

EJERCICIOS DE JAVASCRIPT

EJERCICIO 1

Escribe un programa que pida dos números al usuario usando “prompt” y escriba en pantalla cuál es mayor. Para averiguar cuál es mayor se debe definir una función que recibe los dos números y devuelve el mayor.

La sintaxis de “prompt” es:

```
let a = prompt("Introduce algo");
```

Lo que hace es mostrar una ventana con el texto “Introduce algo” y deja escribir en esa ventana un valor que luego devuelve.

SOLUCIÓN:

```
function esMayor(num1,num2){
    if(!num1 || !num2) return -1;
    if(num1>num2)
        return num1;
    else if(num2>num1)
        return num2;
    else return 0;
}

//CÓDIGO DEL PROGRAMA
let n1=prompt("Dame un número");
let n2=prompt("Dame otro número");
let resu=esMayor(n1,n2);

if(resu===-1)
    console.log("Me faltan numeros");
else if(resu!==0) console.log(`El mayor es ${resu}`);
else console.log("Son iguales");
```

EJERCICIO 2

Escribe un programa que pida un número al usuario y diga si es divisible por 2. Usar una función que reciba el número y devuelva true si es divisible por 2 y false en otro caso.

```
function esDivisibleDos(num){
    if(!num) return false;
    if(num%2==0)
        return true;
```

```
    else return false;
}

let numero=prompt("Dame un número");
if(esDivisibleDos(numero))
    console.log("Es divisible");
else
    console.log("No es divisible");
```

EJERCICIO 3

Escribe un programa que pida la edad al usuario y nos dice si es mayor de edad o en caso contrario dice “demasiado joven”. Si se introduce algo que no es un número también dice “Error. No es un número”. Si se introduce una edad menor de 1 o mayor de 140 también da error de edad incorrecta. Usar una función que reciba la edad y devuelva los textos correspondientes.

```
function esMayorEdad(edad){
    if(!edad) return "Error no es un número";
    if(edad<1 || edad>140) return "Error: edad incorrecta";
    if(edad>=18)
        return "Mayor de edad";
    else
        return "Demasiado joven";
}

let edad=prompt("Dame tu edad");
console.log(esMayorEdad(edad));
```

EJERCICIO 4

Dado un array de salarios, escribe una función llamada “ModificarSalario” que reciba 3 parámetros:

- un array de salarios
- otra función B que se aplicará a cada elemento de ese array de salarios y que permitirá aplicarle una subida de salarios a todos, una bajada, o incluso una subida a los que cumplan x condiciones. Para probarlo, tendréis que inventar al menos dos funciones B diferentes:
 - Una por ejemplo, que aumente un 10% el valor que se le pasa
 - Otra lo reduce en un 5% el valor que se le pasa
- Una letra que indica a qué elementos hay que modificar el salario:
 - ‘T’ → sería a todos
 - ‘P’ → los pares
 - ‘500’ → los de menos de 500 euros

SOLUCIÓN:

```
//DEFINICIÓN DE LAS FUNCIONES
let aumentar10Salario = function (numero) { return 1.1 * numero; }
let reducir5Salario = function (numero) { return 0.95 * numero; }

function modificarSalario(arraySalarios,
                          funcionModificacion,
                          aQuien) {
  switch (aQuien) {
    case 'T': for (let i = 0; i < arraySalarios.length; i++) {
      arraySalarios[i] =
        funcionModificacion(arraySalarios[i]).toFixed(2);
    }
    break;
    case 'P': for (let i=0; i < arraySalarios.length; i+=2){
      arraySalarios[i] =
        funcionModificacion(arraySalarios[i]).toFixed(2);
    }
    break;
    case '500': for (let i=0; i<arraySalarios.length; i++){
      if(arraySalarios[i]<500)
        arraySalarios[i]=
          funcionModificacion(arraySalarios[i]).toFixed(2);
    }
    break;
  }
}

//PROGRAMA PRINCIPAL
let salariosEmpresa = [450, 1500, 300, 990, 3500, 2489, 1188];

// Pruebo a aumentar un 10% a los de menos de 500
modificarSalario(salariosEmpresa,aumentar10Salario, '500');
console.log(`Array con subida de sueldo-MENOS 500: ${salariosEmpresa}`)

//Pruebo a aumentar un 10% salario a todos
modificarSalario(salariosEmpresa, aumentar10Salario, 'T');
console.log(`Array con subida de sueldo-TODOS: ${salariosEmpresa}`);

// Pruebo a aumentar un 10% sólo a los pares
modificarSalario(salariosEmpresa, aumentar10Salario, 'P');
console.log(`Array con subida de sueldo-PARES: ${salariosEmpresa}`);

// Pruebo a disminuir a todos un 5%
```

```
modificarSalario(salariosEmpresa, reducir5Salario, 'T');  
console.log(`Array con reducción de sueldo-TODOS: ${salariosEmpresa}`)
```

OJO → SOLUCIÓN QUE NO FUNCIONA: usando for-of

```
//DEFINICIÓN DE LAS FUNCIONES  
let aumentar10Salario = function (numero) { return 1.1 * numero; }  
let reducir5Salario = function (numero) { return 0.95 * numero; }  
  
function modificarSalario(arraySalarios,  
                           funcionModificacion,  
                           aQuien) {  
  switch (aQuien) {  
    case 'T': for(let sueldo of arraySalarios){  
               sueldo = funcionModificacion(sueldo).toFixed(2);  
               console.log(sueldo);  
            }  
            break;  
    case 'P': for (let sueldo of arraySalarios){  
               sueldo = funcionModificacion(sueldo).toFixed(2);  
               console.log(sueldo);  
            }  
            break;  
    case '500': for (let sueldo of arraySalarios){  
                 if(sueldo<500){  
                   sueldo=funcionModificacion(sueldo).toFixed(2);  
                   console.log(sueldo);  
                 }  
            }  
            break;  
  }  
}  
  
//PROGRAMA PRINCIPAL  
let salariosEmpresa = [450, 1500, 300, 990, 3500, 2489, 1188];  
// Pruebo a aumentar un 10% a los de menos de 500  
modificarSalario(salariosEmpresa,aumentar10Salario, '500');  
console.log(`Array con subida de sueldo-MENOS 500: ${salariosEmpresa}`)
```

```
//Pruebo a aumentar un 10% salario a todos
modificarSalario(salariosEmpresa, aumentar10Salario, 'T');
console.log(`Array con subida de sueldo-TODOS: ${salariosEmpresa}`);

// Pruebo a aumentar un 10% sólo a los pares
modificarSalario(salariosEmpresa, aumentar10Salario, 'P');
console.log(`Array con subida de sueldo-PARES: ${salariosEmpresa}`);

// Pruebo a disminuir a todos un 5%
modificarSalario(salariosEmpresa, reducir5Salario, 'T');
console.log(`Array con reducción de sueldo-TODOS: ${salariosEmpresa}`)
```

OTRA SOLUCIÓN: usando el método MAP de los ARRAYS:

```
//DEFINICIÓN DE LAS FUNCIONES
let aumentar10Salario = function (numero) { return 1.1 * numero; }
let reducir5Salario = function (numero) { return 0.95 * numero; }

function modificarSalario(arraySalarios,
                           funcionModificacion,
                           aQuien) {

    let arrayResu;
    switch (aQuien) {

        case 'T': arrayResu=arraySalarios.map(funcionModificacion);
                   break;
        case 'P': arrayResu=arraySalarios.map((numero,pos)=>{
                        if(pos%2 == 0) return funcionModificacion(numero);
                    });
                   break;
        case '500': arrayResu=arraySalarios.map(numero=>{
                        if(numero<500) return funcionModificacion(numero);
                    });
                   break;
    }

    return arrayResu;
}

//PROGRAMA PRINCIPAL
let salariosEmpresa = [450, 1500, 300, 990, 3500, 2489, 1188];
```

```
// Pruebo a aumentar un 10% a los de menos de 500
console.log(`Array con subida de sueldo-
MENOS 500: ${modificarSalario(salariosEmpresa,aumentar10Salario, '500')}`)

//Pruebo a aumentar un 10% salario a todos
console.log(`Array con subida de sueldo-
TODOS: ${modificarSalario(salariosEmpresa, aumentar10Salario, 'T')}`);

// Pruebo a aumentar un 10% sólo a los pares
console.log(`Array con subida de sueldo-
PARES: ${modificarSalario(salariosEmpresa, aumentar10Salario, 'P')}`);

// Pruebo a disminuir a todos un 5%
console.log(`Array con reducción de sueldo-
TODOS: ${modificarSalario(salariosEmpresa, reducir5Salario, 'T')}`)
```

EJERCICIO 5

Modifica el ejercicio 4 para que las funciones B de subir y bajar el sueldo sean genéricas y permitan subir o bajar el sueldo un porcentaje que se le indicará por parámetro. Para eso debéis convertirlas en funciones que devuelvan otra función.

```
//DEFINICIÓN DE LAS FUNCIONES
let aumentarSalario = function(porcentaje) {
    return function (numero) {
        return numero*(1+porcentaje/100);
    }
};

let reducirSalario = function(porcentaje){
    return function (numero) {
        return numero*(1-porcentaje/100);
    }
};

function modificarSalario(arraySalarios,
                           funcionModificacion,
                           aQuien) {
    switch (aQuien) {
        case 'T': for (let i = 0; i < arraySalarios.length; i++) {
                    arraySalarios[i] =
                        funcionModificacion(arraySalarios[i]).toFixed(2);
                }
                break;
        case 'P': for (let i=0; i < arraySalarios.length; i+=2){
                    arraySalarios[i] =
```

```
        funcionModificacion(arraySalarios[i]).toFixed(2);
    }
    break;
case '500': for (let i=0; i<arraySalarios.length; i++){
    if(arraySalarios[i]<500)
        arraySalarios[i]=
            funcionModificacion(arraySalarios[i]).toFixed(2);
    }
    break;
}

}

//PROGRAMA PRINCIPAL
let salariosEmpresa = [450, 1500, 300, 990, 3500, 2489, 1188];

// Pruebo a aumentar un 10% a los de menos de 500
modificarSalario(salariosEmpresa,aumentarSalario(10), '500');
console.log(`Array con subida de sueldo 10%-MENOS 500: ${salariosEmpresa}`)

//Pruebo a aumentar un 5% salario a todos
modificarSalario(salariosEmpresa, aumentarSalario(5), 'T');
console.log(`Array con subida de sueldo 5%-TODOS: ${salariosEmpresa}`);

// Pruebo a aumentar un 20% sólo a los pares
modificarSalario(salariosEmpresa, aumentarSalario(20), 'P');
console.log(`Array con subida de sueldo 20%-PARES: ${salariosEmpresa}`);

// Pruebo a disminuir a todos un 10%
modificarSalario(salariosEmpresa, reducirSalario(10), 'T');
console.log(`Array con reducción de sueldo 10%-TODOS: ${salariosEmpresa}`)
```