

# Examen Práctico de JavaScript

Duración: 3 horas

Completa las siguientes tareas utilizando JavaScript. Puedes utilizar HTML y CSS solo para la estructura básica y el estilo, pero el enfoque principal debe ser en JavaScript. Asegúrate de comentar tu código para explicar tu lógica.

Debes crear un juego en el que adivines el número (de 1 a 100) que ha “pensado” el ordenador.

---

## Parte 1: Configuración del Juego (30 minutos)

1. Crea un archivo HTML básico que contenga un formulario con un campo de entrada para que el usuario adivine un número y un botón para enviar su respuesta. También incluye un área para mostrar mensajes al usuario (por ejemplo, "Demasiado alto", "Demasiado bajo", "¡Correcto!").
2. Genera un número aleatorio entre 1 y 100 cuando se carga la página. Este será el número que el usuario debe adivinar. Utiliza `Math.random()` para esto.

## Parte 2: Control del Formulario y Lógica del Juego (1 hora)

3. Agrega un evento al botón que capture el valor ingresado por el usuario cuando se envía el formulario.
4. Compara el número ingresado por el usuario con el número aleatorio generado. Utiliza condicionales (`if`, `else if`, `else`) para determinar si el número es demasiado alto, demasiado bajo o correcto. Muestra un mensaje apropiado en el área designada.
5. Implementa un contador de intentos que se incremente cada vez que el usuario envíe una respuesta. Muestra el número de intentos realizados en el área de mensajes.

## Parte 3: Mejora del Juego (1 hora)

6. Agrega un botón de reinicio que permita al usuario reiniciar el juego. Este botón debe generar un nuevo número aleatorio y restablecer el contador de intentos y los mensajes.

7. Implementa un límite de intentos (por ejemplo, 10 intentos). Si el usuario no adivina el número en ese límite, muestra un mensaje que indique que ha perdido.
8. En caso de perder (superar el número de intentos), revela el número correcto en el área de mensajes.

#### Parte 4: Manipulación del DOM (30 minutos)

9. Mejora la interfaz de usuario utilizando JavaScript para cambiar dinámicamente el estilo de los mensajes (por ejemplo, cambiar el color del texto según si el mensaje es de éxito o error).
  10. Agrega un historial de intentos que muestre todos los números que el usuario ha ingresado hasta el momento. Utiliza un elemento de lista (<ul> o <ol>) para mostrar este historial y actualízalo cada vez que el usuario envíe un nuevo intento.
- 

#### Criterios de Evaluación

- Funcionalidad: El juego debe funcionar correctamente según las especificaciones.
- Uso de JavaScript: Se evaluará el uso adecuado de variables, condicionales, bucles y manipulación del DOM.
- Código Limpio: El código debe estar bien estructurado y comentado.
- Interactividad: La experiencia del usuario debe ser fluida y agradable.