

Cheatsheet

Taller Web - 2018

HTML

<html>

Elemento raíz del documento.

<head>

Determina el área de encabezado del documento. No es visible

<title>

Indica el título del documento. Se puede ver como el texto que figura en la ventana/tab del navegador. Este tag debe estar dentro de <head>

<body>

Determina el área de cuerpo del documento. Dentro del body se colocan todos los elementos que el browser va a renderizar

<h1> <h2> <h5>

Indica tipos de títulos. Similar a Word con "Título 1", "Subtítulo 1", "Subtítulo 2", etc. Vienen pre-formateados, pero por CSS se puede modificar el estilo.

<table>

Indica un área de tabla. Dentro de la tabla habrá filas y celdas.

Algunos atributos necesarios para el taller:

- id: identificador único del elemento en el documento
- border: indica formato de borde para toda la table. Valores: 1,2,3... etc. El estilo que se le da es muy rústico, pero sirve para poder visualizar la tabla.

<tr>

Indica una fila dentro de una tabla.

<td>

Indica una celda dentro de una fila de una tabla.

Atributo útil:

- colspan: Acepta valores enteros 1,2,3..etc. Este atributo sirve para “mergear” (o combinar) celdas dentro de una fila. Similar a la opción “combinar celdas” de word.

Indica una referencia a una imagen.

Atributo útil:

- src: indica el path/ruta a la imagen

<div>

Determina un bloque dentro del documento. No es una división visual, sino más bien estructural.

<link>

Sirve para referir a recursos externos. Usualmente se utiliza para incluir hojas de estilos (CSS).

Atributos útiles:

- rel: indica cómo está relacionado el recurso con el documento actual. Para hojas de estilos usamos “**stylesheet**”.
- type: indica el tipo de archivo del recurso externo. Para CSS usamos **text/css**.
- href: indica la ubicación del recurso externo.

<input>

Indica un elemento que acepta entrada de información por parte del usuario. Hay muchos tipos: input, password, etc etc.

Atributo útil para el taller:

- type: indica el tipo de input. Vamos a usar **checkbox**.
- onclick: una referencia a una función de JavaScript que se ejecutará al hacer click sobre el elemento.
- checked: si está presente el atributo, el input (checkbox) se mostrará marcado. Si no está presente el atributo estará desmarcado.

<label>

Indica un área de texto simple (etiqueta)

<select>

Indica un selector tipo combo/dropdown.

Atributos útiles:

- `onchange`: una referencia a una función JavaScript que se ejecutará cuando cambie el valor seleccionado por el usuario.
- `disabled`: si el atributo está presente, el selector está deshabilitado.

<option>

Indica opciones dentro de un `<select>`.

Atributo útil:

- `value`: es el valor (no lo que el usuario ve) de dicha opción.

Ej: `<option value="op1">Opción 1</option>`

CSS

display: <flex | none>

Indica cómo se mostrará un elemento.

FlexBox

https://developer.mozilla.org/es/docs/Web/CSS/CSS_Flexible_Box_Layout/Usando_las_cajas_flexibles_CSS

flex-direction: <row | column>

Para elementos con `display: flex`, indica si será una orientación horizontal (row) o vertical (column)

`flex-grow`: <número>

Para un elemento dentro de un bloque con `display: flex`, indica cuánto crecerá proporcionalmente al crecer el bloque.

Para el taller usaremos el valor **1** para indicar que se ajuste al máximo posible.

font-family

Indica la tipografía. Algunas tipografías standard son: **Arial**, **Helvetica**, etc.

font-size

Indica tamaño de la tipografía. Se puede indicar en varias unidades: pt, px, etc. No debe haber espacio entre el valor y la unidad.

Ej: **12px**;

color: <hexa rgb>

Indica el color de la fuente. Los colores se indican en formato hexa-rgb: **#RRGGBB**.

Para encontrar colores fácilmente: https://www.w3schools.com/colors/colors_picker.asp

padding:

Indica un espacio entre el borde del bloque y el contenido. Se puede indicar en diferentes unidades, pero para el taller usaremos **px**. Ej: **12px**;

padding-top, **padding-bottom**, **padding-left**, **padding-right** se utilizan para indicar el padding de cada uno de los lados del bloque.

width / height

Ancho / Alto del bloque. En el taller usaremos pixels: Ej: **width: 12px**;

border: <solid | dotted | etc> <tamaño en px> <color>

Indica el estilo del borde del bloque. Existe **border-top**, **border-bottom**, **border-left** y **border-right** para los lados correspondientes. Ej: **border-bottom: solid 1px #0000FF**;

DOM API

document.getElementById(<id>)

Devuelve un elemento matcheando por identificador único.

document.querySelectorAll(<selector>)

Devuelve un conjunto de **elementos** matcheado por el selector indicado. El formato de selector es el mismo utilizado en CSS.

[array].forEach(item => { <codigo> })

para cada **item** del **array** se ejecuta el **código** indicado. Funciona como un iterador de elementos.

[elemento].classList

Devuelve un listado con las clases (css) del **elemento**.

Tip: para activar/desactivar una clase se puede usar

<elemento>.classList.toggle(<miClase>, <true|false>).

[elemento].setAttribute(<atributo>, <valor>)

Setea el valor de un atributo para el elemento.

[elemento].removeAttribute(<atributo>)

Remueve el atributo del elemento.

[elemento].innerHTML = <código html>

Fuerza el contenido de elemento, mediante el seteo del código HTML interno.