

### Práctico N° : 3

#### Javascript

##### Introducción

1. Escriba un bloque de código utilizando Javascript donde se observe la diferencia entre “var”, “let” y “const” al momento de definir variables. Explique las diferencias entre “var” y “let” e indique cuál es el “problema” que puede surgir al usar var.
2. Explique lo que es el término “hoisting” en Javascript y de 2 ejemplos de código sobre este término.
3. Defina una misma función utilizando las distintas formas que provee javascript (funciones flechas, funciones anónimas, etc).
4. Manejo de arreglos. Dado el siguiente arreglo y utilizando las funciones que Javascript provee (filter, map, forEach, reduce, etc), muestre el código que permite resolver los siguientes enunciados:

Arreglo: [1, 2, 3, 4, 5, 6, 11, 23, 1, 989, 0, 1, 111, 645, 50, 45]

- a. Retornar el menor elemento del arreglo.
  - b. Retornar la suma de todos los elementos del arreglo.
  - c. Retornar un arreglo donde cada elemento es el doble que en el arreglo original.
  - d. Retornar un arreglo con los elementos que sean mayores o iguales a 10.
  - e. Retornar el índice (en caso de existir) del primer elemento mayor a 10.
5. Definir e instanciar una clase Automóvil, que posea las siguientes propiedades: ruedas, puertas, color y velocidad. Añadir un método que permita aumentar la velocidad del automóvil.

## Eventos

Dada la siguiente estructura HTML, añade las siguientes funcionalidades:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Práctico N° 3</title>
  <style>
    * {
      margin: 0;
      padding: 0;
    }
    header,
    footer {
      width: 100%;
      background-color: bisque;
      padding: 10px 0;

      display: grid;
      place-items: center;
    }
    footer {
      position: absolute;
      bottom: 0;
    }
  </style>
</head>
<body>
  <header>Practico N° 3 Javascript</header>
  <main>
    <p id="saludo"></p>
```

```
<button onclick="saludar()">SALUDAR</button>
<button id="getDataBtn">GET DATA</button>
</main>
<footer>UNJS - Programación Web 2024</footer>
</body>
<!-- insertar AQUÍ referencia a archivo Javascript -->

</html>
```

1. Al hacer click en el botón "SALUDAR", debe aparecer un texto en la web que salude al usuario (usar el elemento <p> del HTML).
2. A los 5 segundos de cargada la página, el color de fondo del header y footer deben cambiar automáticamente.
3. Al hacer click en el botón "GET DATA", realizar una consulta HTTP a '<https://jsonplaceholder.typicode.com/posts>' y mostrar una lista desordenada en la página con los primeros 10 elementos de la respuesta.  
En caso de no existir el elemento HTML para mostrar la lista, crearlo e insertarlo mediante Javascript.

### Formularios

1. Validación: Construya un formulario de "Registro" para una página web ficticia con HTML, conformado por los inputs:
  - a. Nombre de Usuario
  - b. Correo
  - c. Contraseña
  - d. Repetir ContraseñaAñade utilizando Javascript validación sobre estos campos, tal que:
  - e. Nombre de Usuario: tenga una longitud mínima de 3 caracteres, y no deje ingresar nombres preexistentes como "Ana", "Pepe", "Pancho".

- f. Correo: validar que sea una dirección de correo válida. Pueden usar expresiones regulares.
  - g. Contraseña: debe poseer una longitud mínima de 8 caracteres.
  - h. Repetir contraseña: el valor ingresado en este campo debe coincidir con el valor ingresado en el campo anterior (campo contraseña).
2. Envío de datos: Tome la información del formulario construido en el inciso previo (1.) y realice el envío de estos datos a la siguiente dirección <https://jsonplaceholder.typicode.com/posts>. Una vez finalizado el envío de datos, informe si el proceso fue exitoso o si ocurrió un error durante el envío de datos.

Identifique el método http correspondiente para el envío de la información.

Documentación de la API: <https://jsonplaceholder.typicode.com/guide/>.

### Entretenimiento

En esta sección del práctico se construirá el famoso juego “TA-TE-TI” o “TIC-TAC-TOE”.

- 1. Utilizando HTML y CSS construya un tablero 3x3 el cual será utilizado para llevar a cabo el juego.
- 2. Utilizando Javascript agregue funcionalidad interactiva al juego de manera que:
  - a. Se pueda identificar a quien corresponde el turno actualmente.
  - b. Se permite colocar la “ficha” (X/O) del jugador correspondiente al hacer clic en una casilla del tablero.
  - c. Indicar si hubo un ganador según las reglas del juego (fichas iguales en diagonal, vertical u horizontal).
  - d. Permitir comenzar un nuevo juego.

### Opcional (solo para aprendizaje del alumno)

- e. Permitir jugar utilizando el teclado (colocar piezas, moverse por el tablero).
- f. Poder aumentar el tamaño del tablero según indique el jugador. Por ejemplo, pasar de un tablero 3x3 a uno 4x4, es decir, aumentar las dimensiones del mismo.
- g. Poder jugar contra la máquina (sin IA).