

COMPATIBILIDAD

Tipos de navegadores web soportan las distintas características de CSS.
 @supports. (Chrome, Firefox, Safari, Edge)
 prefijos (ej: -webkit-, -moz-, -ms-).

LENGUAJE CSS**CÁLCULOS CSS**

Explica el uso de funciones CSS para realizar cálculos matemáticos, incluyendo funciones para máximos y mínimos, redondeo, funciones trigonométricas y exponentiales.

FUNCION
(calc())**SIGNO**
abs() / sign()

Funciones para obtener valores máximos y mínimos (max(), min()), redondear números (round(), floor(), ceil()), y realizar cálculos trigonométricos (sin(), cos(), tan()) y exponentiales (pow(), sqrt()).
Unidades mixtas: Puedes combinar diferentes unidades de medida en un mismo cálculo (ej: width: calc(100% - 20px)).

LÓGICA CSS

Introduce el uso de condicionales con la función if().

Menciona la regla @function para crear funciones personalizadas en CSS.

Explora la función random() para generar números aleatorios.

Describe la función attr() para acceder a los valores de los atributos HTML desde CSS.

Función CSS	Descripción	Ejemplo
pow()	Devuelve la base A elevada al exponente B.	pow(A, B)
sqrt()	Devuelve la raíz cuadrada de A.	sqrt(A)
hypot()	Devuelve la raíz cuadrada de A^2+B^2 + ...	hypot(A, B, ...)
log()	Devuelve el logaritmo de A en base B (si se omite, es e)	log(A, B)
exp()	Devuelve el valor exponencial de A.	exp(A)

SELECTORES CSS

Este capítulo profundiza en los selectores, que permiten aplicar estilos a elementos específicos. Se cubren selectores básicos, de atributos, combinadores, pseudoclases como :hover y :nth-child, y pseudoelementos como ::before y ::after. También se discuten metodologías para escribir CSS más manejable.

SELECTORES

Selector de clase (.clase): Aplica estilos a todos los elementos con una clase específica.

Selector de ID (#id): Aplica estilos a un elemento con un ID específico

Selector universal (*): Aplica estilos a todos los elementos.

Selector de atributo: input[type="text"] { border: 1px solid #ccc; }

Combinaciones: Descendiente (A B), Hijo directo (A > B), Hermano adyacente (A + B) Hermano general (A ~ B)

PSEUDOELEMENTOS CSS

:**before**: Inserta contenido antes del contenido de un elemento.

:**after**: Inserta contenido después del contenido de un elemento.

:**p:first-line**: Aplica estilos a la primera línea de un bloque de texto.

:**p:first-letter**: Aplica estilos a la primera letra de un bloque de texto.

:**li:marker**: Aplica estilos a los marcadores de los elementos de lista.

:**input::placeholder**: Aplica estilos al texto de sugerencia (placeholder) de los campos de entrada.

Permite modificar los estilos

```
textarea {
  min-width: 400px;
  min-height: 100px;
  font-size: 1.25rem;
}

::spelling-error {
  background: darkred;
  color: white;
}
```

CALIDAD DE CÓDIGO

Organización: Utiliza saltos de línea y comentarios para hacer el código legible.

Divide el código en secciones lógicas.

Considera usar preprocesadores CSS (Sass, Less)

Utiliza abbreviaciones CSS (por ejemplo, margin en lugar de margin-top, margin-right)

Evita el uso excesivo de !important.

:**not(selector)**: Selecciona todos los elementos que no coinciden con el selector dado

:**has(selector)**: Selecciona elementos que contienen al menos un elemento que coincide con el selector dado.

TIPOGRAFIAS

La elección de una tipografía adecuada, su tamaño, color, espacio entre letras, interlineado y otras características pueden dotar de una parte fundamental, de forma consciente o inconsciente, en la percepción en la que una persona interpreta o accede a los contenidos de una página.

TIPOS

font-family: Especifica la fuente para un elemento.

font-size: Establece el tamaño de la fuente.

font-weight: Define el grosor de la fuente. Los valores comunes son normal, bold y números del 100 al 900.

font-style: Controla el estilo de la fuente (normal, italic, oblique).

line-height: Establece la altura de la línea.

letter-spacing: Ajusta el espacio entre las letras.

Valor	Descripción
none	Elimina cualquier tipo de decoración.
underline	La línea de decoración se establece por abajo (subrayado).
overline	La línea de decoración se establece por encima.
line-through	La línea de decoración es una línea tachada.

FUENTES VARIABLES

font-weight: Permite rangos de valores.

font-stretch: Controla el ancho de la fuente (condensed, expanded, etc.).

font-style: Permite inclinar la fuente.

font-variation-settings: Proporciona control de bajo nivel sobre las características de la fuente variable.

CARGA TIPOGRAFIA

@font-face: Permite cargar fuentes personalizadas.

font-display: Controla cómo se muestra la fuente mientras se carga (auto, block, swap, fallback, optional).

font-variant: Controla variantes tipográficas como versalitas.

Unidades de medida tamaños

LENGUAJE CSS

LAYOUTS

MAQUETACIÓN

display: Define el tipo de caja de un elemento (block, inline, inline-block, flex, grid, etc.)

float: Posiciona un elemento a la izquierda o a la derecha de su contenedor, permitiendo que otros elementos fluyan a su alrededor.

clear: Especifica qué lados de un elemento no deben flotar junto a otros elementos flotantes.

Propiedad	Descripción
column-gap	Establece un hueco o espacio entre columnas con el tamaño de las líneas divisorias.
column-rule-width	Establece el tamaño de la línea divisoria entre columnas.
column-rule-style	Establece el estilo de la línea divisoria entre columnas.
column-rule-color	Establece el grosor de la línea divisoria entre columnas.
column-rule	Propiedad de atajo de las tres anteriores.

GRID CSS

Contenedor Grid: Se establece con `display: grid;`

Definición de filas y columnas: Utilizando `grid-template-rows` y `grid-template-columns`.

Ubicación de elementos: Mediante `grid-row` (filas) y `grid-column`.

grid-gap: Define el espacio entre las celdas del grid.

grid-area: Permite colocar elementos en áreas específicas del grid.

```
.grid {
  display: grid;
  grid-template-columns: 1fr 2fr;
  grid-template-rows: auto;
  gap: 10px;
}
```

FLEX CSS

Contenedor Flex: Se define estableciendo `display: flex;` en el elemento padre.

Dirección: Controlada por `flex-direction` (row, column, etc.).

Alineación: `justify-content` (eje principal) y `align-items` (eje cruzado) controlan la alineación de los elementos hijos.

Valor	Descripción
start	Agrupa los ítems al inicio del eje principal.
end	Agrupa los ítems al final del eje principal.
center	Agrupa los ítems al centro del eje principal.
space-between	Distribuye los ítems dejando espacio entre ellos.
space-around	Distribuye los ítems dejando espacio alrededor de ellos.

POSICIONAMIENTOS CSS

position: static; Valor por defecto. El elemento se posiciona según el flujo normal del documento.

position: relative; El elemento se posiciona relativo a su posición normal. Se utilizan top, right, bottom, left para ajustar su posición.

position: absolute; El elemento se posiciona relativo a su ancestro posicionado más cercano.

position: fixed; El elemento se posiciona relativo al viewport (la ventana del navegador) y permanece fijo incluso al hacer scroll.

```
body {
  height: 250px;
}

.container {
  font-size: 3rem;
  background: grey;
}

.element {
  background: black;
  color: white;
  font-size: 1.5rem;
  padding: 25px;
  width: 200px;
  position: sticky;
  right: 0;
  top: 0;
}
```

```
table {
  border: 1px solid #000;
  border-collapse: separate;
  border-spacing: 5px;
}
```

INTERFAZ DE USUARIO

CONTORNOS

outline: resalta sin afectar tamaño/posición.

Propiedad	Significado
outline-width	Tamaño de los bordes. Tamaño predefinido o específico.
outline-style	Idem a los bordes. Estilo de la línea divisoria.
outline-color	Invierte el color de la misma.
outline-offset	Desplazamiento del contorno.

BARRAS DE DESPLAZAMIENTO

Personaliza scrollbar (estética/usabilidad).

```
.element-1 {scrollbar-color: deeppink indigo;scrollbar-width: thin;scrollbar-gutter: auto;}.
element-2 {scrollbar-color: deeppink indigo;scrollbar-width: thin;scrollbar-gutter: stable both-edges;}
```

cursor: cambia icono del puntero.

```
.element {
  width: 200px;
  height: 200px;
  background: indigo;
  cursor: help;
}
```

SCROLL SUAVE

scroll-behavior: smooth
Transición fluida en anclas.

CSS SNAP SCROLL

scroll-snap-type + scroll-snap-align: detención precisa (sliders, galerías).

Scroll Snapping Alignment

Vertical



REPRESENTACIÓN DE DATOS

TABLAS CSS

Estiliza con `border`, `background`, `padding`, `text-align`.

LISTAS CSS

`list-style-type`/`image`/`position`: personalización completa.

CONTADORES CSS

`counter-reset`, `counter-increment`, `content`: numeración automática.

Propiedades	Descripción
counter-reset	Resetea el contador <code>id</code> al valor inicial (0 por defecto).
counter-increment	Incrementa un contador, indicándole su <code>id</code> de referencia.
counter-set	Similar a <code>counter-reset</code> , pero no realiza incrementos si el

LENGUAJE CSS

EFEKTOS

SOMBRA

Texto
text-shadow: resalta o estiliza texto.
`.text {text-shadow: 4px 4px; /*`
 Equivalente a `4px 4px 0 black */text-`
`shadow: 4px 4px 2px; /*`
 Equivalente a `4px 4px 2px black */text-`
`shadow: 4px 4px 5px #0005;}`

PIXEL ART

Iguales a las generales.
`.box {`
`box-shadow: 5px 5px 10px black;}`
`.drop`
`{filter: drop-shadow(5px 5px 10px black);}`

CAJAS

Cajas
box-shadow: sombra con desplazamiento y color.
`.element {box-shadow: 5px 5px 0;`
`box-shadow: 5px 5px 2px;`
`box-shadow: 5px 5px 10px;`
`box-shadow: 5px 5px 40px;`

SOMBRA IDÉNTICAS

box-shadow múltiple sin blur → pixel art.
`.pixel {`
`background: black; width: 32px; height:`
`32px; box-shadow:`
`32px 0 red,`
`64px 0 gold;`

FILTROS

Filtros CSS
`filter: desenfoque, brillo, grises, etc.`
`img {`
`filter: grayscale(100%);`
`transition: filter 1s;`
`width: 200px;`
`}`
`img:hover {`
`filter: grayscale(0%);`

MODOS DE FUSIÓN

mix-blend-mode, background-blend-mode:
 mezcla visual entre capas.
`body {`
`background: deeppink;`
`}`
`img {`
`mix-blend-mode: multiply;`
`}`

MÁSCARAS Y RECORTES

FORMAS BÁSICAS

clip-path: círculos, polígonos, elipses.

CLIP-PATH

Recorta y define partes visibles.

FLOAT

Ubica elemento y hace fluir texto.

SHAPE-OUTSIDE

Define cómo el texto rodea la forma.

RESPONSIVE

Diseño que adapta la web a distintos dispositivos.

BASES DEL RESPONSIVE DESIGN

Uso de unidades relativas, diseño fluido y media queries.

```
<header></header>
<div class="page">
  <div class="menu"></div>
  <div class="content"></div>
</div>
<footer></footer>
```

CSS CONTAINER QUERIES

@media: aplica estilos según condiciones del dispositivo.

<code>@media (<condición >)</code>	Si se cumple la condición, se aplican los estilos de su interior.
<code>@media not (<condición >)</code>	Si no se cumple la condición, se aplican los estilos de su interior.
<code>@media only (<condición >)</code>	Si se cumple la condición y es un navegador moderno, se aplican los estilos.
<code>@media (<condición >) and (<condición >)</code>	Se se cumplen ambas condiciones, se aplican los estilos.

CSS MEDIA FEATURES

Condiciones como width, orientation, prefers-color-scheme.

```
@media (aspect-ratio ≤ 6/2) {
  .container {
    background: red;
    color: white;
  }
}
```

CSS CONTAINER QUERIES

Estilos basados en el tamaño del contenedor, no del viewport.

```
.container {
  container-name: parent;
  container-type: inline-size;
}
```

PREFERENCIAS DE USUARIO

Detecta preferencias como modo oscuro o reducción de movimiento.

Preferencia de usuario	Valores	¿Qué detecta?
prefers-color-scheme	dark light	El esquema de colores (modo claro o oscuro).
prefers-reduced-motion	reduce no-preference	Animaciones reducidas (sin o con animaciones).
prefers-reduced-data	reduce no-preference	Ahorro de datos (consumo reducido de datos).
prefers-reduced-transparency	reduce no-preference	Transparencias reducidas (sin o con transparencias).
prefers-contrast	more less custom no-preference	Contraste preferido (Mayor o menor contraste).
forced-colors	active none	Forzar colores (requiere prefers-contrast a custom).

MEDIOS IMPRESOS

Estilos específicos para impresión con **@media print**

```
@media print {
  /* Reglas de CSS
}
```

MEDIOS PAGINADOS

```
@page {
  size: 8.27in 11.69in; /* DIN A4 */
  margin: 0.5in 0.5in 0.5in 0.5in;
}
```

Estilos para documentos paginados, como PDFs, usando **@page**.

LENGUAJE CSS

TRANSFORMACIONES Y DIBUJOS

TRANSFORMACIONES CSS

transform: aplica cambios visuales como mover, rotar, escalar o deformar elementos.

```
.element {
  width: 100px;
  height: 100px;
  background: indigo;
  transform: rotate(45deg);
}
```

TRANSLACIONES 2D

translate(x, y): desplaza elementos en ejes X e Y.

Funciones	Significado
translateX(x)	Traslada el elemento una distancia de SIZE x horizontalmente.
translateY(y)	Traslada el elemento una distancia de SIZE y verticalmente.
translate(x, y)	Propiedad de atajo de las dos anteriores.
translate(x)	Equivalente a translate(x, 0)

ROTACIONES 2D

rotate(ángulo): gira elementos alrededor de un punto.

ESCALADO 2D

scale(x, y): cambia el tamaño de los elementos en X e Y.

```
.image {
  transform: scaleX(-1); /* Imagen espejo en horizontal */
  transform: scaleY(-1); /* Imagen espejo en vertical (boca abajo) */

  transform: scale(-1); /* Equivalente a las dos anteriores a la vez */
}
```

DEFORMACIONES 2D

skew(x, y): inclina elementos en los ejes X e Y.

```
.element {
  transform: skewX(-5deg);
  transform: skewY(25deg);

  transform: skewX(5deg) skewY(15deg);
}
```

TRANSFORMACIONES 3D

rotateX(), rotateY(), rotateZ(): rotaciones en 3D.

```
.container {
  transform-style: preserve-3d;
  transform: rotateX(2deg) rotateY(2deg) rotateZ(2deg);
  perspective: 300px;
}
```

PERSPECTIVAS CSS

perspective: añade profundidad a las transformaciones 3D.

Propiedades	Formato	Significado
perspective	none SIZE distance	Punto de fuga para los elementos hijos.
perspective-origin	50% 50% POSITION posX posY	Punto de origen de la perspectiva.

DIBUJAR CON CSS

Crear ilustraciones usando solo HTML y CSS, sin imágenes externas.

Método	Tiempo de creación	Peso (Tamaño)
Dibujar visualmente y exportar en imagen	Bajo	Pesado
Dibujar visualmente y exportar en imagen vectorial	Bajo	Ligero-Medio
Crear directamente código SVG y CSS	Alto	Ligero
Crear directamente código HTML y CSS	Alto	Medio

ESTRUCTURA HTML

Organizar elementos HTML para formar la base del dibujo.



ESTRUCTURA CSS

Aplicar estilos para dar forma, color y posición a los elementos.

FORMAS CON CSS

Utilizar propiedades como **border-radius**, **transform** y **clip-path** para crear figuras.