

# HTML 5

---

Profesor: Eduardo Palacio

# Compatibilidad con HTML 4.x

- Html 5 es totalmente compatible con las versiones anteriores hasta el punto de poder cambiar simplemente el doctype directamente por HTML.
  - `<!DOCTYPE html>`
- Además de etiquetas incorpora nuevas funcionalidades por medio de APIs.

# Novedades

- Se implementan etiquetas semánticas que en html4.x eran muy comunes por e los desarrolladores como:
  - `<div id="nav"></div>` por `<nav></nav>`
- Nueva multimedia `<audio>` `<video>`
- Se quita `<object>` como elemento grafico genérico y se usa `<canvas>` que nos permite dibujar en el.
- Se redefinen elementos como `<a>`, `<cite>` y `<menu>`
- Se incorpora la api MathML para formulas matemáticas
- Permite almacenamiento local en el lado del cliente.
- Permite geolocalización

# HTML 5 es una API

- Además de las etiquetas y la geolocalización html 5 define por fin claramente como deben de interactuar las etiquetas con javaScript mediante le modelo DOM.
  - EJ: además de definir etiquetas como <video>, <audio> define su correspondiente api para para estos objetos (detectar el soporte para diferentes formatos, reproducir...)
  - Es decir se pueden controlar mediante javascript

# Semántica

- HTML 5 incorpora etiquetas semánticas.
  - <https://www.w3schools.com/tags/default.asp>
- Podemos partirlas en dos grupos:
  - Las que tenemos pero se mejora si comportamiento mediante javascript como <video>, <audio> , <canvas>
  - Etiquetas semánticas para estructurar el sitio Web
    - en lugar de usar <div id="..."> <nav>, <header>, <footer>, <aside>...

# La cabecera HTML 5

- El DOCTYPE define concretamente, en el documento que estoy creando, las etiquetas que se pueden utilizar, los atributos de cada etiqueta y el tipo de valores que puede tener cada atributo.
- Pasamos de las 15 posibilidades de antes a una:
  - `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>`
  - `<!DOCTYPE html>`

# Elemento raiz

- La definición típica es:
  - `<html xmlns=http://www.w3.org/1999/xhtml lang="en" xml:lang="en">`
- En html 5 no es necesario definirla porque solo tiene un espacio de nombres y esta en ingles
  - `<html lang="en">` (el valor del atributo lang puede ser “es” u otro idioma)

# Head

- Es el primer hijo del raíz (HTML)
- El elemento head contiene los metadatos que aportan información extra sobre la página, como su título, descripción, autor, etc. Además, puede incluir referencias externas a contenidos necesarios para que el documento se muestre y comporte de manera correcta (como hojas de estilos o *scripts*). *Este elemento ha sufrido pequeñas variaciones, pero que debemos tener en cuenta:*



# Ejemplo HEAD

- `<head>`
  - `<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />`
  - `<title>Mi SitioWeb</title>`
  - `<link rel="stylesheet" type="text/css" href="miestilo.css" />`
  - `<link rel="alternate" type="application/atom+xml" title="Mi SitioWeb" href="/misitio/" />`
  - `<link rel="search" type="application/opensearchdescription+xml" title="Mi SitioWeb" href="opensearch.xml" />`
  - `<link rel="favoriteicon" href="/images/favicon.ico" />`
- `</head>`

# HEAD: Meta

- Meta:
  - Importante establecer la configuración de caracteres se hace de forma reducida:
  - `<meta charset="utf-8">`

# HEAD: Link

- Link: son una manera de acceder o declarar contenido externo al documento actual, que puede cumplir distintos objetivos:
  - Es una hoja de estilo contiene las reglas CSS que su navegador debe aplicar al presente documento.
  - Es un feed que contiene el mismo contenido que esta página, pero en un formato estándar (RSS).
  - Es una traducción de esta página en otro idioma.
  - Es el mismo contenido que esta página, pero en formato PDF.
  - Es el próximo capítulo de un libro en línea de la cual esta página es también una parte.

# HEAD: Link

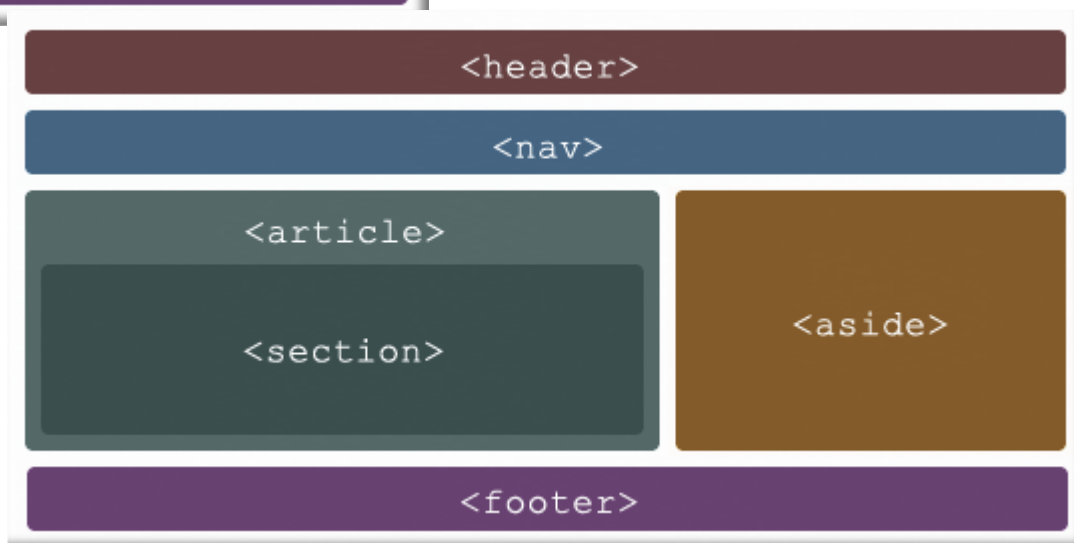
- Link se divide en dos categorías:
  - Recursos que mejoran el documento (css,js...)
  - Los enlaces a otros documentos
- Todos funcionan como antes pero con css cambia un poquito:
  - La forma típica de llamar a una css es
    - `<link rel="stylesheet" href="miestilo.css" type="text/css" />`
      - Ya solo hay transformación de tipo css por lo que podemos eliminar el final.
    - `<link rel="stylesheet" href="miestilo.css">`

# Etiquetas semánticas



← HTML 4.x

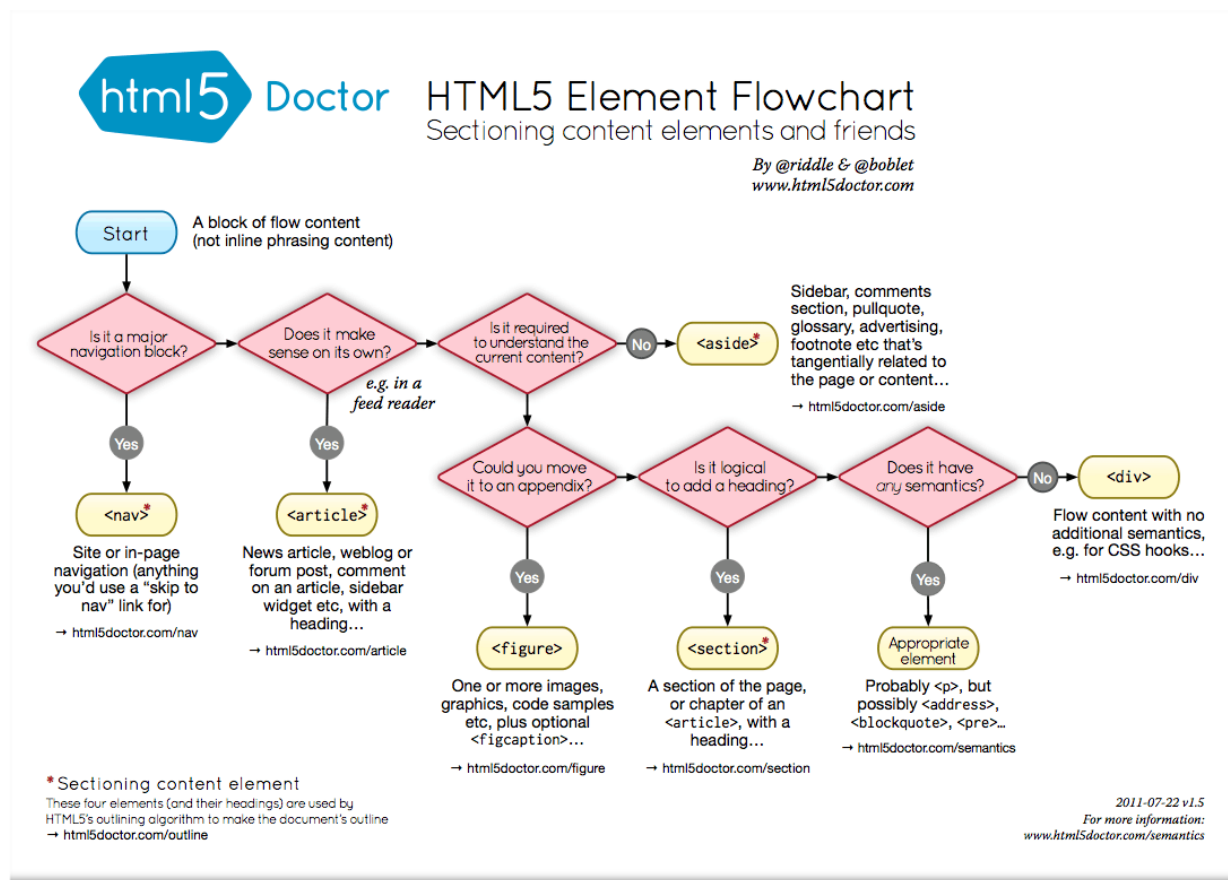
HTML 5 →



# Etiquetas semánticas de estructura de página.

- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>

# Algoritmo para seleccionar las etiquetas correctamente



# Novedades de las etiquetas

- <header>
  - <a href="/"><img src=logo.png alt="home"></a>
  - <hgroup>
    - <h1>Title</h1>
    - <h2 class="tagline">subtitulo.</h2>
  - </hgroup>
- </header>



# Novedades de las etiquetas

- `<nav>`
  - `<ul>`
    - `<li><a href="#">home</a></li>`
    - `<li><a href="#">blog</a></li>`
    - `<li><a href="#">gallery</a></li>`
    - `<li><a href="#">about</a></li>`
  - `</ul>`
- `</nav>`

# Novedades de las etiquetas

- Ver las etiquetas:
  - <article>
  - <section>
  - <aside>
  - <Figure>

# Elementos de formulario

- HTML 5 añade dos nuevos métodos update y delete
- También añade 12 nuevos tipos de campos input.
- Algunos de los nuevos campos input ya incluyen la validación por lo que no es necesario usar JavaScript.

# Tipos input nuevos

- `<input type="email">` se valida y el atributo “multiple” permite escribir una lista de direcciones, puede ser vacío o no (required)
- `<input type="url">`
- `<input type="date">` los datos enviados cumplen con la norma ISO [http://es.wikipedia.org/wiki/ISO\\_8601](http://es.wikipedia.org/wiki/ISO_8601)
- `<input type="time">`
- `<input type="datetime-local">`
- `<input type="month">`
- `<input type="week">`

# Tipos input nuevos

- `<input type="number">` podemos usar los atributos min, max y step
- `<input type="range">` podemos usar los atributos min, max y step
- `<input type="tel">` NO valida, pero adapta el teclado de los móviles.
- `<input type="search">`
- `<input type="color">`

# Nuevos atributos de campos input

- autocomplete, min, max, multiple, pattern, autofocus, placeholder, required y step
- Además de estos tenemos list que referencia a otro elemento ambos deben de tener el mismo id.
- Ej:

```
<input id="titulo" type="text" list="mylista"/>
```

```
<datalist id="mylista">
```

```
  <option label="Sr" value="Sr">
```

```
  <option label="Sra" value="Sra">
```

```
</datalist>
```

# Nuevos atributos de campos input

- Placeholder es la pequeña leyenda que aparece en los campos de texto antes de escribir.
- Required indica que debe de ser rellenado a ntes de enviar.
- Pattern permite poner una expresión regular que debe de cumplir el valor del campo
  - `<input pattern="[0-9][A-Z]{3}" name="part" title="Es un numero seguido de tres mayúsculas.">`
- Multiple permite que el input se rellene con varios valores separados por comas
  - `<input type="file" multiple="multiple">`

# Nuevos elementos de formulario

- `<progress>` mostrar avance o progreso en la ejecución de una tarea (típicamente relleno de formulario). Podemos usar los atributos `value`, `max` y `position`
  - `<progress value="5" max="20">5</progress>`
- `<meter>` muy parecido a `progress` pero para escalas conocidas (longitud, masa, peso, uso de disco). Podemos usar los atributos `value`, `low`, `max`, `optimum` entre otros.
  - Ej:
    - `<p>Tu puntuacion es:`
      - `<meter value="91" min="0" max="100" low="40" high="90" optimum="100">A+</meter>`
    - `</p>`



# Curiosidad con formularios

- Si se realiza un formulario pero por alguna razón de diseño no entra algún campo dentro de `<form></form>`
- Se puede usar el `id` del form para indicar que el elemento externo pertenece a este formulario.
  - Ej:
    - `<form id="foo">`
      - `<input type="text">`
      - ...
    - `</form>`
    - ...
    - `<textarea form="foo"></textarea>`
    - En este caso, el elemento `<textarea>` se encuentra fuera del formulario, pero realmente pertenece a él ya que hemos definido el atributo `form` con el identificador del formulario.

- Fin