**Mi primera página web**

Alejandro Quiceno Pérez

Institución Universitaria Digital de Antioquia

Tecnología en Desarrollo de Software

Ingeniería Web I - PREICA2401B020025

Antonio Jesús Valderrama

Unidad 2 - Evidencia de aprendizaje 2

Medellín, mayo 25 de 2024

**Introducción:**

Los estilos CSS (Cascading Style Sheets) son un lenguaje utilizado para describir la presentación de un documento escrito en HTML o XML. CSS controla el aspecto visual de una página web, permitiendo definir colores, fuentes, márgenes, alineaciones y otros aspectos de diseño.

¿Para qué se usan?

CSS se utiliza para separar el contenido de la estructura visual, facilitando el mantenimiento y la actualización del diseño de un sitio web. Con CSS, se puede lograr un diseño consistente en todas las páginas de un sitio, mejorar la accesibilidad y garantizar que el sitio se vea bien en diferentes dispositivos y tamaños de pantalla.

¿Qué aportan a una página web?

CSS aporta una apariencia profesional y atractiva, mejora la experiencia del usuario (UX) al hacer la navegación más intuitiva y rápida, y optimiza el rendimiento del sitio web al reducir el tiempo de carga. Además, facilita la adaptabilidad y flexibilidad del diseño, permitiendo a los desarrolladores realizar cambios estéticos sin modificar el contenido HTML.

**Resumen explicativo del código CSS:**

* **styles.css**: Este código CSS define el estilo de una página web con un enfoque en la tipografía, el layout de navegación y la presentación del logo

1. Se importa la fuente: "Shadows Into Light" desde Google Fonts, que se usará para el título del logo
2. Se resetea el margen y el relleno de todos los elementos a 0 para eliminar el espacio predeterminado. Se aplica el modelo de caja border-box para incluir el borde y el padding en el ancho y alto total de los elementos.

Se establece la fuente global "Courier New".

1. **container-logo**  
   - display: flex: Utiliza el modelo de caja flexible para alinear el contenido del logo.

- justify-content: center: Centra los elementos hijos horizontalmente.

- align-items: center: Centra los elementos hijos verticalmente.

- gap: 25px: Añade un espacio de 25px entre los elementos hijos.

- h1: Establece el tamaño de fuente a 50px y aplica la fuente "Shadows Into Light".

1. **nav**- width: 100%: Hace que la barra de navegación ocupe todo el ancho de la página.

- display: flex: Utiliza el modelo de caja flexible.

- justify-content: space-between: Distribuye el espacio disponible entre los elementos, ubicándolos en los extremos.

- padding: 20px 80px: Añade un padding de 20px arriba y abajo, y 80px a los lados.

- align-items: center: Alinea verticalmente los elementos en el centro.

1. **Lista de Navegación (ul, li, a)**

- **ul:** Usa flex para alinear los elementos de la lista horizontalmente.

- **li:** Añade un margen de 20px a los lados y elimina los estilos predeterminados de la lista.

- **a:** **Estilos para los enlaces:**

\* Elimina el subrayado.

\* Establece el color del texto a negro.

\* Define el tamaño de fuente a 20px y aplica negrita.

\* Añade una transición suave de 0.2s para los cambios de estilo.

\* a:hover: Cambia el color del texto cuando el ratón pasa por encima.

**home.css**

* Contenedor principal (.container):

1. margin-top: 30px; Añade un espacio de 30 píxeles en la parte superior del contenedor, separándolo de cualquier contenido que esté encima.
2. display: flex; Utiliza el modelo de diseño flexible (flexbox) para organizar las tarjetas.
3. justify-content: center; Centra las tarjetas horizontalmente dentro del contenedor.
4. gap: 20px; Añade un espacio de 20 píxeles entre las tarjetas.
5. flex-wrap: wrap; Permite que las tarjetas se envuelvan a la siguiente línea si no caben en una sola fila, manteniendo un diseño limpio y ordenado.

* Tarjetas (.inventory-item)

1. background-color: #fff;: Establece el fondo de la tarjeta en blanco, lo que le da un aspecto limpio.
2. border-radius: 8px;: Redondea ligeramente las esquinas de la tarjeta para un aspecto más suave y moderno.
3. box-shadow: 0px 0px 14px rgba(0, 0, 0, 0.1);: Añade una sombra sutil alrededor de la tarjeta para darle profundidad y hacerla resaltar del fondo.
4. overflow: hidden;: Asegura que el contenido que se desborde del área de la tarjeta se oculte, manteniendo el diseño limpio.
5. width: 300px;: Fija el ancho de la tarjeta a 300 píxeles.
6. margin: 20px;: Añade un margen alrededor de cada tarjeta para separarlas unas de otras.
7. transition: transform 0.2s ease-in-out, box-shadow 0.2s ease-in-out;: Añade una transición suave cuando la tarjeta cambia de estado (por ejemplo, al pasar el ratón por encima).

* Efectos Interacción (.inventory-item:hover)

1. transform: translateY(-10px);: Mueve la tarjeta hacia arriba 10 píxeles, creando un efecto de elevación.
2. box-shadow: 0 8px 16px rgba(0, 0, 0, 0.2);: Aumenta la sombra, haciendo que la tarjeta parezca más elevada y destacada.

* Imagen

1. width: 400px; y height: 300px;: Establecen las dimensiones fijas de la imagen.
2. object-fit: cover;: Asegura que la imagen cubra completamente el área asignada sin distorsionarse, recortando si es necesario.

* Producto Tarjeta (.inventory-details)

1. padding: 16px;: Añade un relleno interno para separar el contenido del borde del contenedor.
2. text-align: center;: Centra el texto dentro del contenedor.

* Titulo producto (.inventory-details h2)

1. font-size: 1.5em;: Establece un tamaño de fuente grande para que el título destaque.
2. margin: 0 0 10px 0;: Elimina el margen superior y añade un margen inferior de 10 píxeles para espaciarlo del texto siguiente.
3. color: #333;: Aplica un color gris oscuro para una buena legibilidad.

* Precio del producto

**p:**

1. font-size: 1.2em;: Establece un tamaño de fuente moderadamente grande para el precio.
2. color: #666;: Usa un color gris medio.
3. margin: 0;: Elimina los márgenes predeterminados.

**span:**

1. color: #28a745;: Aplica un color verde brillante para destacar el precio.
2. font-weight: bold;: Usa negrita para dar énfasis al precio.

**store.css**

* Contenedor principal de tarjetas (.container)

1. height: 80vh;: Establece la altura del contenedor al 80% de la altura de la ventana del navegador.
2. display: flex;: Utiliza flexbox para organizar las tarjetas.
3. justify-content: center;: Centra las tarjetas horizontalmente dentro del contenedor.
4. align-items: center;: Centra las tarjetas verticalmente dentro del contenedor.
5. gap: 20px;: Añade un espacio de 20 píxeles entre las tarjetas.
6. flex-wrap: wrap;: Permite que las tarjetas se envuelvan a la siguiente línea si no caben en una sola fila, manteniendo un diseño ordenado.

* Estilos tarjetas (.inventory-item)

1. background-color: #fff;: Establece el fondo de la tarjeta en blanco.
2. border-radius: 8px;: Redondea ligeramente las esquinas de la tarjeta.
3. box-shadow: 0px 0px 14px rgba(0, 0, 0, 0.301);: Añade una sombra suave alrededor de la tarjeta.
4. overflow: hidden;: Asegura que el contenido que se desborde del área de la tarjeta se oculte.
5. width: 300px;: Fija el ancho de la tarjeta a 300 píxeles.
6. height: 470px;: Fija la altura de la tarjeta a 470 píxeles.
7. margin: 20px;: Añade un margen alrededor de cada tarjeta para separarlas.
8. transition: transform 0.2s ease-in-out, box-shadow 0.2s ease-in-out;: Añade una transición suave cuando la tarjeta cambia de estado.

* Efectos

1. box-shadow: 0px 0px 14px rgba(0, 0, 0, 0.747);: Aumenta la sombra, haciendo que la tarjeta parezca más destacada.

* Imagen

1. width: 400px; y height: 300px;: Establecen las dimensiones fijas de la imagen.
2. object-fit: cover;: Asegura que la imagen cubra completamente el área asignada sin distorsionarse, recortando si es necesario.

* Detalles del producto

1. .inventory-details:

* padding: 16px;: Añade un relleno interno.
* text-align: center;: Centra el texto.

1. h2:

* font-size: 1.5em;: Establece un tamaño de fuente grande.
* margin: 0 0 10px 0;: Elimina el margen superior y añade un margen inferior.
* color: #333;: Usa un color gris oscuro.

1. p:

* font-size: 1.2em;: Establece un tamaño de fuente moderadamente grande.
* color: #666;: Usa un color gris medio.
* margin: 0;: Elimina los márgenes.

1. span:

* color: #28a745;: Usa un color verde brillante.
* font-weight: bold;: Usa negrita.
* Boton

1. --primary-color, --secondary-color, --hover-color, --arrow-width, --arrow-stroke: Variables CSS para colores y tamaños.
2. margin: 0 auto;: Centra el botón horizontalmente.
3. border: 0;: Elimina el borde del botón.
4. border-radius: 20px;: Redondea las esquinas del botón.
5. color: var(--secondary-color);: Establece el color del texto del botón.
6. padding: 1em 1.8em;: Añade espacio interno.
7. background: var(--primary-color);: Establece el color de fondo del botón.
8. display: flex;: Utiliza flexbox para el contenido del botón.
9. transition: 0.2s background;: Añade una transición suave para cambios de fondo.
10. align-items: center;: Alinea el contenido verticalmente.
11. gap: 0.6em;: Añade espacio entre el texto y el icono.
12. font-weight: bold;: Usa negrita.
13. font-size: 15px;: Establece el tamaño de fuente.
14. .arrow-wrapper: Usa flexbox para centrar la flecha.
15. .arrow: Define la barra horizontal de la flecha.
16. margin-top: 1px;, width: var(--arrow-width);, background: var(--primary-color);, height: var(--arrow-stroke);, position: relative;, transition: 0.2s;.
17. .arrow::before: Añade una flecha mediante pseudoelemento.
18. content: "";, box-sizing: border-box;, position: absolute;, border: solid var(--secondary-color);, border-width: 0 var(--arrow-stroke) var(--arrow-stroke) 0;, display: inline-block;, top: -3px;, right: 3px;, transition: 0.2s;, padding: 3px;, transform: rotate(-45deg);.
19. button:hover: Cambia el color de fondo del botón al pasar el ratón.
20. button:hover .arrow: Cambia el color de la flecha al pasar el ratón.
21. button:hover .arrow:before: Ajusta la posición de la flecha al pasar el ratón

* Footer

1. text-align: center;: Centra el texto del pie de página.
2. padding: 10px;: Añade espacio interno.
3. position: fixed;: Fija el pie de página en la parte inferior de la ventana.
4. bottom: 0;: Asegura que el pie de página esté al fondo.
5. width: 100%;: Hace que el pie de página abarque todo el ancho de la ventana.
6. font-weight: bold;: Usa negrita para el texto del pie de página.

**newProduct.css**

* Seccion principal

1. width: 800px;: Establece el ancho de la sección a 800 píxeles, proporcionando un área bien definida para el contenido.
2. height: 600px;: Fija la altura de la sección a 600 píxeles.
3. margin: 0 auto;: Centra la sección horizontalmente dentro de su contenedor padre.
4. box-shadow: 0px 0px 17px rgba(0, 0, 0, 0.562);: Añade una sombra alrededor de la sección para darle un efecto de profundidad.
5. border-radius: 20px;: Redondea las esquinas de la sección, dándole un aspecto más suave y moderno.

* Titulo Seccion

1. text-align: center;: Centra el texto del encabezado horizontalmente.
2. padding-top: 80px;: Añade un relleno superior al encabezado para separarlo del borde superior de la sección.
3. margin: 80px 0;: Añade un margen superior e inferior de 80 píxeles para espaciar el encabezado del resto del contenido

* Formulario

1. margin: 0 auto;: Centra el formulario horizontalmente dentro de la sección.
2. width: 90%;: Establece el ancho del formulario al 90% del ancho de su contenedor padre.
3. display: flex;: Utiliza el modelo de caja flexible para organizar los elementos del formulario.
4. flex-direction: column;: Coloca los elementos del formulario en una columna.
5. margin-bottom: 10px;: Añade un margen inferior para espaciar las etiquetas de los campos de entrada.
6. font-weight: bold;: Hace que el texto de las etiquetas sea negrita, destacándolas.
7. padding: 10px;: Añade relleno interno para hacer que los campos sean más fáciles de usar.
8. margin-bottom: 20px;: Añade un margen inferior para espaciar los campos entre sí.
9. border: 1px solid #ccc;: Añade un borde de 1 píxel con un color gris claro, dándole definición al campo.
10. border-radius: 5px;: Redondea ligeramente las esquinas de los campos.
11. padding: 10px 20px;: Añade relleno interno para hacer que el botón sea más fácil de usar.
12. background-color: #645bff;Establece el color de fondo del botón a un tono morado.
13. font-size: 20px;: Establece el tamaño de fuente del texto en el botón.
14. font-weight: bold;: Hace que el texto del botón sea negrita, destacándolo.
15. color: #fff;: Establece el color del texto a blanco.
16. border: none;: Elimina cualquier borde predeterminado del botón.
17. border-radius: 5px;: Redondea las esquinas del botón.
18. cursor: pointer;: Cambia el cursor al pasar sobre el botón, indicando que es clicable.
19. transition: background-color 0.3s ease;: Añade una transición suave para el cambio de color de fondo.
20. input[type="submit"]:hover { background-color: #111; }: Cambia el color de fondo del botón a negro cuando se pasa el ratón sobre él, proporcionando un efecto visual interactivo.

**Razones de Uso de Estos Estilos**

1. Apariencia y Consistencia

Reset de Margen y Padding: Asegura que todos los elementos en la página web comienzan con un margen y padding uniformes. Esto elimina las inconsistencias predeterminadas entre diferentes navegadores, lo que facilita el diseño coherente y predecible.

2. Flexibilidad y Diseño Centrado

Flexbox (en .container-logo y nav): Flexbox es una herramienta poderosa para crear layouts flexibles y adaptables. Permite centrar elementos tanto horizontal como verticalmente, y distribuir el espacio de manera eficiente. Esto es especialmente útil para el diseño de navegaciones y contenedores de logotipos, donde la alineación precisa es crucial.

3. Identidad de Marca y Estética

Tipografía Específica (fuente importada y font-family): La selección de fuentes específicas ayuda a transmitir la identidad de la marca. La fuente "Shadows Into Light" para el título añade un toque personalizado y distintivo, mientras que "Courier New" da un aspecto clásico y profesional. Esto contribuye a una experiencia de marca cohesiva y memorable para los visitantes.

4. Navegación Intuitiva y Experiencia del Usuario

Estilización de Enlaces de Navegación (ul, li, a): Los enlaces de navegación claramente visibles y estilizados mejoran la usabilidad. Al eliminar el subrayado predeterminado y utilizar negrita, los enlaces se vuelven más fáciles de leer. La transición suave de color al pasar el ratón por encima proporciona una retroalimentación visual agradable, mejorando la interacción del usuario.

5. Accesibilidad y Adaptabilidad

Diseño Responsivo y Distribución de Espacio: Utilizar padding y margin específicos asegura que los elementos estén espaciados adecuadamente, adaptándose a diferentes tamaños de pantalla. Esto es crucial para garantizar que la página web sea accesible y se vea bien en dispositivos móviles, tablets y computadoras de escritorio.

6. Interactividad y Estética Visual

Transiciones y Hover Effects: Las transiciones suaves y los efectos al pasar el ratón (hover) hacen que la interacción con la página web sea más dinámica y atractiva. Esto no solo mejora la experiencia del usuario, sino que también puede dirigir la atención hacia elementos importantes, como enlaces de navegación o llamados a la acción.

7. Coherencia Visual

Estilización de Iconos SVG: Ajustar el color y el tamaño de los iconos SVG asegura que se integren perfectamente con el diseño general del sitio. Esto mantiene la coherencia visual y asegura que todos los elementos gráficos estén alineados con la estética de la marca.

**Conclusión:**

En conclusión el css es fundamental para el diseño y la funcionalidad de una página web. No solo permite crear un aspecto visual atractivo y profesional, sino que también permite la mejora de la experiencia de usuario (UX) al facilitar una navegación más intuitiva y consistente.

Separar el contenido CSS simplifica el mantenimiento y la actualización del sitio, haciendo que los cambios de diseño sean más eficientes. Además, garantiza que el sitio sea accesible y se adapte correctamente a diferentes dispositivos, lo cual es crucial en el entorno digital actual.

Pagina desplegada en el servidor de GitHub: <https://alejandro-quiceno-perez.github.io/IUDigital_IngenieriaWeb1/index.html>