



Introducción al Desarrollo Web

Ing. Marco Aedo López

Fundamentos del desarrollo Frontend

CSS

Tema 5

Objetivos

- Conocer conceptos básicos del CSS, su sintaxis, sus principales selectores y propiedades
- Entender el modelo de caja (Box Model)
- Manejar propiedades de posicionamiento y display
- Comprender la aplicación de CSS a formularios

10. Propiedad display: Flex

- Flexbox (CSS Flexible Box Layout Module) es un módulo de CSS para manejar un **layout unidimensional** (en fila o en columna) que facilita la organización, alineación y distribución de los elementos dentro de un contenedor
- Incluso cuando su tamaño es dinámico o desconocido
- Proporciona un modelo mucho más intuitivo y potente para crear **layouts responsivos**

10. Propiedad display: Flex

- Contenedor flexible (flex container)
 - Se define al aplicar
`display: flex;` o `display: inline-flex;` a un elemento
 - Sus hijos se convierten automáticamente en elementos flexibles (flex items)
- Eje principal y eje cruzado
 - Main axis (eje principal): definido por `flex-direction`. Puede ser horizontal (row) o vertical (column)
 - Cross axis (eje cruzado): perpendicular al eje principal

```
<div class="contenedor">
  <div class="item">1</div>
  <div class="item">2</div>
  <div class="item">3</div>
</div>

.contenedor {
  display: flex;          /* Activa flexbox */
  flex-direction: row;     /* Dirección horizontal (por defecto) */
  justify-content: space-around; /* Distribuye espacio entre ítems en eje principal */
  align-items: center;    /* Centra verticalmente en el eje cruzado */
  height: 150px;
  border: 2px solid black;
}

.item {
  flex-basis: 60px;        /* Tamaño base fijo */
  background: lightblue;
  text-align: center;
  padding: 10px;
}
```

1

2

3

1
2
3

```
<div class="galeria">
  <div class="tarjeta">Tarjeta 1</div>
  <div class="tarjeta">Tarjeta 2</div>
  <div class="tarjeta">Tarjeta 3</div>
  <div class="tarjeta">Tarjeta 4</div>
  <div class="tarjeta">Tarjeta 5</div>
  <div class="tarjeta">Tarjeta 6</div>
</div>

.galeria {
  display: flex;
  flex-flow: row wrap;      /* fila + salto de línea si no caben */
  justify-content: space-evenly; /* Distribuye el espacio de forma uniforme */
  align-items: stretch;      /* Todas las tarjetas tienen la misma altura */
  gap: 20px;                 /* Espacio entre tarjetas */
  padding: 20px;
  border: 2px solid ■#333;
  background: □#f8f8f8;
}

.tarjeta {
  flex-grow: 1;              /* Crecen para ocupar espacio sobrante */
  flex-shrink: 1;             /* Se encogen si falta espacio */
  flex-basis: 350px;          /* Ancho base de cada tarjeta */
  background: □lightblue;
  text-align: center;
  padding: 30px 10px;
  border-radius: 10px;
  box-shadow: 0 2px 6px □rgba(0,0,0,0.2);
}
```

Tarjeta 1

Tarjeta 2

Tarjeta 3

Tarjeta 4

Tarjeta 5

Tarjeta 6

10. Propiedad display: Grid

- Grid (CSS Grid Layout Module) es un módulo de CSS para manejar un **layout bidimensional** (filas y columnas)
- Se diferencia en que Flexbox es unidimensional
- Permite organizar elementos en **filas y columnas** de manera muy precisa y flexible

10. Propiedad display: Grid

- Contenedor Grid (grid container)
 - Se define aplicando `display: grid;` a un elemento
 - Sus hijos se convierten en ítems de la cuadrícula (grid items)
- Definición de filas y columnas
 - `grid-template-columns`
 - `grid-template-rows`
- Con líneas de cuadrícula (grid lines):
 - `grid-column: inicio / fin;`
 - `grid-row: inicio / fin;`

```
<div class="contenedor">
  <div class="item">1</div>
  <div class="item">2</div>
  <div class="item">3</div>
  <div class="item">4</div>
  <div class="item">5</div>
  <div class="item">6</div>
</div>
```

```
.contenedor {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr; /* 3 columnas iguales */
  grid-template-rows: auto auto;      /* filas automáticas */
  gap: 10px;
}

.item {
  background: lightblue;
  padding: 20px;
  text-align: center;
}
```

1

2

3

4

5

6

```
<div class="grid-container">
  <div class="item">1</div>
  <div class="item">2</div>
  <div class="item">3</div>
  <div class="item">4</div>
  <div class="item">5</div>
  <div class="item">6</div>
</div>
```

```
.grid-container {
  display: grid;
  grid-template-columns: repeat(3, 100px); /* 3 columnas */
  grid-template-rows: repeat(2, 80px);      /* 2 filas */
  gap: 5px;
  border: 2px solid black;
}

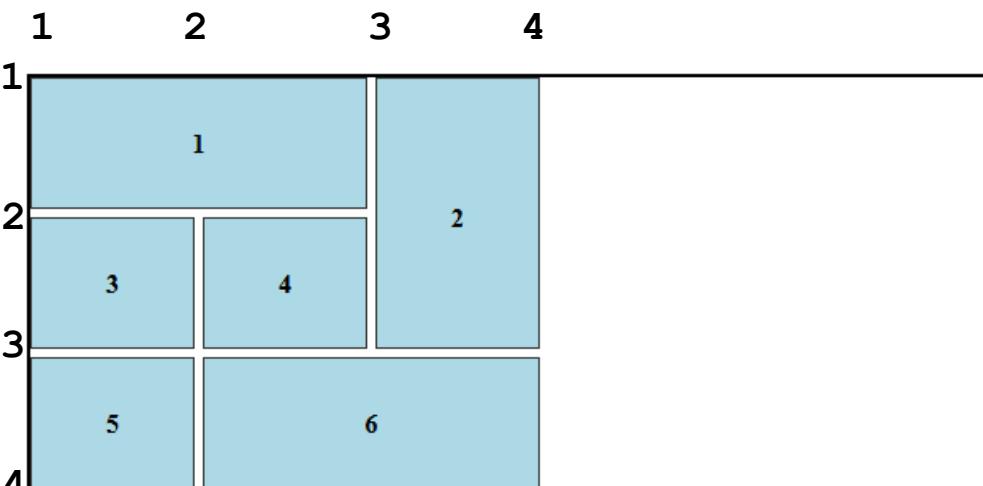
.item {
  background: lightblue;
  border: 1px solid #333;
  font-weight: bold;
  text-align: center; /* centrar horizontal */
}
```

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |

```

<div class="grid-container">
  <div class="item item1">1</div>
  <div class="item item2">2</div>
  <div class="item item3">3</div>
  <div class="item item4">4</div>
  <div class="item item5">5</div>
  <div class="item item6">6</div>
</div>

```



```

.grid-container {
  display: grid;
  grid-template-columns: repeat(3, 100px); /* 3 columnas */
  grid-template-rows: repeat(2, 80px);      /* 2 filas */
  gap: 5px;
  border: 2px solid black;
}

.item {
  background: lightblue;
  border: 1px solid #333;
  font-weight: bold;
  text-align: center; /* centrar horizontal */
}

/* Ejemplo: ocupar más espacio con grid-column y grid-row */
.item1 {
  grid-column: 1 / 3; /* Ocupa de la línea 1 a la 3 → dos columnas */
  grid-row: 1 / 2;    /* Solo la primera fila */
  line-height: 80px; /* Centrado vertical */
}

.item2 {
  grid-column: 3 / 4; /* Solo la tercera columna */
  grid-row: 1 / 3;    /* Ocupa ambas filas */
  line-height: 170px; /* Centrado vertical (2 filas = 160px + gap ≈ 170px) */
}

.item3 {
  line-height: 80px;
}

.item4 {
  line-height: 80px;
}

.item5 {
  line-height: 80px;
}

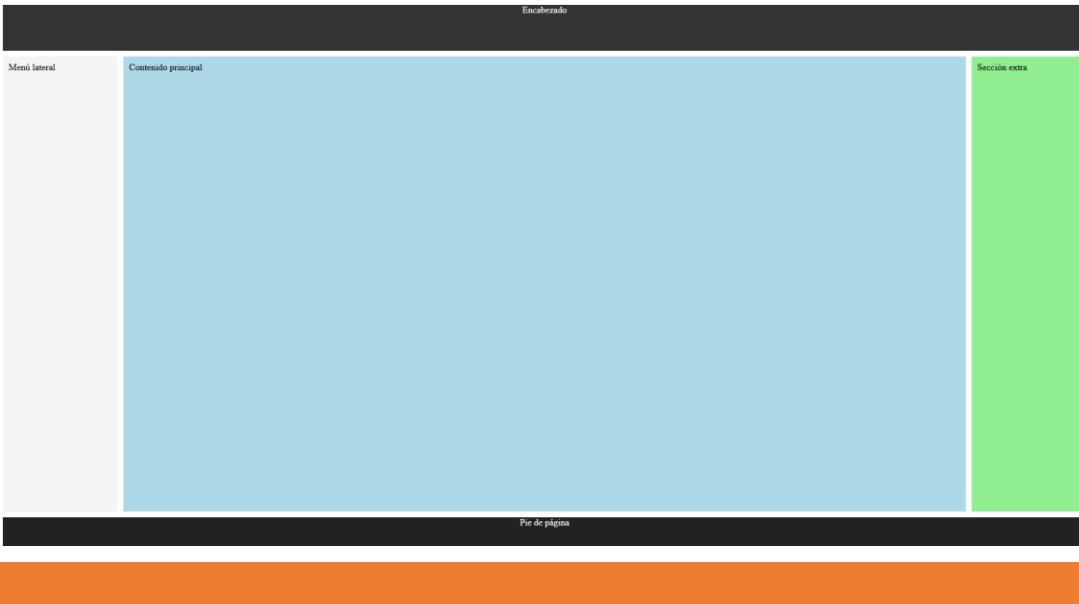
.item6 {
  grid-column: 2 / 4; /* Ocupa de la línea 2 a la 4 → dos columnas */
  grid-row: 3 / 4;    /* Solo la segunda fila */
  line-height: 80px;
}

```

```

<div class="contenedor">
  <header class="encabezado">Encabezado</header>
  <nav class="menu">Menú lateral</nav>
  <main class="contenido">Contenido principal</main>
  <aside class="extra">Sección extra</aside>
  <footer class="pie">Pie de página</footer>
</div>

```



```

.contenedor {
  display: grid;
  grid-template-columns: 200px 1fr 200px; /* 3 columnas */
  grid-template-rows: 80px 1fr 60px;      /* 3 filas */
  grid-template-areas:
    "encabezado encabezado encabezado"
    "menu contenido extra"
    "pie pie pie";
  gap: 10px;
  min-height: 100vh;
}
/* === ÁREAS DEFINIDAS === */
.encabezado {
  grid-area: encabezado;
  background: black #333;
  color: white;
  text-align: center;
}
.menu {
  grid-area: menu;
  background: #f4f4f4;
  padding: 10px;
}
.contenido {
  grid-area: contenido;
  background: lightblue;
  padding: 10px;
}
.extra {
  grid-area: extra;
  background: lightgreen;
  padding: 10px;
}
.pie {
  grid-area: pie;
  background: black #222;
  color: white;
  text-align: center;
}

```

11. Propiedad position

- Define cómo un elemento se posiciona dentro del flujo del documento
- Establece el punto de referencia para moverlo usando las propiedades top, right, bottom y left
- ¿Cómo quieres que calcule la ubicación de este elemento?

11. Propiedad position

| Valor | Descripción | Comportamiento con top, right, bottom, left |
|-------------------------------|---|---|
| static (<i>por defecto</i>) | El elemento se coloca según el flujo normal del documento | Ignora las propiedades de desplazamiento |
| relative | Mantiene su posición original en el flujo, pero puede moverse desde ahí usando top, left, etc. | Sí, relativo a su posición original |
| absolute | Se saca del flujo y se posiciona en relación con el contenedor posicionado más cercano (relative, absolute, fixed o sticky) | Sí, relativo al contenedor posicionado |
| fixed | Se saca del flujo y se posiciona en relación con la ventana del navegador (viewport). No se mueve al hacer scroll | Sí, relativo al viewport |
| sticky | Actúa como relative hasta que el scroll alcanza un límite definido por top, left, etc., y luego se "pega" como fixed | Sí, según su estado en el scroll |

11. Propiedad position

```
<h1>Ejemplo de position</h1>
```

```
<div class="contenedor">
    <div class="caja static">static</div>
    <div class="caja relative">relative</div>
    <div class="caja absolute">absolute</div>
    <div class="caja fixed">fixed</div>
    <div class="caja sticky">sticky</div>
</div>
```

```
<p class="alto">
    Desplázate hacia abajo para ver cómo se comporta fixed y sticky.
</p>
```

```
body {  
    font-family: Arial, sans-serif;  
    margin: 0;  
    padding: 20px;  
}  
  
.contenedor {  
    position: relative; /* Para que absolute y sticky tengan referencia */  
    height: 400px;  
    border: 2px solid ■#333;  
    padding: 10px;  
}  
  
.caja {  
    width: 100px;  
    height: 100px;  
    color: □white;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
    font-weight: bold;  
}  
  
/*para que se note el efecto*/  
.alto{  
    height: 1000px;  
}
```

```

/* 1. static */
.static {
  background: gray;
  position: static; /* valor por defecto */
}

/* 2. relative */
.relative {
  background: blue;
  position: relative;
  top: 20px; /* se mueve desde su posición original */
  left: 20px;
}

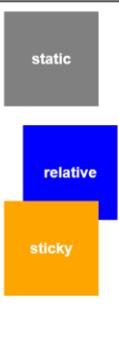
/* 3. absolute */
.absolute {
  background: green;
  position: absolute;
  top: 50px; /* se posiciona respecto al contenedor */
  right: 10px;
}

/* 4. fixed */
.fixed {
  background: red;
  position: fixed;
  top: 10px; /* se mantiene siempre visible en la ventana */
  right: 10px;
}

/* 5. sticky */
.sticky {
  background: orange;
  position: sticky;
  top: 0; /* se pega al llegar a la parte superior del contenedor */
}

```

Ejemplo de position



absolute

fixed



Desplázate hacia abajo para ver cómo se comporta fixed y sticky.

fixed

Desplázate hacia abajo para ver cómo se comporta fixed y sticky.



fixed

Desplázate hacia abajo para ver cómo se comporta fixed y sticky.

12. Pseudoclases

- Palabras clave que se añaden a los selectores para especificar un **estado especial** o condición de un elemento
- Permiten aplicar estilos específicos basados en dicho estado
- Empiezan con dos puntos (:) y se usan para aplicar estilos según interacciones, posición o condiciones del elemento

12. Pseudoclases: de interacción

| Pseudoclase | Descripción | Ejemplo |
|-------------|--|---|
| :hover | Aplica estilos cuando el puntero está sobre el elemento | <pre>a:hover { color: red; }</pre> |
| :active | Se activa mientras el elemento está siendo presionado | <pre>button:active { background: green; }</pre> |
| :focus | Se activa cuando el elemento tiene el foco (ej. un input seleccionado) | <pre>input:focus { border-color: blue; }</pre> |
| :visited | Se aplica a enlaces que ya fueron visitados | <pre>a:visited { color: purple; }</pre> |

```
<body>
  <a href="https://www.unsa.edu.pe">Ir a mi Universidad</a>
</body>
```

Ir a mi Universidad

Ir a mi Universidad

Ir a mi Universidad

Ir a mi Universidad

```
a {
  color: blue;
  text-decoration: none;
  font-size: 18px;
  padding: 5px 10px;
  border: 2px solid transparent;
  transition: all 0.3s ease;
}

/* Cuando el enlace ya fue visitado */
a:visited {
  color: purple;
}

/* Cuando el mouse está encima */
a:hover {
  color: white;
  background-color: blue;
  border-color: blue;
}

/* Cuando se hace clic y se mantiene presionado */
a:active {
  background-color: darkblue;
  border-color: darkblue;
}

/* Cuando el enlace tiene el foco (tab o clic) */
a:focus {
  outline: none;
  color: black;
  border-color: orange;
  background-color: lightyellow;
}
```

12. Pseudoclases: estructurales

| Pseudoclase | Descripción | Ejemplo CSS | Ejemplo HTML |
|----------------------|--|--|--|
| :first-child | Selecciona el primer hijo de un elemento padre | p:first-child { color: red; } | <div> <p>Rojo</p><p>Normal</p> |
| :last-child | Selecciona el último hijo de un elemento padre | p:last-child { color: blue; } | <div><p>Normal</p> <p>Azul</p></div> |
| :only-child | Selecciona un elemento si es el único hijo de su padre | p:only-child { font-weight: bold; } | <div> <p> Negrita (único hijo)</p></div> |
| :nth-child(n) | Selecciona el n-ésimo hijo de su padre (sin importar el tipo de elemento) | li:nth-child(2) { color: green; } | 1 2 Verde 3 |
| :nth-last-child(n) | Selecciona el n-ésimo hijo desde el final | li:nth-last-child(1) { color: purple; } | 12 3 Púrpura |
| :first-of-type | Selecciona el primer hijo de un tipo específico dentro del padre | p:first-of-type { text-decoration: underline; } | <div><h2>Título</h2><p> Subrayado </p> <p>Normal</p></div> |
| :last-of-type | Selecciona el último hijo de un tipo específico dentro del padre | p:last-of-type { font-style: italic; } | <div><p>Normal</p> <p> Cursiva </p></div> |
| :only-of-type | Selecciona un elemento si es el único de su tipo dentro del padre | p:only-of-type { background: yellow; } | <div><h2>Título</h2> <p> Fondo amarillo </p></div> |
| :nth-of-type(n) | Selecciona el n-ésimo elemento de un tipo específico dentro del padre | p:nth-of-type(2) { color: orange; } | <div> <p>1</p><p> Naranja </p><p>3</p></div> |
| :nth-last-of-type(n) | Selecciona el n-ésimo elemento de un tipo específico desde el final | p:nth-last-of-type(1) { color: brown; } | <div> <p>1</p><p>2</p><p> Café </p></div> |

```

<div>
  <p>Primer párrafo (rojo, subrayado)</p>
  <p>Segundo párrafo (fondo gris)</p>
  <p>Tercer párrafo (último → azul, cursiva, tamaño 20px)</p>
  <p> Este es <span>un span solitario (único hijo → fondo amarillo)</span> que resalta</p>
</div>

<ul>
  <li>Elemento 1</li>
  <li>Elemento 2 (segundo → negrita)</li>
  <li>Elemento 3 (último → verde)</li>
</ul>

```

Primer párrafo (rojo, subrayado)

Segundo párrafo (fondo gris)

Tercer párrafo (último → azul, cursiva, tamaño 20px)

Este es un span solitario (único hijo → fondo amarillo) que resalta

- Elemento 1
- Elemento 2 (segundo → negrita)
- Elemento 3 (último → verde)

```

/* 1. Primer hijo */
p:first-child {
  color: ■red;
}
/* 2. Último hijo */
p:last-child {
  color: ■blue;
}
/* 3. Único hijo */
span:only-child {
  background: ■yellow;
}
/* 4. Segundo hijo de cualquier tipo */
li:nth-child(2) {
  font-weight: bold;
}
/* 5. Último hijo contado desde el final */
li:nth-last-child(1) {
  color: ■green;
}
/* 6. Primer párrafo dentro del div */
p:first-of-type {
  text-decoration: underline;
}
/* 7. Último párrafo dentro del div */
p:last-of-type {
  font-style: italic;
}
/* 8. Único párrafo dentro del contenedor */
div:only-of-type p {
  border: 1px solid ■black;
}
/* 9. Segundo párrafo dentro del div */
p:nth-of-type(2) {
  background: ■lightgray;
}
/* 10. Último párrafo dentro del div contado desde el final */
p:nth-last-of-type(1) {
  font-size: 20px;
}

```

13. Pseudoelementos

- Permiten dar estilo a una parte específica de un elemento que normalmente no podrías seleccionar directamente con HTML
- Se escriben con dos puntos dobles ::

13. Pseudoelementos

| Pseudoelemento | Descripción | Ejemplo |
|----------------|---|--|
| ::first-line | Selecciona la primera línea de un texto en un elemento de bloque. | <pre>p::first-line { font-weight: bold; }</pre> |
| ::first-letter | Selecciona la primera letra del texto. Muy usado para iniciales decorativas. | <pre>p::first-letter { font-size: 2em; color: red; }</pre> |
| ::before | Inserta contenido antes del contenido real de un elemento. | <pre>h1::before { content: "★ "; color: gold; }</pre> |
| ::after | Inserta contenido después del contenido real de un elemento. | <pre>h1::after { content: " ✓ "; color: green; }</pre> |
| ::selection | Selecciona la parte del texto que el usuario resalta con el mouse. | <pre>p::selection { background: yellow; }</pre> |
| ::marker | Estiliza los marcadores de listas (,). | <pre>li::marker { color: red; font-size: 20px; }</pre> |
| ::placeholder | Da estilo al texto de placeholder en un <input> o <textarea>. | <pre>input::placeholder { color: gray; font-style: italic; }</pre> |
| ::backdrop | Estiliza el fondo que aparece detrás de un elemento en pantalla completa (fullscreen). | <pre>::backdrop { background: black; }</pre> |

<p>

Este es un párrafo de ejemplo para mostrar el uso de pseudoelementos en CSS.
Si seleccionas este texto, verás que cambia el color de fondo.

</p>

<h2>Subtítulo con extras</h2>

<input type="text" placeholder="Escribe tu nombre...">

- Elemento uno
- Elemento dos
- Elemento tres

Este es un párrafo de ejemplo para mostrar el uso de pseudoelementos en CSS. Si seleccionas este texto

* Subtítulo con extras ***

Escribe tu nombre...

- Elemento uno
- Elemento dos
- Elemento tres

```
/* Primera letra decorada */  
p::first-letter {  
    font-size: 2em;  
    font-weight: bold;  
    color: ■darkred;  
}  
  
/* Primera línea en cursiva */  
p::first-line {  
    font-style: italic;  
}  
  
/* Agregar contenido antes y después de un título */  
h2::before {  
    content: "* ";  
    color: ■gold;  
}  
  
h2::after {  
    content: " ***";  
    color: ■green;  
}  
  
/* Personalizar placeholder */  
input::placeholder {  
    color: ■gray;  
    font-style: italic;  
}  
  
/* Estilizar marcador de listas */  
li::marker {  
    color: ■blue;  
    font-size: 1.2em;  
}  
  
/* Personalizar selección de texto */  
p::selection {  
    background: ■yellow;  
    color: ■black;  
}
```

14. Propiedad transition

- Permite que ciertos cambios en los estilos se realicen **de forma gradual** durante un intervalo de tiempo, en lugar de aplicarse de manera instantánea
- Es como ponerle “animación” a un cambio para crear **transiciones suaves**

`transition`: propiedad duración función_de_tiempo retraso;

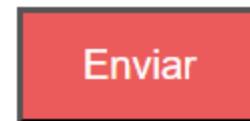
`transition`: background-color 0.5s ease;

`transition`: width 1s ease;

14. Propiedad transition

```
button {  
    background-color: #rgb(21, 228, 239);  
    color: #white;  
    padding: 10px 20px;  
    transition: background-color 0.5s ease;  
}
```

```
button:hover {  
    background-color: #rgb(236, 91, 91);  
}
```



15. Propiedad transform

- **Aplica transformaciones visuales** a un elemento, modificando su forma, tamaño, rotación o posición **sin alterar su flujo en el documento**
- Puede mover, rotar, escalar o inclinar un elemento, pero su “caja” original permanece en el mismo lugar

`transform: función1(valor) función2(valor) ...;`

15. Propiedad transform

| Función | Ejemplo | Descripción |
|--------------------------|-----------------------|---|
| translate(x, y) | translate(50px, 20px) | Mueve el elemento en el eje X y Y |
| translateX(n) | translateX(100px) | Mueve solo en horizontal |
| translateY(n) | translateY(-50px) | Mueve solo en vertical |
| rotate(ángulo) | rotate(45deg) | Rota el elemento alrededor de su centro |
| scale(x, y) | scale(1.5, 2) | Escala el tamaño en horizontal y vertical |
| scaleX(n) | scaleX(2) | Escala solo en horizontal |
| scaleY(n) | scaleY(0.5) | Escala solo en vertical |
| skew(x, y) | skew(20deg, 10deg) | Inclina el elemento en X e Y |
| skewX(n) | skewX(15deg) | Inclina solo en horizontal |
| skewY(n) | skewY(-15deg) | Inclina solo en vertical |
| matrix(a, b, c, d, e, f) | — | Aplica una transformación combinada usando una matriz 2D (avanzado) |

15. Propiedad transform

```
<p>
    Lorem ipsum, dolor sit amet consectetur adipisicing elit. Repudiandae, exercitationem?
</p>
```

“Lorem ipsum, dolor sit amet consectetur adipisicing elit. Repudiandae, exercitationem?”

```
<p class="caja">
    Lorem ipsum, dolor sit amet consectetur adipisicing elit. Repudiandae, exercitationem?
</p>
```

```
.caja {  
    width: 100px;  
    height: 100px;  
    background-color: royalblue;  
    transform: rotate(45deg) scale(1.2) translateX(200px);  
}
```



16. Tabla con CSS

Países y Capitales

| País | Capital |
|-----------|------------------|
| Perú | Lima |
| Argentina | Buenos Aires |
| México | Ciudad de México |
| España | Madrid |
| Japón | Tokio |

Países y Capitales

| País | Capital |
|-----------|------------------|
| Perú | Lima |
| Argentina | Buenos Aires |
| México | Ciudad de México |
| España | Madrid |
| Japón | Tokio |

```
body {
    font-family: Arial, sans-serif;
    padding: 20px;
    background-color: #f4f6f8;
}

h2 {
    text-align: center;
    color: #333;
}

table {
    width: 60%;
    margin: auto;
    border-collapse: collapse;
    background: white;
    box-shadow: 0 2px 8px rgba(0,0,0,0.1);
}

th, td {
    padding: 12px;
    border: 1px solid #ddd;
    text-align: left;
}

th {
    background-color: #3498db;
    color: white;
}

tr:nth-child(even) {
    background-color: #f2f2f2;
}

tr:hover {
    background-color: #d6eaf8;
}
```

17. Formulario con CSS

Formulario de Registro

Nombre: Correo electrónico: País: Mensaje:

Formulario de Registro

Nombre:

Correo electrónico:

País:

Mensaje:

```
<body>
  <form>
    <h2>Formulario de Registro</h2>

    <label for="nombre">Nombre:</label>
    <input type="text" id="nombre" name="nombre" placeholder="Ingresa tu nombre" required>

    <label for="email">Correo electrónico:</label>
    <input type="email" id="email" name="email" placeholder="ejemplo@correo.com" required>

    <label for="pais">País:</label>
    <select id="pais" name="pais" required>
      <option value="">Seleccione...</option>
      <option value="peru">Perú</option>
      <option value="mexico">México</option>
      <option value="argentina">Argentina</option>
      <option value="espana">España</option>
    </select>

    <label for="mensaje">Mensaje:</label>
    <textarea id="mensaje" name="mensaje" rows="4" placeholder="Escribe tu mensaje..."></textarea>

    <button type="submit">Enviar</button>
  </form>
```

```
body {
    font-family: Arial, sans-serif;
    background-color: #f4f6f8;
    padding: 20px;
}

form {
    background-color: #fff;
    max-width: 400px;
    margin: auto;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0,0,0,0.1);
}

h2 {
    text-align: center;
    color: #333;
}

label {
    display: block;
    margin-top: 10px;
    font-weight: bold;
}

input, select, textarea, button {
    width: 100%;
    padding: 8px;
    margin-top: 5px;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}

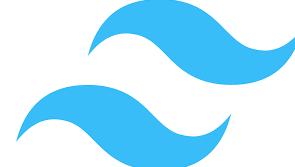
input:focus, select:focus, textarea:focus {
    border-color: #3498db;
    outline: none;
}

button {
    background-color: #3498db;
    color: white;
    border: none;
    margin-top: 15px;
    cursor: pointer;
}

button:hover {
    background-color: #217dbb;
}
```

18. Frameworks

- Conjunto de archivos, reglas y utilidades ya predefinidas en CSS que permiten ahorrar tiempo al diseñar y maquetar sitios web
- **Bootstrap**
 - El más popular y usado
 - Incluye grid basado en flexbox y utilidades rápidas
 - Viene con componentes como menús, tarjetas, modales, alertas
- **Tailwind**
 - En lugar de usar componentes prefabricados, ofrece **clases pequeñas** que construyen diseños a medida
 - Muy flexible, ideal para proyectos modernos y personalización total



18. Frameworks

- Ventajas
 - Acelera el desarrollo
 - Garantiza compatibilidad entre navegadores
 - Diseño responsive de forma automática
 - Permite enfocarse más en la lógica del sitio que en los detalles del estilo
- Desventajas
 - Puede agregar código innecesario (peso extra)
 - Riesgo de que todos los sitios se vean iguales
 - Requiere aprender la convención del framework
 - Menor personalización si se usa “tal cual”

19. Ejemplos de CSS

```
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sin CSS</title>

</head>
<body>
    <h1>Mi Página Web</h1>
    <p>Bienvenido a mi página. Aquí aprenderás sobre el poder del CSS.</p>
    <a href="#">Leer más</a>

    <h2>Artículos</h2>
    <ul>
        <li>HTML Básico</li>
        <li>Introducción a CSS</li>
        <li>JavaScript para Principiantes</li>
    </ul>
</body>
</html>
```

Mi Página Web

Bienvenido a mi página. Aquí aprenderás sobre el poder del CSS.

[Leer más](#)

Artículos

- HTML Básico
- Introducción a CSS
- JavaScript para Principiantes

```
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Con CSS</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <h1>Mi Página Web</h1>
    <p>Bienvenido a mi página. Aquí aprenderás sobre el poder del CSS.</p>
    <a href="#">Leer más</a>

    <h2>Artículos</h2>
    <ul>
        <li>HTML Básico</li>
        <li>Introducción a CSS</li>
        <li>JavaScript para Principiantes</li>
    </ul>
</body>
</html>
```

```
body {  
    font-family: Arial, sans-serif;  
    background-color: #f4f4f4;  
    margin: 0;  
    padding: 0;  
}  
  
h1 {  
    background-color: #007BFF;  
    color: white;  
    padding: 20px;  
    text-align: center;  
}  
  
p {  
    font-size: 18px;  
    text-align: center;  
    margin: 20px;  
}  
  
a {  
    background-color: #28a745;  
    color: white;  
    padding: 10px 20px;  
    text-decoration: none;  
    border-radius: 5px;  
}  
  
a:hover {  
    background-color: #218838;  
}  
  
h2 {  
    margin-top: 40px;  
    text-align: center;  
    color: #333;  
}  
  
ul {  
    max-width: 400px;  
    margin: 20px auto;  
    padding: 0;  
    list-style: none;  
}  
  
li {  
    background: white;  
    padding: 10px;  
    margin: 5px 0;  
    border-radius: 5px;  
    box-shadow: 0px 2px 5px rgba(0,0,0,0.1);  
}
```

Mi Página Web

Bienvenido a mi página. Aquí aprenderás sobre el poder del CSS.

Leer más

Artículos

HTML Básico

Introducción a CSS

JavaScript para Principiantes

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sin CSS</title>

</head>
<body>
    <header>
        <h1>Noticias de Tecnología</h1>
    </header>

    <main>
        <article>
            <h2>Artículo 1</h2>
            <p>La inteligencia artificial está cambiando el mundo.</p>
            <footer>
                <a href="#">Leer más</a>
            </footer>
        </article>

        <article>
            <h2>Artículo 2</h2>
            <p>Se lanza un nuevo framework de JavaScript muy prometedor.</p>
            <footer>
                <a href="#">Leer más</a>
            </footer>
        </article>

        <article>
            <h2>Artículo 3</h2>
            <p>CSS sigue evolucionando con nuevas funciones y propiedades.</p>
            <footer>
                <a href="#">Leer más</a>
            </footer>
        </article>
    </main>
</body>
</html>
```

Noticias de Tecnología

Artículo 1

La inteligencia artificial está cambiando el mundo.

[Leer más](#)

Artículo 2

Se lanza un nuevo framework de JavaScript muy prometedor.

[Leer más](#)

Artículo 3

CSS sigue evolucionando con nuevas funciones y propiedades.

[Leer más](#)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Con CSS</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <header>
        <h1>Noticias de Tecnología</h1>
    </header>

    <main>
        <article>
            <h2>Artículo 1</h2>
            <p>La inteligencia artificial está cambiando el mundo.</p>
            <footer>
                <a href="#">Leer más</a>
            </footer>
        </article>

        <article>
            <h2>Artículo 2</h2>
            <p>Se lanza un nuevo framework de JavaScript muy prometedor.</p>
            <footer>
                <a href="#">Leer más</a>
            </footer>
        </article>

        <article>
            <h2>Artículo 3</h2>
            <p>CSS sigue evolucionando con nuevas funciones y propiedades.</p>
            <footer>
                <a href="#">Leer más</a>
            </footer>
        </article>
    </main>
</body>
</html>
```

```
body {
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
    margin: 0;
    padding: 0;
}

header {
    background-color: #007BFF;
    color: white;
    padding: 20px 0;
    text-align: center;
}

main {
    text-align: center;
    padding: 20px;
}

article {
    display: inline-block;
    vertical-align: top;
    background: white;
    border-radius: 10px;
    box-shadow: 0px 4px 10px rgba(0,0,0,0.1);
    width: 300px;
    margin: 10px;
    padding: 20px;
    transition: transform 0.3s, box-shadow 0.3s;
}

article:hover {
    transform: translateY(-5px);
    box-shadow: 0px 6px 15px rgba(0,0,0,0.2);
}

article h2 {
    margin-top: 0;
    color: #007BFF;
}

article p {
    color: #651313;
    line-height: 1.5;
}

footer a {
    display: inline-block;
    margin-top: 10px;
    background-color: #28a745;
    color: white;
    padding: 8px 15px;
    border-radius: 5px;
    text-decoration: none;
}

footer a:hover {
    background-color: #218838;
}
```

Noticias de Tecnología

Artículo 1

La inteligencia artificial está cambiando el mundo.

[Leer más](#)

Artículo 2

Se lanza un nuevo framework de JavaScript muy prometedor.

[Leer más](#)

Artículo 3

CSS sigue evolucionando con nuevas funciones y propiedades.

[Leer más](#)

Gracias!!!