

UNIDAD 2.

Desarrollo del Front end.

(El estudiante utilizará los elementos de una hoja de estilos y frameworks de hojas de estilos para la creación de un documento HTML.)

Tema II. Hojas de estilos en cascada (CSS).

Evidencia de aprendizaje Tema II –RESUMEN

Temas	Saber	Saber hacer	Resultado de Aprendizaje
Hojas de estilos en cascada (CSS)	Describir los conceptos y reglas de usabilidad de interfaces Web. Describir los elementos de hojas de estilo en cascada. (CSS y CSS3) Identificar conceptos de diseño responsivo. Identificar frameworks de diseño responsivo.	Diseñar la presentación de documentos HTML en hojas de estilo en cascada Desarrollar sitios Web responsivos en frameworks	Los estudiantes comprenden los elementos y la aplicación de las hojas de estilo en documentos HTML.
Evidencia de aprendizaje			
A partir de un examen práctico con un tiempo definido, desarrollar una página web que contenga: el conjunto de páginas web estáticas y responsivas usando HTML, CSS, JS, Frameworks para el diseño responsivo.			

Información sobre el profesor

Profesor	Correo	Días de Clase
Bernardo Prado Díaz	Tenor_prado@yahoo.com.mx	Lunes y Martes

Las Hojas de Estilo en Cascada, o **CSS** por sus siglas en inglés, son la herramienta que usamos para dar estilo a los documentos HTML. Si HTML es el esqueleto de tu página web, CSS es el vestuario, el maquillaje y la decoración que la hace atractiva y presentable. CSS se encarga del color, la tipografía, el diseño, los espacios y las animaciones, separando por completo la estructura (HTML) del diseño.

1. Conceptos Fundamentales de CSS

Reglas y Selectores

Una regla CSS se compone de un **selector** y un bloque de declaraciones. El selector "apunta" a los elementos HTML que quieres estilizar, y las declaraciones definen las propiedades y sus valores.

CSS

```
selector {  
  
  propiedad: valor;  
  
}
```

- **Selectores de Elemento:** Seleccionan todas las etiquetas de un tipo. Ejemplo: `p { color: gray; }` estiliza todos los párrafos.
- **Selectores de Clase:** Seleccionan elementos con un atributo class específico. Se usa un punto. Ejemplo: `.resaltado { font-weight: bold; }`
- **Selectores de ID:** Seleccionan un elemento único con un id específico. Se usa un numeral. Ejemplo: `#titulo-principal { font-size: 2.5rem; }`

Modelo de Caja (Box Model)

Cada elemento en HTML es una caja rectangular, y el Modelo de Caja es el concepto fundamental que describe cómo se calculan el tamaño y el espacio de estos elementos. Está compuesto por cuatro partes principales:

- **Content (Contenido):** El área donde se muestra el texto o las imágenes.
- **Padding (Relleno):** El espacio entre el contenido y el borde.
- **Border (Borde):** La línea alrededor del padding y el contenido.
- **Margin (Margen):** El espacio alrededor del borde, que separa el elemento de otros elementos.

La Cascada y la Especificidad

El término "Cascada" se refiere a cómo el navegador decide qué reglas de estilo aplicar cuando hay múltiples reglas que podrían aplicarse a un mismo elemento. La **Especificidad** es el factor clave en esta decisión:

- Un selector de ID (`#id`) es más específico que un selector de clase (`.class`).
- Un selector de clase (`.class`) es más específico que un selector de elemento (`p`).

- Los estilos en línea (<p style="...">) son los más específicos y anulan todos los demás.
-

2. Elementos de CSS3 y Diseño Responsivo

CSS3 y Propiedades Avanzadas

CSS3 introdujo muchas propiedades que transformaron el diseño web moderno, permitiendo crear interfaces más complejas y atractivas sin necesidad de imágenes.

- **box-shadow:** Añade sombras a las cajas.
- **border-radius:** Permite crear esquinas redondeadas.
- **flexbox y grid:** Son sistemas de diseño avanzados para crear layouts complejos de forma sencilla y eficiente. **Flexbox** es ideal para diseños unidimensionales (filas o columnas), mientras que **Grid** es para diseños bidimensionales.

Diseño Responsivo

El diseño responsivo es la práctica de crear sitios web que se adaptan a diferentes tamaños de pantalla (móviles, tabletas, ordenadores). Los conceptos clave son:

- **Unidades Flexibles:** Usar unidades relativas como % (porcentaje) o vw (viewport width) en lugar de píxeles fijos (px) para que los elementos escalen proporcionalmente.
- **Media Queries (@media):** Son reglas que aplican estilos solo si se cumple una condición, como el ancho de la pantalla. Esto te permite cambiar el diseño por completo para dispositivos más pequeños.

Frameworks

Los frameworks CSS, como **Bootstrap** o **Tailwind CSS**, son bibliotecas de estilos predefinidos. Su principal ventaja es que agilizan el desarrollo web al proporcionar clases listas para usar que ya implementan un diseño responsivo, lo que te permite construir interfaces complejas de manera mucho más rápida.

3. Ejercicio Guiado en Visual Studio Code

A continuación, encontrarás un ejercicio completo en un solo archivo HTML que integra todos los conceptos discutidos, incluyendo un diseño responsivo.

Instrucciones para VS Code:

1. Asegúrate de tener instalada la extensión **Live Server**.
2. Crea un nuevo archivo llamado estilos.html.
3. Copia y pega el código completo a continuación.
4. Haz clic derecho en el editor y selecciona "**Open with Live Server**".
5. Una vez que la página esté en tu navegador, intenta **cambiar el tamaño de la ventana**

```
<!--
  EJERCICIO GUIADO DE CSS Y DISEÑO RESPONSIVO

  Este documento HTML demuestra cómo usar CSS para dar estilo a los
  elementos
  y cómo implementar un diseño que se adapta a diferentes tamaños de
  pantalla.
-->
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Introducción a CSS y Diseño Responsivo</title>

  <!--
    Hoja de estilos interna:
    Aquí definiremos todas nuestras reglas CSS.
  -->
  <style>
    /* Selector Universal (*):
       Aplica estilos a todos los elementos. Aquí lo usamos para un
       "reset" básico,
       eliminando márgenes y paddings por defecto del navegador.
    */
    * {
      box-sizing: border-box; /* Incluye padding y borde en el
tamaño total */
      margin: 0;
      padding: 0;
      font-family: Arial, sans-serif;
    }

    /* Selector de Elemento (body):
       Estiliza el cuerpo de la página.
```

```

*/
body {
    background-color: #f4f4f4;
    color: #333;
    line-height: 1.6;
}

/* Selector de Clase (.container):
   Crea un contenedor centrado y con un ancho máximo.
*/
.container {
    max-width: 1100px;
    margin: auto;
    overflow: hidden;
    padding: 0 20px;
}

/* Selector de ID (#header):
   Estiliza la cabecera. Es un ID porque solo habrá una cabecera
principal.
*/
#header {
    background: #333;
    color: #fff;
    padding-top: 20px;
    min-height: 70px;
    border-bottom: #0779e4 3px solid;
    text-align: center;
}

#header h1 {
    margin-bottom: 20px;
}

/* Selector de Elemento (h2, p):
   Estilos para los títulos y párrafos.
*/
h2 {
    text-align: center;
    margin-top: 30px;
    margin-bottom: 10px;
    color: #0779e4;
}

p {

```

```

        text-align: justify;
        margin-bottom: 15px;
    }

    /*
        Demostración del Modelo de Caja:
        Estiliza un elemento con padding, border y margin.
    */
    .box {
        background: #fff;
        padding: 20px;          /* Espacio interno */
        border: 2px solid #ccc;  /* Borde */
        margin: 20px 0;         /* Espacio externo */

        /* Propiedades de CSS3 */
        border-radius: 10px;     /* Esquinas redondeadas */
        box-shadow: 0 5px 15px rgba(0,0,0,0.1); /* Sombra */
    }

    /* Estilos para los elementos del formulario:
        Se usan selectores de tipo para darles un aspecto uniforme.
    */
    input[type="text"], input[type="email"], textarea {
        width: 100%;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 5px;
        margin-bottom: 15px;
    }

    .btn-submit {
        display: block;
        width: 100%;
        padding: 10px;
        background: #0779e4;
        color: #fff;
        border: none;
        border-radius: 5px;
        cursor: pointer;
        transition: background 0.3s ease; /* Transición suave en hover
    */

    }

    .btn-submit:hover {
        background: #055ba3;
    }

```

```

    }

    /*
        Uso de Flexbox para un layout moderno y flexible:
        Permite alinear y distribuir elementos en un contenedor.
    */
    .flex-container {
        display: flex;
        justify-content: space-between; /* Espacia los elementos
uniformemente */
        flex-wrap: wrap; /* Permite que los elementos pasen a la
siguiente línea */
        gap: 20px;
        margin-top: 20px;
    }

    .flex-item {
        background: #fff;
        padding: 20px;
        border-radius: 10px;
        box-shadow: 0 5px 15px rgba(0,0,0,0.1);
        flex: 1; /* Permite que el elemento crezca y se encoja */
        min-width: 250px;
    }

    /*
        MEDIA QUERIES - El corazón del Diseño Responsivo:
        Estas reglas aplican cuando el ancho de la ventana es menor o
igual a 600px.
    */
    @media (max-width: 600px) {
        /* Para pantallas pequeñas, cambiamos la dirección del flexbox
        para que los elementos se apilen verticalmente.
        */
        .flex-container {
            flex-direction: column;
        }
    }
</style>
</head>
<body>

    <header id="header">
        <h1>Conceptos de CSS y CSS3</h1>
    </header>

```

```

<div class="container">

  <section class="box">
    <h2>El Modelo de Caja</h2>
    <p>
      Este elemento demuestra cómo las propiedades de CSS
      definen el Modelo de Caja.
      Tiene un <strong>padding</strong> para el espacio interno,
      un <strong>border</strong> para el espacio externo,
      y un <strong>margin</strong> para el espacio externo.
    </p>
    <p>
      Además, se aplican propiedades de CSS3 como <code>border-
      radius</code> y <code>box-shadow</code> para
      mejorar su apariencia visual.
    </p>
  </section>

  <h2>Diseño Responsivo con Flexbox</h2>
  <div class="flex-container">
    <div class="flex-item">
      <h3>Elemento 1</h3>
      <p>Este es un elemento flexible que se adapta al espacio
      disponible. En pantallas grandes, se mostrará junto a otros elementos.</p>
    </div>
    <div class="flex-item">
      <h3>Elemento 2</h3>
      <p>En dispositivos móviles, este elemento se apilará
      debajo del anterior gracias a una media query.</p>
    </div>
    <div class="flex-item">
      <h3>Elemento 3</h3>
      <p>Intenta redimensionar la ventana de tu navegador para
      ver cómo cambia el diseño.</p>
    </div>
  </div>

  <section class="box">
    <h2>Formulario de Contacto</h2>
    <form action="#">
      <label for="nombre">Nombre:</label>
      <input type="text" id="nombre" name="nombre" required>

      <label for="email">Email:</label>

```

```
        <input type="email" id="email" name="email" required>

        <label for="mensaje">Mensaje:</label>
        <textarea id="mensaje" name="mensaje" rows="4"></textarea>

        <button type="submit" class="btn-submit">Enviar</button>
    </form>
</section>

</div>

</body>
</html>
```