

Selectores:

- **Selector universal (*)**: selecciona todos los elementos de la página.
- **Selector de etiqueta (p, div, h1)**: aplica estilos a todos los elementos de ese tipo.
- **Selector de clase (.nombre)**: selecciona elementos con una clase específica.
- (class="nombre")
- **Selector de ID (#nombre)**: selecciona un único elemento con un identificador único.
- (id="nombre")

Selectores de atributos

- [atributo]: selecciona elementos que tengan ese atributo.
- [atributo="valor"] : selecciona elementos con un valor específico.
- [atributo^="valor"] : selecciona elementos cuyo atributo empieza por ese valor.
- [atributo\$="valor"] : selecciona elementos cuyo atributo termina con ese valor.
- [atributo*="valor"] : selecciona elementos cuyo atributo contiene ese valor.

Tipos de declaraciones:

- propiedad **text-align** alineación de texto.
- propiedad **font-family** tipo de letra, si no puede "Roboto" será "sans-serif".
- propiedad **background-color** color de fondo.
- propiedad **color** color de texto.
- list-style
- disc Círculo sólido (por defecto)
- circle Círculo hueco
- square Cuadrado
- Text-decoration: underline; Propiedad que indica que el texto se mostrará subrayado.

Unidades:

- Entero.

Ejemplo: line-height: 2;

- ❖ Establece la altura de la caja de la línea de texto al doble del tamaño de la fuente.

Número decimal, utiliza el punto decimal.

Ejemplo: opacity: 0.75.

- ❖ Establece la transparencia de un elemento, 1.0 es completamente sólido, 0.0 es completamente transparente.

Dimensión: número con una unidad de medida asociada, por ejemplo px (píxeles).

Ejemplo: margin: 10px.

Porcentaje: una fracción de un total, por ejemplo, width="75%" indica que el ancho del elemento será un 75% del real.

Unidades absolutas:

Representan longitudes sin importar el contexto del componente al que hacen referencia o del tamaño de la pantalla. Las más usadas son:

- cm centímetro.
- mm milímetro.
- in pulgada (2,54 cm).
- px píxel (1/96 de in).
- pt puntos (1/72 de in).

Unidades relativas:

- em: en base al tamaño de letra del elemento padre.
- rem: en base al tamaño de letra del elemento raíz.
- ex: altura de la fuente actual.
- ch: ancho del carácter "0" de la fuente.
- vw: % del ancho de la ventana gráfica.
- vh: % del alto de la ventana gráfica.
- vmin: % de la dimensión más pequeña de la ventana gráfica.
- vmax: % de la dimensión más grande de la ventana gráfica.

Colores:

- Por nombre: red, blue, green, etc...
- Por valor RGB: rgb(red, green, blue), rgb(0-255, 0-255, 0-255).
Ejemplo: rgb(100, 75, 0).

- Por valor hexadecimal: #valorhexadecimal. Ejemplo: #ff0001.
- Por valor HSL: hsl(matiz, saturación, luminosidad). Ejemplo: hsl(25, 100%, 50%).

Imagen:

background-image: url("huerta.jpg").

Orden de importancia:

1. !important ,prioridad máxima
2. style, en la propia etiqueta prioridad muy elevada
3. Selectores de ID, prioridad elevada
4. Selectores de clase, pseudo-clase y atributo prioridad media
5. Selector de elemento, prioridad baja
6. Selector universal ,prioridad muy baja
7. Orden en el documento HTML, última prioridad

Pseudoclases:

Estado y validación de formularios:

- :active → El elemento ha sido activado por el usuario.
- :checked → Afecta a elementos <input> de tipo radio o checkbox o a elementos <option> cuando han sido marcados.
- :default → Elemento de un formulario marcado como predeterminado.
- :disabled → El elemento está deshabilitado.
- :enabled → El elemento está habilitado.
- :empty → El elemento no tiene hijos.
- :focus → El elemento (de un formulario) tiene el foco.
- :focus-within → El elemento (de un formulario) o uno de los elementos contenidos tiene el foco.
- :hover → El cursor del ratón se encuentra sobre el elemento.
- :indeterminate → El estado es indeterminado. Aplica a <input> tipo checkbox/radio y a <progress>.
- :in-range → El valor de un campo <input> está dentro del rango definido por min y max.
- :out-of-range → El valor de un campo <input> está fuera del rango definido por min y max.
- :invalid → El elemento (de un formulario) no es válido.

- :valid → El elemento (de un formulario) sí es válido.
- :optional → El elemento (<input>, <select>, <textarea>) no tiene el atributo required.
- :required → El elemento es obligatorio.
- :read-only → El elemento no es editable.
- :read-write → El elemento es editable.

Posición y tipo

- :first-child → El primer elemento entre un grupo de hermanos.
- :last-child → El último elemento entre un grupo de hermanos.
- :nth-child(n) → El elemento en la posición n entre sus hermanos. Se puede usar odd y even.
- :nth-last-child(n) → El elemento en la posición n contando desde el final.
- :first-of-type → El primer elemento de un tipo entre sus hermanos.
- :last-of-type → El último elemento de un tipo entre sus hermanos.
- :nth-of-type(n) → El elemento de un tipo en la posición n entre sus hermanos.
- :nth-last-of-type(n) → El elemento de un tipo en la posición n contando desde el final.
- :only-child → El elemento no tiene hermanos.
- :only-of-type → El elemento no tiene hermanos del mismo tipo.

Enlace y estructura

- :link → El enlace no ha sido visitado. :visited → El enlace ha sido visitado.
- :target → El elemento referenciado por un enlace interno.
- :lang(idioma) → El elemento está en un idioma específico, por ejemplo :lang(es).
- :not(selector) → Selecciona elementos que **no** coinciden con el selector dado.
- :root → El elemento raíz del documento (normalmente).

Específicas para enlaces

- `a:link` → Es el estado por defecto de un enlace. `a:visited` → Se activa cuando el enlace ha sido visitado (aunque en navegadores modernos está limitado por seguridad).
- `a:hover` → Se activa cuando se pasa el ratón por encima del enlace.
- `a:focus` → Se activa cuando el enlace recibe el foco (por ejemplo, al navegar con teclado).
- `a:active` → Se activa en el momento de la activación del enlace (cuando se hace clic).

Agrupación de selectores:

Ejemplos 1:

```
p, a, span, em {
    text-decoration: underline;
}
```

Bordes:

- **`border-width`** → Define el grosor del borde (ej. `thin`, `medium`, `thick` o valores en `px/rem`). **`border-style`** → Establece el estilo del borde (`solid`, `dashed`, `dotted`, `double`, `groove`, `ridge`, `inset`, `outset`, `none`).
- **`border-color`** → Cambia el color del borde (puede ser un color, `transparent`, o valores `RGBA/HEX`).
- **`border`** → Propiedad abreviada que combina ancho, estilo y color (ej. `border: 2px solid red;`).
- **`border-radius`** → Redondea las esquinas del borde (ej. `border-radius: 10px;`). **`border-image`** → Permite usar una imagen como borde.
- **`border-image-slice`** → Define cómo se corta la imagen para aplicarla al borde.
- **`border-image-repeat`** → Controla si la imagen se repite, se estira o se ajusta.
- **`border-image-width`** → Ajusta el ancho de la imagen usada como borde.

Específicas por lado:

- **border-top, border-right, border-bottom, border-left** → Permiten definir el borde de cada lado de manera independiente.
- **Border-top-left-radius,**
- **border-top-right-radius,**
- **border-bottom-left-radius,**
- **border-bottom-right-radius**

Redondean esquinas específicas.

Listas:

- **list-style-type** → Define el tipo de marcador (viñeta o numeración).
Ejemplos: disc, circle, square, decimal, lower-roman, upper-alpha, none. **list-style-position** → Controla la posición del marcador respecto al texto.
 - outside → el marcador queda fuera del bloque de texto.
 - inside → el marcador queda alineado dentro del bloque.
- **list-style-image** → Permite usar una imagen personalizada como viñeta.
Ejemplo: list-style-image: url("icono.png").

Tablas:

border-collapse

Determina si los bordes de las celdas de la tabla están fusionados o no.
Valores posibles: collapse, separate.

border-spacing

Determina la distancia entre bordes adyacentes.
Valores posibles: cualquier unidad válida en CSS (px, em, etc.).

caption-side

Determina dónde aparecerá el texto contenido en el elemento .
Valores posibles: top, bottom.

empty-cells

Indica si las celdas vacías deben mostrarse o no.
Valores posibles: show, hide.

table-layout

Determina el algoritmo que utilizar para diseñar celdas, filas y columnas.

Valores posibles: auto, fixed.

vertical-align

Determina la alineación vertical del contenido de las celdas de una tabla.

Valores posibles: top, middle, bottom.

Texto:

color

Determina el color del texto.

Valores posibles: cualquier valor válido de color (nombre, HEX, RGB, etc.).

font-family

Determina el tipo de letra que se va a utilizar.

Valores posibles: cualquier fuente válida o familia de fuentes.

font-size

Determina el tamaño del texto.

Valores posibles: medium, xx-small, x-small, small, large, x-large, xx-large, smaller, larger, unidades CSS (px, em, %, etc.).

font-style

Determina el estilo del texto.

Valores posibles: normal, italic, oblique.

font-weight

Determina el grosor del trazo de los textos.

Valores posibles: normal, bold, bolder, lighter, 100 a 900.

text-transform

Transforma el texto.

Valores posibles: capitalize, uppercase, lowercase.

text-decoration

Agrega una decoración al texto en forma de línea.

Valores posibles: combinación de `text-decoration-line`, `text-decoration-style`, `text-decoration-color`.

text-decoration-line

Dibuja una línea horizontal sobre el texto.

Valores posibles: `underline`, `overline`, `line-through`.

text-decoration-style

Determina el tipo de línea.

Valores posibles: `solid`, `double`, `dotted`, `dashed`, `wavy`.

text-decoration-color

Determina el color de la línea.

Valores posibles: cualquier color válido.

text-shadow

Añade una sombra al texto.

Valores posibles: color + desplazamiento en eje X e Y + desenfoque.

text-align

Determina la alineación del texto.

Valores posibles: `left`, `right`, `center`, `justify`.

line-height

Determina el alto de línea de los textos.

Valores posibles: número sin unidad, número con unidad, porcentaje.

letter-spacing

Determina el espacio entre letras.

Valores posibles: cualquier unidad válida en CSS.

word-spacing

Determina el espacio entre palabras.

Valores posibles: cualquier unidad válida en CSS.

Maquetación de la página:

Propiedades del flex:

Display

Permite que el elemento se convierta en un contenedor Flexbox, haciendo posible la distribución flexible de sus *flex items* o elementos hijos.

- **flex** → Los *flex items* se organizan según el eje principal.
- **inline-flex** → El contenedor se comporta como un elemento en línea, no provoca salto de línea y su tamaño se ajusta a lo que contiene.

Flex-direction

Indica la forma en la que se colocarán los *flex items* con respecto al eje principal (por defecto row, de izquierda a derecha).

- **row-reverse** → Los *flex items* se colocan de derecha a izquierda.
- **column** → Los *flex items* se colocan de arriba a abajo. El eje principal pasa a ser vertical y el eje transversal horizontal.
- **column-reverse** → Los *flex items* se colocan de abajo a arriba. El eje principal pasa a ser vertical y el eje transversal horizontal.

Flex-wrap

Indica las líneas o columnas que ocuparán los *flex items* (por defecto nowrap, una única línea, si no caben se encogen).

- **wrap** → En varias líneas o columnas según la dirección activa.
- **wrap-reverse** → En varias líneas o columnas en orden inverso a la dirección activa.

Justify-content

Indica la alineación de los *flex items* con respecto al eje principal.

- **flex-start** → Al inicio del eje principal (por defecto).
- **flex-end** → Al final del eje principal.
- **center** → Al centro del eje principal.
- **space-between** → Misma separación entre ellos, sin espacio al inicio ni al final.
- **space-around** → Misma separación entre ellos, con espacio al inicio y al final (la mitad del espacio entre ítems).
- **space-evenly** → Exactamente la misma separación entre ellos y con el inicio y final del eje principal.
- **align-items**

Indica cómo se alinean los *flex items* con respecto al **eje transversal** cuando aparecen en una sola línea.

Valores principales:

- **stretch** → Por defecto, ocupan todo el espacio disponible en el eje transversal.
- **flex-start** → Se alinean al inicio del eje transversal.
- **flex-end** → Se alinean al final del eje transversal.
- **center** → Se alinean al centro del eje transversal.
- **baseline** → Se alinean siguiendo la línea base del último texto.

align-content

Indica cómo se alinean los *flex items* con respecto al **eje transversal** cuando aparecen en más de una línea.

Valores principales:

- **stretch** → Por defecto, ocupan todo el espacio disponible en el eje transversal.
- **flex-start** → Se alinean al inicio del eje transversal.
- **flex-end** → Se alinean al final del eje transversal.
- **center** → Se alinean al centro del eje transversal.
- **space-between** → Dejan la misma separación entre ellos, sin espacio al inicio ni al final transversal.
- **space-around** → Dejan la misma separación entre ellos y con el inicio y final transversal. El espacio al inicio y final es la mitad que el espacio total entre los *flex items*.
- **space-evenly** → Dejan exactamente la misma separación entre ellos y con el inicio y final transversal. El espacio al inicio y final es igual al espacio total entre los *flex items*.

Propiedades asociadas a los flex items:

Order

Indica el orden de los *flex items* dentro del contenedor, cambiando su orden visual según el valor asignado.

- Valor predeterminado: **0**.
- Se pueden usar valores **negativos** (sitúan el ítem antes del orden :0) y **positivos** (lo sitúan después).
- Solo funciona si el contenedor es `display: flex`.

Flex-grow

Permite especificar la proporción en la que un *flex item* aumentará su tamaño respecto a los demás cuando el contenedor tenga espacio disponible.

- Valor predeterminado: **0** (no aumenta automáticamente).
- Acepta valores **0, positivos o negativos**.
- Si varios ítems tienen distintos valores, el espacio disponible se reparte proporcionalmente según su `flex-grow`.
- Solo funciona si el contenedor es `display: flex`.

Flex-shrink

Permite especificar la proporción en la que un *flex item* disminuirá su tamaño si fuera necesario, en relación con los demás, cuando el contenedor no tenga suficiente espacio para mostrar todos.

- Acepta valores **negativos, positivos o cero**.
- Si varios ítems tienen distintos valores, la reducción se reparte proporcionalmente (similar a `flex-grow`).
- Solo funciona si el contenedor es `display: flex`.

Flex-basis

Indica el **tamaño inicial** de los *flex items* dentro del contenedor antes de aplicar aumentos o reducciones.

- Tiene prioridad sobre `width` o `height`, salvo que su valor sea `auto` (por defecto).
- Solo funciona si el contenedor es `display: flex`.

Flex

Propiedad abreviada que combina **`flex-grow`**, **`flex-shrink`** y **`flex-basis`** en una sola declaración.

- Sintaxis: `flex: grow shrink basis;`
- Ejemplo: `flex: 1 0 auto;`

Align-self

Permite alinear cada *flex item* de manera independiente al resto.

- Ignora la alineación en el eje transversal establecida por `align-items` en el contenedor, aplicándose solo al ítem

concreto.

- Admite los mismos valores que `align-items`:
 - **stretch** → Ocupa todo el espacio disponible en el eje transversal.
 - **flex-start** → Se alinea al inicio del eje transversal.
 - **flex-end** → Se alinea al final del eje transversal.
 - **center** → Se alinea al centro del eje transversal.
 - **baseline** → Se alinea siguiendo la línea base del texto.
- Solo funciona si el contenedor es `display: flex`.

Box-sizing: border-box;

`box-sizing: border-box;` sirve para que el **ancho y alto de un elemento incluyan el padding y el borde**, evitando que se sumen aparte y se desajuste el tamaño total.

Grid layout

Propiedades asociadas al contenedor Grid:

Display

Convierte un elemento en un **contenedor Grid**, permitiendo distribuir sus hijos (*grid items*) en una rejilla.

Valores principales:

- **grid** → Convierte el elemento en contenedor de grid capaz de contener *grid items* o hijos directos.
- **inline-grid** → El contenedor se comporta como un elemento en línea, no provoca salto de línea y se ajusta al contenido.

Grid-template-columns

Permite definir la **estructura de columnas** de la rejilla.

- Se pueden indicar tantos valores como columnas queramos incluir.
- Los valores pueden ser unidades relativas (`fr`, `%`), absolutas (`px`, `em`), funciones (`minmax()`, `repeat()`) o simplemente `auto`.

Grid-template-rows

Permite definir el **número de filas** del Grid y el **alto de cada una**.

- La forma de indicar lo que ocupará verticalmente cada fila es similar a `grid-template-columns`.
- Se pueden usar unidades relativas (`fr`, `%`), absolutas (`px`, `em`), funciones (`minmax()`, `repeat()`) o `auto`.

Gap (antes grid-gap)

Permite definir el **espacio entre filas y columnas** en un Grid.

Sintaxis:

gap: ;

- **row-gap** → Espacio entre filas.
- **column-gap** → Espacio entre columnas.
- **justify-items**: alinea los grid items según el eje horizontal. Los valores permitidos son: start, end, center y stretch (por defecto).
- **align-items**: alinea los grid items según el eje vertical. Los valores permitidos son: start, end, center y stretch (por defecto).

Propiedades asociadas a los grid items:

- **grid-column-start** y **grid-column-end**: sirven para controlar en qué columna comienza y termina (sin ocuparla) un grid item dentro de una línea de la cuadrícula. Son muy útiles para posicionar grid item sin depender de `grid-template-areas`. Las líneas de la cuadrícula se numeran empezando en 1 de izquierda a derecha.
- **grid-row-start** y **grid-row-end**: similares a las propiedades anteriores pero indicando filas que ocupan en vez de columnas.