

Parte del código	TypeScript (scripts.ts)	JavaScript (scripts.js)	Diferencia
Declaración de randomNumber	let randomNumber: number = Math.floor(Math.random() * 100) + 1;	var randomNumber = Math.floor(Math.random() * 100) + 1;	TS usa let y tipado number, JS usa var sin tipo.
Declaración de variables DOM	let guesses = document.querySelector(".guesses") as HTMLParagraphElement;	var guesses = document.querySelector(".guesses");	TS usa as HTMLParagraphElement para tipar y asegurar el elemento.
Campo de entrada	let guessField: HTMLInputElement = document.querySelector(".guessField")!;	var guessField = document.querySelector(".guessField");	TS especifica HTMLInputElement y usa ! (no nulo).
Contador de intentos	let guessCount: number = 1;	var guessCount = 1;	TS tipa como number.
Botón de reinicio	let resetButton: HTMLButtonElement;	var resetButton;	TS tipa como HTMLButtonElement.
Función checkGuess	function checkGuess(): void { ... }	function checkGuess() { ... }	TS especifica que no retorna nada (void).
Conversión a número	const userGuess: number = Number(guessField.value);	var userGuess = Number(guessField.value);	TS usa const y tipado number.
Selección y limpieza de párrafos	const resetParas = document.querySelectorAll(".resultParas p");	var resetParas = document.querySelectorAll(".resultParas p");	TS usa const y tipado implícito como NodeListOf<HTMLParagraphElement>.
Eliminación del botón	(resetButton.parentNode as HTMLElement)?.removeChild(resetButton);	`(_a = resetButton.parentNode) === null	
Eventos	guessSubmit.addEventListener("click", checkGuess);	igual	Misma lógica, pero en TS guessSubmit está tipado.
Final del juego	Lógica idéntica	Lógica idéntica	Sin cambios funcionales, solo tipado.

Aspecto	Código Mdn	Código TS de JS
Declaración de variables	Mezcla let y const, algunas redefiniciones (guessField aparece dos veces).	Usa solo var, sin redefiniciones.
Código extra/pruebas	Incluye variables de prueba (name, hello, greeting), concatenaciones, console.log(i), modificaciones de estilo a guesses, asignaciones como guessField.value = "Hola".	No contiene código de prueba ni modificaciones extra de estilo, solo lo necesario para el juego.
Función checkGuess()	Se define dos veces: una como marcador de posición (alert) y luego con la lógica completa.	Definida una sola vez directamente con la lógica final.
Limpieza de párrafos (resetGame() y al inicio)	Usa bucle for clásico para limpiar párrafos. No usa encadenamiento opcional al eliminar el botón.	Usa .forEach() para limpiar párrafos. Usa encadenamiento opcional (?.) al eliminar el botón.
Función Estilos en DOM	Cambia estilos de guesses (fondo amarillo, fuente 200%, padding, sombra).	Solo cambia el fondo de lastResult (verde, rojo, blanco).
Estructura general	Menos ordenada, con código repetido y pruebas mezcladas con lógica del juego.	Estructurada, con comentarios descriptivos y sin código innecesario.
Comentarios	Casi no tiene explicaciones en el código.	Incluye comentarios claros en cada sección del código.