

# T2Actividad5: Crear una calculadora con operaciones básicas

## Ejercicio: Calculadora con operaciones básicas y manejo de errores

### Enunciado

El objetivo del ejercicio es:

1. Crear un objeto `calculator` con las siguientes propiedades y métodos:
  - **Propiedades:**
    - `a` : Almacena el primer valor introducido por el usuario.
    - `b` : Almacena el segundo valor introducido por el usuario.
  - **Métodos:**
    - `read()` : Solicita los valores `a` y `b` mediante `prompt` y los almacena en las propiedades del objeto.
    - `sum()` : Calcula y retorna la suma de `a` y `b`.
    - `mul()` : Calcula y retorna el producto de `a` y `b`.
    - `rest()` : Calcula y retorna la resta de `a` menos `b`.
    - `divi()` : Calcula y retorna el cociente de `a` entre `b`. Si `b` es 0, muestra un mensaje de error.
    - `modu()` : Calcula y retorna el módulo de `a` entre `b`. Si `b` es 0, muestra un mensaje de error.
2. Solicitar al usuario que introduzca la operación que desea realizar:
  - Operaciones válidas: `"sum"`, `"mul"`, `"rest"`, `"divi"`, `"modu"`.
  - Si la operación no es válida, mostrar un mensaje de error mediante `alert`.
3. Mostrar el resultado de la operación seleccionada en un `alert`.

---

### Análisis del Problema

1. **Definición del objeto:**
  - Crear un objeto `calculator` que almacene los valores introducidos y tenga métodos para realizar las operaciones matemáticas solicitadas.
2. **Entrada de datos:**
  - Pedir al usuario los valores `a` y `b` mediante `prompt`.
  - Pedir al usuario que introduzca la operación a realizar.
3. **Validación de la operación:**
  - Usar una estructura `switch` para seleccionar la operación.
  - Mostrar un mensaje de error si la operación no es válida.
4. **Cálculo y salida:**
  - Calcular el resultado de la operación seleccionada y mostrarlo en un `alert`.
  - Manejar errores como la división por cero.

---

### Solución Propuesta

Se implementó una solución en JavaScript que utiliza:

- Un objeto `calculator` con las propiedades y métodos indicados.

- Una función principal `ejecutarCalculadora()` para orquestar el flujo del programa.
- Un manejo adecuado de errores, como la división por cero y operaciones inválidas.

## Código Fuente

### HTML

El código puede ser ejecutado directamente sin un archivo HTML específico.

### JavaScript

El código JavaScript es el siguiente:

```
const calculator = {
  a: 0,
  b: 0,

  read() {
    this.a = parseFloat(prompt("Introduce el valor de a:"));
    this.b = parseFloat(prompt("Introduce el valor de b:"));
  },

  sum() {
    return this.a + this.b;
  },

  mul() {
    return this.a * this.b;
  },

  rest() {
    return this.a - this.b;
  },

  divi() {
    if (this.b === 0) {
      alert("Error: División por cero no permitida.");
      return undefined;
    }
    return this.a / this.b;
  },

  modu() {
    if (this.b === 0) {
      alert("Error: División por cero no permitida.");
      return undefined;
    }
    return this.a % this.b;
  },
};

function ejecutarCalculadora() {
  calculator.read();
  const operacion = prompt(
    'Introduce la operación a realizar: "sum", "mul", "rest", "divi", "modu"'
  );
```

```

    ).toLowerCase();

    let resultado;
    switch (operacion) {
        case "sum":
            resultado = calculator.sum();
            break;
        case "mul":
            resultado = calculator.mul();
            break;
        case "rest":
            resultado = calculator.rest();
            break;
        case "divi":
            resultado = calculator.divi();
            break;
        case "modu":
            resultado = calculator.modu();
            break;
        default:
            alert("Error: Operación no válida.");
            return;
    }

    if (resultado !== undefined) {
        alert(`El resultado de la operación "${operacion}" es: ${resultado}`);
    }
}

ejecutarCalculadora();

```

## Pruebas Realizadas

### 1. Caso válido:

- Entrada: `a = 10, b = 5, operación = "sum"`.
- Salida esperada: `El resultado de la operación "sum" es: 15`.

### 2. División por cero:

- Entrada: `a = 10, b = 0, operación = "divi"`.
- Salida esperada: `Error: División por cero no permitida.`

### 3. Operación inválida:

- Entrada: `a = 10, b = 5, operación = "invalid"`.
- Salida esperada: `Error: Operación no válida.`

## Conclusión

El programa cumple con los requisitos del enunciado:

- Implementa todas las operaciones solicitadas.
- Maneja errores como la división por cero y operaciones no válidas.
- Permite al usuario interactuar mediante `prompt` y muestra resultados con `alert`.