~~SEO~~

esto no es SEO, es hacer las cosas bien

Artista: Persona que hace algo con suma perfección.

Sois artistas, creáis "algo" de la "nada".
Si queréis que se reconozca vuestro trabajo, tenéis que hacerlo con perfección, como *Michelangelo*.

Webmaster

Si vas a **crear** en Internet,
has de saber **cómo funciona** Internet.

Hacer las cosas bien

- Al desarrollar un sitio web hay que tener en cuenta:
  - Arquitectura de la Información:
    Cómo se va a organizar la información un sitio web.
  - Estructura de la Información:
    Cómo se va a organizar el contenido de un sitio web.
  - Analítica:
    Cómo se va a medir la información de un sitio web.
  - Buenas prácticas:
    Cómo se están usando las distintas herramientas que se ponen a disposición para los webmasters.

Carpetas y ficheros

- El sistema de ficheros de Windows "no distingue" entre mayúsculas y minúsculas, pero el de UNIX sí.
- En Windows "no hay" ficheros sin extensión, en UNIX sí.

- En Internet **las URL** son el identificador único, por lo que ha de cuidarse al 100%.
  Lo ideal es **que nunca cambie**.

URL

http://www.example.com/

https://www.example.com/

https://www.example.com/12345

https://www.example.com/12345/

https://www.example.com/12345/example

https://www.example.com/12345/example/

https://www.example.com/12345/example.html

URL

http://www.example.com/	← carpeta
https://www.example.com/	← carpeta
https://www.example.com/12345	← fichero
https://www.example.com/12345/	← carpeta
https://www.example.com/12345/example	← fichero
https://www.example.com/12345/example/	← carpeta
https://www.example.com/12345/example.html	← fichero

Canonical

http://example.com/

https://example.com/

http://www.example.com/

https://www.example.com/index.php

Todas estas páginas son la misma. Hay que elegir cuál es la correcta e indicarlo.

Canonical

RFC 6596: The Canonical Link Relation

No hay que inventar, hay que leerse el RFC 6596 y ahí está la respuesta a cómo se ha de aplicar.

https://www.casares.blog/rfc6596-canonical/

Organizando la información

- Hay que *clusterizar* la información
- En una web hay que *clusterizar* los mismos tipos de contenido, que se suelen resumir en "tipos de plantilla".
- Para *clusterizar* hay que agrupar contenidos en distintos dominios, hostname o carpetas.

Dominios: gTLD, ccTLD...

- Un dominio es un nombre para que un usuario se acuerde de un sitio web.

- Hay dominios genéricos (.com, .net...) y dominios localizados (.es, .fr, .us...)

SubDominios

- Un subdominio es considerado un "sitio web diferente", pero dependiente del sitio principal.

- Los subdominios se suelen organizar por idiomas, países, servicios...

eng.example.com | us.example.com | api.example.com

Carpetas

- Las carpetas agregan contenidos de un mismo *cluster* dentro de un sitio web.

- Habitualmente son servicios distintos, categorías, o elementos diferenciadores entre varios tipos de contenido.

example.com/coches/ | example.com/motos/

example.com/blog/ | example.com/search/

Hay mil maneras de organizar

Cada proyecto en Internet ha de tener su propia manera de crear su propia Arquitectura de la Información.

Hay mil maneras de organizar

- Si eres un proyecto focalizado a país (soy la Ford), es probable que use su ccTLD para cada lugar.

www.ford.es | www.ford.fr | www.ford.us

Hay mil maneras de organizar

- Si soy un medio de comunicación sobre tecnología, quizá organice la información por idioma:

www.tech.com | es.tech.com | fr.tech.com

Hay mil maneras de organizar

- Si tengo servicios que no tienen nada que ver, por ejemplo un e-commerce:

www.shop.com | api.shop.com | blog.shop.com

- Dentro de la tienda, creo, por ejemplo, carpetas:

www.shop.com/coches/ | www.shop.com/motos/

- Dentro de la API, creo, por ejemplo, carpetas:

api.shop.com/v1/ | api.shop.com/v2/

Hay mil maneras de organizar

- Pero también hay que saber cuándo dejar de crear subniveles y crear "filtros".

example.com/coches/

example.com/coches/2/

example.com/coches/?page=2

example.com/coches/ford/

example.com/coches/ford/2/

example.com/coches/ford/?page=2

Atención

Arquitectura de la paginación

- HTML 5 ya nos da la solución: Link Types.

Los buscadores, con esto, saben tomar decisiones.

rel-first | rel-next | rel-prev | rel-last

```
<link rel="canonical" href="https://www.example.com/pagina/?page=3">
<link rel="first" href="http://www.example.com/pagina/">
<link rel="prev" href="http://www.example.com/pagina/?page=2">
<link rel="next" href="http://www.example.com/pagina/?page=4">
<link rel="last" href="http://www.example.com/pagina/?page=12">
```

https://developer.mozilla.org/en-US/docs/Web/HTML/Link_types

Arquitectura huérfana

example.com/news/12345/de/que/sirve/esto/news.html

- No sirve de nada crear carpetas sin contenido.

¿Esto sí que funciona? (Al menos, podría)
example.com/news/12345/

¡Esto no funciona!
example.com/news/12345/de/que/sirve/

Contenidos *caducables*

- Si sabemos que una página va a desaparecer… ¿por qué no avisar previamente de ello?

```
unavailable_after: [RFC-850 date/time]
```

```
<meta name="robots" content="unavailable_after: 25 Jun 2010 15:00:00 PST">
```

home + listados + fichas

- Un sitio web se compone, habitualmente de 4 tipos de páginas:
  - Página principal
  - Listados
  - Fichas
  - Páginas "estáticas"

- Existen sistemas genéricos de Arquitectura de la Información que siempre funcionan, aunque hay que buscar el adecuado para tu proyecto.

home + listados + fichas

example.com/

example.com/robots.txt
example.com/sitemap.xml

example.com/site/
example.com/site/legal/
example.com/site/contact/

home + listados + fichas

Un producto puede pertenecer a muchas categorías...

example.com/coches/

example.com/tienda/material-escolar/
example.com/tienda/boligrafos/
example.com/tienda/dibujo/

example.com/producto/boligrafo-hb-con-goma/

home + listados + fichas

Un producto puede pertenecer a una categoría...

example.com/coches/

example.com/coches/ford/
example.com/coches/ford/fiesta/

example.com/coches/ford/fiesta/1.5-tdi/
example.com/coches/ford/fiesta/1.5-tdi/fotos/
example.com/coches/ford/fiesta/1.5-tdi/especificaciones/

Códigos HTTP

- Una vez más, todo está documentado en el RFC 2616.
- Los códigos van del 1xx al 5xx
  - 100's: Internos para continuar
  - 200's: Correcto
  - 300's: Redirecciones
  - 400's: Errores web
  - 500's: Errores servidor
- Cuanto más alto es el número, más "grave" es el problema. Es mejor una redirección que un error web, y que un error de servidor.

Códigos HTTP

- Hay que devolver el código correcto en cada momento (que para eso se inventaron).

101 Switching Protocols

- Se usa, últimamente, para pasar de HTTP/1.1 a HTTP/2.0 (en su día del 1.0 al 1.1)

200 OK

- La página se ha cargado correctamente.

Códigos HTTP

300 Multiple Choices

- Redirecciones en las que hay múltiples opciones.
- Se puede mandar una única URL.
- Se puede cachear.

301 Moved Permanently

- La URL se ha movido permanentemente.
- La URL actual ha de sustituirse por la nueva.
- Se puede cachear.

Códigos HTTP

302 Found

- Existe una redirección con respuesta, pero esta URL la siguiente vez puede ser diferente.
- Los motores de búsqueda atribuyen el contenido de la URL de destino al contenido de la URL en cuestión (contenidos duplicados).
- Se puede cachear.

304 Not Modified

- El contenido de la URL es el mismo que se consultó la última ocasión.
- Es la respuesta de elementos cacheados.

Códigos HTTP

307 Temporary Redirect

- Cada vez que se llama la URL puede devolver un contenido distinto.
- Los motores de búsqueda atribuyen el contenido de la URL de destino al contenido de la URL en cuestión (contenidos duplicados).
- Los navegadores suelen avisar de esta redirección.

403 Forbidden

- El contenido no es accesible ya que requiere autorización.
- Si el usuario está autenticado, se suele devolver un 200.

Códigos HTTP

404 Not Found

- La URL no existe, pero podría existir en un futuro.
- Puede ser un "error temporal".

410 Gone

- La Url no existe y es imposible que vaya a volver a existir.
- Es un "error definitivo".

Códigos HTTP

500 Internal Server Error

- Existe un error en el servidor que impide resolver la petición.

503 Service Unavailable

- El servidor actualmente está en mantenimiento o sobre cargado.
- Debe enviarse la cabecera Retry-After indicando cuándo debe volverse a intentar la solicitud.

robots.txt

- ¿Dónde tiene acceso un robots de búsqueda?
- Por defecto hay que dejarle acceder a todo (nunca se han de bloquear scripts u hojas de estilo)
- Sólo se deberían bloquear "carpetas" generales en las que no haya interés que aparezca en los resultados de búsqueda (términos y condiciones…)
- Las páginas que tienen identificador no han de bloquearse (si se devuelve un 403) -login, etcétera-

meta-robots

- index / noindex:
  Permite que esa página aparezca (o no) en los resultados de búsqueda.
- follow / nofollow:
  Permite seguir (o no) todos los enlaces de esa página.
- noarchive:
  Impide que se guarde una copia de la página en el buscador (que se le deje ver la copia al usuario).
- nosnippet:
  Impide que se muestre el snippet en los resultados de búsqueda. Si hay meta-description, no se muestra.

meta-referrer

- no-referrer:
  No se envía URL de referencia.
- origin:
  Se envía la URL de referencia completa.
- no-referrer-when-downgrade:
  No se envía la URL si se pasa a un nivel de seguridad inferior (de https a http).
- origin-when-crossorigin:
  Se envía la URL complete si es el mismo hostname, pero solo el hostname si es distinto.
- same-origin:
  Se envía la URL complete solo si es al mismo hostname.
- strict-origin:
  Sólo se manda el hostname si la URL es del mismo sitio y seguro.
- strict-origin-when-cross-origin:
  Se manda la URL complete a cualquier sitio, siempre que tenga el mismo nivel de seguridad o superior.
- unsafe-URL:
  Manda la Url complete en cualquier situación.

Sitemaps XML

- El fichero de SitemapsXML, es un mapa del sitio en formato XML en el que se incluyen las páginas del sitio. En principio está focalizado a mostrar aquellos contenidos "nuevos", aunque se pueden incluir todos los contenidos.
- Debe devolver siempre una cabecera text/xml.
- Reglas:
  - Máximo 50.000 URL (si hay más: Sitemaps Index)
  - No debe incluir URL que estén bloqueadas en el robots.txt
  - Recomendado solo enviar "listados" y "fichas".

Enlaces

- rel-external:
  Cuando hagamos un enlace externo al sitio (un sitio que no sea de nuestra propiedad -sitio hermano-) añadiremos el siguiente código al enlace.
- rel-help:
  Para aquellas páginas de ayuda o de soporte, utilizaremos el siguiente código en el enlace
- target-_blank + rel-noopener + rel-noreferrer + rel-external:
  Todos los enlaces que se abran en ventana nueva (externa), llevarán conjuntamente los siguientes códigos.

HTML5

- Os podéis leer la documentación oficial, o la documentación de gente que desarrolla y usa la tecnología:

https://www.w3.org/TR/html/

https://developer.mozilla.org/docs/Web/HTML

Si desarrollad en web, has de saberte eso casi de memoria.

HTML vs. XHTML

Hay que centrarse exclusivamente en HTML 5 (en la medida de lo posible, en HTML 5.2, si los navegadores le dan soporte)

HTML 5 ¿por qué?

Semántica

- La gran importancia de HTML 5 después de 10 años con HTML 4, es su foco en que las máquinas comprendan mejor los contenidos y su organización.
- Viene acompañado de mejoras en JavaScript y CSS.
- Todo lo que no sea HTML 5, está obsoleto.

Algunas "chorraditas"

- En el código fuente, los "ampersand" han de escribirse &, sobre todo en los "links".
- Hay elementos que no permiten "cosas dentro": un <kbd> no puede incluir un <section>. Es ilógico.
- Cada elemento tiene un rol. Cuando lo cambiemos, ha de tener sentido: <hr role="cell"> un separador no puede ser una celda.
- Un es un block, un es un inline. Ninguno aporta "nada" al mundo, solo dar formato.

Ante la duda, la más...

...lógica.

¿Tienes dudas de si un código HTML es correcto?
https://validator.w3.org/

Ahí tienes siempre la respuesta.

NOTA: Aplica el sentido común en los errores. Hay cosas lógicas, hay otras que pueden tener que ver con JavaScript y ser correctas aunque diga que no.

Cosas del HTML 5

- Aunque permite quitar "organización", lo mejor es ser organizado.

```
<DOCTYPE html>
<p>Hola mundo!</p>
```

Esto es un sitio web válido, pero es "mejor" esto:

```
<DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
</head>
<body>
 <p>Hola Mundo!</p>
</body>
</html>
```

Atributos globales

accesskey
autocapitalize
class
contenteditable
contextmenu
data-*
dir
draggable
dropzone
hidden
id
is
itemid
itemprop
itemref
itemscope
itemtype
lang
slot
spellcheck
style
tabindex
title
translate

Atributo global: *title*

The title attribute represents advisory information for the element, such as would be appropriate for a tooltip.

On a link, this could be the title or a description of the target resource;

On an image, it could be the image credit or a description of the image;

On a paragraph, it could be a footnote or commentary on the text;

On a citation, it could be further information about the source;

On interactive content, it could be a label for, or instructions for, use of the element...

El primer elemento: html

- Pues eso, cuando se comienza una página, el primer elemento es el <html>.

La meta-información

- El primer elemento tras el <html> ha de ser el <head>.
- En los metadatos encontramos:
  - <title>
  - <base>
  - <link>
  - <meta>
  - <style>

Las secciones

- La primera y mayor sección es el <body>, a partir de aquí tenemos subsecciones.

<article>

- The article element represents a complete, or self-contained, composition in a document, page, application, or site. This could be a magazine, newspaper, technical or scholarly article, an essay or report, a blog or other social media post.

Las secciones

<section>

- The section element represents a generic section of a document or application. A section, in this context, is a thematic grouping of content. Each section should be identified, typically by including a heading (h1-h6 element) as a child of the section element.

¿Article o Section?

- A section forms part of something else.
- An article is its own thing.
- But how does one know which is which? Mostly the real answer is "it depends on author intent".
- For example, one could imagine a book with a "Granny Smith" chapter that just said "These juicy, green apples make a great filling for apple pies."; that would be a section because there'd be lots of other chapters on (maybe) other kinds of apples.
- On the other hand, one could imagine a tweet or tumblr post or newspaper classified ad that just said "Granny Smith. These juicy, green apples make a great filling for apple pies."; it would then be articles because that was the whole thing.

Las secciones

<nav>

- The nav element represents a section of a page that links to other pages or to parts within the page: a section with navigation links.

Las secciones

<aside>

- The aside element represents a section of a page that consists of content that is tangentially related to the content of the parenting sectioning content, and which could be considered separate from that content. Such sections are often represented as sidebars in printed typography.

Las secciones

<h1> - <h6>

- These elements represent headings for their sections.
- These elements have a rank given by the number in their name. The h1 element has the highest rank, the h6 element has the lowest rank, and two elements with the same name have equal rank.

Las secciones

<main>

- The header element represents introductory content for its nearest ancestor main element or sectioning content or sectioning root element. A header typically contains a group of introductory or navigational aids.

Las secciones

<footer>

- The footer element represents a footer for its nearest ancestor main element or sectioning content or sectioning root element. A footer typically contains information about its section, such as who wrote it, links to related documents, copyright data, and the like.

Los grupos

<p>
<address>
<hr>
<pre>
<blockquote>

<dl>
<dt>
<dd>
<figure>
<figcaption>
<main>

Elementos semánticos

<a>

<small>
<s>
<cite>
<q>
<dfn>
<abbr>
<ruby>
<rb>
<rt>
<rtc>
<rp>
<data>
<time>
<code>
<var>
<samp>
<kbd>
<sub>
<sup>
<i>
<b>
<u>
<mark>
<bdi>
<bdo>

<wbr>

Los enlaces

- La etiqueta <a> es quizá de las más importantes de Internet. Básicamente porque la red se basa en ello.
  - Puedes añadirle download para forzar que el link se descargue (por ejemplo en PDF y similares).
  - No tiene porqué llevar un href.
  - Puedes indicarle la política de "referrer" con referrerpolicy.
- NOTE: Anchor tags are often abused with the onclick event to create pseudo-buttons by setting href to "#" or "javascript:void(0)" to prevent the page from refreshing.

vs <i>

- ¿Cuál es la diferencia entre uno y otro?

vs <i>

- <i> se usa para mostrar en cursiva por pantalla.
- <i> no es anidable.

- se usa para dar énfasis semántico a algo.
- sí es anidable.
- no da importancia, da énfasis.

```
<p><em>Cats are <em>cute</em> animals!</em></p>
```

vs <b>

- ¿Cuál es la diferencia entre uno y otro?

vs <b>

- <b> se usa ara mostrar negrita por pantalla
- <b> no es anidable.

- se usa para dar importancia, seriedad o urgencia.
  - Importancia: en una cabecera, caption o parágrafo para distinguir.
  - Seriedad: para marcar algo a destacar.
  - Urgencia: para leerse antes que otras partes.

<s>

- No se ha de usar para "tachar" algo (para eso tenemos el CSS.
- Indica que un contenido ya no es relevante o preciso.
- Si el contenido es sustituido, se ha de usar <del> (normalmente en conjunción con <ins>).

<blockquote>, <cite> y <q>

- <cite> hace referencia a un trabajo creativo.
- <q> representa una frase citada de otro elemento.
- <blockquote> es el contenedor de un elemento con citas.

```
<p><q>This is correct, said Hillary.</q> is a quote from the popular daytime TV drama <cite>When Ian became Hillary</cite>.</p>
```

<time>

- Si hay que indicar un formato de fecha en cualquier formato, se indica con <time>.
- Debe ir acompañado de datetime.

```
<p>Published <time property="published"
datetime="2009-08-29">two days ago</time>.</p>
```

<mark>

- Se usa como un marcador de esos amarillos... Para resaltar algo.

```
<p>Look around and you will find, no-one's really
<mark>colour</mark> blind.</p>
```


- No sirve para nada, pero es útil para ponerle elementos globales.

```
<pre><code class="lang-php">
<span class="keyword">if</span> (<span
class="ident">a</span> == 0) <span
class="ident">b</span> = 1;
</code></pre>
```

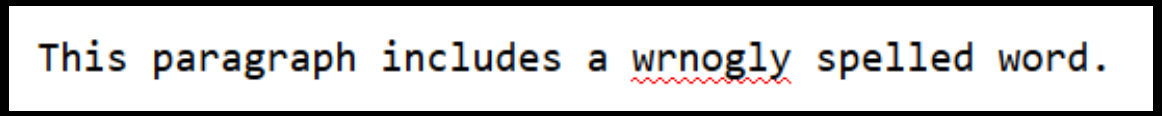

 y <wbr>

- En los textos a veces hay que hacer saltos de línea.
- Con
 forzamos un salto de línea.
- Con <wbr> damos la oportunidad de indicar que ahí puede haber un salto de línea.

<u>

- ¡No! No es para "subrayar" (underline) nada como antiguamente se hacía (HTML 3).
- Se ha de usar para identificar un texto no articulado pero renderizado (sí, una cosa rara).

```
<p>This text includes a <u class="spell">wrnogly</u>
spelled word.</p>
```




```
u.spelling {text-decoration: red wavy underline;}
```

Contenido editado

- <ins> representa un elemento añadido al contenido.
- <del> representa un elemento eliminado del contenido.

- Suelen ir acompañados de cite y datetime para indicar de dónde se saca la información del cambio y cuándo se ha realizado el cambio.

Contenido incrustado (imagen)

- `<picture>` provee varias fuentes de una imagen.
  - `<source>` indica las varias posibles fuentes de una imagen.
  - `<img>` indica la imagen.

```
<picture>
 <source media="(min-width: 32em)"
srcset="large.jpg">
 <img src="small.jpg" alt="Un árbol en la montaña.">
</picture>
```

Contenido incristado (vídeo)

- <video> indica la existencia de un video y en general permite automáticamente mostrar un reproductor HTML.
  - <source> al igual que en imágenes, distintas fuentes.
  - <track> permite indicar los subtítulos.

```
<video controls autoplay>
 <source src='video.mp4' type='video/mp4;
codecs="avc1.42E01E, mp4a.40.2"'>
 <source src='video.ogv' type='video/ogg; codecs="theora,
vorbis"'>
 <track kind=subtitles src=brave.en.vtt srclang=en
label="English">
</video>
```

Otros elementos incrustables

- <iframe> para incluir contenido de otra página.
- <embed> para incluir un elemento habitualmente no-HTML.
- <object> para incluir un recurso extreno tratado según su formato.
  - <param> para configurar el objeto.
- <audio> igual que el <video>.
- <map> para crear zonas clicables en una imagen
  - <area> indica las partes del <map>.

Las tablas

- Además de la <table> en sí, podemos indicarle un texto explicativo con su <caption>.
- Pero además podemos indicar <col>umnas... Y agruparlas con <colgroup>.
- Las tablas pueden tener su cabecera <thead>, su contenido <tbody> y su pie <tfoot>.
- Además de sus filas <tr> y sus celdas normales <td> y de cabecera <th>.

Formularios

<form>
<label>
<input>
<button>
<select>
<datalist>
<optgroup>
<option>
<textarea>
<output>
<progress>
<meter>
<fieldset>
<legend>

Elementos de entrada <input>

button
checkbox
color
date
datetime-local
email
file
hidden
image
month
number
password
radio
range
reset
search
submit
tel
text
url
week

Que se pueden autocomplete=""

off
on
name
honorific-prefix
given-name
additional-name
famili-name
honorific-suffix
nickname
email
username
new-password
current-password
organizartion-title
organization

street-address
address-line1(2/3)
country
country-name
postal-code
cc-name
cc-given-name
cc-additional-name
cc-family-name
cc-number
cc-exp
cc-exp-month
cc-exp-year
cc-csc
cc-type

transaction-currency
transaction-amount
language
bday
bday-day
bday-month
bday-year
sex
tel
tel-country-code
tel-extensión
url
photo
...

Y podemos cambiar el "teclado"

- Sobre todo en los dispositivos móviles, para que al introducir el contenido se nos adapte el teclado.
- Usando inputmode con alguna de estas opciones:
  - none
  - text
  - decimal => input type-number
  - numeric => input type-number
  - tel => input type-tel
  - email => input type-email
  - url => input type-url

Elementos interactivos

- Podemos usar los <details> con su subelemento <summary>.

```
<details>
 <summary><label for=fn>Name &
Extension:</label></summary>
 <p><input type=text id=fn name=fn value="Pillar
Magazine.pdf">
 <p><label><input type=checkbox name=ext checked>
Hide extension</label>
</details>
```

Elementos interactivos

Aún hay más

- Esto es quizá "lo más importante" pero hay muchísimo más.
- Si te dedicas a trabajar con HTML (ya seas "front-end" o "back-end" developer) has de saber qué permite y qué no el HTML.
- De la misma forma tenemos todo lo relacionado con `<script>` (JavaScript) o `<style>` (CSS).
- Saber jQuery no significa saber JavaScript.

Y cómo hago que las máquinas...

...entiendan lo que yo quiero?

Google está construido por ingenieros, y los ingenieros usan los RFC y otros documentos oficiales como base y si les falta algo, lo inventan.

Es por esto que es importante el uso de meta-información a todos los niveles.

Schema.org

- Schema.org is a collaborative, community activity with a mission to create, maintain, and promote schemas for structured data on the Internet, on web pages, in email messages, and beyond.

Existe una extensa lista de "formatos" que son comprensibles para las máquinas:

https://schema.org/docs/full.html

Algunos de los interesantes

- Article: An article, such as a news article or piece of investigative report. Newspapers and magazines have articles of many different types and this is intended to cover them all.
- SocialMediaPosting: A post to a social media platform, including blog posts, tweets, Facebook posts, etc.
- TechArticle: A technical article - Example: How-to (task) topics, step-by-step, procedural troubleshooting, specifications, etc.

Algunos de los interesantes

- APIReference: Reference documentation for application programming interfaces (APIs).
- Blog: A blog.
- Book: A book.
- Comment: A comment on an item - for example, a comment on a blog post. The comment's content is expressed via the text property, and its topic via about, properties shared with all CreativeWorks.
  - Answer: An answer offered to a question; perhaps correct, perhaps opinionated or wrong.

Algunos de los interesantes

- Menu: A structured representation of food or drink items available from a FoodEstablishment.
- Review: A review of an item - for example, of a restaurant, movie, or store.
- Event: An event happening at a certain time and location, such as a concert, lecture, or festival. Ticketing information may be added via the offers property. Repeated events may be structured as separate Event objects.

Algunos de los interesantes

- Brand: A brand is a name used by an organization or business person for labeling a product, product group, or similar.
- ComputerLanguage: This type covers computer programming languages such as Scheme and Lisp, as well as other language-like computer representations. Natural languages are best represented with the Language type.
- PaymentMethod: A payment method is a standardized procedure for transferring the monetary amount for a purchase. Payment methods are characterized by the legal and technical structures used, and by the organization or group carrying out the transaction.

Algunos de los interesantes

- GenderType: An enumeration of genders.
- Flight: An airline flight.
- Invoice: A statement of the money due for goods or services; a bill.
- BreadcrumbList: A BreadcrumbList is an ItemList consisting of a chain of linked Web pages, typically described using at least their URL and their name, and typically ending with the current page.

Algunos de los interesantes

- JobPosting: A listing that describes a job opening in a certain organization.
- Offer: An offer to transfer some rights to an item or to provide a service — for example, an offer to sell tickets to an event, to rent the DVD of a movie, to stream a TV show over the internet, to repair a motorcycle, or to loan a book.
- AggregateOffer: When a single product is associated with multiple offers (for example, the same pair of shoes is offered by different merchants), then AggregateOffer can be used.

Algunos de los interesantes

- Order: An order is a confirmation of a transaction (a receipt), which can contain multiple line items, each represented by an Offer that has been accepted by the customer.
- Rating: A rating is an evaluation on a numeric scale, such as 1 to 5 stars.
- AggregateRating: The average rating based on multiple ratings or reviews.
- Reservation: Describes a reservation for travel, dining or an event. Some reservations require tickets.

Algunos de los interesantes

- OpeningHoursSpecification: A structured value providing information about the opening hours of a place or a certain service inside a place.
- Ticket: Used to describe a ticket to an event, a flight, a bus ride, etc.
- Organization: An organization such as a school, NGO, corporation, club, etc.
- LocalBusiness: A particular physical business or branch of an organization. Examples of LocalBusiness include a restaurant, a particular branch of a restaurant chain, a branch of a bank, a medical practice, a club, a bowling alley, etc.

Algunos de los interesantes

- Restaurant: A restaurant.
- Hotel: A hotel is an establishment that provides lodging paid on a short-term basis.
- Person: A person (alive, dead, undead, or fictional).
- Place: Entities that have a somewhat fixed, physical extension.
- Airport: An airport.

Algunos de los interesantes

- Product: Any offered product or service. For example: a pair of shoes; a concert ticket; the rental of a car; a haircut; or an episode of a TV show streamed online.
- IndividualProduct: A single, identifiable product instance (e.g. a laptop with a particular serial number).
- ProductModel: A datasheet or vendor specification of a product (in the sense of a prototypical description).
- Car: A car is a wheeled, self-powered motor vehicle used for transportation.

Cómo se informan

- Lo mejor es hacer uso de JSON-LD aunque en algunos casos se puede incorporar con *microdata* dentro del HTML 5.

```
<script type="application/ld+json">
{
  "@context": "http://schema.org",
  "@type": "Organization",
  "url": "http://www.example.com",
  "name": "Unlimited Ball Bearings Corp.",
  "contactPoint": {
    "@type": "ContactPoint",
    "telephone": "+1-401-555-1212",
    "contactType": "Customer service"
  }
}
</script>
```

Algunos de estos son importantes

- Review
- Rating
- AggregateRating

Los tiene en cuenta Google. Ayudan a que aparezcan "estrellitas" en los resultados de búsqueda.

https://developers.google.com/search/docs/data-types/review

Algunos de estos son importantes

- Organization
- Place
- PostalAddress

(y otros, dependiendo del tipo de negocio)

Los tiene en cuenta Google. Ayudan a que mejoren los datos de Google Maps y en los resultados de búsqueda.

https://developers.google.com/search/docs/data-types/local-business

Algunos de estos son importantes

- VideoObject

Los vídeos ayudan a ser indexados y mostrados en resultados de búsqueda.

https://developers.google.com/search/docs/data-types/video

Arquitectura y Navegación

- SiteNavigationElement
- AboutPage
- ContactPage
- Brand
- BreadcrumbList

Seguramente, el más sencillo e interesante es este último.

BreadcrumbList

```
<script type="application/ld+json"> {
  "@context": "http://schema.org",
  "@type": "BreadcrumbList",
  "itemListElement":
  [ {
    "@type": "ListItem",
    "position": 1,
    "item": {
      "@id": "https://example.com/dresses",
      "name": "Dresses"
    }

  }, {
    "@type": "ListItem",
    "position": 2,
    "item": {
      "@id": "https://example.com/dresses/real",
      "name": "Real Dresses"
    }
  } ]
} </script>
```

EOF