**Documento de Trabajo de Electiva profesional I**

Jairo Andrés Montenegro

Jefferson Montenegro

Fernando Andrés Montenegro

Facultad de Ingenierías, Fundación Universitaria de Popayán

Electiva profesional I

Luis vejarano

21 de noviembre del 2024

TABLA DE CONTENIDO

[Introducción 3](#_Toc183033472)

[SEGUNDO CORTE 3](#_Toc183033473)

[Explicación del HTML 3](#_Toc183033474)

[Explicación del Javascript 4](#_Toc183033475)

# Introducción

# SEGUNDO CORTE

Explicación del HTML

* <head>: Contiene la información sobre la página, como la codificación de caracteres y el viewport para hacer que la página sea responsiva. También incluye el título de la página y estilos CSS en línea.
* <style>: En esta parte hicimos la definición de los estilos CSS para mejorar la apariencia visual de la página:
  + body: Aquí definimos la fuente, el color del texto, la alineación del texto y los márgenes superiores.
  + button: En esta parte se define el estilo de los botones, incluyendo el color de fondo, el tamaño de la fuente, el margen, el cursor, y los efectos al pasar el mouse.
  + footer: con este hicimos la definición del estilo del pie de página, incluyendo el margen superior y el tamaño y color de la fuente.
  + entrada-numero: Define el margen del contenedor que incluye el campo de entrada.
* <body>: Contiene el contenido principal de la página:
  + <h1>: Insertamos el título principal de la página.
  + <p>: lo utilizamos para poner un párrafo descriptivo de lo que iría a continuación.
  + <div class="entrada-numero">: Contiene un campo de entrada (<input>) para que el usuario ingrese un número y un botón para mostrar las tablas.
  + <div id="tablas">: Contenedor ID que incluye divisiones donde se mostrarán las tablas de multiplicar generadas por los diferentes ciclos.
  + <footer>: Pie de página que contiene un mensaje sobre el propósito de nuestro proyecto.

Explicación del Javascript

* mostrarTablas:
  + Obtiene el número ingresado por el usuario del campo de entrada.
  + Llama a cada una de las funciones para mostrar las tablas de multiplicar utilizando el número proporcionado.
* **Funciones de Ciclos**:
  + mostrarTablaFor(numero):
    - En la iteración se crea una lista de elementos <li> con los resultados de la multiplicación.
  + mostrarTablaForIn(numero):
    - Crea un objeto tabla con los resultados de la multiplicación.
    - Itera sobre las claves del objeto tabla para crear una lista de elementos <li> con los resultados.
  + mostrarTablaForOf(numero):
    - Crea un array tabla con los resultados de la multiplicación.
    - Itera sobre los valores del array tabla para crear una lista de elementos <li> con los resultados.
  + mostrarTablaWhile(numero):
    - Itera desde 1 hasta 10, creando una lista de elementos <li> con los resultados de la multiplicación.
  + mostrarTablaDoWhile(numero):
    - Itera desde 1 hasta 10, creando una lista de elementos <li> con los resultados de la multiplicación.
* document.addEventListener("DOMContentLoaded", function() { ... }):
* En este caso, se utiliza para ejecutar una función cuando el contenido del documento se ha cargado completamente (DOMContentLoaded).

OBJETOS

En programación, un objeto es una entidad que agrupa datos (propiedades) y funciones (métodos) relacionados. Puedes pensar en un objeto como una representación de algo del mundo real.

Por ejemplo, si estamos modelando autos, un objeto "Coche" podría tener propiedades como "marca", "modelo" y "año", junto con métodos como "arrancar" o "detenerse".

Los objetos nos permiten organizar y estructurar nuestro código de manera más eficiente y lógica.

* Propiedades:

Las propiedades son características o atributos de un objeto. Representan los datos asociados a ese objeto.

Por ejemplo, en un objeto "Persona", las propiedades podrían ser "nombre", "edad" y "género".

Accedemos a las propiedades de un objeto utilizando la notación de punto (por ejemplo, miCoche.marca) o la notación de corchetes (por ejemplo, miCoche["marca"]).

* Métodos:

Los métodos son funciones asociadas a un objeto. Representan las acciones que el objeto puede realizar.

Siguiendo con el ejemplo del objeto "Coche", podríamos tener un método llamado "arrancar" que inicia el motor del coche.

Los métodos se definen dentro del objeto y se invocan utilizando la notación de punto (por ejemplo, miCoche.arrancar()).

DESARROLLO DEL EJEMPLO

Proceso de creación de un buscador interactivo de marcas de autos utilizando HTML, JavaScript. El objetivo del ejercicio es permitir a los usuarios seleccionar una marca de automóvil y mostrar su logotipo junto con información relevante sobre la marca.

* A continuación, se detallan los pasos seguidos para implementar el buscador:
  + Preparación de Recursos:
    - Se recopilaron los logotipos de diferentes marcas de automóviles en formato PNG.
    - Se creó una carpeta llamada **logos** para almacenar los logotipos.
  + Creación del Archivo HTML (index.html):
    - Se estructuró un archivo HTML básico con los siguientes elementos:
      * Un título descriptivo.
      * Un elemento <select> para que el usuario elija una marca.
      * Un <div> con un id para mostrar los resultados (logotipo y descripción).
  + Creación del Archivo JavaScript (script.js):
    - Se definió un objeto llamado marcas que contiene información sobre cada marca (nombre, logotipo, descripción, año de fundación).
    - Se agregó un evento al <select> para detectar la selección del usuario.
    - Cuando el usuario selecciona una marca, se actualiza el logotipo y la descripción en la página.
* Resultados

El resultado final es un buscador interactivo que muestra logotipos y detalles de diferentes marcas de automóviles. Al seleccionar una marca, el logotipo se actualiza automáticamente, y se muestra una breve descripción junto con el año de fundación.

* Conclusiones

Este ejercicio demuestra cómo combinar HTML, JavaScript para crear una experiencia interactiva en una página web. La modularidad del código permite mantener la flexibilidad y facilita futuras mejoras o adiciones.

* Referencias

Documentación de HTML

Documentación de JavaScript

Nota: Este documento fue creado con fines educativos y como parte de un ejercicio práctico. Los logotipos utilizados son ficticios y se proporcionan solo como ejemplo.