

# SEO y HTML 5



# ¿por qué estoy yo aquí?



javiercasares.com



kisslab.com

kisslab.com

javier.casares@kisslab.com

2

# qué falta para el HTML 5

---

El **HTML 5** es la nueva versión, tras más de 10 años, de la forma en la que se construyen las páginas de Internet, y uno de los responsables de esta nueva versión es uno de los trabajadores de **Google**.

Con este nuevo formato podremos conectarnos a dispositivos externos, geolocalizar a los usuarios, cachear información en el navegador, el AJAX se mejorará, los formularios aumentan la capacidad de la Web 2.0...

El único navegador que hoy en día no soporta HTML 5 es Internet Explorer, pero con una línea de JavaScript y un par en los CSS se puede activar todo su potencial.

Los buscadores como Google o Bing ya son capaces de interpretar parcialmente algunos de los elementos más interesantes para SEO del HTML 5.

# las metaetiquetas

**HTML 5** establece los meta-tags que ha de llevar una página, de forma que el resto puede obviarse:

- *application-name*: se usa en el caso de que el contenido no sea una página web, por lo que, en general, no se usará.
- *author*: es el nombre de la persona o empresa creadora del contenido.
- *description*: sólo ha de haber uno por página y es una breve descripción de lo que encontraremos en ella; recomiendo que tenga entre 125 y 175 caracteres, que es lo que aparecerá en los resultados de los buscadores.
- *generator*: es el programa que ha creado la página y que, personalmente, recomiendo no incluir.
- *keywords*: una serie de conceptos separados por comas que indican qué tendría que encontrar un usuario en la página; personalmente recomiendo entre 4 y 6 conceptos.

# las metaetiquetas

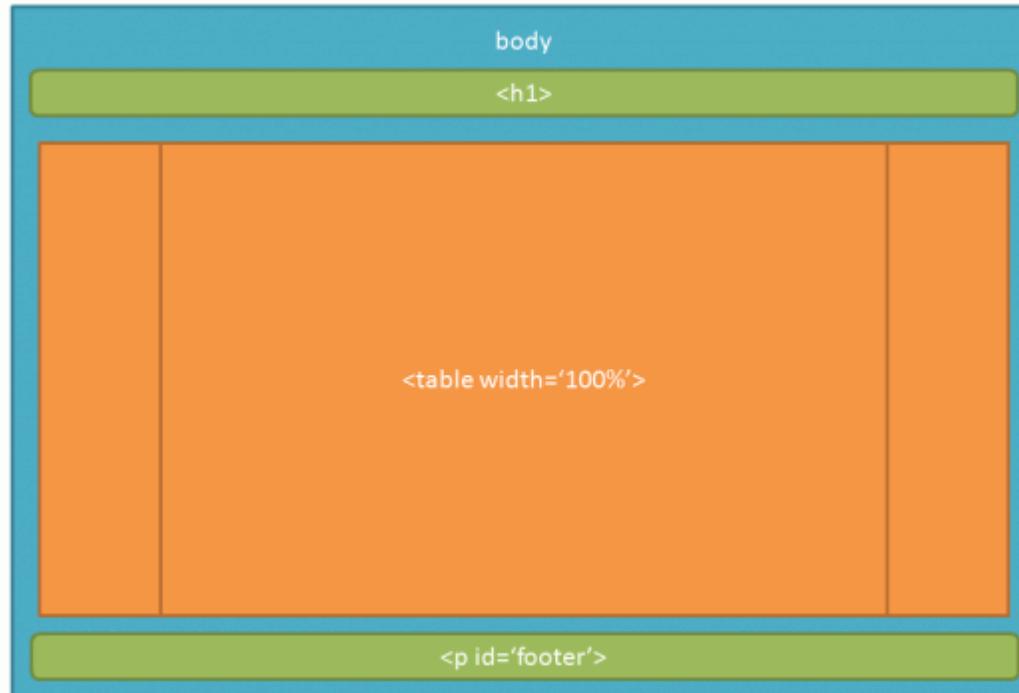
- *robots*: es el que se utiliza para [bloquear a los robots de búsqueda](#).
- *audience*: una lista de palabras que definirían el tipo de personas que sí o no pueden ver los contenidos.

El *http-equiv* que hace referencia a algunas cabeceras:

- *content-language*: define el idioma que tendrá la página en general.
- *content-type*: es el tipo de contenido MIME que tendrá la página; habitualmente *text/html*.
- *default-style*: establece cuál es la hoja de estilos CSS por defecto.
- *refresh*: indica que esa página ha de refrescarse cada N segundos. Si se le indica después una URL se actualizará hacia esa dirección.

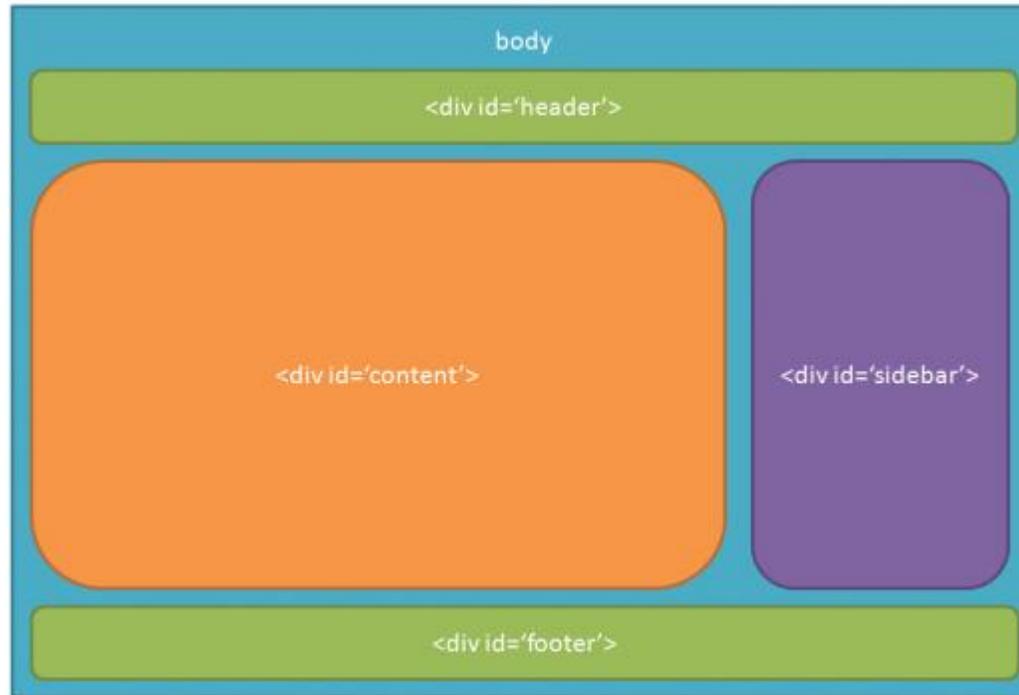
# el formateo de las páginas

## Estructura básica con HTML 3



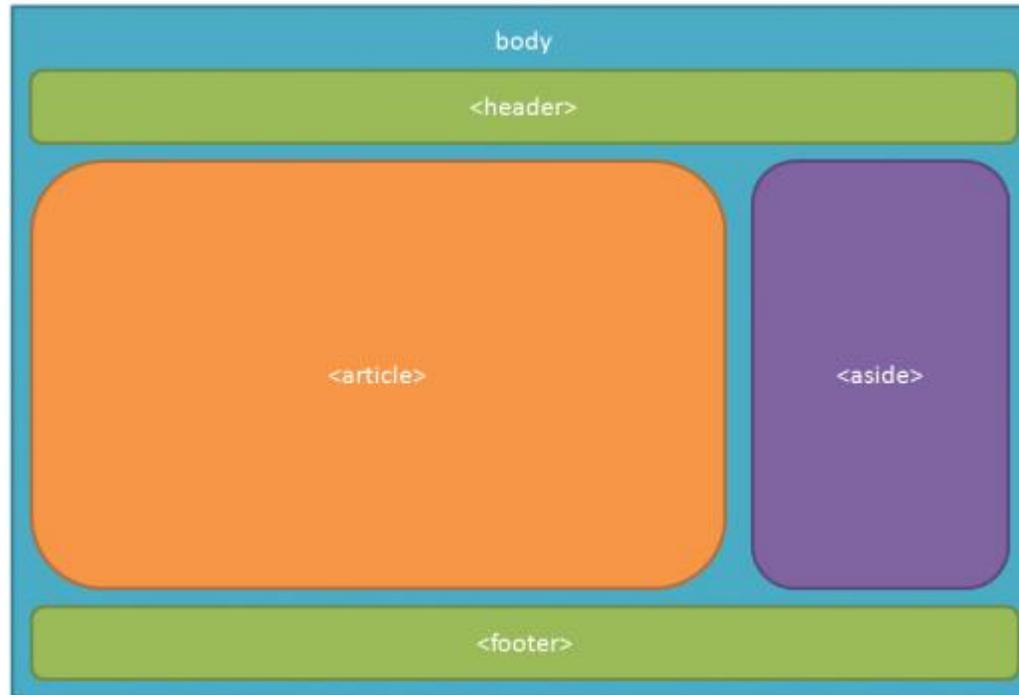
# el formateo de las páginas

Estructura básica con HTML 4



# el formateo de las páginas

Estructura básica con HTML 5



# las nuevas etiquetas contenedoras

---

*<header>*

La inicio de los distintos elementos...

*<footer>*

El final de los distintos elementos

*<article>*

Lo más importante de la página (o sea, lo que hay que «posicionar»).

*<aside>*

Los contenidos relacionados con el contenido principal (article)

# las nuevas etiquetas contenedoras

---

*<section>*

Las diferentes secciones con «identidad propia» donde organizar contenidos.

*<nav>*

El bloque que agrupa los enlaces

*<figure>*

El bloque que agrupa imágenes, elementos multimedia...

# la importancia de los enlaces

- sigue existiendo el «href»
- con el «target» podemos abrir las páginas en nuevas ventanas:
  - \_self
  - \_parent
  - \_top
  - \_blank
- el «media» el sitio en el que se abrirá:
  - screen
  - tty
  - tv
  - projection
  - handheld
  - print
  - braille
  - aural
  - all

# la importancia de los enlaces

- indicamos el idioma de destino con «`hreflang`»
- el «`type`» permite indicar el MIME del contenido de destino
- `rel="alternate"`: elemento alternativo
- `rel="archives"`: página histórica
- `rel="author"`: creador del contenido
- `rel="bookmark"`: URL única del «`article`»
- `rel="external"`: la URL no forma parte del documento
- `rel="help"`: ayuda al contenido actual
- `rel="license"`: la licencia de ese artículo
- `rel="nofollow"`: los contenidos no tienen nada que ver con el destino
- `rel="noreferrer"`: no se ha de enviar el «`referrer`» por el navegador
- `rel="search"`: se enlaza a una página de resultados de búsqueda
- `rel="sidebar"`: contenidos secundarios relacionados con el principal
- `rel="tag"`: enlace al «`tag`» del contenido (no usar en nubes de tags)
- `rel="index"`, `rel="up"`, `rel="first"`, `rel="last"`, `rel="next"`, `rel="prev"`...

# formularios para la web 2.0

Los `<input>` evolucionan para permitir más contenidos:

- *hidden*: Es un contenido oculto que suele corresponder a una serie de caracteres alfanuméricos. Este valor no puede ser manipulado por el usuario.
- *text*: Campo de texto que no permite saltos de línea. Si se añade el atributo *value* será el contenido original del campo.
- *search*: Campo de texto que no permite saltos de línea y que indica que será usado como contenido de una búsqueda. Si se añade el atributo *value* será el contenido original del campo.
- *tel*: Campo de texto que indica un número de teléfono. No tiene ningún formato en particular, por lo que se pueden incluir espacios o guiones, por ejemplo.
- *url*: Campo de texto en el que se pueden incluir direcciones URL absolutas. Se puede hacer una especie de *autosuggest* junto a otros elementos (ver ejemplo más abajo).
- *email*: Campo de texto que permite incluir correo electrónico. Principalmente si se incluye el atributo *value* permite una única dirección, y si es el *multiple* puede incluir varias.

# formularios para la web 2.0

- *password*: Campo de texto, pero que aparece oculto para el usuario y así proteger la información.
- *datetime*: Permite una fecha y hora válida en un formato como *0037-12-13T00:00Z*, *1979-10-14T12:00:00.001-04:00* o *8592-01-01T02:09+02:09*. Si se indica un número, serán los milisegundos transcurridos desde el 01/01/1970.
- *date*: Permite indicar una fecha. El atributo *value* indica la fecha en cuestión y el *min* y *max* los límites del rango de fecha. El atributo *step* marca los milisegundos (por defecto 1 día) entre un valor y otro. El formato sería *2010-03-09*.
- *month*: Permite indicar un mes. Tiene los mismos atributos que *date*. El formato sería *2010-03*.
- *week*: Permite indicar la semana del año. Tiene los mismos atributos que *date*. El formato sería *2010-W03* (tercera semana).
- *time*: Permite indicar una hora. Tiene los mismos atributos que *date*. El formato sería *17:23* y opcionalmente los segundos, *17:23:59*.
- *datetime-local*: Es igual que el *datetime* pero no permite indicar ningún tipo de desfase horario.

# formularios para la web 2.0

- *number*: Representa un valor numérico. Este puede ser negativo (si empieza con un guión -), seguido de un número y con decimales (si se le indica un punto .) y para acabar, un exponente, empezando con una letra e, seguida de un símbolo positivo + o negativo – y la cantidad exponencial. Un ejemplo completo sería: *-12345.678e+90*.
- *range*: El tipo *range* permite un *slide* en donde el número (valor) no es lo 100% importante, sino que es el formato en el que se muestra. Un poco más abajo, un ejemplo.
- *color*: Este elemento representa un color. El formato es del estilo *#A0F59E* (o sea, el RGB).
- *checkbox*: Este elemento ya existía, y es el que permite elegir unas cuantas opciones de un grupo de ellas.
- *radio*: Al igual que el anterior, permite elegir de un grupo, pero sólo una de las opciones.
- *file*: Éste permite la opción de subir archivos. Con él se pueden seleccionar archivos que luego serán enviados al sistema. Si se le indica el atributo *multiple* se podrán seleccionar varios ficheros a la vez. Además, si se indica el atributo *accept* se puede limitar ficheros según su MIME-type, por ejemplo con *audio/\**, *video/\** o *image/\**.

# formularios para la web 2.0

- *submit*: Este es el botón que permite el envío de información al sistema. Puede ir acompañado de otros atributos que hacen que “ese botón” haga cosas que no son las generales del formulario. Los parámetros son: *formaction* (que hace ese botón), *formenctype* (la codificación), *formmethod* (la forma de envío), *formnovalidate* (la validación) y *formtarget* (en qué “ventana” se abre).
- *image*: Es igual que el *submit* pero permite ser una imagen en vez de un botón en sí. Para ello ha de ir acompañado del atributo *src* que indica la dirección URL de la imagen.
- *reset*: Otro botón que si pulsas deja los valores del formulario en su valor inicial.
- *button*: Un botón que “no hace nada”. Suele utilizarse habitualmente con funciones JavaScript que son los que realizan la acción.

# algunos ejemplos de HTML 5

- *javiercasares.com*

Uno de los primeros blogs hispanos en gestionarse con HTML 5 (aunque el WordPress a veces hace malas pasadas)...

- *tumanitas.com*

La primera web comercial íntegramente en HTML 5. Aunque tiene algunos fallos estructurales, en el momento del cambio aumentó el tráfico en 4 semanas en un 50%.



---

# ¿alguna pregunta?

