

ACTIVIDAD 1

OBJETOS PREDEFINIDOS: STRING, MATH, DATE, ARRAY Y CLASES DEFINIDAS POR EL USUARIO

EJERCICIO 1

Realiza una función `giraPalabras` que codifique un texto dando la vuelta a todas las palabras que tenga este.

El script **pedirá** a través del **prompt** una frase o texto y deberá escribir el mismo texto, pero con las palabras del revés (OJO, no la frase entera del revés, se gira palabra a palabra) a modo del siguiente ejemplo:

```
Hola, me llamo María →, aloH em omall aÍraM
```

(la salida mostrará tanto la frase original como el resultado de la codificación).

PISTA: Los arrays tienen métodos para invertir el orden, los string no

EJERCICIO 2

Vamos a suponer que una contraseña es segura si cumple las siguientes condiciones:

- Tiene un mínimo de 6 caracteres y como mucho 20.
- Contiene una minúscula, una mayúscula y un dígito.
- No contiene 3 caracteres iguales seguidos (ej. "...aaa...").

Escribe una función `comprobadorPassword` que reciba una cadena (la contraseña), y devuelva el mínimo número de cambios necesarios (inserciones, borrados o reemplazos) para hacerla segura. Si ya es segura el número será 0. Los espacios serán ignorados.

```
comprobadorPassword ("Maria!") → 1
// 6 caracteres en total, se necesita un dígito para una contraseña segura
// 1 cambio como mínimo

comprobadorPassword ("maria23!") → 1
// 7 caracteres en total, se necesita añadir una mayúscula.
// 1 cambio como mínimo

comprobadorPassword ("MARIAMARIAMARIAMARIA!") → 3
// 21 caracteres en total, sólo letras mayúsculas, se necesita borrar un carácter,
// y reemplazar dos caracteres, 1 con un dígito, y otro con una minúscula.
// 3 cambios como mínimo

comprobadorPassword ("Mariaaaa!!1") → 1
// Contiene un carácter que se repite más de 3 veces seguidas
// Es necesario reemplazar un carácter intermedio por otro (ej. "a3aa" o añadir un
// carácter en el medio "aa2aa"
// 1 cambio como mínimo
```

EJERCICIO 3: FECHAS

Escribe una función `tieneViernes13` que reciba un mes y un año y devuelva verdadero si ese mes contine un viernes 13.

Usa la función anterior desde otra función llamada `cuentaViernes13` que devuelva cuántos viernes 13 hay en un año determinado.

Ejemplo de uso:

```
console.log(cuentaViernes13(2023)); //Devolverá 2
```

EJERCICIO 4: ARRAYS

Escribe una función `validaSudoku` que reciba un array de 2-D que represente un sudoku resuelto y devuelva true si la solución es correcta, falso en caso contrario. Para ser un sudoku válido:

- Cada fila ha de contener los dígitos 1 al 9 una y sólo una vez.
- Cada columna ha de contener los dígitos 1 al 9 una y sólo una vez.
- Cada trozo de 3x3 ha de contener los dígitos 1 al 9 una y sólo una vez.

6	3	9	5	7	4	1	8	2
5	4	1	8	2	9	3	7	6
7	8	2	6	1	3	9	5	4
1	9	8	4	6	7	5	2	3
3	6	5	9	8	2	4	1	7
4	2	7	1	3	5	8	6	9
9	5	6	7	4	8	2	3	1
8	1	3	2	9	6	7	4	5
2	7	4	3	5	1	6	9	8

```
validaSudoku([
  [ 6, 3, 9, 5, 7, 4, 1, 8, 2 ],
  [ 5, 4, 1, 8, 2, 9, 3, 7, 6 ],
  [ 7, 8, 2, 6, 1, 3, 9, 5, 4 ],
  [ 1, 9, 8, 4, 6, 7, 5, 2, 3 ],
  [ 3, 6, 5, 9, 8, 2, 4, 1, 7 ],
  [ 4, 2, 7, 1, 3, 5, 8, 6, 9 ],
  [ 9, 5, 6, 7, 4, 8, 2, 3, 1 ],
  [ 8, 1, 3, 2, 9, 6, 7, 4, 5 ],
  [ 2, 7, 4, 3, 5, 1, 6, 9, 8 ]
]) → true
```

```
sudokuValidator([
  [ 1, 1, 2, 4, 8, 9, 3, 7, 6 ],
  [ 7, 3, 9, 2, 5, 6, 8, 4, 1 ],
  [ 4, 6, 8, 3, 7, 1, 2, 9, 5 ],
  [ 3, 8, 7, 1, 2, 4, 6, 5, 9 ],
  [ 5, 9, 1, 7, 6, 3, 4, 2, 8 ],
  [ 2, 4, 6, 8, 9, 5, 7, 1, 3 ],
  [ 9, 1, 4, 6, 3, 7, 5, 8, 2 ],
  [ 6, 2, 5, 9, 4, 8, 1, 3, 7 ],
  [ 8, 7, 3, 5, 1, 2, 9, 6, 4 ]
]) → false
```

EJERCICIO 5

Escribe un simulador de un robot. Las instalaciones de prueba de una fábrica de robots necesitan un programa para verificar los movimientos de los robots.

Los robots tienen tres movimientos posibles:

- Gire a la derecha (D)
- Gire a la izquierda (I)
- Avance (A)

Los robots se colocan en una cuadrícula infinita hipotética, mirando en una dirección particular (norte, este, sur u oeste) en un conjunto de coordenadas $\{x,y\}$, por ejemplo, $\{3,8\}$, con coordenadas que aumentan hacia el norte y el este.

Luego, el robot recibe una serie de instrucciones, momento en el que el centro de pruebas verifica la nueva posición del robot y en qué dirección apunta.

La cadena de letras "DAAIAI" significa:

- *Gire a la derecha*
- *Avanzar dos veces*
- *Gire a la izquierda*
- *Avanzar una vez*
- *Gira a la izquierda una vez más*

Digamos que un robot comienza en $\{7, 3\}$ mirando al norte. Luego, ejecutar este flujo de instrucciones debería dejarlo en $\{9, 4\}$ mirando hacia el oeste.

```
const robot = new Robot(7, 3, 'norte');
const instrucciones = "DAAIAI";
robot.ejecuta(instrucciones);
console.log(robot.toString());
```

La salida será:

`(9, 4) mirando al oeste.`