

## Ejercicios en HTML CSS y JAVASCRIPT.

código para crear un generador de combinaciones de la Lotería Primitiva española, con HTML, CSS y JavaScript. Incluye un diseño atractivo y la opción de generar más combinaciones o salir.

### Algoritmo.

Este código está dividido en tres secciones principales: HTML, CSS y JavaScript.

#### HTML (Estructura)

El HTML define la estructura de la página.

- `<!DOCTYPE html>` y `<html lang="es">`: Declaran el tipo de documento y el idioma de la página.
  - `<head>`: Contiene metadatos como la codificación de caracteres (`<meta charset="UTF-8">`), la configuración de la vista para dispositivos móviles (`<meta name="viewport">`) y el título de la página (`<title>`).
    - También se enlaza la fuente de Google Fonts "Montserrat" para un diseño moderno.
    - El CSS está incrustado dentro de la etiqueta `<style>` para simplificar el ejemplo.
  - `<body>`: Contiene todo el contenido visible de la página.
    - `<div class="container">`: Es el contenedor principal que agrupa todos los elementos del generador, centrándolos y aplicándoles estilos de caja.
    - `<h1>`: El título principal de la aplicación.
    - `<div class="buttons-container">`: Contiene los dos botones principales.
      - `<button id="generateButton">`: Botón para generar una nueva combinación.
      - `<button id="exitButton">`: Botón para "salir" del programa (realmente informa al usuario cómo cerrar la pestaña).
    - `<div class="result-box" id="resultBox">`: Aquí se mostrarán los números de la lotería generados. Inicialmente, muestra un mensaje instructivo.
    - `<div class="explanation">`: Una sección que explica las reglas de la Lotería Primitiva y cómo funciona el generador.
  - `<script>`: Al final del `<body>`, incluye el código JavaScript para la interactividad de la página.
- 

#### CSS (Estilos)

El CSS es el encargado de dar un aspecto atractivo y moderno a la aplicación.

- **Fuentes**: Se importa la fuente "Montserrat" de Google Fonts.
- **body**: Define el fondo de la página con un degradado animado, centra el contenido y establece la tipografía base. Incluye un efecto de "fondo pulsante" (`::before` y `@keyframes pulsateBackground`) para un toque visual.
- **.container**: Estiliza la caja principal con un fondo translúcido, bordes redondeados, sombra y transiciones para un efecto interactivo al pasar el ratón.
- **h1**: Estilos para el título principal.

- **.buttons-container y button:** Estilizan los botones de "Generar" y "Salir" con colores, sombras, bordes redondeados y efectos de transición al pasar el ratón (:hover) y al hacer clic (:active).
- **.result-box:** Estiliza la caja donde se muestran los resultados, con un borde punteado y un fondo claro.
- **.number-ball, .complementary-ball, .reintegro-ball:** Estos estilos definen el aspecto de los números individuales como "bolas de lotería" con colores degradados diferentes para los números principales, el complementario y el reintegro, haciendo que se distingan visualmente. Incluyen un efecto de escala al pasar el ratón.
- **.number-group:** Agrupa las bolas de números para una mejor organización visual.
- **.explanation:** Estilos para la sección de texto explicativo, con un fondo ligeramente translúcido y sombra.

## JavaScript (Funcionalidad)

El JavaScript añade la lógica para generar los números y manejar la interacción del usuario.

- **document.addEventListener('DOMContentLoaded', () => { ... });** Asegura que el código JavaScript se ejecute una vez que todo el contenido HTML de la página ha sido cargado.
- **const generateButton = document.getElementById('generateButton');** Obtiene referencias a los elementos HTML por su id.
- **function generatePrimitiva():**
  - Crea un Set llamado `numbers` para almacenar los 6 números principales. Un Set garantiza que no haya números duplicados automáticamente.
  - Usa un bucle `while` para generar números aleatorios entre 1 y 49 hasta que se tengan 6 números únicos.
  - `Math.floor(Math.random() * 49) + 1` genera un número aleatorio entre 1 y 49.
  - Convierte el Set a un Array y lo ordena (`sort((a, b) => a - b)`) para mostrar los números de forma ascendente.
  - Genera el **número Complementario**: Se asegura de que sea un número entre 1 y 49 y que no esté ya en el conjunto de los 6 números principales.
  - Genera el **número Reintegro**: Un número aleatorio entre 0 y 9.
  - Llama a `displayResult()` para mostrar los números generados en la página.
- **function displayResult(numbers, complementary, reintegro):**
  - Actualiza el contenido HTML de `resultBox` para mostrar los números principales, el complementario y el reintegro.
  - Usa `map()` para crear dinámicamente los elementos `div` con la clase `number-ball` para cada número.
- **generateButton.addEventListener('click', generatePrimitiva);** Asigna la función `generatePrimitiva` para que se ejecute cada vez que se haga clic en el botón "Generar Combinación".

- `exitButton.addEventListener('click', () => { ... });`
  - Asigna una función al botón "Salir".
  - Muestra un `confirm` (una ventana de diálogo de confirmación del navegador) preguntando al usuario si realmente quiere salir.
  - Si el usuario confirma, se deshabilita los botones y se muestra un mensaje informativo, ya que por seguridad, los navegadores no permiten que JavaScript cierre la ventana o pestaña de forma programática sin interacción del usuario.
- `generatePrimitiva()` ;: Se llama a esta función una vez al cargar la página para mostrar una combinación inicial sin necesidad de hacer clic.

---

Este generador se crea en un solo código embebido.

Vista del resultado.

