

Profesor:

*Neiner Maximiliano*

1. Realizar una aplicación que muestre los siguientes mensajes utilizando **console.log()**:

*HOLA MUNDO!!!*

*Puedo mostrar comillas 'simples'*

*Y comillas "dobles"*

**Nota:** El mensaje se mostrará en una sola instrucción. Utilice caracteres de escape. Emplee plantillas de string (tilde invertido).

2. Cree una aplicación que muestre, a través de un **Array**, los nombres de los meses de un año y el número al que ese mes corresponde. Utilizar una estructura repetitiva para escribir en la consola (**console.log()**).
3. Realizar una función que reciba un parámetro requerido de tipo numérico y otro opcional de tipo cadena. Si el segundo parámetro es recibido, se mostrará tantas veces por consola, como lo indique el primer parámetro. En caso de no recibir el segundo parámetro, se mostrará el valor inverso del primer parámetro.
4. Realizar una función que reciba un número y que muestre (por consola) un mensaje como el siguiente:

*El número 5 es impar*, siendo 5 el número recibido como parámetro.

5. Guardar su nombre y apellido en dos variables distintas. Dichas variables serán pasadas como parámetro de la función **MostrarNombreApellido**, que mostrará el apellido en mayúscula y el nombre solo con la primera letra en mayúsculas y el resto en minúsculas. El apellido y el nombre se mostrarán separados por una coma (,).

**Nota:** Utilizar **console.log()**

6. Realizar una función que reciba como parámetro un número y que retorne el cubo del mismo.

**Nota:** La función retornará el cubo del parámetro ingresado. Realizar una función que invoque a esta última y permita mostrar por consola el resultado.

7. Se necesita mostrar por consola los primeros 20 números primos. Para ello realizar una **función**.

**Nota:** Utilizar **console.log()**

8. Crear una función que realice el cálculo factorial del número que recibe como parámetro.

**Nota:** Utilizar `console.log()`

9. Realizar una **función** que solicite (por medio de un parámetro) un número. Si el número es positivo, se mostrará el factorial de ese número, caso contrario se mostrará el cubo de dicho número.

**Nota:** Reutilizar la función que determina el factorial de un número y la que calcula el cubo de un número.

10. Definir una **función** que muestre información sobre una cadena de texto que se le pasa como argumento. A partir de la cadena que se le pasa, la función determina si esa cadena está formada sólo por mayúsculas, sólo por minúsculas o por una mezcla de ambas.

11. Definir una **función** que determine si la cadena de texto que se le pasa como parámetro es un palíndromo, es decir, si se lee de la misma forma desde la izquierda y desde la derecha. Ejemplo de palíndromo complejo: *"La ruta nos aporoto otro paso natural"*.

12. Crear una función que reciba como único parámetro una cadena que contenga el día, mes y año de nacimiento de una persona (con formato *dd-mm-yyyy*). La función mostrará por consola a que signo corresponde dicha fecha de nacimiento.

**Nota:** Para descomponer la fecha recibida como parámetro utilice la función `split`.

13. Un **número de Smith** es un número entero tal que la suma de sus dígitos es igual a la suma de los dígitos de los números restantes tras la factorización en primos (la factorización debe estar escrita sin exponentes, repitiendo los números todas las veces necesarias). Por ejemplo,  $378 = 2 \times 3 \times 3 \times 3 \times 7$  es un número de Smith en base 10, porque  $3 + 7 + 8 = 2 + 3 + 3 + 3 + 7$ . Por definición, se deben contar los dígitos de los factores. Por ejemplo, 22 en base 10 es  $2 \times 11$ , y se deben contar los tres dígitos: 2, 1, 1. Por lo tanto 22 es un número de Smith porque  $2 + 2 = 2 + 1 + 1$

**Nota:** Utilice tres funciones, una realiza la comparación, otra descompone el número en sus factores primos y suma los coeficientes, y la última función suma cada término.