

## Guía de Laboratorio 4

### Propiedad display

#### 1) Objetivo del laboratorio

Comprender y aplicar los diferentes valores de la propiedad display en CSS para controlar el flujo, la alineación y la disposición de los elementos HTML en una página web.

#### 2) Desarrollo

##### 1. `display: block;`

Los elementos **ocupan todo el ancho disponible y comienzan en una nueva línea**.

Ejemplo:

```
<h3>Ejemplo display: block</h3>
<div class="block">Bloque 1</div>
<div class="block">Bloque 2</div>

<style>
.block {
    display: block;
    background: lightblue;
    padding: 10px;
    margin: 5px 0;
}
</style>
```

##### 2. `display: inline;`

Los elementos **no inician una nueva línea y solo ocupan el espacio necesario**.

Ejemplo:

```
<h3>Ejemplo display: inline</h3>
<span class="inline">Texto 1</span>
<span class="inline">Texto 2</span>

<style>
.inline {
    display: inline;
    background: lightgreen;
    padding: 5px;
    margin: 5px;
}
</style>
```

### 3. **display: inline-block;**

Se comporta **como inline** (no salta de línea), pero **permite ancho, alto, márgenes y paddings**.

```
<h3>Ejemplo display: inline-block</h3>
<div class="inline-block">Caja 1</div>
<div class="inline-block">Caja 2</div>

<style>
.inline-block {
    display: inline-block;
    background: gold;
    width: 100px;
    height: 50px;
    margin: 5px;
    text-align: center;
    line-height: 50px;
}
</style>
```

### 4. **display: none;**

El elemento **no se muestra en la página, y no ocupa espacio**.

```
<h3>Ejemplo display: none</h3>
<div class="visible">Visible</div>
<div class="hidden">Oculto</div>

<style>
.visible {
    background: lightcoral;
    padding: 10px;
}
.hidden {
    display: none;
}
</style>
```

### 5. **display: flex;**

Crea un **contenedor flexible**, Permite alinear y distribuir los elementos hijos fácilmente.

```
<h3>Ejemplo display: flex</h3>
<div class="flex-container">
    <div>Caja 1</div>
    <div>Caja 2</div>
    <div>Caja 3</div>
</div>

<style>
.flex-container {
    display: flex;
    background: #e0e0e0;

```

```
gap: 10px;
justify-content: space-around;
}
.flex-container div {
background: lightseagreen;
padding: 20px;
color: white;
}
</style>
```

## 6. **display: grid;**

Crea un **contenedor en cuadrícula**, ideal para maquetar layouts.

```
<h3>Ejemplo display: grid</h3>
<div class="grid-container">
<div>1</div>
<div>2</div>
<div>3</div>
<div>4</div>
</div>

<style>
.grid-container {
display: grid;
grid-template-columns: repeat(2, 100px);
gap: 10px;
background: #ddd;
padding: 10px;
}
.grid-container div {
background: steelblue;
color: white;
text-align: center;
padding: 20px;
}
</style>
```

## 3) Código completo

```
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Ejemplos de display en CSS</title>
<style>
body {
font-family: Arial, sans-serif;
margin: 40px;
background-color: #f5f5f5;
```

```
color: #333;
}

h1 {
    text-align: center;
    color: #0077aa;
}

section {
    background: white;
    border-radius: 10px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    padding: 20px;
    margin-bottom: 30px;
}

h2 {
    color: #005b88;
    border-bottom: 2px solid #0077aa;
    padding-bottom: 5px;
}

.example {
    margin-top: 10px;
    margin-bottom: 15px;
}

code {
    background: #eee;
    padding: 2px 4px;
    border-radius: 4px;
    color: #c7254e;
    font-family: monospace;
}

p {
    margin-bottom: 10px;
}

/* 1 BLOCK */
.block {
    display: block;
    background: lightblue;
    padding: 10px;
    margin: 5px 0;
    border: 1px solid #0088cc;
}
```

```
/* 2 INLINE */
.inline {
  display: inline;
  background: lightgreen;
  padding: 5px;
  margin: 5px;
  border: 1px solid #009933;
}

/* 3 INLINE-BLOCK */
.inline-block {
  display: inline-block;
  background: gold;
  width: 100px;
  height: 50px;
  margin: 5px;
  text-align: center;
  line-height: 50px;
  border: 1px solid #cc9900;
}

/* 4 NONE */
.visible {
  background: lightcoral;
  padding: 10px;
  border: 1px solid #cc0000;
}

.hidden {
  display: none;
}

/* 5 FLEX */
.flex-container {
  display: flex;
  background: #e0e0e0;
  gap: 10px;
  justify-content: space-around;
  padding: 10px;
  border: 1px solid #999;
}

.flex-container div {
  background: lightseagreen;
  padding: 20px;
  color: white;
```

```
border-radius: 5px;
text-align: center;
flex: 1;
}

/* 6 GRID */
.grid-container {
  display: grid;
  grid-template-columns: repeat(2, 100px);
  gap: 10px;
  background: #ddd;
  padding: 10px;
  border: 1px solid #999;
  justify-content: center;
}

.grid-container div {
  background: steelblue;
  color: white;
  text-align: center;
  padding: 20px;
  border-radius: 5px;
}
</style>
</head>
<body>

<h1>Ejemplos de la propiedad <code>display</code> en CSS</h1>

<!-- 1 BLOCK -->
<section>
  <h2>1. display: block;</h2>
  <p>Los elementos de tipo <code>block</code> ocupan todo el ancho disponible y comienzan en una nueva línea.</p>
  <div class="example">
    <div class="block">Bloque 1</div>
    <div class="block">Bloque 2</div>
  </div>
</section>

<!-- 2 INLINE -->
<section>
  <h2>2. display: inline;</h2>
  <p>Los elementos <code>inline</code> no iniciaran una nueva línea y solo ocupan el espacio necesario.</p>
  <div class="example">
    <span class="inline">Texto 1</span>
  </div>
</section>
```

```
<span class="inline">Texto 2</span>
<span class="inline">Texto 3</span>
</div>
</section>

<!-- 3 INLINE-BLOCK -->
<section>
    <h2>3. display: inline-block;</h2>
    <p>Se comporta como <code>inline</code> (no salta de línea), pero permite
definir ancho y alto.</p>
    <div class="example">
        <div class="inline-block">Caja 1</div>
        <div class="inline-block">Caja 2</div>
        <div class="inline-block">Caja 3</div>
    </div>
</section>

<!-- 4 NONE -->
<section>
    <h2>4. display: none;</h2>
    <p>El elemento no se muestra en la página ni ocupa espacio.</p>
    <div class="example">
        <div class="visible">Elemento visible</div>
        <div class="hidden">Elemento oculto</div>
        <p><em>(El segundo elemento está oculto con <code>display:
none;</code>)</em></p>
    </div>
</section>

<!-- 5 FLEX -->
<section>
    <h2>5. display: flex;</h2>
    <p>Permite distribuir los elementos hijos de forma flexible y alinearlos
fácilmente.</p>
    <div class="example flex-container">
        <div>Caja 1</div>
        <div>Caja 2</div>
        <div>Caja 3</div>
    </div>
</section>

<!-- 6 GRID -->
<section>
    <h2>6. display: grid;</h2>
    <p>Permite crear diseños en cuadrícula, ideales para estructuras complejas.</p>
    <div class="example grid-container">
        <div>1</div>
```

```
<div>2</div>
<div>3</div>
<div>4</div>
</div>
</section>

</body>
</html>
```

### 3) Ejercicio: Mini-landing “Cursos Tech” con display y position

#### Objetivo

Construir una página sencilla con:

- **Header fijo** (position: fixed)
- **Hero** con un botón posicionado (position: absolute)
- **Tarjetas en columna** usando **display: inline-block** (no flex/grid)
- **Insignias** en esquina con **position: absolute**
- **Etiqueta “Sección”** que se pega al hacer scroll con **position: sticky**
- **Botón “volver arriba”** con **position: fixed**
- Mostrar/ocultar texto con **display: none/block** usando el *checkbox hack*

#### Reglas

- No usar display: flex ni display: grid.
- Puedes usar: block, inline, inline-block, none y position: static/relative/absolute/fixed/sticky.
- Sin librerías externas.

#### Resultado

The screenshot shows the homepage of the CursosTech website. At the top, there's a blue header bar with the logo "CursosTech" on the left and navigation links "Inicio", "Cursos", and "Contacto" on the right. Below the header is a hero section with the title "Aprende hoy, construye mañana" and a subtitle "Programación, datos y seguridad con proyectos reales." A blue "Ver cursos" button is centered below the subtitle. Underneath the hero section is a black banner labeled "Cursos destacados". Three course cards are displayed: "Desarrollo Web" (HTML, CSS y JS desde cero.), "Ciencia de Datos" (Python, análisis y visualización), and "Seguridad Web" (Vulnerabilidades y hardening). Each card has a "Nuevo" badge, a "Top" badge, or a "-20%" badge, and a "Ver más" button. Below the hero section, there's a "Contacto" form with an input field for email and a "Enviar" button. A small note says "escribe a contacto@cursostech.test".

## Código HTML

```
<!doctype html>
<html lang="es">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Cursos Tech – Demo Display & Position</title>
  <link rel="stylesheet" href="ejercicio.css" />
</head>
<body>
  <!-- Header fijo -->
  <header class="site-header">
    <h1 class="logo">CursosTech</h1>
    <nav class="nav">
      <a href="#hero">Inicio</a>
      <a href="#cursos">Cursos</a>
      <a href="#contacto">Contacto</a>
    </nav>
  </header>

  <!-- Hero -->
  <section id="hero" class="hero">
    <h2>Aprende hoy, construye mañana</h2>
    <p>Programación, datos y seguridad con proyectos reales.</p>
    <a class="cta" href="#cursos">Ver cursos</a>
  </section>
```

```
<!-- Separador pegajoso -->
<div class="section-tag">Cursos destacados</div>

<!-- Tarjetas -->
<main id="cursos" class="cards">
    <!-- Card 1 -->
    <article class="card">
        <span class="badge">Nuevo</span>
        <h3>Desarrollo Web</h3>
        <p>HTML, CSS y JS desde cero.</p>

        <label class="toggle">
            <input type="checkbox" />
            <span class="toggle-text">Ver más</span>
            <div class="more">
                <p>Incluye prácticas, retos semanales y feedback.</p>
            </div>
        </label>
    </article>

    <!-- Card 2 -->
    <article class="card">
        <span class="badge">Top</span>
        <h3>Ciencia de Datos</h3>
        <p>Python, análisis y visualización.</p>

        <label class="toggle">
            <input type="checkbox" />
            <span class="toggle-text">Ver más</span>
            <div class="more">
                <p>Casos reales con datasets públicos.</p>
            </div>
        </label>
    </article>

    <!-- Card 3 -->
    <article class="card">
        <span class="badge">-20%</span>
        <h3>Seguridad Web</h3>
        <p>Vulnerabilidades y hardening.</p>

        <label class="toggle">
            <input type="checkbox" />
            <span class="toggle-text">Ver más</span>
            <div class="more">
                <p>Laboratorios de práctica guiados.</p>
            </div>
        </label>
    </article>

```

```
</div>
</label>
</article>
</main>

<!-- Contacto (para ancla de navegación) -->
<section id="contacto" class="contacto">
  <h2>Contacto</h2>
  <p>escribe a contacto@cursostech.test</p>
</section>

<!-- Botón volver arriba -->
<a href="#hero" class="go-top" aria-label="Volver arriba">▲</a>
</body>
</html>
```

## Código CSS

```
***** Reset mínimo *****/
* { box-sizing: border-box; }
body { margin: 0; font-family: system-ui, sans-serif; line-height: 1.5; color: #222; }

***** Header fijo *****/
.site-header {
  position: fixed; /* permanece visible */
  top: 0; left: 0; right: 0;
  background: #0b5cff;
  color: #fff;
  padding: 10px 16px;
  border-bottom: 2px solid #06369b;
  z-index: 10;
}
.logo { display: inline; font-size: 18px; margin: 0; }
.nav { display: inline; float: right; }
.nav a {
  color: #fff; text-decoration: none;
  display: inline-block; /* para padding horizontal */
  padding: 6px 10px;
}
.nav a:hover { text-decoration: underline; }

***** Hero *****/
.hero {
  padding: 120px 16px 64px; /* espacio por header fijo */
  text-align: center;
```

```
background: #eef4ff;
position: relative; /* referencia para .cta absolute */
}
.hero h2 { margin: 0 0 8px; }
.hero p { margin: 0 0 24px; color: #444; }
.hero .cta {
  position: absolute; /* ejemplo de absolute */
  left: 50%; transform: translateX(-50%);
  bottom: 16px;
  display: inline-block;
  background: #0b5cff; color: #fff; text-decoration: none;
  padding: 10px 16px; border-radius: 6px;
}

/***** Etiqueta de sección pegajosa *****/
.section-tag {
  position: sticky; /* se "pega" al top al hacer scroll */
  top: 56px; /* debajo del header fijo */
  background: #111; color: #fff;
  padding: 8px 16px;
  border-top: 1px solid #333;
  border-bottom: 1px solid #333;
  z-index: 5; /* encima del contenido normal */
}

/***** Tarjetas con inline-block *****/
.cards {
  padding: 24px 16px 80px;
  text-align: center; /* para centrar el flujo inline-block */
}

.card {
  display: inline-block; /* SIN flex/grid */
  width: 300px; /* ancho fijo de demo */
  margin: 8px;
  text-align: left;
  background: #fff;
  border: 1px solid #ddd;
  border-radius: 10px;
  padding: 16px;
  vertical-align: top; /* alinear alturas distintas */
  position: relative; /* contenedor para .badge absolute */
}

.badge {
  position: absolute; /* esquina de la tarjeta */
  top: -10px; right: -10px;
```

```
background: #ff3b3b; color: #fff;
padding: 6px 10px;
font-size: 12px;
border-radius: 999px;
}

***** Mostrar/ocultar con display (checkbox hack) *****/
.toggle { display: block; margin-top: 12px; }
.toggle input { display: none; } /* ocultamos el checkbox */
.toggle .toggle-text {
  display: inline-block;
  background: #f0f0f0; padding: 6px 10px; border-radius: 6px; cursor: pointer;
}
.toggle .more { display: none; margin-top: 8px; color: #444; }
.toggle input:checked + .toggle-text + .more { display: block; }

***** Contacto *****/
.contacto { padding: 48px 16px; background: #fafafa; }

***** Botón volver arriba (fixed) *****/
.go-top {
  position: fixed; /* siempre visible */
  right: 16px; bottom: 16px;
  text-decoration: none;
  background: #111; color: #fff;
  padding: 10px 12px; border-radius: 8px;
  opacity: 0.8;
}
.go-top:hover { opacity: 1; }

***** Pequeños ajustes responsivos sin flex/grid *****/
@media (max-width: 980px) {
  .card { width: 45%; } /* 2 columnas usando inline-block */
}
@media (max-width: 640px) {
  .card { width: 100%; } /* 1 columna (stack por block width) */
  .nav { float: none; display: block; margin-top: 6px; }
}
```