

Universidad Mariano Gálvez de Guatemala

Facultad de Ingeniería en Sistemas de Información y Ciencias de la Computación.

CURSO: Desarrollo Web.

CATEDRATICO: Carmelo Estuardo Mayen Monterroso.

SEMESTRE: Octavo.



TAREA: Formulario e Investigación.

ALUMNO: Milton Adrián Martínez Ávila.

CARNÉ: 1790-22-9129.

CHIQUMULILLA, JULIO 2025

<input> Con sus respectivos types.

El elemento <input> es uno de los más importantes en los formularios HTML. Se utiliza para permitir que el usuario introduzca diferentes tipos de datos. Su comportamiento cambia según el valor del atributo type, lo que permite crear campos especializados para distintos propósitos. A continuación te detallo todos los tipos:

Tipos comunes de <input>:

“text”

Sirve para ingresar texto en una sola línea. Es el más utilizado y básico, ideal para nombres, apellidos, títulos, etc.

“password”

Permite ingresar texto oculto. Lo que se escribe no se muestra en pantalla, lo que lo hace útil para contraseñas u otros datos privados.

“email”

Específico para direcciones de correo electrónico. El navegador puede validar automáticamente si el formato del correo es válido (por ejemplo, que tenga @ y un dominio).

“url”

Utilizado para ingresar direcciones web. Al igual que el de correo, puede validar el formato para que sea una URL real.

“tel”

Para ingresar números de teléfono. No valida el formato, pero en dispositivos móviles muestra un teclado numérico.

“number”

Permite ingresar solo números. Se pueden establecer límites mínimos, máximos y pasos de incremento. Es útil para edades, cantidades, precios, etc.

“range”

Muestra una barra deslizante. El usuario puede moverla para elegir un valor dentro de un rango definido. Se usa para calificaciones, volumen, niveles, etc.

“search”

Similar a un campo de texto, pero diseñado específicamente para búsquedas. Puede incluir características como un botón para borrar rápidamente el texto.

“color”

Muestra una paleta para elegir un color. Devuelve un valor en formato hexadecimal (como los colores en diseño web).

“date”

Abre un selector de calendario para elegir una fecha (día, mes y año).

“time”

Permite seleccionar una hora específica del día.

“datetime-local”

Combina fecha y hora en un solo campo. Se usa cuando se necesita registrar un evento con ambos datos.

“month”

Permite seleccionar solo un mes y un año, sin necesidad del día.

“week”

Permite seleccionar una semana del año, lo que puede ser útil para agendas, reportes semanales, etc.

“checkbox”

Crea una casilla que se puede marcar o desmarcar. Se usa para opciones múltiples, como aceptar términos, elegir intereses, etc.

“radio”

Crea una opción circular. Se usa cuando el usuario debe elegir solo una opción entre varias posibles (por ejemplo, sexo o nivel educativo).

“file”

Permite seleccionar uno o más archivos del dispositivo del usuario para subirlos. Es útil para enviar imágenes, documentos, currículums, etc.

“hidden”

Es un campo que no se muestra en pantalla. Su valor se envía junto con el formulario. Se usa para guardar datos que el usuario no necesita ver ni modificar (por ejemplo, identificadores de sesión).

“image”

Crea un botón con imagen que se comporta como un botón de envío. Es usado cuando se quiere personalizar la apariencia del botón.

“submit”

Es un botón que envía el formulario al servidor. Generalmente se coloca al final del formulario.

“reset”

Botón que restablece todos los campos del formulario a sus valores iniciales. Es útil si se quiere dar al usuario la opción de borrar todo fácilmente.

“button”

Botón que no hace nada por sí mismo, pero puede ser programado con JavaScript para realizar acciones específicas (por ejemplo, mostrar u ocultar elementos, abrir ventanas emergentes, etc.).

<textarea> Área de texto

Este elemento sirve para que el usuario escriba texto largo o en varias líneas. Es diferente al <input type="text"> porque aquí se puede usar Enter para cambiar de renglón.

¿Para qué se usa?

Para escribir mensajes largos, como comentarios, descripciones, reseñas, observaciones, preguntas, etc. Es útil cuando se espera que el usuario escriba más de una frase o cuando se necesita una explicación detallada.

Características importantes:

Puedes definir cuántas líneas visibles tendrá el área (aunque puede agrandarse con el mouse).

Puedes establecer un texto guía para que el usuario sepa qué escribir.

Se puede limitar el número máximo de caracteres.

Puede estar deshabilitado o ser solo de lectura.

<select> Menú desplegable

El elemento <select> permite al usuario elegir una o varias opciones de una lista predefinida.

¿Para qué se usa?

Para ofrecer opciones como países, ciudades, profesiones, categorías, etc.

Es útil cuando hay que mostrar muchas opciones sin ocupar mucho espacio.

Se puede hacer que el usuario elija solo una o varias (marcando múltiples opciones).

Características importantes:

Se compone de varias opciones individuales (<option>) que el usuario puede seleccionar.

Se pueden agrupar las opciones por categorías (<optgroup>), lo que es útil si hay muchas.

Puede configurarse para que permita seleccionar varias opciones al mismo tiempo.

Se puede desactivar para que no esté disponible temporalmente.

CSS

CSS (cascading style sheets), conocido en español como hojas de estilo en cascada, es una poderosa herramienta que se integra con HTML para proporcionar a los desarrolladores y usuarios un control avanzado sobre la presentación visual de las páginas web. CSS permite separar el contenido de la presentación, facilitando la creación de diseños consistentes y atractivos.

Con CSS, los diseñadores pueden definir cómo deben mostrarse los distintos elementos de una página, como los encabezados, enlaces, imágenes y tablas. Además, estas hojas de estilo pueden aplicarse de manera global a múltiples páginas, lo que simplifica el mantenimiento y la actualización del diseño de un sitio web completo. Esta capacidad de optimización también contribuye a mejorar el posicionamiento en buscadores (SEO), ya que permite un código más limpio y estructurado.

Origen y evolución de CSS.

El origen de CSS (Cascading Style Sheets) se remonta a 1994, cuando Håkon Wium Lie, trabajando en el CERN, propuso la idea de un lenguaje que permitiera separar el contenido de la presentación en documentos web. En esa época, el HTML se utilizaba principalmente para estructurar contenido, sin muchas capacidades de diseño visual. Lie se unió a Bert Bos, quien también estaba explorando soluciones similares, y juntos refinaron la propuesta inicial. Su objetivo era crear un sistema que permitiera a los desarrolladores aplicar estilos de manera más eficiente y flexible. En 1996, el World Wide Web Consortium (CW3) publicó la primera especificación oficial de CSS, conocida como CSS1, marcando el inicio de una nueva era en el diseño web.

Desde su creación, CSS ha transformado el diseño y desarrollo web. Esta evolución ha facilitado la separación del contenido y la presentación, mejorando tanto la eficiencia del desarrollo como la experiencia del usuario. A continuación, se detalla la evolución de CSS a través de sus principales versiones y características.

- **CSS1 (1996)**

Introducción: CSS1 fue la primera versión oficial, publicada por el W3C en diciembre de 1996.

Características: Proporcionaba capacidades básicas de estilo, como fuentes, colores, y espaciado.

Permitía a los desarrolladores separar el contenido del diseño visual.

- **CSS2 (1998)**

Mejoras: Introducido en mayo de 1998, CSS2 amplió significativamente las capacidades de CSS1.

Nuevas características: Añadió soporte para hojas de estilo para diferentes medios (impresión, pantalla, etc.), posicionamiento absoluto y relativo, y estilos para tablas.

Accesibilidad: Mejoras en la accesibilidad y en la presentación para diferentes dispositivos.

- **CSS2.1 (2011)**

Revisión: CSS2.1 fue una revisión de CSS2, que corrigió errores y eliminó características poco utilizadas.

Estándar estable: Se convirtió en el estándar estable, siendo ampliamente adoptado por los navegadores.

- **CSS3 (A partir de 2011)**

Módulos: CSS3 introdujo un enfoque modular, permitiendo que diferentes partes del estándar evolucionaran a diferentes ritmos.

Nuevas capacidades: Incluyó características avanzadas como sombras, bordes redondeados, gradientes, transiciones, animaciones, y soporte para fuentes web.

Responsive design: Media queries, una parte crucial de CSS3, permitieron el diseño responsivo, adaptando el diseño a diferentes tamaños de pantalla.

- **CSS4 (En desarrollo)**

Estado: Aunque no existe un «CSS4» oficial como tal, el término se utiliza para referirse a las nuevas características que están siendo desarrolladas y añadidas a CSS.

Características emergentes: Incluyen mejoras en selectores, capacidades de layout avanzadas como grid y flexbox, y nuevas funciones para animaciones y efectos visuales.

Frameworks de Estilos de CSS.

Los frameworks de CSS son colecciones de estilos y utilidades predefinidas que aceleran el desarrollo de interfaces coherentes y responsivas. Entre los más populares destacan:

- **Bootstrap**

Ofrece un sistema de cuadrícula (grid) flexible, componentes listos (botones, menús, formularios) y utilidades para espaciados, tipografía y colores. Muy extendido y con amplia documentación.

- **Tailwind CSS**

Basado en clases utilitarias atómicas (“utility-first”), permite componer estilos en el mismo HTML con pequeñas clases (p. ej. margen, color, tamaño) para construir diseños personalizados sin salir del marcado.

- **Bulma**

Ligero y modular, utiliza Flexbox para su grid y componentes. Su sintaxis es sencilla, con nombres de clases semánticas y sin dependencias de JavaScript.

- **Foundation**

Ofrece un sistema de cuadrícula avanzado, componentes accesibles y plugins JavaScript opcionales. Está orientado tanto al prototipado rápido como a proyectos de producción robustos.

- **Materialize / Material UI**

Inspirados en las “Material Design” de Google, proporcionan componentes con animaciones y estilos que siguen esta guía de diseño, facilitando interfaces modernas y consistentes.

Media Querys de CSS.

Las Media Queries son una herramienta esencial en CSS que nos permite adaptar nuestros diseños web a diferentes dispositivos y resoluciones. Son como instrucciones especiales que podemos agregar a nuestro código para aplicar estilos específicos según las características del dispositivo que se está utilizando para ver nuestra página.

Imaginemos que estamos creando un sitio web y queremos que se vea bien tanto en computadoras de escritorio como en teléfonos móviles. Con las Media Queries, podemos establecer reglas de estilo específicas para cada dispositivo. Podemos definir tamaños de fuente, márgenes, disposición de elementos y muchas otras propiedades que se ajusten mejor a cada pantalla.

Importancia del uso de las Media Queries.

La importancia de las Media Queries en el desarrollo web actual radica en que cada vez más personas navegan en Internet desde sus teléfonos y tabletas. Los sitios web responsivos son esenciales para proporcionar una experiencia de usuario óptima en cualquier dispositivo. Las Media Queries nos permiten lograr esto, ya que nos dan el control para adaptar nuestros diseños a diferentes tamaños de pantalla y resoluciones.

Al crear sitios web responsivos, podemos garantizar que nuestros contenidos se vean correctamente, sin desbordamientos o elementos desproporcionados. Al utilizar Media Queries, podemos ajustar automáticamente la apariencia de nuestros diseños para que se adapten perfectamente a cualquier dispositivo, brindando a los usuarios una experiencia de navegación fluida y agradable.

Enlace del Repositorio del Formulario.

<https://github.com/martinez132/Formulario.git>