

Cambio de color:

```
▼ Cambio de color/cambio_color_erroneo.js
... @@ -1,5 +1,5 @@
1 1  function cambiarColor() {
2 2      const colores = ["red", "blue", "green", "yellow", "purple"];
3 -   const indice = Math.random() * colores.length;
3 +   const indice = Math.floor(Math.random() * colores.length);
4 4      document.body.style.backgroundColor = colores[indice];
5 5  }
```

El número aleatorio generado no será un entero, por lo que al usar `Math.floor`, convertimos el número al entero más cercano.

También podríamos usar `Math.ceil`, ya que técnicamente ambos hacen lo mismo.

`Math.floor` transforma números como 4.9 o 4.1 en 4, y `Math.ceil` transforma números como 4.1 o 4.9 en 5

Contador:

```
▼ Contador/contador_erroneo.js
... @@ -1,5 +1,5 @@
1 + let contador = 0;
1 2  function incrementar() {
2 -   let contador = 0;
3 3      contador++;
4 4      document.getElementById("contador").innerText = contador;
5 5  }
```

Inicializamos el contador dentro de la función, esto significa que cada vez que se hace clic en el botón, pasa de 0 a 1 y si se hace clic nuevamente, simplemente lo vuelve a hacer. Al inicializarlo fuera de la función, realmente podemos permitirle crecer.

Hola Mundo:

```
▼ Hola Mundo/hola_mundo_erroneo.js
... @@ -1,3 +1,3 @@
1 1  function cambiarMensaje() {
2 -   document.getElementById("mensaje").innerText = "¡Hola Mundo desde JavaScript!";
2 +   document.getElementById("mensaje").innerText = "¡Hola Mundo desde JavaScript!";
3 3  }
```

El nombre era malo. "mensaje" a "mensaje".

Suma:

```
Suma/suma_erronea.js
... @@ -1,6 +1,6 @@
1 1 function calcularSuma() {
2 2     const num1 = document.getElementById('num1').value;
3 3     const num2 = document.getElementById('num2').value;
4 -   const suma = num1 + num2;
4 +   const suma = parseFloat(num1) + parseFloat(num2);
5 5     document.getElementById('resultado').innerText = `La suma es: ${suma}`;
6 6 }
```

Antes de sumar, tenemos que transformar el num1 y num2 a Float/Int.
También, podemos poner el parseFloat en la creación de num1 y num2.

Valicación:

```
... @@ -1,7 +1,11 @@
1 - function validarFormulario(){
2 -   if(validarFormulario2()){
1 + function validarFormulario()
2 + {
3 +   if(validarFormulario2())
4 +   {
3 5       alert("Formulario correcto!");
4 -   }else{
6 +   }
7 +   else
8 +   {
5 9       alert("Formulario incorrecto!");
6 10    }
7 11  }
@@ -14,16 +18,16 @@ function validarFormulario2() {
14 18     if (nombre.length < 3) {
15 19         mensaje.innerText = "El nombre debe tener al menos 3 caracteres.";
16 20         mensaje.style.color = "red";
17 -     return true;
21 +     return false;
18 22     }
19 23
20 24     if (!email.includes('@') || !email.includes('.')) {
21 25         mensaje.innerText = "El email no es válido.";
22 26         mensaje.style.color = "red";
23 -     return true;
27 +     return false;
24 28     }
25 29
26 30     mensaje.innerText = "Formulario enviado correctamente.";
27 31     mensaje.style.color = "green";
28 -     return false;
32 +     return true;
29 33 }
```

El return estaba malo, estaba dando instrucciones falsas. En esta situación, estaba dando true cuando tenía que dar false, y false cuando tenía que dar true.