

This repository | Search

Pull requests Issues Marketplace Gist

jleiva / magki

Watch 1 Star 0 Fork 0

<> Code Issues 0 Pull requests 0 Projects 0 Wiki Insights

Branch: master magki / docs / HTML /

Create new file Upload files Find file History

Jose Leiva V1 HTML Latest commit 623745f 4 days ago

..

readme.md V1 HTML 4 days ago

readme.md

## HTML Coding Standards

A manera general queremos:

- Indentar con 2 espacios, no tabs.
- Elementos anidados deben ser indentados una vez (2 espacios).
- Usar comillas dobles ( " " ), nunca comillas simples, en atributos. `<div class="valor"></div>`
- No incluir "trailing slash" ( / ) en elementos que se autocierran, es opcional según la especificación de HTML5. `<img src="" alt="">`
- No omitir etiquetas de cierre. ejem: `</li> &acute; </body>` .
- Separar atributos por un espacio.
- Siempre usar igual y comillas dobles para valores de atributos. `<img class="valor" alt="valor" src="">`
- Siempre incluir etiquetas `html` , `head` , y `body` .
- Siempre usar minúscula para nombre de etiquetas y atributos.
- Siempre incluir el atributo `alt` para imagenes.
- Separar responsabilidades. Mantener markup (HTML), estilos (CSS) y comportamiento (JS) separado.
- Utilice HTML de acuerdo a su propósito. Utilice elementos para lo que se han creado. Por ejemplo, utilice elementos de encabezado ( `h1` - `h6` ) para encabezados, elementos `p` para párrafos, elementos `a` para anclajes, etc.

## HTML5 doctype

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

## Block-level elements

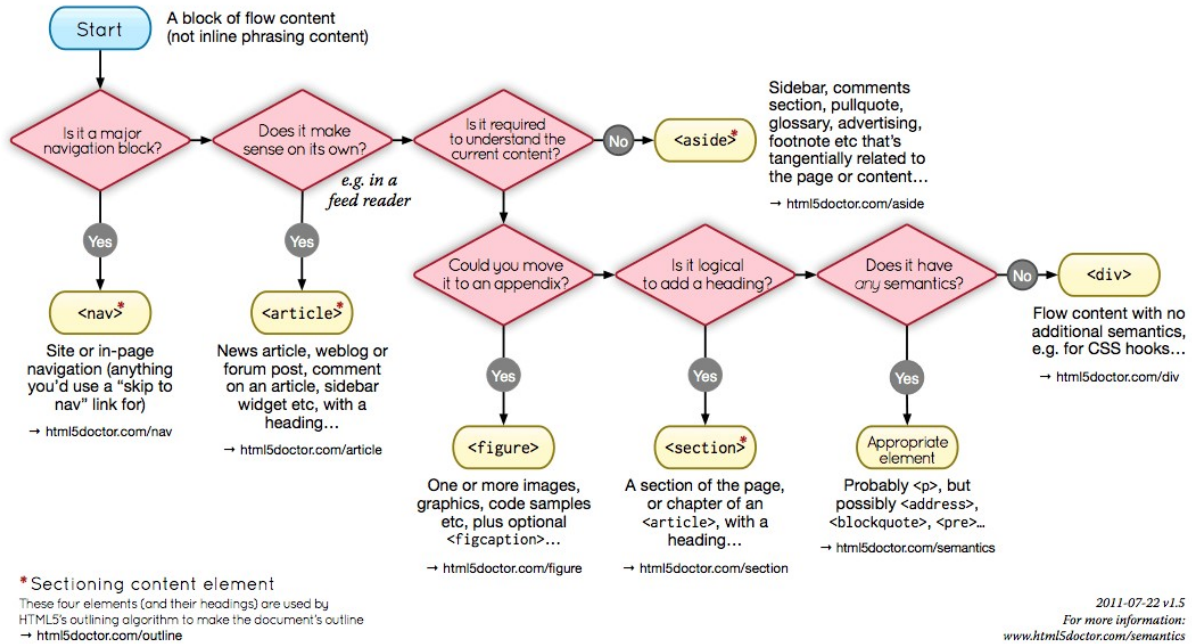
Use elementos apropiados de HTML5 para bloques; en caso de duda, leer sobre el elemento en [html5rocks.com](http://html5rocks.com), [html5doctor](http://html5doctor) ó [MDN](http://mdn) y si no queda claro, usar `div`s



## HTML5 Element Flowchart

Sectioning content elements and friends

By @riddle & @boblet  
www.html5doctor.com



## Atributo de lenguaje

```
<html lang="es">
  <!-- ... -->
</html>
```

## Character encoding

```
<head>
  <meta charset="UTF-8">
</head>
```

## IE compatibility mode

```
<meta http-equiv="X-UA-Compatible" content="IE=Edge">
```

## Inclusion de CSS y JavaScript

Siguiendo la especificación de HTML5, no es obligatorio especificar el `type` cuando se incluye CSS y JavaScript como `text/css` y `text/javascript`, cuando estos valores son los default.

```
<!-- External CSS -->
<link rel="stylesheet" href="code-guide.css">

<!-- In-document CSS -->
```

```
<style>
  /* ... */
</style>

<!-- JavaScript -->
<script src="code-guide.js"></script>
```

## Reducir markup

Siempre que sea posible, evitar escribir elementos "padre" innecesarios.

```
<!-- Not so great -->
<span class="avatar">
  
</span>

<!-- Better -->

```

## Formularios

Siempre incluir un atributo `for` para elementos `label`.

```
<!-- Bad Example -->
<label><input type="radio" name="input" value="first"> First</label>

<!-- Good Example -->
<input type="radio" name="input" id="input-1" value="first">
<label for="input-1">First</label>
```

No usar el atributo `placeholder` para etiquetas; siempre usar un elemento `label`.

```
<!-- Bad Example -->
<input type="text" id="name" placeholder="Enter your name">

<!-- Good Example -->
<label for="name">Name</label>
<input id="name" type="text" placeholder="Jane Doe">
```

## Comentarios

Siempre son una buena idea.

```
<!-- Good single line comment -->

<!--
Good example of a multi-line comment.
If your comment takes up multiple lines, please do this.
-->
```

## Nombre de Clases y Ids

Usar nombre semanticos para Clases y Ids, evitar nombres de "presentación".

```
<!-- Bad Example -->
<button type="submit" class="button-green">Submit</button>
```

```
<!-- Good Example -->  
<button type="submit" class="action-primary">Submit</button>
```

## Fuentes

- [HTML Style Guide](#)
- [Code Guide by @mdo](#)

