

ACTIVIDAD 1

SCRIPTS BÁSICOS

1. Crea un script que pida al usuario un número entero positivo. A continuación, se mostrará en un mensaje por consola el sumatorio de dicho número.

Función	Recibe	Devuelve	Descripción
esEntero	c:cadena	booleano	Devuelve verdadero si c puede convertirse a un número entero y falso en caso contrario (un número es entero si al parsearlo con parseInt devuelve el mismo número).
esPositivo	numero:entero	Booleano	Devuelve verdadero si numero es mayor que 0
sumatorio	numero:entero	entero	Devuelve el sumatorio de numero ($0+1+2+\dots+n$)

En caso de que el usuario no haya introducido un número entero positivo, se mostrará un mensaje de error indicativo y se pedirá de nuevo. Dicho proceso se repetirá hasta que el número introducido sea correcto.

2. Crea un script que pida 4 números enteros positivos e indique si dichos números están en orden estrictamente creciente.

Función	Recibe	Devuelve	Descripción
ordenCreciente	n1:entero n2:entero n3:entero n4:entero	boolean	Devuelve verdadero si los números recibidos como parámetros están en orden estrictamente creciente, y falso en caso contrario.

En caso de que alguno de los números introducidos no sea entero positivo, se mostrará un mensaje de error indicativo y se pedirá de nuevo. Dicho proceso se repetirá hasta que el número introducido sea correcto.

3. Crea un script que pida al usuario un número entero entre 1 y 100. A continuación, se mostrará un mensaje que indique si el número:
 - a. Es divisible entre 2.
 - b. Es divisible entre 3.
 - c. Es divisible entre 5.
 - d. Es divisible entre 10.

Indicaciones:

- Un número es divisible entre N cuando el resto de dividir el número entre N es 0.
- Puede ser útil declarar como **constante** ciertos valores fijos en el ejercicio

```
const MAXIMO = 100;  
const MINIMO = 1;  
const DIVISORES = [2,3,5,10];
```

- Se mostrará un único mensaje que indique entre qué números es divisible el número. Además, en caso de que no sea divisible entre ninguno de los citados, se mostrará un mensaje indicativo.

Función	Recibe	Devuelve	Descripción
esDivisible	dividendo:entero divisor:entero	booleano	Devuelve verdadero si el primer parámetro es divisible entre el segundo y falso en caso contrario

Haz las siguientes pruebas:

- Introduce 24. Tiene que devolver “divisible entre 2, 3”
- Introduce 17. Tiene que devolver “no es divisible entre 2, 3, 5, 10”

En caso de que el número no esté entre 1 y 100 o no se introduzca un número, se mostrará un mensaje de error indicativo y se pedirá de nuevo. Dicho proceso se repetirá hasta que el número introducido sea correcto.

- Implementa un script con una pequeña calculadora que haga las siguientes operaciones con enteros:
 - Pedir un operando.
 - Pedir otro operando
 - Pedir la operación a realizar (puede ser suma, resta, multiplicación o división o FIN para finalizar).

A continuación, se mostrará en consola el resultado de la operación.

Para realizar distintas tareas según el operador introducido encaja a la perfección el uso de switch:

```
switch(operador){
  case "+":
    //completar
    break;

  case "-": //completar
  case "*": //completar
  case "/": //completar
  case "FIN": //completar
```

Es necesario validar:

- Que el operador sea uno de los permitidos (+, -, *, /, FIN).
- En caso de que el operador sea la división, que el segundo operando no sea nulo.

Se realizarán operaciones mientras el operador introducido sea distinto de FIN, en cuyo caso el script finalizará. Además, en caso de que alguna de las validaciones sea incorrecta, se mostrará un mensaje indicativo y se seguirá pidiendo el dato hasta que este sea correcto.

Función	Recibe	Devuelve	Descripción
validaOperador	operador:cadena	booleano	Valida que el operador sea permitido
divisionValida	dividendo:entero divisor:entero	booleano	Valida que la división sea válida.

5. Implementa un script gestione un control de acceso básico a una aplicación. Se pedirán (en dos ventanas distintas) un login y una contraseña:

En función de la validez de las mismas, se informará al usuario acerca de uno de los siguientes mensajes:

- *El nombre de usuario y la contraseña son correctas*
- *El nombre de usuario no existe*
- *El nombre de usuario no coincide con la contraseña.*

Para comprobar los datos, se usarán dos constantes globales de tipo array. La primera de ellos contiene 5 nombres de usuario, mientras que la segunda de ellas contiene las contraseñas de dichos usuarios. La contraseña en la posición i -ésima del array de contraseñas se corresponderá con el login ubicado en la posición i -ésima del array de logins.

Por ejemplo, benito tendrá la contraseña 67890.

pepe	paco	juan	benito	manuel
12345	23456	09876	67890	666666

El usuario tendrá 5 intentos para introducir correctamente el nombre de usuario y la contraseña.

Función	Recibe	Devuelve	Descripción
existeLogin	login:cadena	booleano	Valida que el login exista
passwordCorrecto	password:cadena login:cadena	Booleano	Valida que el password correspondiente al login proporcionado sea correcto.

- La función **passwordCorrecto** comprobará si la contraseña almacenada en la misma posición que el nombre de usuario coincide con la introducida:

```
if (PASSWORDS[indice]==password)
    correcto=true;
```