

## Ejercicios de funciones 2

1. Un programa que tenga varias funciones:
  - Una función `crearArray()` que se encargue de crear un array con un tamaño random entre 1 y 7 y lo devuelva.
  - Una función `llenarArray()` que reciba un array por parámetros y lo rellene con números random entre 1 y 15.
  - Una función `mostrarArray()` que reciba un array por parámetros y se encargue de mostrar por consola (o por el `document.write`) cada uno de los elementos del array.

El flujo principal se encarga de ejecutar estas funciones secuencial mente.

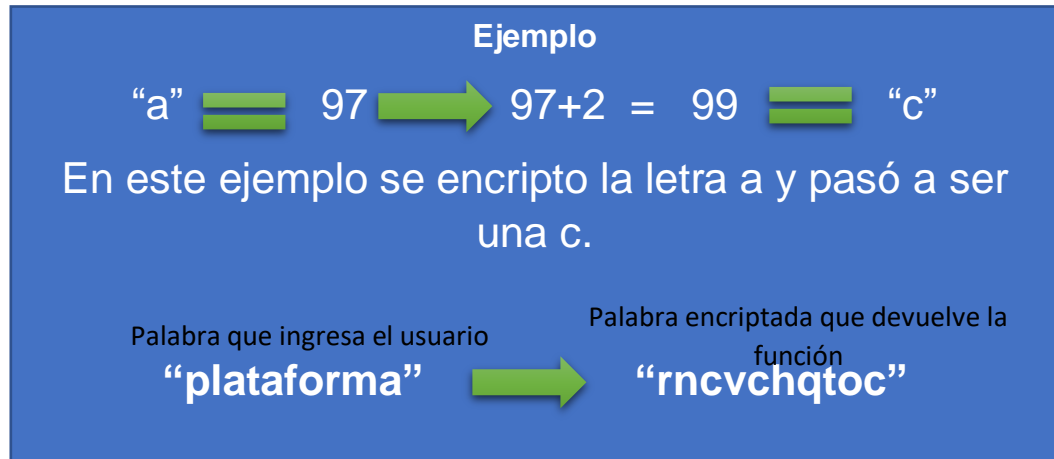
2. Una función que se encargue de ordenar tres números pasados por parámetros y los muestre con `alert()` (no hay que retornar).
3. Un sistema de gestión de alumnos de un colegio. Los alumnos tienen la siguiente estructura:

```
alumno{  
    nombre,  
    apellido,  
    direccion,  
    telefono,  
    email  
}
```

Las operaciones son:

- `insertarAlumno()` recibe el array de alumnos y se encarga de pedir los datos, crear el objeto y agregarlo al array.
- `eliminarAlumno()` recibe el array y el nombre del alumno a eliminar. se encarga de buscar al alumno y de eliminarlo del array.
- `existeAlumno()` recibe el array y el nombre del alumno. Se encarga de comprobar si existe el alumno en el array. De ser así retorna `true`, de lo contrario retornará `false`.

4. Una función que reciba como argumento una palabra que escriba el usuario por teclado y se encargue de encriptarlo y de devolverlo. (Para encriptarlo, la función recorrerá la palabra y a cada letra le sumará un 2 al código ASCII).



5. Al ejercicio anterior agregarle una función que se encargue de desencriptar (en lugar de sumar 2, se le restará para desencriptar).
6. Un juego de adivinar un número secreto.
- Los números se almacenan en un array de 7 posiciones (es decir, guarda 8 números).
  - primero se le pedirá al jugador ingresar dichos números y al terminar, el programa deberá elegir uno de esos números (ese número que elija será el número secreto que el jugador debe adivinar).
  - Luego de todo esto el programa avisará de que empezó el juego y le va pidiendo de forma iterativa números.
  - Cuando el usuario ingrese un número el programa debe decirle si el número que ingresó es mayor o menor al número secreto.
  - El programa deja de pedir números cuando el usuario adivine el número secreto.
  - Al adivinar, se mostrará por pantalla las felicitaciones y el número de intentos.

7. Una función que reciba un string y retorne un array que tenga en cada posición las letras del string.

**Ejemplo**

```
const array = convertir("hola");  
console.log(array);
```



A green arrow points from the code to a green box containing the array `['h', 'o', 'l', 'a']`.

8. Un sistema que pida un DNI y calcule la letra correspondiente al DNI ingresados por el usuario.

Para esto el programa le pedirá el DNI al usuario y se lo da a una función que devuelva la letra correspondiente al DNI.

Para calcular la letra, se tomará el resto de la división del DNI entre 23 (es decir, no se tomará el resultado, solo el resto de la división).

$43017108 / 23 = 1870309.043478261$  (resultado de la división)  
 $43017108 \% 23 = 1$  (resto de la división)

Se tomará el resto y se buscará en un array de letras, la letra correspondiente al resto.

POSICION	LETRA
0	T
1	R
2	W
3	A
4	G
5	M
6	Y
7	F
8	P
9	D
10	X
11	B
12	N
13	J
14	Z
15	S
16	Q
17	V
18	H
19	L
20	C
21	K
22	E

Por ejemplo, si introduzco 70588387, el resultado sera de 7 que corresponde a 'F'.

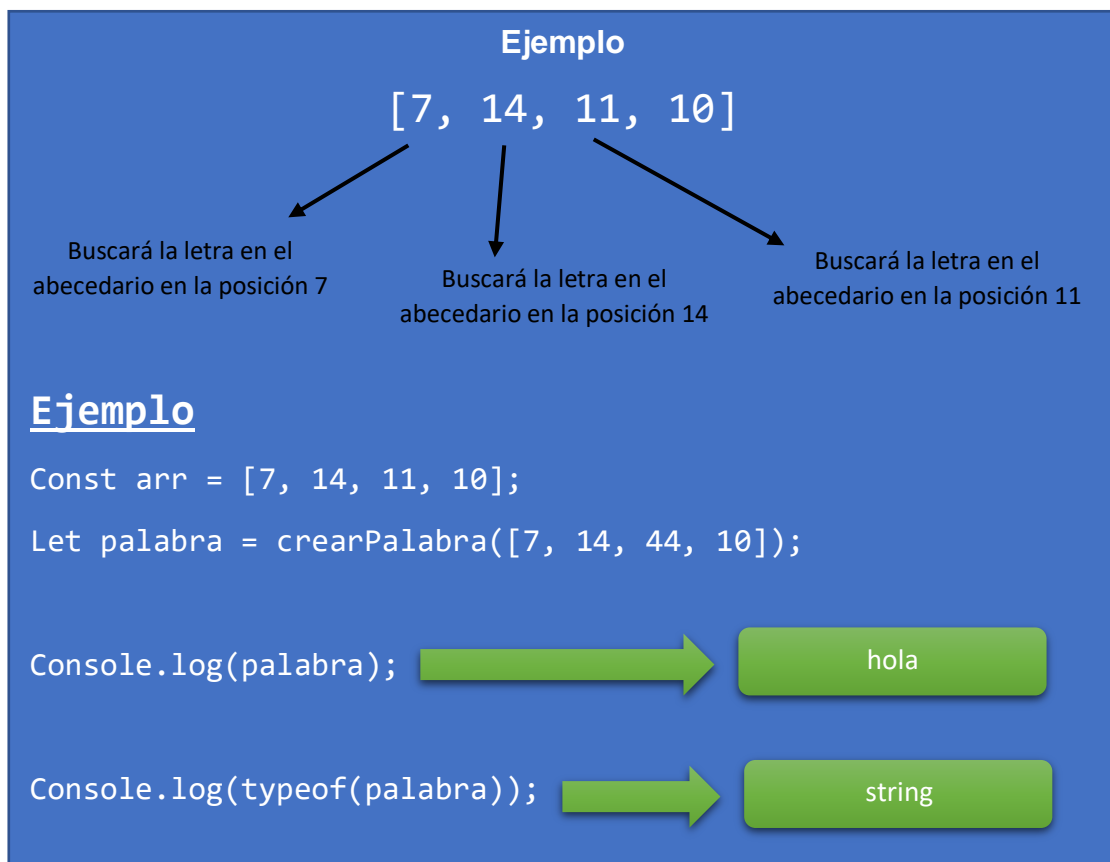
El resto, no el resultado

## 9. Un sistema en donde pueda armar palabras escribiendo números.

Para esto, habrá una función que reciba un array de números (por ejemplo [7, 14, 11, 0]).

Esa función tendrá en su interior un array llamada abecedario con todas las letras del abecedario (exceptuando la ñ).

Luego va a ir armando un nuevo array en donde cada posición tendrá una letra del abecedario. El array que recibe por parámetro guarda las posiciones.



## **Ejercicios ya hechos (pero con funciones)**

En este apartado se volverán a ver algunos los ejercicios que ya completaste antes, pero esta vez vas a resolverlos usando funciones. Esto servirá para que aprendas a identificar de que funciones hacer, cuando, etc. Es decir, no se mencionará como implementar funciones, sino que vas a pensar como lo implementarías vos.

- 1.** Un programa que solicite al usuario la cantidad de números que quiere ingresar. Luego pedir uno a uno los números y almacenarlos en un array. Luego de eso, el programa deberá sumar todos los números del array y mostrará el resultado por pantalla.
- 2.** Un programa en donde se tengan dos arrays, uno que guarde números pares y otro que guarde números impares. Luego pedir de forma iterativa números que serán guardados en los arrays teniendo en cuenta que, si el número es par, se guarda en el array de pares y si es impar, se guarda en el array de impares. El programa termina cuando el usuario ingrese un cero y muestra todos los arrays.

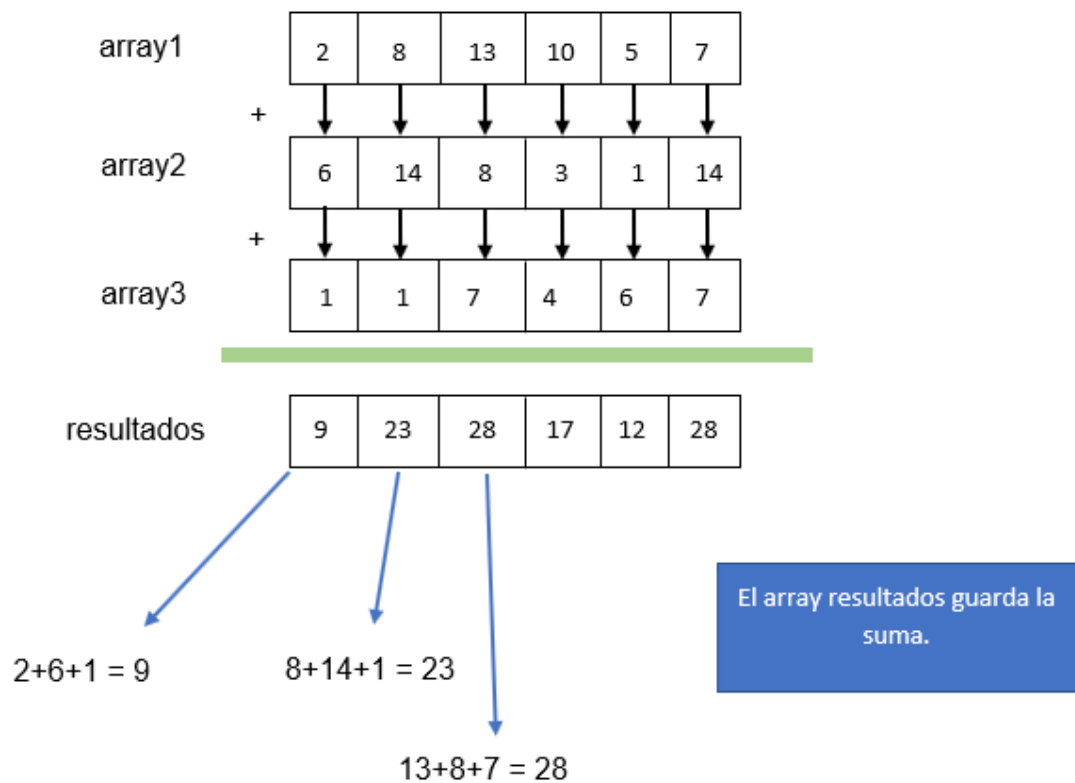
- 3.** Hacer tres arrays con un tamaño random entre 1 y 10. Ojo, para hacer esto, primero se genera un número aleatorio y luego se crean los tres arrays y el número aleatorio que ya fue generado será el tamaño de los tres.

Luego, llenarlos con números aleatorios entre 1 y 15.

Para finalizar, crear un cuarto array en donde la primera posición guardará la suma entre el primer elemento de los tres arrays. La segunda posición guarda la suma del segundo elemento de los tres arrays y así hasta completar el cuarto array.

Mostrar el array de los resultados.

#### Ejemplo



**4.** Dado un array de colores que contiene: azul, amarillo, rojo, verde y rosa. Pedir al usuario que ingrese un color y comprobar si se encuentra en el array. Si no se encuentra, agregarlo con `push()` en la posición que quiera el usuario.

**5.** Una compañía dedicada al alquiler de autos cobra un monto fijo de \$300000 para los primeros 300km. Para los que tienen entre 300km y 1000km recorridos, cobra un monto adicional de \$15000 por cada km. Para más de 1000km cobra un monto adicional de \$10000 por cada km. Al monto final se incluye un IVA del 20%. Determinar el monto a pagar por el alquiler y el monto final con el impuesto incluido. (la compañía te pregunta cuantos Km hiciste y según ese número hace el cálculo).

**6.** Parte 1:

En un almacén se hacen dos descuentos:

- Descuento del 10% al precio final si el cliente compra entre 5 y 10 productos.

- Descuento del 5% al precio final si el cliente compra más de 10 productos.

Determinar según la cantidad de productos que quiera el cliente, qué descuento se aplica al precio final y mostrarlo por consola.

Parte 2:

Supongamos que el cliente quiera comprar x cantidad de un producto que vale \$67. Al código ya escrito agregarle el calcular el precio final con el descuento aplicado y mostrarle al cliente cuánto paga por la cantidad que eligió de la siguiente manera: "total a pagar por x productos: \$x"

**7.** Hacer un programa que solicite un número que denominaremos *divisor* y luego, de forma iterativa, va a solicitar un segundo número el cual irá comprobando si es divisible por el *divisor*. El programa termina cuando se ingresa cero.

8. Un programa que muestre los números pares del 1 al 1000.
9. Preguntarle al usuario si quiere saber el área de un rectángulo o el de un triángulo. Luego pedirle los datos (base y altura) y mostrarle el área de la figura que pidió.
10. Pedir al usuario una cantidad de mb y luego convertirlo a kb. Mostrar el resultado por pantalla.

1mb -> 1024kb

2mb ->  $1024 * 2 = 2048$ kb