



Programación Imperativa

Hagamos una calculadora

Con el conocimiento que adquirimos hasta el momento, estamos en condiciones de desarrollar nuestra primera aplicación: una calculadora. La misma consta de diferentes niveles de dificultad. Te acompañaremos para superarlos.



Calculadora - Nivel I

1. Crear un archivo `calculadora.js` que contendrá las distintas operaciones a realizar de nuestra calculadora.
2. Crear una función `sumar`, esta deberá recibir dos parámetros y retornar la suma de los mismos.

```
function sumar(num1, num2) {  
    return num1 + num2;  
}
```

3. Crear una función `restar`, la cual deberá recibir dos parámetros y retornar la resta del primero menos el segundo.

```
function restar(num1, num2) {  
    return num1 - num2;  
}
```



4. Crear una función *multiplicar*, la misma deberá recibir dos parámetros y retornar el resultado de su multiplicación.

```
function multiplicar(num1, num2) {  
    return num1 * num2;  
}
```

5. Crear una función *dividir*, la cual recibirá dos parámetros y retornar el resultado de la división del primero sobre el segundo.

```
function dividir(num1, num2) {  
    return num1 / num2;  
}
```



Calculadora - Nivel II

Superado el Nivel I, agregamos un poco más de dificultad, probaremos nuestras funciones. Te recomendamos usar la función *prompt()* con la que estuvimos trabajando en la semana para pedirle al usuario que ingrese los números con los que posteriormente ejecutaremos las funciones de nuestra calculadora. Recordá guardar esos datos del usuario en diferentes variables.

1. En el archivo `calculadora.js` —debajo de nuestras 4 funciones hechas con anterioridad— creá un *console.log* en el cual vas a poner un string para indicar que debajo de él comenzarás a probar las funciones, algo así como un “separador de código”. Por ejemplo:

```
console.log("----- Testeo de Operaciones / Calculadora -----")
```

```
console.log("***** NIVEL II - Testeo de Calculadora*****");
```



2. Ejecutar la función que permite sumar y la función que permite restar pasando como argumentos dos números cualesquiera. Mostrar en consola los resultados.

```
let rdoSuma = sumar(6,4);  
console.log("El resultado de la suma es: " + rdoSuma);  
  
let rdoResta = restar(16,4);  
console.log("El resultado de la resta es: " + rdoResta);
```

3. Ejecutar la función que permite multiplicar pasando como argumentos dos números cualesquiera. Mostrar en consola el resultado.

```
let rdoMultiplicacion = multiplicar(2,20);  
console.log("El resultado de la multiplicacion es: " +  
rdoMultiplicacion);
```

4. Ejecutar la función que permite dividir pasando como argumentos dos números cualesquiera. Mostrar en consola el resultado.

```
let rdoDivision = dividir(100,5);  
console.log("El resultado de la division es: " +  
rdoDivision);
```

5. Ejecutar la función que permite dividir pasando, ahora, como uno de los dos argumentos, el número **cero**. Mostrar en consola el resultado.

```
let rdoDivision = dividir(100,0);  
  
console.log("El resultado de la division es: " +  
rdoDivision);
```

En el punto 5, se nos requiere ejecutar la función *dividir*, pero si pasamos el cero como divisor, nos arroja un error. No te preocupes. La idea es familiarizarte con que, durante toda tu carrera como programador —y durante tu proceso formativo—, vas a encontrarte con errores. El objetivo es tratar de aprender a lidiar con esa frustración de una manera lo más productiva posible. ¡No te desanimes!



Calculadora - Nivel III - Funciones extras

Si llegamos hasta aquí, podemos quedarnos tranquilos con lo realizado. A continuación desarrollaremos funciones extras a nuestra calculadora, en las cuales utilizaremos las funciones ya creadas. Podemos realizar la prueba de estas luego de desarrollar cada una.

1. Crear una función que se llame *cuadradoDeUnNumero*, la cual recibe un número como parámetro y deberá retornar ese número elevado al cuadrado.

Importante: *cuadradoDeUnNumero()* deberá utilizar la función *multiplicar()* para calcular el cuadrado del parámetro ingresado a *cuadradoDeUnNumero()*

```
function cuadradoDeUnNumero(num) {  
    return multiplicar(num,num)  
}
```

2. Crear la función *promedioDeTresNumeros*, la misma deberá calcular el promedio de 3 números, que serán ingresados por parámetro.

Importante: *promedioDeTresNumeros()* deberá utilizar algunas funciones previamente creadas de nuestra calculadora.



```
function promedioDeTresNumeros(num1, num2, num3) {  
    let sum = sumar(num1, num2);  
    return sumar(sum, num3) / 3;  
}
```

3. Crear la función *calcularPorcentaje*. Esta recibirá dos parámetros: el número total y el porcentaje que se le quiere calcular. Deberá retornar el x% del total. Ejemplo: *calcularPorcentaje(300, 15)* debe retornar 45, ya que es el 15% de 300. **Importante:** *calcularPorcentaje()* deberá utilizar algunas funciones previamente creadas de nuestra calculadora.

```
function calcularPorcentaje(numero, porcentaje) {  
    return multiplicar(numero, dividir(porcentaje, 100));  
}
```

4. Crear una función *GeneradorDePorcentaje* que recibe dos parámetros y deberá retornar el porcentaje del primero con respecto al segundo. **Importante:** *GeneradorDePorcentaje()* deberá utilizar algunas funciones previamente creadas de nuestra calculadora. Ejemplo: *GeneradorDePorcentaje(2, 200)* deberá retornar 1, ya que 2 es el 1% de 200.

```
function generadorDePorcentaje(num1, num2) {  
    let primeraOperacion = multiplicar(num1, 100);  
    let segundaOperacion = dividir(primerOperacion, num2);  
    return segundaOperacion;  
}
```

¡Felicidades! Logramos aplicar y comprender conceptos que nos servirán a lo largo de nuestra carrera. Ahora nos queda pensar en mejoras para nuestra calculadora con los conocimientos que adquiriremos en el futuro.