag de apertura
 - `<nombre-tag>` (va entre `<>`).
 - Un end tag o tag de cierre
 - `</nombre-tag>` (va entre `</>`)
 - Al añadir un tag alrededor de un contenido, tiene el nombre de MARCADO (función de html).
- **Contenido:**
 - texto que es marcado por un tag.

Unidad N°3 Creamos una página simple

Algunas cosas:

- No todos los elementos html tienen contenido, estos son llamados elementos vacíos y solo tienen el tag de apertura (ejemplo imagen - filmina posterior).
- Los tag no son case sensitive, es decir que `<h1></h1>` y `<H1></H1>` se refieren al mismo tag.



Unidad N°3 Creamos una página simple

Volviendo al paso 2: Vamos a definir la estructura mínima que todo documento HTML necesita.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Cualquier titulo</title>
  </head>
  <body>
    El contenido de la página aca.
  </body>
</html>
```

Unidad N°3 Creamos una página simple

Analizemos las partes:



Unidad N°3 Creamos una página simple

1. La primera línea es la declaración del tipo de documento e identifica que el documento es un documento HTML5, esto le indica a los navegadores que deberían interpretar el documento según esa versión del estándar.
2. El tag `<html>` contiene a todos los elementos del documento, por ende al documento entero. No debería ser contenido por otro tag.
3. Todo documento dentro de `<html></html>` es dividido en dos partes:
 - a. `<head></head>`
 - b. `<body></body>`




Unidad N°3 Creamos una página simple

El head contiene información descriptiva sobre la página tales como el título, links a archivos css, scripts y otra “meta” información:

4. El elemento `<meta></meta>` da información acerca del documento, por ejemplo el charset encoding (charset), el cual define la colección de letras , números y símbolos.

5. El `<title></title>` también se incluye en el head y es un elemento obligatorio en todo documento, lo que hace es indicar el título el cual se ve en el nombre de la pestaña o bien cuando se agrega una página a los marcadores del navegador..



Unidad N°3 Creamos una página simple

6. Por último `<body></body>`, es el elemento que contiene todo lo que se quiere mostrar en el navegador.



Unidad N°3 Creamos una página simple

Ahora si para terminar el paso 2:

Escribimos la estructura mínima en nuestro archivo index.html y luego dentro del elemento `<body></body>` copiamos el contenido que escribimos anteriormente.

Cambiamos el título por Black Goose Bistro.

Guardamos el archivo.



Project

index.html

Solucion

blackgoose.png

index.html

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="utf-8">
5      <title>Cualquier titulo</title>
6    </head>
7    <body>
8      Black Goose Bistro
9
10     El restaurante
11     El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar
12
13     Catering
14     Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para clu
15
16     Ubicación y Horarios
17     Villa María, Córdoba;
18     Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am
19   </body>
20 </html>
21
```

index.html 0 0 0 20:8

LF 2 deprecations UTF-8 HTML (Angular) master +21 4 updates

Unidad N°3 Creamos una página simple

Para finalizar el paso 2:

Hacemos nuevamente doble click en el archivo o refrescamos la pestaña del navegador.



Black Goose Bistro El restaurante El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar los ingredientes más frescos. Catering Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para club de bridge a recaudadores de fondos corporativos elegantes. Ubicación y Horarios Villa María, Córdoba; Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am

Unidad N°3 Creamos una página simple

Hasta el momento parece que todo sigue igual, solo cambio el título del documento en la pestaña, pero ahora aunque no se note se le empezó a definir una estructura al documento lo cual es muy importante.



Unidad N°3 Creamos una página simple

Paso 3: Dar estructura al texto- identificar elementos de texto

El propósito de HTML es dar **SIGNIFICADO Y ESTRUCTURA** al contenido.

Para estructurar el contenido se debe elegir el elemento html que de la descripción más significativa para ese contenido. Esto se llama **marcado semántico**.



Unidad N°3 Creamos una página simple

Por ejemplo, El título o titular más importante en la página debería marcarse con el elemento `<h1>` (header level 1) el cual se usa para indicar títulos importante en la página.

- Puede ser que por ahora suenan raro pero veamos un ejemplo en nuestro ejercicio y expliquemos mejor.



Unidad N°3 Creamos una página simple

Finalizando el paso 3:

- Agregamos un `<h1>` a la primera línea Black Goose Blstro.
- Para los próximos títulos un `<h2>` para cada uno.
- Los títulos están seguidos de texto en párrafos, por ende, se deben marcar con el elemento `<p>` (párrafo).
- Agregamos al final un elemento `` con el texto gracias por la visita!!
- Guardamos el archivo y refrescamos la página.



Black Goose Bistro

El restaurante

El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar los ingredientes más frescos.

Catering

Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para club de bridge a recaudadores de fondos corporativos elegantes.

Ubicación y Horarios

Villa María, Córdoba; Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am

Gracias por la visita!!!

Unidad N°3 Creamos una página simple

Podemos introducir ahora un nuevo concepto **Elementos en línea y en bloque**

- Elementos en bloques: Un elemento en bloque ocupa todo el espacio de su elemento padre (contenedor), creando así un "bloque". Al finalizar empieza una nueva línea generando un espacio entre este y el próximo elemento. Pueden contener a otros elementos en bloques (depende) y elementos en línea.
- Elemento en línea: Un elemento en línea ocupa sólo el espacio delimitado por las etiquetas que definen el elemento en línea. No empieza una nueva línea, si a su lado hay otro elemento de línea y entran en el ancho de la página se ponen uno al lado del otro.

Unidad N°3 Creamos una página simple

Los elementos como títulos (h1) y párrafos son elementos en bloque.



Unidad N°3 Creamos una página simple

Elementos como el span, imaganes y otros que veremos a continuación son elementos en línea.

Ubicación y Horarios

span | 144.38 x 17 | ba; Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am
Gracias por la visita!!!

Unidad N°3 Creamos una página simple

Otra cosa que podemos concluir es que los navegadores asignan estilos por defectos a un contenido marcado con html.

Esto lo hacen porque cada navegador tiene definido su propio CSS.

Más adelante en el paso 5 vamos a cambiar un poco el estilo.



Unidad N°3 Creamos una página simple

Paso 4: Agregamos una imagen

Antes de poder agregar una imagen es importante explicar dos aspectos:

- Elementos html vacíos.
- atributos de elementos html.



Unidad N°3 Creamos una página simple

Elementos vacíos

Estos elementos solo tienen el tag de apertura o inicio y pueden o no contener uno o más atributos. Tienen la siguiente forma:

```
<elemento atributo1="" atributo2="">
```

El elemento para las imágenes es un elemento vacío. Le dice al navegador que obtenga un archivo de imagen desde el servidor y que la “dibuje” en el flujo de texto donde se encuentra este elemento.

Por ende es un elemento en línea



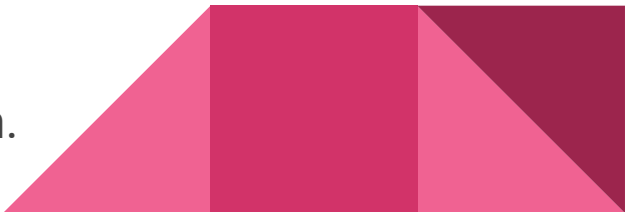
Unidad N°3 Creamos una página simple

Atributos: Son instrucciones que modifican un elemento. Tienen la forma:

atributo="valor"

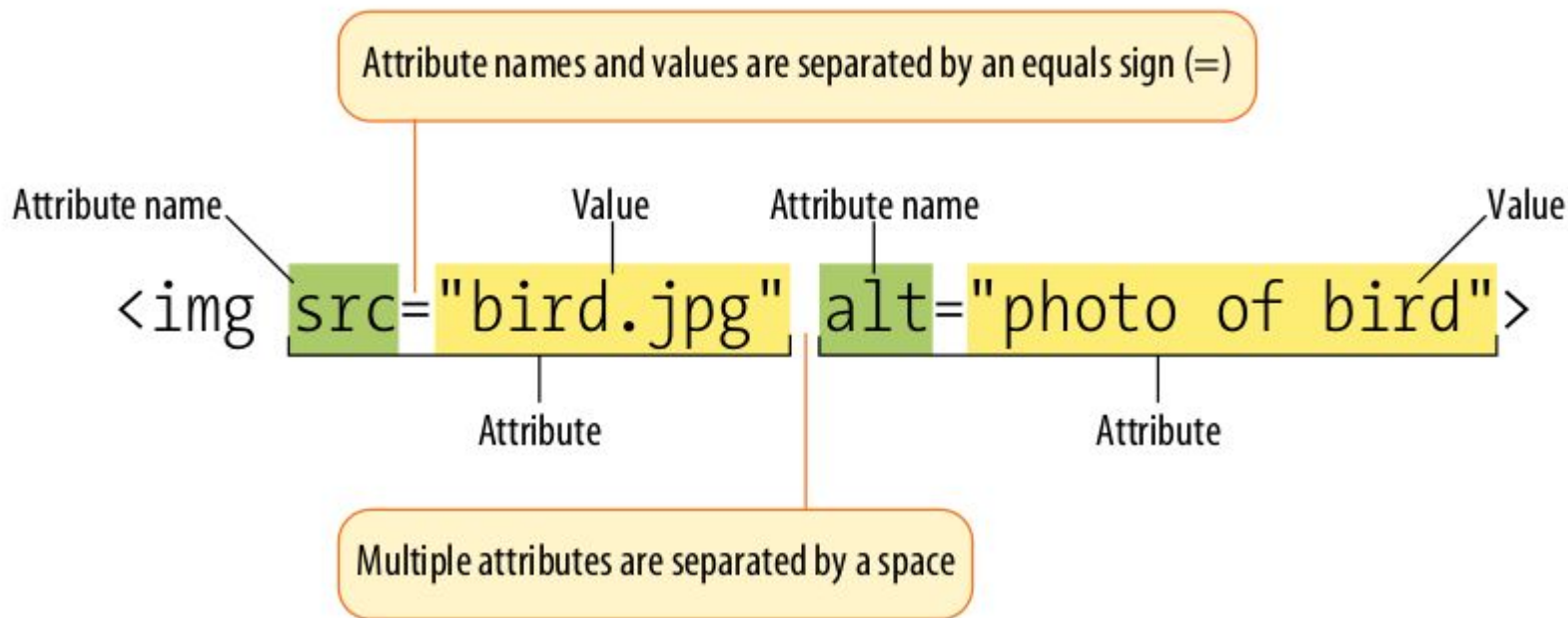
- Se ubican en el tag de apertura.
- Múltiples atributos se separan por espacios en blanco.
- Algunos atributos son requeridos otros no, depende del elemento.

Por ejemplo el elemento tiene dos atributos obligatorios:

- src (source): que indica la url de la imagen.
 - alt: texto que se muestra si no se recupera la imagen.
- 

Unidad N°3 Creamos una página simple

Resumiendo el elemento imagen:



Unidad N°3 Creamos una página simple

Ahora si, agregamos una imagen:

- Copiamos la imagen blackgoose.png de la carpeta clase 1- ejercicio 1 y la pegamos en la misma carpeta donde esta nuestro archivo index.html
- Agregamos un elemento `` dentro del `<h1>` y antes del texto Black Goose Bistro.
- En el atributo `src` escribimos blackgoose.png para indicarle que esa es la imagen a mostrar.
- En el atributo `alt` escribimos Black Goose logo
- guardamos y refrescamos la página.





Black Goose Bistro

El restaurante

El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar los ingredientes más frescos.

Catering

Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para club de bridge a recaudadores de fondos corporativos elegantes.

Ubicación y Horarios

Villa María, Córdoba; Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am

Gracias por la visita!!!

Unidad N°3 Creamos una página simple

Para que quede separado la imagen del título, vamos a agregar un elemento llamado quiebre de línea `
` el cual finaliza una línea y comienza una nueva.

Lo agregamos luego de la etiqueta ``.





Black Goose Bistro

El restaurante

El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar los ingredientes más frescos.

Catering

Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para club de bridge a recaudadores de fondos corporativos elegantes.

Ubicación y Horarios

Villa María, Córdoba; Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am

Gracias por la visita!!!

Unidad N°3 Creamos una página simple

Paso 5: Cambiamos la apariencia con una hoja de estilos

Vamos a cambiar la apariencia de los elementos de texto son algunas reglas simples de estilo. Esto es a manera de introducción, la parte fuerte de css viene una vez que avancemos un poco con los elementos de html.



Unidad N°3 Creamos una página simple

Vamos a agregar la etiqueta `<style></style>` dentro del head.

Esta etiqueta es una de las varias formas para agregar estilos a una página.

```
<head>
  <meta charset="utf-8">
  <title>Black Goose Bistro</title>
  <style>

</style>
</head>
```

Unidad N°3 Creamos una página simple

Copiamos el contenido del archivo style.css que se encuentra en la carpeta clase 1 - ejercicio 1 dentro de la etiqueta style recién agregada.





BLACK GOOSE BISTRO

El restaurante

El Black Goose Bistro ofrece comida informal y cena en un ambiente de moda. El menú cambia regularmente para destacar los ingredientes más frescos.

Catering

Tu te diviertes ... nos encargaremos de la cocina. Black Goose Catering puede manejar eventos de bocadillos para club de bridge a recaudadores de fondos corporativos elegantes.

Ubicación y Horarios


Villa María, Córdoba; Lunes a Jueves 11am-11:30pm, Viernes y sabados 11am-2am

Gracias por la visita!!!

Unidad N°3 Creamos una página simple

En resumen:

html	Es el elemento raíz o base que contiene a todos los otros elementos
head	Identifica la cabecera del documento que contiene información sobre el documento
body	Identifica el cuerpo del documento que contiene al contenido
title	Define el título de la página.
meta	Provee información acerca del documento.



Unidad N°4 Marcando texto

Objetivos:

Una vez marcada la estructura del documento se debe identificar los elementos presentes en el contenido, es importante elegir de acuerdo a la semántica es decir de acuerdo al significado.

Material:

- Clase 1 - ejercicios:
 - 2.1
 - 2.2
 - 2.3



Unidad N°4 Marcando texto

Párrafos

- Tiene la forma `<p>contenido</p>`
- Son elementos en bloque.
- Pueden contener texto, imágenes, otros elementos en línea.
- No puede contener cualquier otro elemento que es mostrado en línea (no es recomendado)

```
<p>Esto es un parrafo</p>
```

```
<p>Esto es otro parrafo</p>
```

Unidad N°4 Marcando texto

Headings (Títulos o titulares)

- Es un elemento para contener o indicar contenido introductorio o un set de links de navegación.
- Hay 6 niveles, la importancia es mayor mientras menor es el número de nivel.

```
<h1>Encabezado nivel 1</h1>  
<h2>Encabezado nivel 2</h2>  
<h3>Encabezado nivel 3</h3>  
<h4>Encabezado nivel 4</h4>  
<h5>Encabezado nivel 5</h5>  
<h6>Encabezado nivel 6</h6>
```



Unidad N°4 Marcando texto

Headings (continua)

- Los motores de búsqueda como google, inspeccionan los headings como parte de su lógica para las búsquedas.
- El h1 es el de mayor peso y es normal o buena práctica empezar con un h1.
- Los navegadores también crean el document outline el cual es usado para indexar contenido tal como lo hace wikipedia.
- Veamos un ejemplo clase 1- ejemplos - ejemplo1.html



Type Design

Serif Typefaces

Serif typefaces have small slabs at the ends of letter strokes. In general, serif fonts can make large amounts of text easier to read.

Baskerville

Description

Description of the Baskerville typeface.

History

The history of the Baskerville typeface.

Georgia

Description and history of the Georgia typeface.

Sans-serif Typefaces

Sans-serif typefaces do not have slabs at the ends of strokes.

Unidad N°4 Marcando texto

Document online generado:

1. Type Design
 1. Serif Typefaces
+ text paragraph
 1. Baskerville
 1. Description
+ text paragraph
 2. History
+ text paragraph
 2. Georgia
+ text paragraph
2. Sans-Serif Typefaces
+ text paragraph

Índice [\[ocultar\]](#)

- 1 Historia
- 2 Partes principales en vehículos automóviles
- 3 Orden de masa en vehiculos automoviles
- 4 Clasificación de vehículos automóviles
 - 4.1 Según Reglamento de Homologación nº 13
 - 4.2 Según Directivas CE 77/143, 88/449, 91/328
- 5 Método de propulsión
 - 5.1 Combustibles
 - 5.2 Accionamiento eléctrico
 - 5.3 Accionamiento hibrido
 - 5.4 Otros sistemas de propulsión
- 6 Datos técnicos de un automóvil que figuran en los catálogos comerciales
- 7 Contaminación
- 8 Véase también
- 9 Referencias
- 10 Enlaces externos

Unidad N°4 Marcando texto

Regla horizontal (cambio de tema)

- Tiene la forma `<hr>`.
- Se usa para indicar que un tema terminó y comienza otro.
- Por lo general se dibuja una línea separadora en el navegador. ejemplo 2

```
<h3>Times</h3>
<p>Description and history of the Times typeface.</p>

<hr>

<h3>Georgia</h3>
<p>Description and history of the Georgia typeface.</p>
```

Unidad N°4 Marcando texto



Times

Description and history of the Times typeface.

Georgia

Description and history of the Georgia typeface.

Unidad N°4 Marcando texto

Listas

HTML ofrece dos tipos de listas:

- Listas no ordenadas.
- Listas ordenadas.

Todos los tipos de listas y los elementos que van dentro de ellas son mostrados como elementos en bloque.

clase 1 -ejemplos- ejemplo 3



Unidad N°4 Marcando texto

Listas no ordenadas

Es una lista que permite listar ejemplos, opciones, o cualquier cosa como una lista donde el orden no importa, por lo general cada elemento de la lista se dibuja con un bullet (punto negro). Tiene la forma:

`` este tag define una lista no ordenada (unordered list)

`` (este tag define un ítem de lista)

``

``



Unidad N°4 Marcando texto

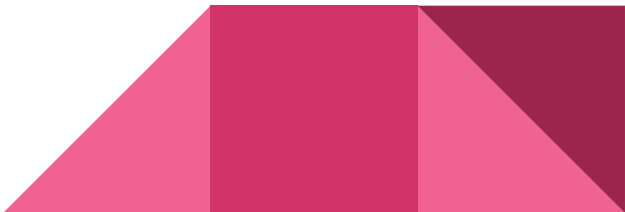
Listas no ordenadas

```
<h3>Lista no ordenada</h3>
<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  <li>Elemento 3</li>
</ul>
```

Lista no ordenada

- Elemento 1
- Elemento 2
- Elemento 3

- Dentro de un solo pueden ir elementos
- Dentro de un elemento puede ir cualquier cosa.



Unidad N°4 Marcando texto

Con css se puede cambiar
la forma en que se muestran
las listas no ordenadas

Serif
Sans-serif
Script
Display
Dingbats

 SERIF

 SANS-SERIF

 SCRIPT

 DISPLAY

 DINGBATS

Serif

Sans-serif

Script

Display

Dingbats

SERIF

SANS-SERIF

SCRIPT

DISPLAY

DINGBATS

Unidad N°4 Marcando texto

Listas ordenadas

Es una lista que permite listar ítems que ocurren en un orden determinado o tiene cierta importancia, ejemplo pasos de una receta, cada ítem tiene asignado un número de orden pero se puede cambiar por letras u otros, tiene la forma:

`` este tag define una lista ordenada (ordered list)

`` (este tag define un ítem de lista)

``

``



Unidad N°4 Marcando texto

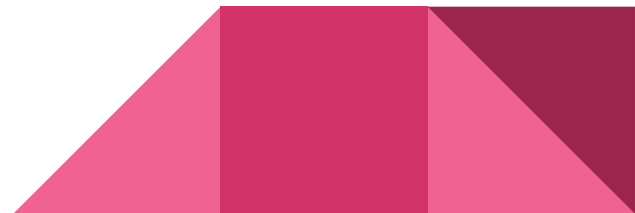
Listas ordenadas

```
<h3>Lista ordenada</h3>
<ol>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  <li>Elemento 3</li>
</ol>
```

Lista ordenada

1. Elemento 1
2. Elemento 2
3. Elemento 3

- Dentro de un solo pueden ir elementos
- Dentro de un elemento puede ir cualquier cosa.



Unidad N°4 Marcando texto

Listas ordenadas

Se puede cambiar el valor de inicio de una lista ordenada con el atributo `start="valor de inicio"`

```
<h3>Lista ordenada inicio distinto a 1</h3>  
<ol start="12">  
  <li>Elemento 1</li>  
  <li>Elemento 2</li>  
  <li>Elemento 3</li>  
</ol>
```

Lista ordenada inicio distinto a 1

12. Elemento 1
13. Elemento 2
14. Elemento 3

Unidad N°4 Marcando texto

Anidando listas: es posible anidar listas dentro de las otras

```
<h3>Lista no ordenada anidada con una ordenada</h3>
<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  <li>Elemento 3:
    <ol>
      <li>Elemento 1</li>
      <li>Elemento 2</li>
      <li>Elemento 3</li>
    </ol>
  </li>
</ul>
```

Lista no ordenada anidada con una ordenada

- Elemento 1
- Elemento 2
- Elemento 3:
 1. Elemento 1
 2. Elemento 2
 3. Elemento 3

Unidad N°4 Marcando texto

```
<h3>Lista ordenada anidada con una no ordenada</h3>
<ol>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  <li>Elemento 3:
    <ul>
      <li>Elemento 1</li>
      <li>Elemento 2</li>
      <li>Elemento 3</li>
    </ul>
  </li>
</ol>
```

Lista ordenada anidada con una no ordenada

1. Elemento 1
2. Elemento 2
3. Elemento 3:
 - Elemento 1
 - Elemento 2
 - Elemento 3



Unidad N°4 Marcando texto

Se puede cambiar el estilo de los índices mostrados en las listas puede ser cambiado mediante css a través de la propiedad `list-style-type`

Ver ejemplo 4



Unidad N°4 Marcando texto

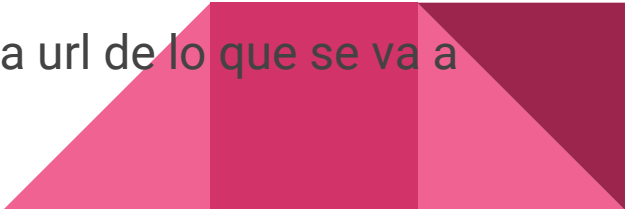
Long quotations (citas largas)

Este elemento es usado para citas largas, testimonios de personas o una cita de copia de algún recurso por parte de otra fuente.

Tiene la forma: `<blockquote></blockquote>`

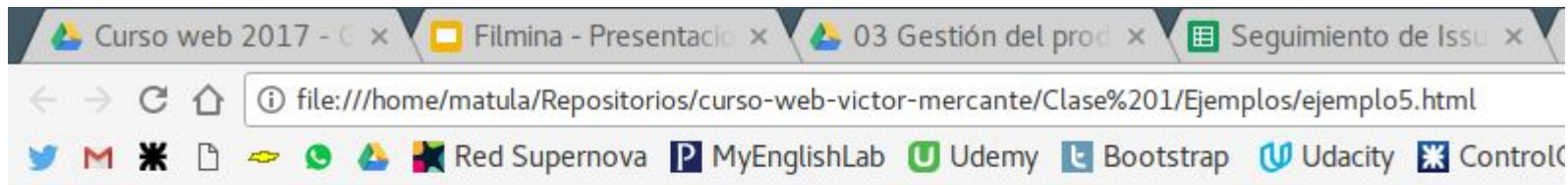
Es buena práctica que lo que va dentro de este elemento sean párrafos, listas, headings, entre otros.

Puede contener el atributo `cite="url"` que permite indicar la url de lo que se va a citar.



Unidad N°4 Marcando texto

```
<p>Albert Einstein una vez dijo:</p>
<blockquote cite="http://www.proverbia.net/citasautor.asp?autor=327">
  <p>La vida es muy peligrosa. No por las personas que hacen el mal,
</blockquote>
```



Albert Einstein una vez dijo:

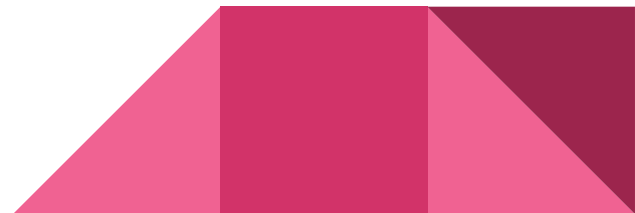
La vida es muy peligrosa. No por las personas que hacen el mal, sino por las que se sientan a ver lo que pasa.

Unidad N°4 Marcando texto

Ejercicio N°2:

En la carpeta clase 1- ejercicios se encuentra la carpeta ejercicio 2, esta carpeta contiene un archivo de texto, el objetivo es aplicar formato a dicho texto con los elementos que vimos hasta ahora:

- headings, listas, párrafos, blockquotes.



Unidad N°4 Marcando texto

Organizando el contenido de la página (código de ejemplo 6):

HTML5 introduce nuevos elementos con diferentes semántica para darle significado a diferentes secciones de la página:

- secciones (section).
- artículos (article).
- navegación (nav).
- contenido relacionado (aside).
- Encabezados (header).
- Pie de página (footer).

*Tener en cuenta compatibilidad navegadores (IE8-9)



Unidad N°4 Marcando texto

Secciones

- Las secciones son usadas para dividir largos documentos en secciones temáticas o identificar secciones temáticas dentro de un único artículo.
- Suelen tener un heading y cualquier otro contenido que tenga sentido agrupar.
- Tiene la forma:

`<section></section>`

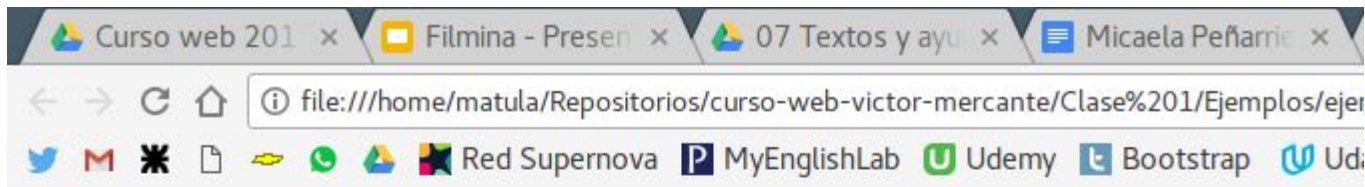


Unidad N°4 Marcando texto

Por ejemplo, usar secciones para dividir un documento con información acerca de técnicas de tipografía

```
<h3>Ejemplo secciones para dividir recursos disponibles sobre un tema</h3>
<section>
  <h2>Libros sobre tipografia</h2>
  <ul>
    <li>libro 1</li>
    <li>libro 2</li>
  </ul>
</section>
<section>
  <h2>Tutoriales de tipografia</h2>
  <p>Estos son los mejores tutoriales en la web:</p>
  <ul>
    <li>tutorial 1</li>
    <li>tutorial 2</li>
  </ul>
</section>
<hr>
```

Unidad N°4 Marcando texto



Ejemplo secciones para dividir recursos disponibles sobre un tema

Libros sobre tipografia

- libro 1
- libro 2

Tutoriales de tipografia

Estos son los mejores tutoriales en la web:

- tutorial 1
- tutorial 2

Unidad N°4 Marcando texto

Artículos

- Especifica contenido independiente y auto contenido que puede estar solo o ser reusado en un contexto diferente.
- Es usado cuando tiene sentido que este solo por su cuenta y pueda ser usado independientemente del resto de los elementos del sitio.
- Usos potenciales:
 - Post en foros y blogs.
 - Noticias de diarios o artículos de revistas.
 - Comentarios
- Tiene la forma: `<article></article>`



Unidad N°4 Marcando texto

Ejemplo la sección de deportes de un diario dividido en artículos

```
<h3>Seccion con varios articulos</h3>
<section >
  <article>
    <h4>Una mirada a la actualidad del futbol</h4>
    <p>...</p>
  </article>
  <article>
    <h4>El tenis Argentino en el mejor momento de la historia</h4>
    <p>...</p>
  </article>
</section>
<hr>
```

Unidad N°4 Marcando texto

Seccion con varios articulos

Una mirada a la actualidad del futbol

...

El tenis Argentino en el mejor momento de la historia

...

Ejemplo:

<https://lamiradadelreplicante.com/2017/05/24/crear-y-leer-codigos-qr-en-linux/>

En este ejemplo el blog usa el arituclo para describir cada post dentro del blog



Unidad N°4 Marcando texto

También es posible dividir un artículo en secciones:

```
<h2>articulo con varias secciones</h2>
<article>
  <h3>Conocer Helvetica</h3>
  <section>
    <h4>Historia de Helvetica</h4>
    <p>Helvetica es una tipografia desarrollada ...</p>
  </section>
  <section>
    <h4>Helvetica actualmente</h4>
    <p>...</p>
  </section>
</article>
<hr>
```

articulo con varias secciones

Conocer Helvetica

Historia de Helvetica

Helvetica es una tipografia desarrollada ...

Helvetica actualmente

...



Unidad N°4 Marcando texto

Aside (sidebars)

- Identifica contenido que está relacionado al contenido que lo rodea.
- Es similar a un sidebar pero no es un sidebar, pero es usada como barra lateral y contienen por ejemplo un glosario, lista de links, publicidad, la biografía del autor, o en aplicaciones web, la información de perfil o enlaces a blogs relacionados.
- En cambio un sidebar si puede ser modelado con este elemento
- Tiene la forma : `<aside></aside>`



Unidad N°4 Marcando texto

```
<h2>Ejemplo aside</h2>
<h3>Web Typography</h3>
<p>Back in 1997, there were competing font formats and tools for making them...
</p>
<p>We now have a number of methods for using beautiful fonts on web pages...
</p>
<aside>
  <h2>Web Font Resources</h2>
  <ul>
    <li><a href="http://typekit.com/">Typekit</a></li>
    <li><a href="http://www.google.com/webfonts">Google Fonts</a></li>
  </ul>
</aside>
<hr>
```

Unidad N°4 Marcando texto

En este ejemplo vemos que no un aside no es un sidebar, no es necesario que esté ubicado como un sidebar, de hecho por defecto css no ubica un aside como un sidebar

Ejemplo aside

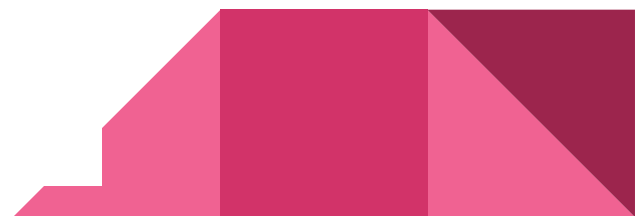
Web Typography

Back in 1997, there were competing font formats and tools for making them...

We now have a number of methods for using beautiful fonts on web pages...

Web Font Resources

- [Typekit](#)
- [Google Fonts](#)



Unidad N°4 Marcando texto

Como la etiqueta aside es candidata para marcar un sidebar, con un poco de css podemos hacer lo siguiente:

Ejemplo aside 2

Web Typography

Back in 1997, there were competing font formats and tools for making them...

We now have a number of methods for using beautiful fonts on web pages...

Web Font Resources

- [Typekit](#)
- [Google Fonts](#)



Unidad N°4 Marcando texto

Aside (continua):

- Por defecto se muestra como un elemento de bloque, para posicionarlo en un lugar distinto, por ejemplo al costado de la página es necesario usar CSS.



Unidad N°4 Marcando texto

Navigation (navegación)

- Permite identificar las maneras de navegar por la página.
- Por lo general contiene links que llevan a otra parte dentro de la misma página.
- Tiene la forma: `<nav></nav>`



Unidad N°4 Marcando texto

```
<h2>Navigation</h2>
<nav>
  <ul>
    <li><a href="#">Home</a></li>
  </ul>
  <ul>
    <li><a href="#">Contacto</a></li>
  </ul>
  <ul>
    <li><a href="#">Recursos</a></li>
  </ul>
</nav>
```

Navigation

- [Home](#)
 - [Contacto](#)
 - [Recursos](#)
-

El elemento `<a>` permite definir un enlace o link a otro recurso, lo veremos más adelante.

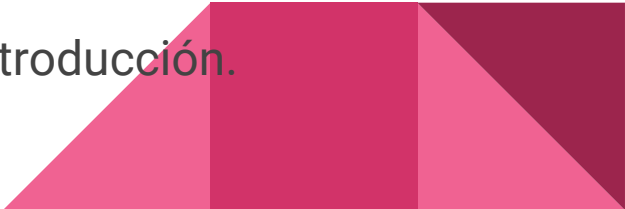
Unidad N°4 Marcando texto

Con un poco de css podemos hacer que el nav tenga mejor apariencia, por ejemplo Twitter:



Unidad N°4 Marcando texto

Headers

- Es usado para el contenido introductorio de una página.
 - Tiene la forma: `<header></header>`
 - Usualmente aparece al principio de una página web o al principio de una sección o artículo.
 - Casi siempre se combina con el elemento nav para que al principio de la página aparezca la navegación de la página.
 - Es el lugar ideal para poner el logo.
 - Contiene cualquier cosa que tenga sentido para la introducción.
- 

Unidad N°4 Marcando texto

```
<h3>Ejemplo de header</h3>
<header>
  
  <h1>Noticiario simpson</h1>
  <nav>
    <ul>
      <li><a href="">Home</a></li>
      <li><a href="">Historia</a></li>
      <li><a href="">Shop</a></li>
    </ul>
  </nav>
</header>
<hr>
```

Unidad N°4 Marcando texto

Ejemplo de header



Noticiario simpson

- [Home](#)
- [Historia](#)
- [Shop](#)

Unidad N°4 Marcando texto

Otro ejemplo

[Inicio](#)[El Lugar](#)[Menú](#)[Reservas](#)[Ubicación](#)[Contacto](#)

Nuestro lugar



Unidad N°4 Marcando texto

Footer (Pie de página)

- Es usado para indicar la información que por lo general va al final de la página o un artículo como:
 - autor.
 - información de copyright
 - documentos relacionados.
 - navegación.
- Puede aplicarse a toda la página o asociarlo a una sección o artículo particular.



Unidad N°4 Marcando texto

Footer (continua)

- Tiene la forma: `<footer></footer>`
- Si es aplicado entre los tag body aplica para toda la página.
- si es contenido dentro de una sección, artículo, nav, aside u otro, solo aplica a esa sección que lo contiene.
- No es necesario que se ubique al final, pero usualmente esa es su posición.



Unidad N°4 Marcando texto

```
<footer>
  <p><small>Copyright &copy;2012 Emanuel Matulich.</small></p>
  <nav>
    <ul>
      <li><a href="">Ir al inicio</a></li>
      <li><a href="">Mapa del sitio</a></li>
    </ul>
  </nav>
</footer>
```

Lo siguiente es el footer

Copyright ©2012 Emanuel Matulich.

- [Ir al inicio](#)
- [Mapa del sitio](#)

Unidad N°4 Marcando texto



[Términos y Condiciones](#) | [Política de Protección de Datos Personales](#) | [Normas de confidencialidad y privacidad](#)



Piedras 1743. C.A.B.A, ARGENTINA

Directora: Ernestina Herrera de Noble

Editor Responsable: Ricardo Kirschbaum

Registro de Propiedad Intelectual: 5205286

Edición N°: 25495



Información fiscal

AGEA DIGITAL

DIARIO OLÉ | ENTREMujeres | GRAN DT | ARGENPROP |
EMPLEOS CLARÍN | DE AUTOS | GUÍA CLARÍN |
CLASIFICADOS | RECEPTORÍA ONLINE |

GRUPO CLARÍN

GRUPO CLARÍN | TN | EL TRECE TV | MITRE | LA 100 |
CIUDAD | CIENRADIO | TYCSports |
LA VOZ DEL INTERIOR | LOS ANDES | VIAPAI | RUMBOS |

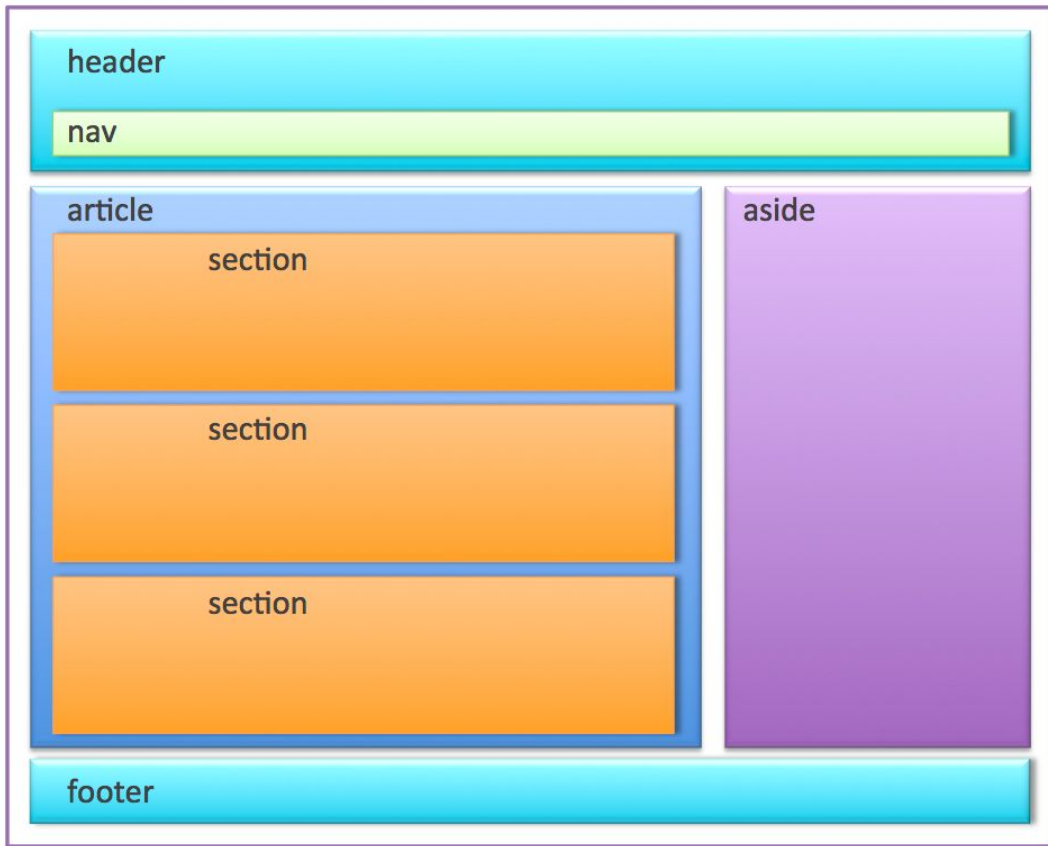
SUSCRIBIRME
CONTÁCTENOS
AYUDA
ARCHIVO

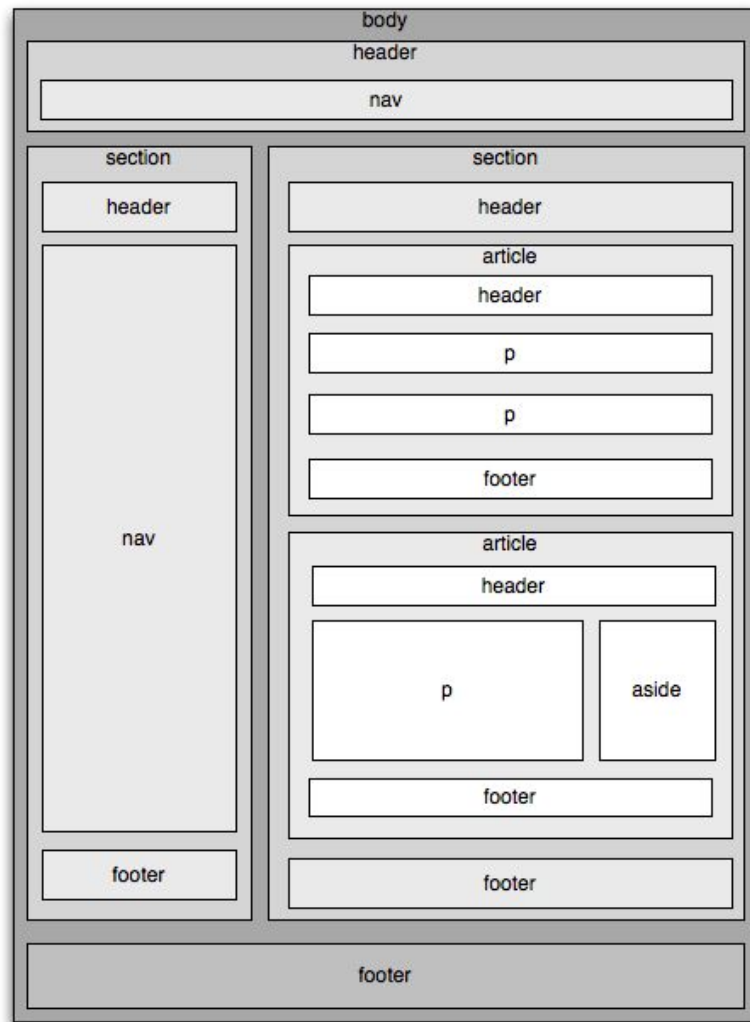
PUBLICIDAD

ANÚNCIESE CON
NOSOTROS

ageadigital

Registro Propiedad Intelectual: 5205286. Propietario Arte Gráfico Editorial Argentino S.A. © 1996-2017 Clarin.com - Clarin Digital - Todos los derechos reservados.





Unidad N°4 Marcando texto

Ver ejemplo <http://tn.com.ar/deportes> y ejemplo facebook



Unidad N°4 Marcando texto

Algunos elementos en línea: (Ejemplo 7)

- a (anchor unidad 5 entra en detalle).
- em
- strong
- small
- short quotations
- time
- line break
- word break

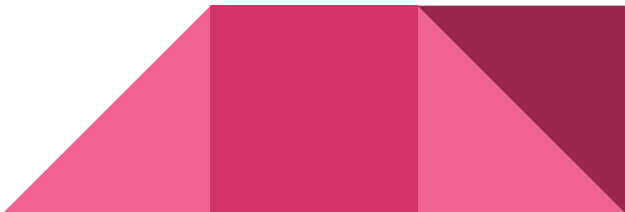


Unidad N°4 Marcando texto

Texto enfatizado (em)

- Permite indicar que una parte del texto tiene énfasis.
- Tiene la forma ``
- Por defecto muestra el texto en italics (cursiva).

Texto importante (strong)

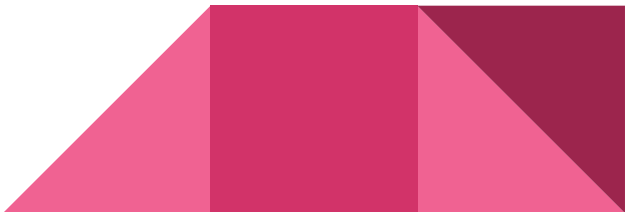
- Indica que una palabra o frase en el texto es importante.
 - Forma: ``
 - Por defecto se muestra en negrita (bold).
- 

Unidad N°4 Marcando texto

small

- Es usado para notas acerca del texto principal, como apéndice, como texto para aclarar referencias numeradas.
- Tiene la forma: `<small></small>`

Short quotations

- Son usadas para marcar citas cortas en alguna línea de un texto, lo contrario a la blockquote que genera un bloque.
 - Tiene la forma `<q></q>`
- 

Unidad N°4 Marcando texto

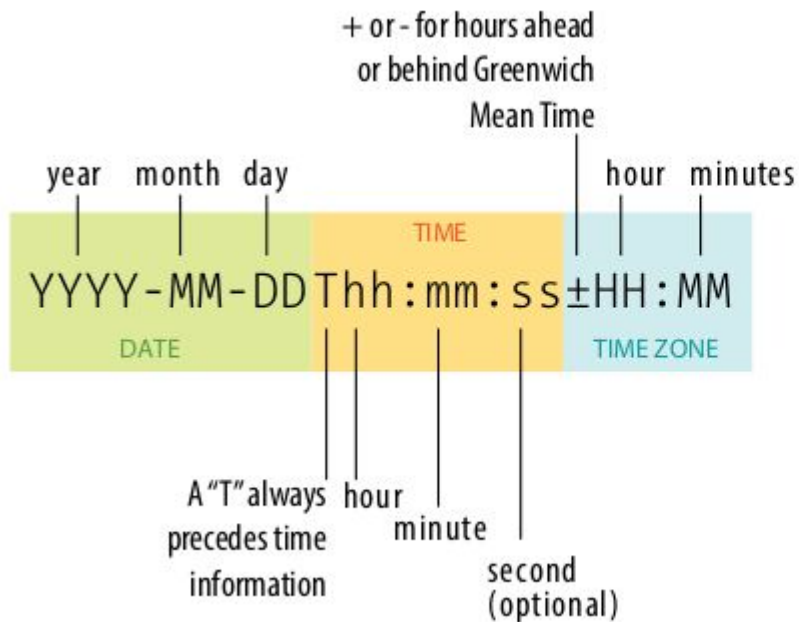
Time

- Permite marcar fechas u horas de manera que permita ser entendida por los usuarios.
- Tiene la forma: `<time datetime="formato"></time>`
- El atributo `datetime` permite especificar el formato con el que se muestra el texto, no el estilo.



Unidad N°4 Marcando texto

El atributo datetime para el tag time tiene la forma



Example:

3pm PST on December 25, 2012

`2012-12-25T15:00-8:00`

Unidad N°4 Marcando texto

Line breaks (quiebre de línea)

- Es un elemento vacío.
- Permite romper líneas, es decir, finaliza el flujo de texto en línea como si fuera un bloque.
- Tiene la forma `
`

Word Breaks (Quiebre de palabras)

- Es usado para marcar una palabra en donde debería ser cortada si es necesario.
- Tiene la forma `<wbr>`

Unidad N°4 Marcando texto

- Ver código y archivo ejemplo7.html

Ejercicio N°3 TAREA:

- En la carpeta clase 1 - ejercicio 3, hay un archivo de texto llamado elementos-en-línea, crear un documento html, copiar en el body el contenido del archivo, y agregar los elementos en líneas que vimos donde sea oportuno.



Unidad N°4 Marcando texto

Elementos genéricos (div y span) [Ejemplo8.html código de ejemplo]

- Si ninguno de los elementos anteriores describe el contenido de la página existen 2 elementos genéricos
- El elemento **div** indica una división de contenido, es un elemento de bloque.
- El elemento **span** indica una palabra o texto para cual no hay un elemento en línea que lo describa.
- A estos elementos se les da significado y contexto a través de dos atributos:
 - id.
 - class
- De igual manera id y class no son obligatorios.



Unidad N°4 Marcando texto

Elemento div

- Es usado para crear una agrupación de elementos o contenidos relacionados.
- Asignando un **id** único o indicando que pertenecen a una clase (**class**), se le da contexto a los elementos que agrupa.
- Tiene la forma:
- `<div id="valor de id" class="valor de clase"></div>`
- Ejemplos:
 - Tarjeta resumen de una noticia.
 - Noticia.
 - Post en facebook.
 - Un Twitter

Unidad N°4 Marcando texto

Ejemplo: tarjeta resumen noticia

*Mostrar variantes, comentando CSS que el div es un elemento en bloque

```
<h3>Uso de div</h3>
<h2>Tarjeta resumen noticia</h2>
<div class="card">
  
  <p><cite>La genialidad de homero</cite>, James Felici</p>
  <p>Una mezcla de genialidades y buena suerte ayudan a homero a ser un gran tipo.
  </p>
</div>
<div class="card" id="gris">
  
  <p><cite>La genialidad de homero</cite>, James Felici</p>
  <p>Una mezcla de genialidades y buena suerte ayudan a homero a ser un gran tipo.
  </p>
</div>
```

Unidad N°4 Marcando texto

Uso de div

Tarjeta resumen noticia



La genialidad de homero, James Felici

Una mezcla de genialidades y buena suerte ayudan a homero a ser un gran tipo.



La genialidad de homero, James Felici

Una mezcla de genialidades y buena suerte ayudan a homero a ser un gran tipo.

Unidad N°4 Marcando texto

Elemento en linea generico Span

- Ofrece las mismas ventajas de un div solo que es usado para palabras o frases.
- No introduce quiebre de línea.
- Es un elemento de línea.
- Puede ser contenidos en otros elementos en línea o en bloque.



Unidad N°4 Marcando texto

Ejemplo de span: No hay un elemento para representar números de teléfonos, pero con un span podemos rodear estos números y a través de la clase le damos un color azul a la letra.

```
<h3>Uso de span</h3>
<div>
  <ul>
    <li>John: <span class="tel">999.8282</span></li>
    <li>Paul: <span class="tel">888.4889</span></li>
    <li>George: <span class="tel">888.1628</span></li>
    <li>Ringo: <span class="tel">999.3220</span></li>
  </ul>
</div>
```

Uso de span

- John: 999.8282
- Paul: 888.4889
- George: 888.1628
- Ringo: 999.3220

Unidad N°4 Marcando texto

Atributos ID y CLASS

- Estos atributos pueden ser usados con cualquier elemento no solamente con div y span.
- El valor para un id o clase debe:
 - empezar con una letra (A-Z a-z) o guión bajo.
 - No puede contener espacios o caracteres especiales.
 - Letras, números, guión medio, están permitidos.
 - Es sensible a mayúsculas, la clase Hola es distinta a la clase hola.



Unidad N°4 Marcando texto

Identificación con ID

- El atributo id es usado para asignar un identificador único a un elemento en el documento.
- Es decir el valor que se le asigna a un id sólo puede ser asignado una vez.

Ejemplo: se asigna como id el ISBN a un div que contiene los datos de un libro.



Unidad N°4 Marcando texto

```
<h3>Uso del atributo id</h3>
<div id="ISBN0321127307">
  
  <p><cite>The Complete Manual of Typography</cite>, James Felici</p>
  <p>A combination of type history and examples of good and bad type.
  </p>
</div>
<div id="ISBN0881792063">
  
  <p><cite>The Elements of Typographic Style</cite>, Robert Bringhurst
  </p>
  <p>This lovely, well-written book is concerned foremost with creating beautiful typography.</p>
</div>
```

Unidad N°4 Marcando texto

Usando un poco de css aplicado
a esos div puede quedar algo como:

Uso del atributo id



The Complete Manual of Typography, James Felici

A combination of type history and examples of good and bad type.



The Elements of Typographic Style, Robert Bringhurst

This lovely, well-written book is concerned foremost with creating beautiful typography.

Unidad N°4 Marcando texto

Se puede usar el id ya que es único para identificar diferentes secciones de la página:

```
<section id="main">
  <!-- Elementos principales -->
</section>
<section id="news">
  <!-- noticias -->
</section>
<aside id="links">
  <!-- lista de links -->
</aside>
```

Unidad N°4 Marcando texto

Clasificaciones con clases (class)

- El atributo class clasifica los elementos en grupos, pero a diferencia del id, varios elementos pueden usar o compartir la misma clase.
- Haciendo a los elementos parte de la clase se puede aplicar estilos a todos ellos escribiendo solo una vez esos estilos.
- Ejemplo: uso de clase listing, description



Unidad N°4 Marcando texto

```
<h3>Uso del atributo class</h3>
<div id="ISBN03211273037" class="listing">
  <header>
    
    <p><cite>The Complete Manual of Typography</cite>, James Felici</p>
  </header>
  <p class="description">A combination of type history and examples of good and bad type.</p>
</div>
<div id="ISBN08817920623" class="listing">
  <header>
    
    <p><cite>The Elements of Typographic Style</cite>, Robert Bringhurst
  </p>
  </header>
  <p class="description">This lovely, well-written book is concerned foremost with creating beautiful typography.</p>
</div>
```



The Complete Manual of Typography, James Felici

A combination of type history and examples of good and bad type.



The Elements of Typographic Style, Robert Bringhurst

This lovely, well-written book is concerned foremost with creating beautiful typography.

Unidad N°4 Marcando texto

Caracteres especiales: Algunos caracteres como las vocales con acento, la letra ñ, símbolos mayor y menor, es conveniente que no sean escritos directos sino que se los codifica o escapa.

Para escapar un carácter se usa el símbolo & + codificación + ;

por ejemplo la letra ó se codifica ó

listado de todos los caracteres:

<http://www.theukwebdesigncompany.com/articles/entity-escape-characters.php>



Unidad N°4 Marcando texto

Ejercicio 4: Usando las imágenes, el archivo de estilo y el archivo con el contenido, tratar de replicar la página mostrada en la imagen.

El objetivo es crear dos clases:

- La clase resumen que engloba la imagen y la primera descripción de la noticia.
- La clase contenido que engloba el resto de la noticia.



Unidad N°5 Agregamos links

Es muy probable que una página tenga que apuntar a otras páginas o recursos, ya sea en el mismo sitio web o a otro.

Esto se realiza a través del elemento anchor (hypertext link), tiene la forma:

```
<a href="URL Página">Ir a la pagina</a>
```

El atributo href indica la URL de la página a la cual se apunta o el recurso.

El texto dentro del elemento se convierte en el link.



Unidad N°5 Agregamos links

El siguiente código (ejemplo9) crea una página que muestra solo un enlace, al hacer click esté redirecciona a la página de Google Argentina.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Links</title>
  </head>
  <body>
    <a href="https://www.google.com.ar">Ir a la Google</a>
  </body>
</html>
```

- Por defecto la página que se especifica en un anchor se va a cargar en la pestaña actual del navegador.

Unidad N°5 Agregamos links

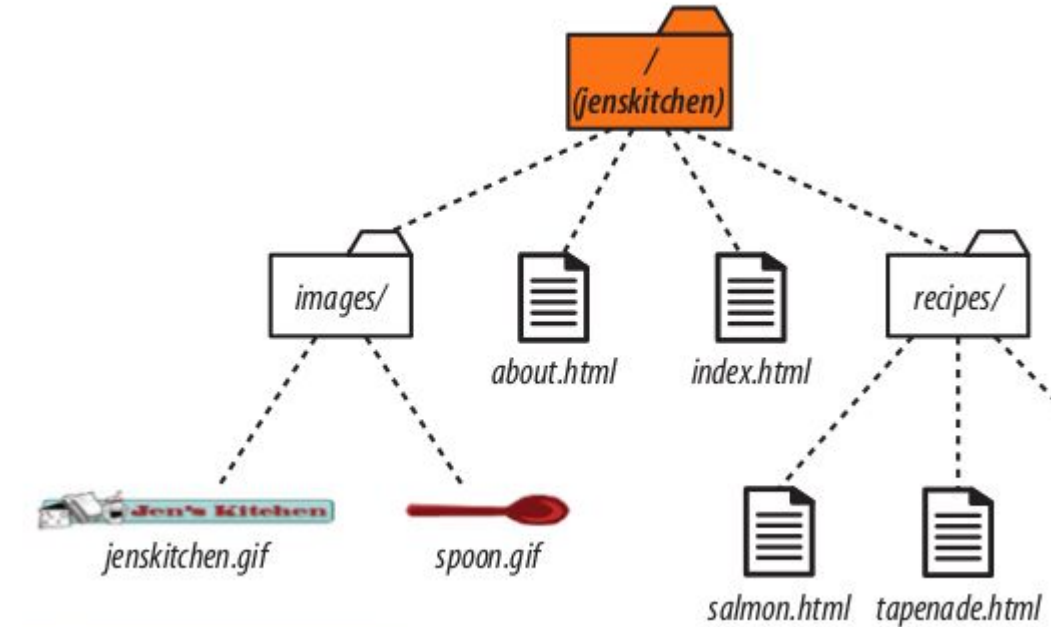
Posibles valores del atributo href

Se pueden especificar dos tipos de URL al atributo href:

- URL absolutas: es la URL completa de una página por ejemplo <http://www.gmail.com>
- URL relativas: esta opción es usada dentro del mismo sitio web por lo general, es más fácil de explicar con un ejemplo visual.



Unidad N°5 Agregamos links



Unidad N°5 Agregamos links

Si desde el archivo index.html se quiere linkear al archivo salmón.html el atributo href tendría la siguiente URL relativa:

```
href="recipes/salmon.html"
```

Es decir que en lugar de especificar la URL entera del sitio solo especificar la dirección del archivo teniendo en cuenta desde que archivo me encuentro.

En este caso el index.html estaba en la misma carpeta que la carpeta recipes, la cual contiene al archivo salmón.html por ende escribiendo recipes y luego /salmon.html indicó que dentro de esa carpeta el archivo apuntado es salmon.html

Unidad N°5 Agregamos links

Ejercicio 5.1 creando links externos:

- Abrir el archivo index.html de la carpeta jenskitchen y hacer que el elemento de la lista con el texto Epicurious apunte a <http://www.epicurious.com> y el elemento The food network que apunte a <http://www.foodnetwork.com>




Unidad N°5 Agregamos links

Linkeando dentro de la misma página

La opción más usada para moverse dentro de una misma página por ejemplo de desde la home a otras páginas de secciones es la opción de la URL relativa.

Con esta opción el navegador busca en servidor donde está nuestra página otra página.

Esto se hace sin especificar el `http://` o el nombre del sitio, se usa el pathname, la notación usada para apuntar a un archivo en particular o directorio donde se encuentra el archivo.



Unidad N°5 Agregamos links

Linkeando dentro de la misma páginas entre directorios

En esta opción la URL relativa apunta a otro archivo que está en la misma carpeta que el archivo desde el cual linkeamos.

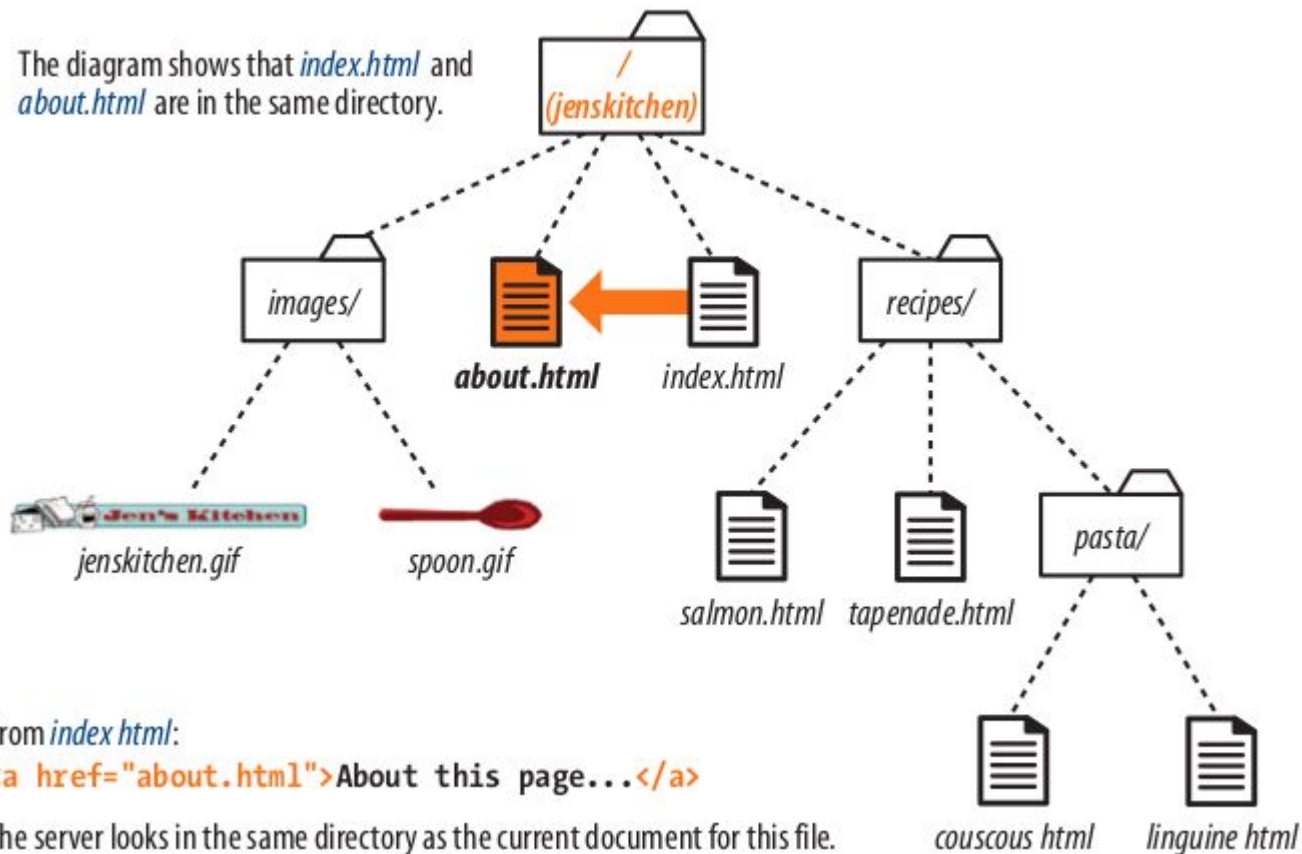
En este caso solo se necesita proveer el nombre del archivo al atributo href. Esto indica que se debe buscar en la misma carpeta del archivo actual.

Veamos un ejemplo:



Unidad N°5 Agregamos links

The diagram shows that *index.html* and *about.html* are in the same directory.



Unidad N°5 Agregamos links

Si desde el archivo index.html queremos apuntar al archivo about.html solo debemos escribir:

```
<a href="about.html">About</a>
```



Unidad N°5 Agregamos links

Ejercicio 5.2 creando links internos:

- Abrir el archivo about.html de la carpeta jenskitchen y hacer que el párrafo Back to the home page apunte al index.html

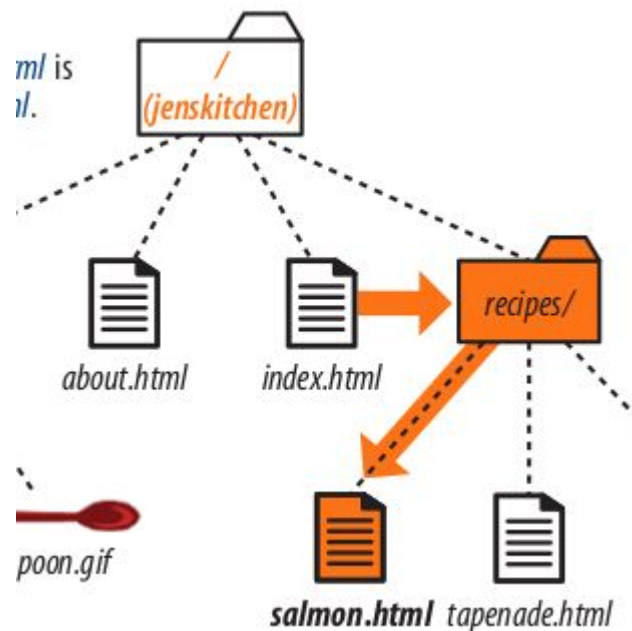


Unidad N°5 Agregamos links

Linkear a Subdirectorios

Supongamos que desde el archivo index.html quiero apuntar al archivo salmon.html en el directorio recipes

En este caso los archivos no estan en el mismo directorio, tenemos que dar el pathname como URL al href.

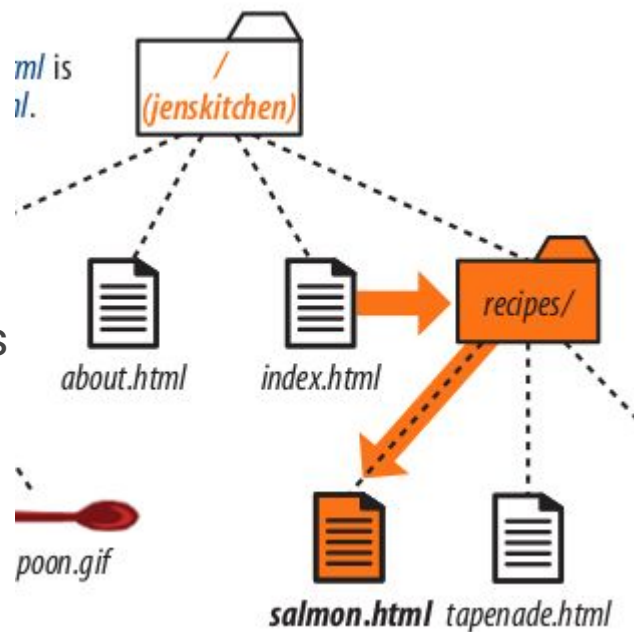


Unidad N°5 Agregamos links

En este caso el archivo salmon.html esta en un subdirectorio llamado recipes. El link desde el archivo index.html tiene que indicar lo siguiente:

Busca en mi directorio un subdirectorio llamado recipes y dentro de este, un archivo llamado salmon.html

`href="recipes/salmon.html"`



Unidad N°5 Agregamos links

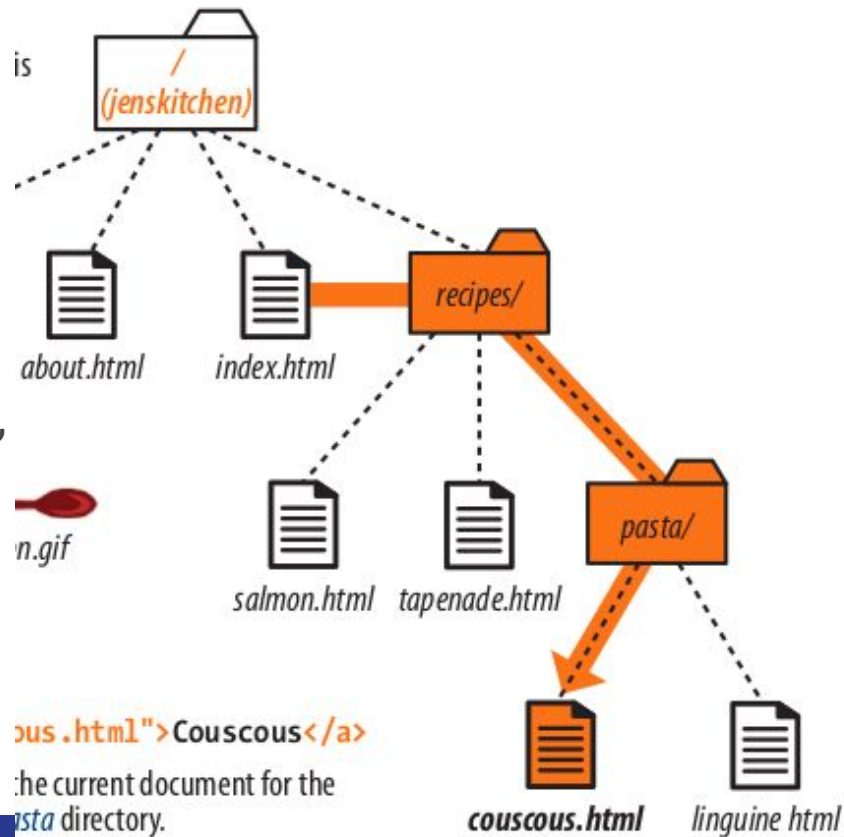
Ejercicio 5.3 linkeando subdirectorios:

Hacer que el li con texto garlic salmon apunte al archivo salon.html y que el li Tapenade apunte a tapenade.html, ambos archivos objetivos están en la carpeta recipes, la cual esta en el mismo directorio que el index.



Unidad N°5 Agregamos links

Supongamos ahora que queremos ir a un archivo (couscous.html) dos directorios más abajo, similar al ejemplo anterior solo se agrega un directorio más en el pathname, en este caso el href es igual a href="recipes/pasta/couscous.html"



Unidad N°5 Agregamos links

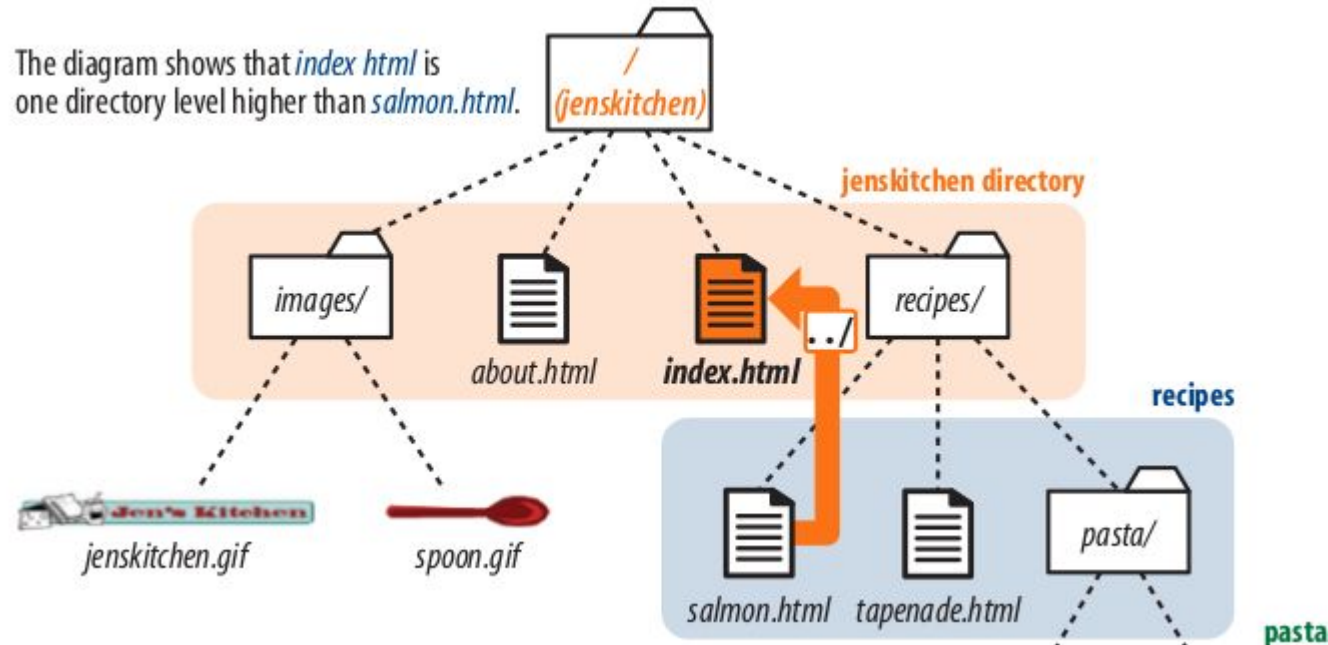
Ejercicio 5.4 linkeando a más subdirectorios:

Hacer que en el archivo index.html el `Couscous with Peas and Mint` apunte a el archivo `couscous.html` y que el `Linguine with Clam Sauce` apunte al archivo `linguine.html`. Ambos archivos destino están en la carpeta pasta la cual esta dentro de la carpeta recipes.



Unidad N°5 Agregamos links

Linkeando a directorios superiores



Unidad N°5 Agregamos links

Supongamos que desde el archivo salmon.html ubicado en la carpeta recipes queremos volver al index.html, es decir, volver a un directorio más arriba.

Para indicar que queremos subir solo un nivel de directorio se usa “../”, esto le indica al navegador que vuelva a la carpeta o directorio anterior, es decir, al directorio que contiene al archivo desde el cual se indica el ../

En este caso desde salmon.html tenemos que escribir un **href=”../index.html”**

Esto indica anda a la carpeta superior (/) y busca el
archivo index.html



Unidad N°5 Agregamos links

Ejercicio 5.5 linkear directorios superiores

En el archivo salmon.html y tapenade.html ubicados en la carpeta recipes agregar un link en cada archivo para volver al archivo index.html,

El link tiene que ser agregado en `<p>[Back to the home page]</p>` en ambos archivos.



Unidad N°5 Agregamos links

Linkeando directorios superiores de más de un nivel

Al igual que con subdirectorios se puede especificar directorios superiores de más de un nivel, en este caso, se necesita usar más de un ../

Por ejemplo, teniendo en cuenta la imagen de la próxima filmina como seria el href para volver desde `couscous.html` el cual esta dentro de `recipes` en la carpeta pasta al archivo `index.html`

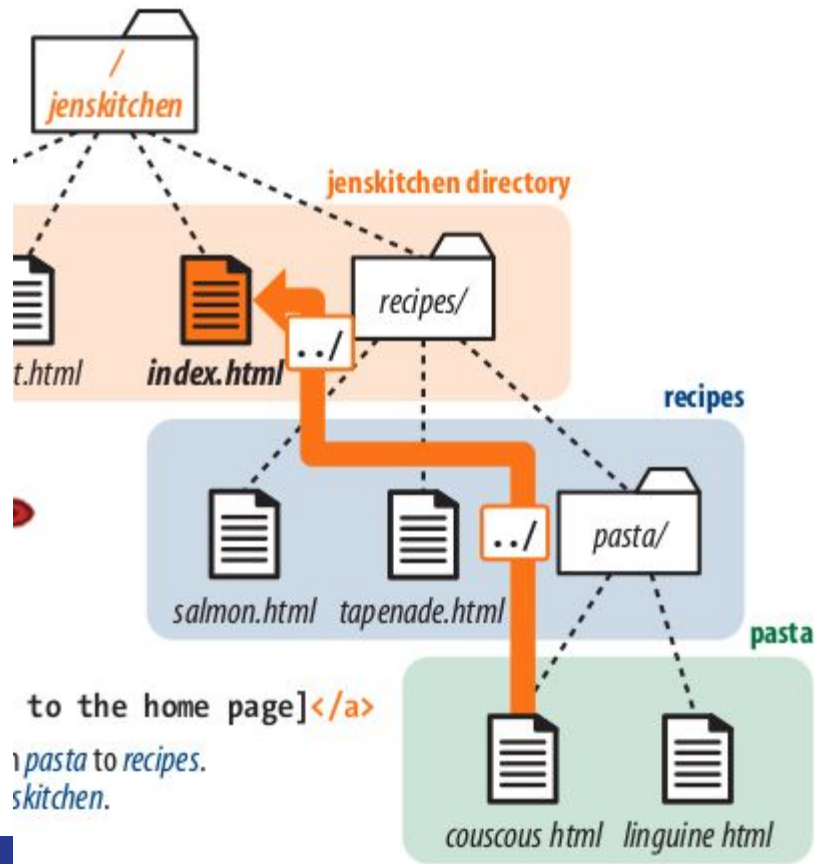


Unidad N°5 Agregamos links

La solución es:

`href="../../index.html"`

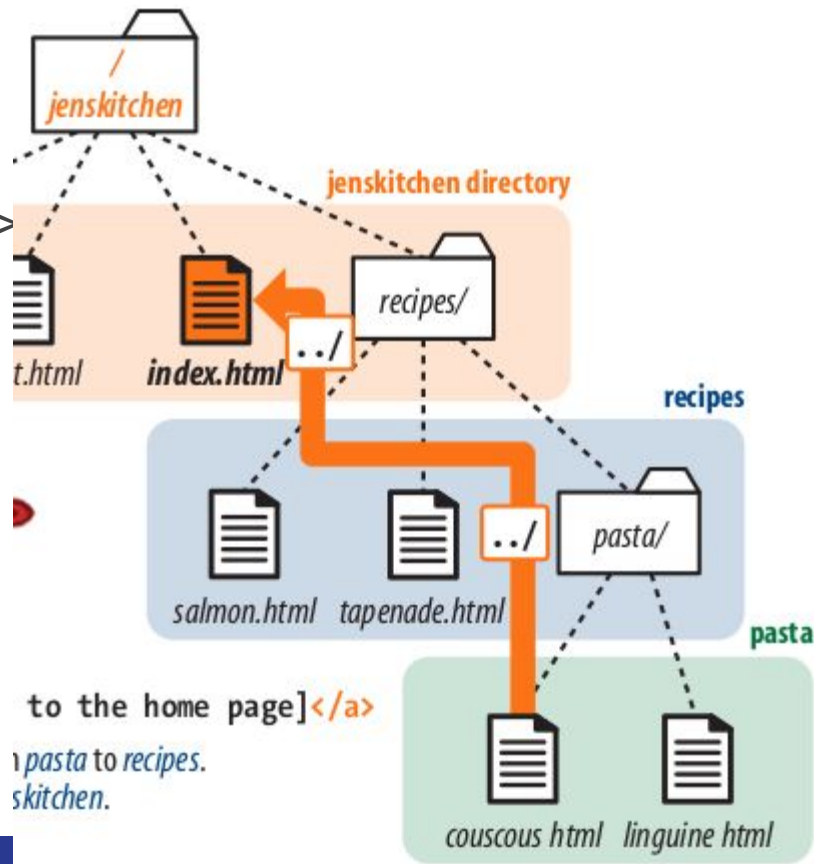
El primer ../ indica que desde couscous.html se vuelva al directorio recipes, el segundo ../ le indica que suba un directorio más desde recipes al directorio /, y en ese directorio busque el archivo index.html



Unidad N°5 Agregamos links

Ejercicio 6.6:

Agregar en los `<p>[Back to the home page]</p>`
de los archivos `cousous.html` y `linguine.html`
un link para volver al archivo `index.html`



Unidad N°5 Agregamos links

Site root y URL relativas

Todo sitio web tiene un directorio **root**, este es el directorio que contiene todos los archivos y subdirectorios de la página.

Otra forma de escribir las URL relativas dentro de un mismo sitio como vimos anteriormente es empezando desde este **directorio root** y luego listando el pathname de subdirectorios y archivos.

Este directorio en las URL relativas se representa con /

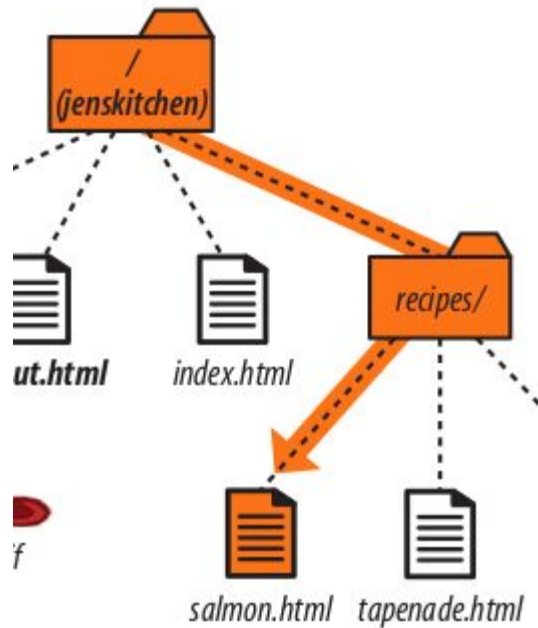


Unidad N°5 Agregamos links

Por ejemplo, si queremos agregar un link desde el index al archivo salmon.html usando el root, el href queda:

`Garlic Salmon`

Esto indica que partiendo desde el directorio / busque el directorio recipes y dentro de este el archivo salmon.html



Unidad N°5 Agregamos links

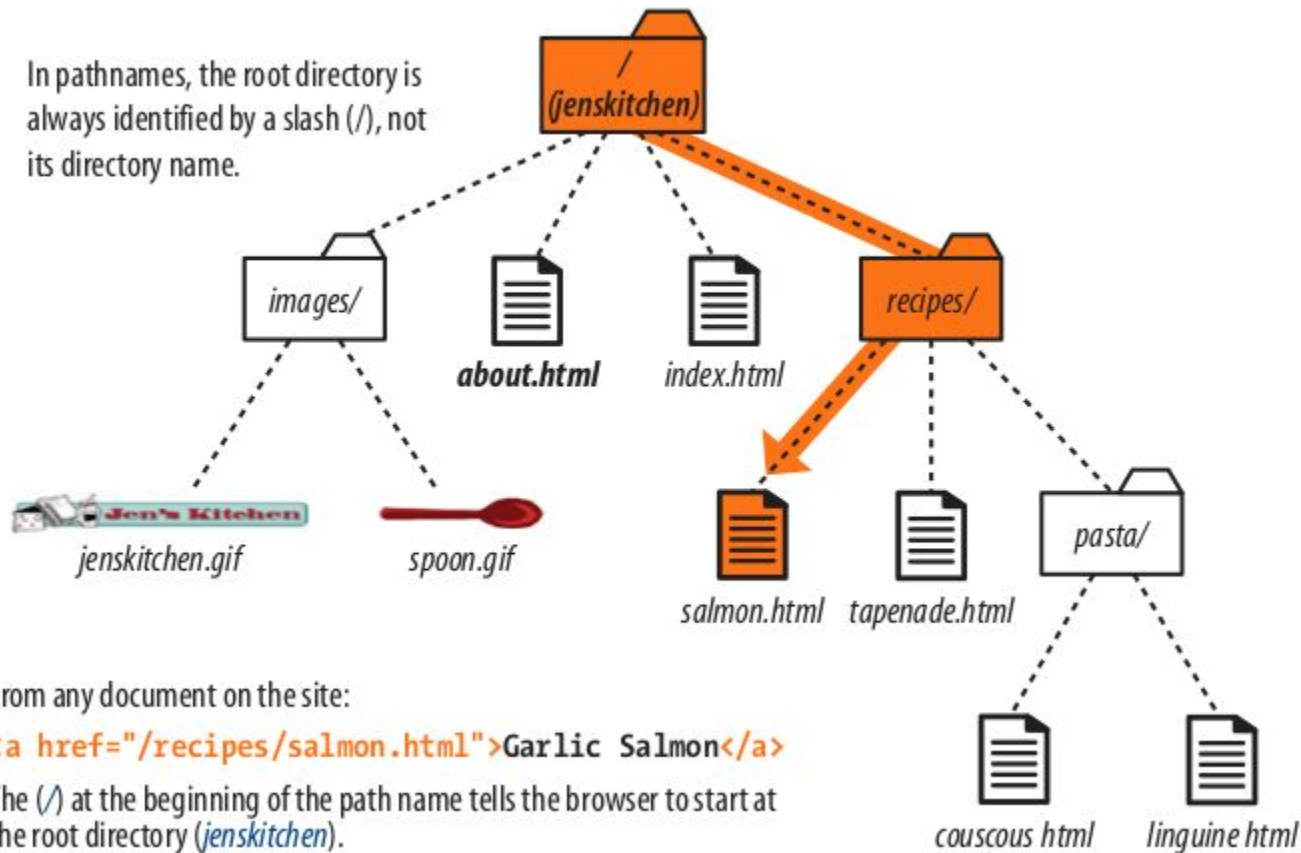
Atributo href y el atributo src de las imágenes

Estos atributos funcionan de la misma manera, es decir que para especificar la ubicación de una imagen dentro del atributo `src="URL imagen"` en la etiqueta ``, todo lo visto anteriormente es válido y de hecho es así como funciona.

Veamos algunos ejemplos considerando la siguiente imagen:



In pathnames, the root directory is always identified by a slash (/), not its directory name.



From any document on the site:

`Garlic Salmon`

The (/) at the beginning of the path name tells the browser to start at the root directory (*jenskitchen*).

Unidad N°5 Agregamos links

Para insertar la imagen jenskitchen.gif en el index necesitamos bajar un nivel de directorio al directorio images:

```

```

Si desde couscous.html queremos agregar la imagen spoon.gif debemos subir 2 niveles de directorios y despues bajar uno para acceder al archivo:

```

```



Unidad N°5 Agregamos links

Ejercicio 5.7 TAREA: agregar los siguientes links:

- De salmon.html a tapenade.html con el nombre Go to tapenade recipe
- De couscous.html a salmon.html con el nombre Try this with Garlic salmon
- De tapenade.html a linguine.html con Try th linguine with clam sauce
- De linguine.html a about.html con about Jen's kitchen
- De tapenade.html a www.allrecipes.com con Go to allrecipe.com



Unidad N°5 Agregamos links

Linkeando a un punto específico de la página

- Permite agregar atajos para scrollear la página.
- Por ejemplo ir desde el inicio a una sección específica o encontrándose al final de la página poder volver al tope.
- Linkear a un punto específico de la página se suele referir como linkear a un fragmento de la página o el documento de la página.



Unidad N°5 Agregamos links

Linkeando a un punto específico de la página

Tiene dos pasos:

- Identificar el punto de destino.
- Linkear al destino



Unidad N°5 Agregamos links

Linkeando a un punto específico de la página

Identificar el punto de destino:

Para crear e identificar un destino “linkeable” se usa el atributo ID para darle al punto o destino objetivo un nombre único con el cual lo identificamos en la página.

Este es el identificador de nuestro fragmento.



Unidad N°5 Agregamos links

Por ejemplo supongamos que tenemos un diccionario y queremos linkear al inicio de cada letra, el cual es un titular con el carácter de la letra, en el caso de la letra B le daremos un ID llamado startB a su heading.

```
<h2 id="startB">B</h2>
```



Unidad N°5 Agregamos links

Linkeando a un punto específico de la página

Linkear al destino:

Como vimos anteriormente para añadir un link se usa el tag anchor `<a>`, en este caso para linkear a un fragmento de la página lo que se hace es especificar en el atributo href el ID del fragmento al cual apuntamos y le agregamos como prefijo al ID el símbolo #, queda de la forma:

```
<a href="#IdDelFragmento">Texto</a>
```



Unidad N°5 Agregamos links

Continuando con el ejemplo del diccionario, para agregar el link al fragmento identificado por el ID startB se necesita el siguiente elemento:

```
<a href="#startB">B</a>
```


Unidad N°5 Agregamos links

Ejercicio 5.8:

Continuando con el ejemplo del diccionario, vamos a crear nuestro propio diccionario, en la carpeta ejercicio 5 >ejercicio 5.8 se encuentra el archivo glossary.html, sede hacer lo siguiente:

- Agregar un ID a cada heading nivel 2 que representa el inicio de una nueva letra.
- Agregar un link en cada letra ubicada en el párrafo inicial para que apunte a cada heading al que se le dio el ID.
- Agregar heading nivel 1 el id de nombre top y al final del documento agregar un link que apunte a dicho id con el texto “TOP”

Unidad N°5 Agregamos links


Linkear a un fragmento en otro documento

Es posible linkear a un fragmento en otro documento, para hacer esto a lo visto anteriormente además de anteponer el # al id se le debe anteponer la URL, ya sea absoluta o relativa:

```
<a href="URL#ID"></a>
```

Por ejemplo: quiero apuntar al fragmento con el id="main " que se encuentra en la página de la Universidad nacional de Villa María,

```
<a href="http://www.unvm.edu.ar/#main"></a>
```



Unidad N°5 Agregamos links

Apuntando a una nueva pestaña del navegador

- Cuando agregamos un link al hacer click en este el recurso al que apunta ese link se carga en la pestaña actual.
- Es posible hacer que un link cargue el recurso al que apunta en una nueva pestaña
- Para esto se usa el atributo **target** del tag `<a>`, en el cual indicamos el nombre de la nueva pestaña:

```
<a href="www.google.com.ar" target="auxiliar"></a>
```

Unidad N°5 Agregamos links

```
<a href="www.google.com.ar" target="auxiliar"></a>
```

- Si repetimos el nombre en este caso, auxiliar, en todos los links de la página para el atributo target, en lugar de abrir una nueva pestaña cada vez que hacemos click en un link, solo se va a abrir una pestaña y en dicha pestaña se irá cargando los recursos a medida que se hace click en distintos links.
- Si queremos que cada link siempre abra una nueva pestaña entonces se setea el atributo **target** con el valor **_blank**

```
<a href="www.google.com.ar" target="_blank"></a>
```

Unidad N° 6 Agregando imagenes

Aunque anteriormente usamos imagenes quedaron algunas cosas por ver en detalle.

En cuanto a su uso las imágenes poseen dos:

- Como elemento de línea a través de html mediante el tag (lo vimos anteriormente).
- Imagen de fondo agregada a través de CSS (próximas unidades).



Unidad N° 6 Agregando imagenes

Formatos de las imágenes válidas para páginas web:

- JPEG
- PNG
- GIF

Dichas imágenes deben ir acompañadas por su prefijo en el nombre del archivo ya sea **.jpg/.jpeg .gif .png** para poder ser reconocidas por el navegador.



Unidad N° 6 Agregando imagenes

Si se tiene imagenes en otro formato hay dos opciones:

- Convertirla a los formatos válidos
- Agregarla como un link, el navegador usará una herramienta externa para visualizarla pero no como elemento en línea dentro de la página sino que en una nueva pestaña.



Unidad N° 6 Agregando imagenes

El tag :

```

```

- Indica al navegador que debe colocar una imagen en línea.
- La imagen se puede colocar en cualquier parte del flujo del texto, ya que es un elemento en línea.

```
<p>I had been wanting to go to Tuscany   
for a long time, and I was not disappointed.</p>
```

I had been wanting to go to Tuscany
and I was not disappointed.



Unidad N° 6 Agregando imagenes

El tag :

- Cuando el navegador ve un elemento hace la petición al servidor para recuperar el archivo de la imagen antes de mostrarla en la página, si no la recupera el texto alternativo es mostrado.
- Los atributos **src** y **alt** son requeridos.
- **src** le indica al navegador la url con la ubicación del archivo de la imagen.
- **alt** provee un texto alternativo que se muestra si el navegador no puede recuperar el archivo de la imagen.



Unidad N° 6 Agregando imagenes

- Es un elemento en línea, por ende se comporta como cualquier otro elemento en línea.
- Si entran otros elementos a su lado se permite y si el navegador cambia de tamaño entonces, estos son achicados o agrandados para llenar el nuevo ancho.



Unidad N° 6 Agregando imagenes

- Por defecto el borde inferior de la imagen se alinea con el texto.



- Esto se puede cambiar con css usando la propiedad float, para colocar la imagen en el margen izquierdo o derecho y que el texto la envuelva.

Unidad N° 6 Agregando imagenes

```
img {  
  float: left;  
  padding: 5px 5px 5px 5px;  
}  
  
<p>  
  <a href="#">  
    <img src="" alt="">  
  </a>  
  texto rodeante  
</p>
```

Pozzarello



The house we stayed in was called Pozzarello and it was built around a cool inside, despite the blistering mid-day heat. This is the view from the window. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Unidad N° 6 Agregando imagenes

Valores de los atributos src y alt:

- **src** significa fuente de la imagen, es la URL de la imagen y al igual que la teoría de links, todas las formas vistas para especificar url son válidas.
- Se suelen organizar las imágenes en una carpeta imágenes dentro del directorio raíz de la página.
- **alt** significa texto alternativo, es requerido que esté presente pero no es necesario que tenga un texto, se puede dejar texto vacío. Ejemplo siguiente

```

```

Unidad N° 6 Agregando imagenes

Que muestran los navegadores si la imagen no es recuperada:

```
<p>If you're  and you know it clap your hands.</p>
```

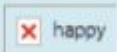


If you're and you know it clap your hands.


With image displayed

If you're happy and you know it clap your hands.


Firefox (Windows and Mac)

If you're  and you know it clap your hands.

Internet Explorer (Windows)

If you're  and you know it clap your hands.

Chrome (Mac & Windows)

If you're  and you know it clap your hands.

Safari (Mac)

Unidad N° 6 Agregando imagenes

Dimensiones de una imagen

- El tag cuenta con dos atributos para especificar las dimensiones de la imagen:
 - **width (ancho).**
 - **height (alto).**
- Ambos atributos indican las dimensiones en **número de píxeles.**
- Si estos atributos no son especificados el navegador muestra la imagen en el tamaño real.
- Estos atributos también pueden modificarse por css.



Unidad N° 6 Agregando imagenes

Dimensiones de una imagen - ver ejemplo 10

```
<div>  
  <h1>tamaño reducido 960x600</h1>  
    
</div>
```


Unidad N° 6 Agregando imagenes

Dimensiones de una imagen

- ver ejemplo 10

tamaño por defecto 1920x1200



tamaño reducido 960x600



Unidad N° 6 Agregando imagenes

- Si las dimensiones en píxeles difieren de las originales, el navegador redimensiona a los pixeles especificados, por ende se puede achicar o agrandar la imagen.
- Esto es peligroso ya que la imagen puede perder calidad o volverse borrosa
- Siempre es ideal usar una herramienta para redimensionar imágenes.
- Ejemplo 11 (problemas de redimensión)



Unidad N° 6 Agregando imagenes

tamaño reducido 960x600



tamaño reducido 400x600 (con problemas)



Unidad N° 6 Agregando imagenes

Ejercicio 6:

1. Agregar al principio del párrafo The tuscan countryside la imagen countryside_thumb ubicada en la carpeta thumbnails con width 100 y height 75.
2. Agregar al principio del párrafo sienna la imagen sienna_thumb ubicada en la carpeta thumbnails con width 75 height 100.
3. En la etiqueta style dentro de la etiqueta head agregar el código css ubicado en el archivo style.css



Unidad N° 6 Agregando imagenes

4. Agregar un link a las imagenes recién agregadas para la primera apunte al archivo countryside.html y la segunda a sienna.html
5. En countryside.html agregar una imagen en el párrafo vacío que está debajo del h1, la imagen está en la carpeta photos con el nombre countryside.jpg, con las dimensiones width 500 y height 375
6. Hacer lo mismo del paso 5 para el archivo sienna.html, pero la imagen es sienna.jpg dentro de la carpeta photos y tiene las dimensiones width 375 y height 500.



Unidad N° 6 Agregando imagenes

7. En el index.html agregar 3 párrafos mas, uno con el titulo Catedral, otro lavender y el último con el título Sweets. Pueden encontrar el texto en texto-adicional.txt
8. Al igual que para los primeros parrafos agregar sus imagenes thumbs, para catedral y sweets tienen width 75 y height 100; para lavender width 100 y height 100.
9. Crear tres archivos con nombres catedral.html, lavender.html y sweets.html, estos deben tener el formato como window.html.
10. Agregar links a las imágenes agregadas en el paso 8 a estos nuevos archivos



Unidad N° 6 Agregando imagenes

11. En cada uno de los archivos html recién creados agregar las imágenes correspondientes de la carpeta photos, para lavender y catedral la imágenes tienen width 500 y height 375; en el caso de sweets tiene width 375 y height 500

