**Fundamentos básicos II**

1. **Tamaño Grande -**Dado un array, escribe una función que cambie todos los números positivos en él, por el string “big”. Ejemplo: grande([-1,3,5,-5]) devuelve [-1, “big”, “big”, -5].
2. **Imprime (print) el menor, devuelve (return) el mayor -** Crea una función que tome un array de números. La función debería **imprimir (print)**el menor valor del array, y devolver (return) el mayor.
3. **Imprime (print) uno, devuelve (return) otro-**Crea una función para un array de números. La función debería **imprimir (print)** el penúltimo valor y **devolver (return)**el primer valor impar.
4. **Doble Visión -** Dado un array, crea una función que devuelva un nuevo array donde cada valor se duplique. Entonces, doble([1,2,3]) debiera devolver [2, 4, 6] sin cambiar el array original.
5. **Contar Positivos -**  Dado un array de números, crea una función para reemplazar el último valor con el número de valores positivos encontrados en el array. Ejemplo, contarPositivos([-1,1,1,1]) cambia el array original y devuelve [-1,1,1,3].
6. **Pares e Impares -**Crea una función que acepte un array. Cada vez que ese array tenga 3 valores impares seguidos, imprime (print) “¡Qué imparcial!”, y cada vez que tenga 3 pares seguidos, imprime (print) “¡Es para bien!”.
7. **Incrementa los Segundos -** Dado un array de números arr, agrega 1 a cualquier otro elemento, específicamente a aquellos cuyo índice es impar (arr[1], arr[3], arr[5], etc). Luego, console.log cada valor del array y devuelve arr.
8. **Longitudes previas -** Pasado un array (similar a decir ‘tomado un array’ o ‘dado un array’) que contiene **strings**, reemplaza cada string con un número de acuerdo cantidad de letras (longitud) del string anterior. Por ejemplo, longitudesPrevias([“programar”,“dojo”, “genial”]) debería devolver [“programar”,9, 4]. **Pista:**¿For loops solo puede ir hacia adelante?
9. **Agrega Siete -** Construye una función que acepte un array. Devuelve un nuevo array con todos los valores del original, pero sumando 7 a cada uno. No alteres el array original. Por ejemplo, agregaSiete([1,2,3) debería devolver [8,9,10] en un nuevo array.
10. **Array Inverso -**Dado un array, escribe una función que invierte sus valores en el lugar. Ejemplo: invertir([3,1,6,4,2)) devuelve el mismo array pero con sus valores al revés, es decir [2,4,6,1,3]. Haz esto sin crear un array temporal vacío. (Pista: necesitarás intercambiar (swap) valores).
11. **Perspectiva: Negativa -** Dado un array crear y devuelve uno nuevo que contenga todos los valores del array original, pero negativos (no simplemente multiplicando por -1). Dado [1,-3,5], devuelve [-1,-3,-5].
12. **Siempre hambriento -**Crea una función que acepte un array e imprima (print) “yummy” cada vez que alguno de los valores sea “comida”. Si ningún valor es “comida”, entonces imprime “tengo hambre” una vez.
13. **Cambiar hacia el centro -** Dado un array, cambia el primer y último valor, el tercero con el ante penútimo, etc. Ejemplo: cambiaHaciaElCentro([true, 42, “Ada”, 2, “pizza”]) cambia el array a [“pizza¨, 42, “Ada”, true]. cambiaHaciaElCentro([1,2,3,4,5,6]) cambia el array a [6,2,4,3,5,1]. No es necesario devolver (return) el array esta vez.
14. **Escala el Array -**Dado un array **arr**y un número**num,**multiplica todos los valores en el array **arr**por el número **num**, y devuelve el array **arr** modificado. Por ejemplo, escalaArray([1,2,3], 3] debería devolver [3,6,9].