

TAREA 2 – UNIDAD 4

OBJETIVOS

- **Consolidar el manejo de css**
- **Trabajar con layouts (de momento flex)**
- **Trabajar con estilos adicionales y efectos**

INTRODUCCIÓN

1. Font-face

@font-face te permite cargar fuentes personalizadas (archivos .woff2, .woff, etc.) y usarlas en tu sitio con font-family como si fueran fuentes del sistema. Es clave para dar identidad tipográfica, controlar pesos, y asegurar consistencia visual.

Sintaxis básica

css

Copy

```
@font-face {  
    font-family: "MiFuente";  
    src: url("../fonts/MiFuente-Regular.woff2") format("woff2"),  
         url("../fonts/MiFuente-Regular.woff") format("woff");  
    font-weight: 400;  
    font-style: normal;  
    font-display: swap;  
}
```

- **font-family:** el nombre con el que referirás a la fuente en tu CSS (p. ej. body { font-family: "MiFuente", system-ui, sans-serif; }).
- **src:** rutas a los archivos de fuente y su formato. WOFF2 es estándar moderno; WOFF es fallback.
- **font-weight:** peso específico que aporta este archivo (400 = normal, 700 = bold). Si es una fuente variable, se usa un rango.
- **font-style:** normal o italic (o oblique).
- **font-display:** cómo renderiza el texto mientras carga la fuente (recomendado: swap).

2. Degradados

Los degradados en CSS son imágenes generadas por el navegador que puedes usar donde colocarías una imagen de fondo (background-image), en bordes con trucos, o incluso para colorear texto mediante background-clip. No requieren archivos externos y son muy personalizables.

- Usa porcentajes para posiciones relativas al tamaño del elemento.
- Usa calc() si necesitas combinar valores.
- turn es otra unidad angular: 1turn = 360deg (ej. from 0.25turn).

Hay tres tipos principales:

- **linear-gradient():** Crea una transición de color a lo largo de una línea recta.

Sintaxis: linear-gradient([ángulo | dirección], parada1, parada2, ...)

Dirección:

- **Con ángulo:**
 - 0deg apunta hacia arriba;
 - 90deg a la derecha;
 - 180deg hacia abajo;
 - 270deg a la izquierda.
- **Con palabras clave:**
 - to right
 - to left
 - to bottom right, etc.

- **radial-gradient():** Genera una transición circular o elíptica desde un punto central o desplazado.

Sintaxis: radial-gradient([forma tamaño] at posición, parada1, parada2, ...)

Forma:

- circle o ellipse

Tamaños:

- closest-side,
- farthest-side
- closest-corner
- farthest-corner

Posición:

- center
- left top
- right 30% 20%, etc.

- **conic-gradient()** : Crea un degradado con forma de “pastel” alrededor de un punto, cambiando el color angularmente.
Sintaxis: conic-gradient([from ángulo] at posición, parada1, parada2, ...)

Puedes combinar varias **capas** separadas por comas. La primera es la más cercana al usuario.

Ejemplo:

background:

```
background-color:  
    radial-gradient(1200px 600px at 10% -10%, #1e293b 0%, transparent  
60%),  
    radial-gradient(800px 500px at 90% 10%, #0ea5e9 0%, transparent  
55%),  
    linear-gradient(180deg, #020617 0%, #0b1220 60%, #0f172a 100%);
```

3. Posicionamiento y superposición

Una de las prácticas más comunes es colocar un elemento interno con position: absolute mientras que su elemento contenedor tiene position: relative

- **Posición absoluta:** Permite ubicar las formas en ubicaciones específicas del contenedor.
 - **Z-index:** Controla la superposición de elementos cuando se superponen.

```
.circulo {  
    z-index: 1;  
}  
}
```

```
.cuadrado {  
    z-index: 2; /* el cuadrado se posicionará delante del círculo */  
}
```

4. Estilos adicionales y efectos

- **Sombras y bordes:** Utiliza box-shadow para crear profundidad.
 - **Gradientes:** Aplica gradientes de fondo para dar un toque artístico.
 - **Transformaciones:** Con transform puedes rotar, escalar o desplazar elementos.

```
.circulo {  
    box-shadow: 0 4px 8px rgba(0,0,0,0.3);  
    transform: scale(1.1);  
}
```

```
.cuadrado {  
    border: 2px solid #c0392b;  
    transform: rotate(15deg);  
}
```

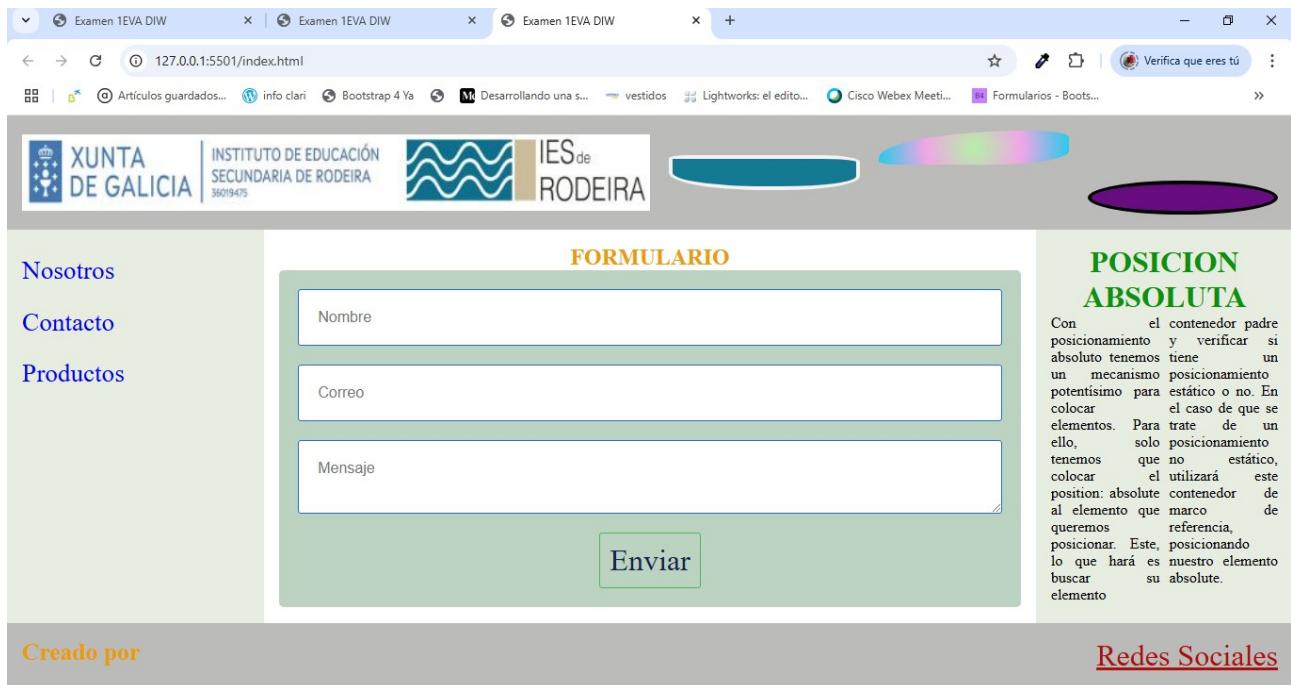
5. Uso de pseudo-elementos

- **Decoración extra:** Se pueden usar ::before y ::after para agregar detalles sin necesidad de más HTML.
- **Ejemplo:** Añadir un “borde interior” o una sombra sin afectar la estructura principal.

```
.cuadrado::before {  
    content: " ";  
    position: absolute;  
    top: 10px;  
    left: 10px;  
    right: 10px;  
    bottom: 10px;  
    border: 2px dashed #fff;  
}
```

SECUENCIA/DESARROLLO

- 1. Ejercicio 1:** Partiendo del archivo **index.html** tienes que crear el archivo **styles.css** para que aparezca el resultado mostrado en la imagen que aparece a continuación.



- 1.** Resultado similar al mostrado en la imagen. Para ello debes utilizar un layout flex: **(6 puntos)**
- 2.** Consideraciones:

1. Código css:

- **Declaración de variables: (0,5 puntos)**
- **Declarar tipos de letras: (0,5 puntos)**
- **Código documentado y estructurado (0,25 puntos)**
- **Reseteo del css: (0,25 puntos)**

2. Propiedades del header:

- **Divs del header (0,5 puntos)**
 - Modificar el borde
 - Color de fondo
 - Disposición
 - start (primer div)
 - stretch (segundo div)
 - end (tercer div)
- **Utilizar selectores child → (0,5 puntos)**

- **Cambios en la lista con viñetas del nav (0,5 puntos)**
 - Quitar la viñeta
 - Modificar tamaño letra
- **Formulario**
 - Debe aparecer cada elemento uno debajo de otro, tal y como se muestra en la imagen de la página. → **(0,5 puntos)**
 - Cambio color tipo de letra → **(0,25 puntos)**
 - Botón hover → **(0,25 puntos)**
- **Aside**
 - Divide el texto en formato dos columnas, texto justificado, tamaño letra del título, color, etc → **(0,5 puntos)**
- **Footer**
 - Modificar la disposición, tipo de letra, tamaño, etc → **(0,5 puntos)**
- **Gradientes y bordes en todo la página (1 punto)**

2. Ejercicio 2: Maqueta la página Web para el tamaño **@media (max-width: 768px)**, y que aparezca tal y como se muestra a continuación en las imágenes. **(4 puntos)**

- **Elementos que cambian:**
 - **Header:** Cambiamos la orientación del layout flex con disposición columna y redimensionamos los tamaños de los componentes.
 - **Main:** Se puede modificar de una forma muy sencilla haciendo que este elemento pase a tener una disposición como un bloque. (**propiedad Display**)

Artículos guardados... info clara Bootstrap 4 Ya >



FORMULARIO

Nombre

Correo

Mensaje

Enviar

[Nosotros](#)

[Contacto](#)

[Productos](#)

POSICIÓN ABSOLUTA

Con el posicionamiento absoluto contenedor padre y verificar si tenemos un mecanismo tiene un posicionamiento estático potentísimo para colocar o no. En el caso de que se trate de elementos. Para ello, solo un posicionamiento no estático, tenemos que colocar el position: utilizará este contenedor de marco absolute al elemento que de referencia, posicionando queremos posicionar. Este, lo que nuestro elemento absolute. hará es buscar su elemento

[Creado por](#)

[Redes Sociales](#)