



Trabajo Práctico grupal de presentación obligatoria.

Parte 1

Crear un repositorio en GitHub con el nombre **pv_tp2_grupo999**, donde 999 es el número del grupo.

Todos los integrantes del grupo deben clonar el repositorio.

Abrir la carpeta clonada en Visual Studio Code.

Crear una carpeta denominada js en esta carpeta ir agregando cada uno de los archivos correspondientes a los ejercicios.

Nombre de archivos:

ejercicio1.js

Las variables deben declararse utilizando exclusivamente **let** o **const**, evitando completamente el uso de **var**, y todas las funciones deben escribirse usando la sintaxis de función flecha (arrow function).

Para los siguientes ejercicios utilizar Code Runner, console.log(), Template strings.

1. Crear una variable constante denominada integrantes y asignarle la cadena de texto con los nombres de los integrantes del grupo.

Mostrar por consola el siguiente texto:

Somos (el valor de la variable) y estamos aprendiendo javascript.

2. Mostrar por consola los primeros 10 números impares.

3. Calcular y mostrar por consola el volumen de una esfera cuyo radio = 4 m

$$V = \frac{4}{3}\pi r^3$$

4. Declare las variables necesarias para almacenar valores numéricos que representan los lados de un rectángulo. Calcular el área y el perímetro del rectángulo y mostrar los resultados por consola. La unidad de medida es cm.

5. Declarar dos variables denominadas horas, minutos, asignar valores a cada una y realizar la conversión para expresar esas horas y minutos en segundos. Mostrar por consola las horas y minutos, luego el resultado de la conversión en segundos.

6. Crear un array denominado edades con al menos 8 edades distintas.

Recorrer el array, mostrar las edades y mostrar el promedio de las edades.

7. Declarar un array denominado nombres con al menos 6 nombres.

Mostrar los nombres del array y obtener el nombre más largo del array nombres.

Las cadenas de textos (String) son listas de caracteres y las listas tienen un largo (length) que indica su tamaño.

8. Declarar un array denominado notas que almacene 10 notas (números enteros). Con el array cargado realice lo siguiente:

- a) Recorrer el array y mostrar cada nota.
- b) Mostrar cuáles notas están aprobadas (≥ 6)
- c) Mostrar cuáles están desaprobadas (< 6).
- d) Calcular el promedio general.
- e) Mostrar la cantidad de notas que están aprobadas.

Ejercicios para usar `prompt`, `document.write()`.

Para desarrollar los siguientes ejercicios debe crear por cada ejercicio un archivo html, dentro de la etiqueta `<body></body>` agregar las etiquetas `<script></script>` y dentro de ellas agregar el código javascript para resolver cada ejercicio. Para mostrar los resultados utilizar **template strings**.

El nombre para cada archivo html debe ser:

ejercicio9.html, etc.

9. Solicitar al usuario que ingrese primero su apellido, luego su nombre. Mostrar un mensaje de bienvenida, se la siguiente forma:

```
¡Hola (nombre apellido).  
Bienvenido a Programación Visual!
```

10. Solicitar al usuario que ingrese la cantidad de números que desea ingresar. Pedir que ingrese esa cantidad de números enteros de forma iterativa, guardar los números en un array. Luego mostrar los números ingresados y el promedio.

11. Solicitar al usuario que ingrese el número de kilómetros recorridos por su vehículo y el número de litros de combustible consumidos. Calcular y mostrar el consumo de combustible por kilómetro.

El resultado se obtiene dividiendo número de litros dividido por número de kilómetros.

12. Solicitar al usuario que ingrese un número.

Usando un **if/else** mostrar un mensaje indicando si el número es par o impar.

13. Escribir una función llamada `mostrarArreglo` que reciba un arreglo numérico como parámetro e imprima cada elemento en una línea a parte. El usuario debe ingresar cuantos números se almacenarán en el array e ingresar cada número de forma iterativa.
 14. Escribir una función llamada `duplicar` que reciba un arreglo de números y retorne un nuevo arreglo donde cada número esté multiplicado por dos (2). El usuario debe ingresar cuantos números se almacenarán en el array e ingresar cada número de forma iterativa.
 15. Escribir una función llamada `retornarMes`, que reciba un número entero y si corresponde a un mes del año devolver el nombre del mes, si no se cumple entonces mostrar un mensaje que no es un mes.
 16. Escribir una función llamada `sumarRango` que reciba dos argumentos numéricos enteros: número inicial y número final. La función debe retornar la suma de los números en ese rango (incluyéndolos).
El número inicial debe ser menor o igual que el número final.
 17. Escribir una función llamada `contarLetraA` que reciba un string y retorne el número de veces que aparece la letra "a".
 18. Escribir una función llamada `calcularMayor` y pasarle como parámetros (`numero1`, `numero2`). Los números son ingresados por el usuario. Dentro de la función mostrar un mensaje que diga cuál de los números ingresados es mayor. En caso de ser iguales, tendrá que indicarlo con otro mensaje.
-

Parte 2.

Uso de eventos y manipulación de DOM

Para cada ejercicio, crear un archivo HTML (ej: `ejercicio19.html`). En la carpeta `js` ir agregando todos los archivos JavaScript.

El código JavaScript ahora debe estar estructurado utilizando módulos de JavaScript.

Cada archivo debe tener un nombre que identifique el ejercicio, por ejemplo: `ejercicio19.js` y `funciones19.js`

En cada archivo HTML, antes de cerrar la etiqueta `<body>`, enlazar el archivo JavaScript correspondiente utilizando la siguiente estructura:

```
<script type="module" src="js/ejercicio19.js"></script>
```

Nota: Se deben usar las palabras claves **export** e **import** para compartir funciones o variables entre archivos.

19. Mostrar en una página html un párrafo de texto y un botón, aplicar estilos css.
Crear un script que capture el evento clic del botón y que cambie el texto del párrafo por otro.

20. Crear una página que muestre 3 campos (input): Nombre, apellido, libreta universitaria.
Crear un script para obtener cada uno de los elementos input creados, mediante el método `querySelector()`, manipular los elementos para obtener su valor con la propiedad JS `value` y mostrar los datos con un `alert` de la siguiente forma:

Los datos ingresados son:
Nombre: María
Apellido: Diaz
Libreta Universitaria: APU999999

21. Construir una página utilizando Html, Css y JavaScript que contenga dos listas desplegables (`<select>`):

La primera lista debe contener una selección de países de América.

La segunda lista debe contener las capitales correspondientes a esos países.

Al seleccionar un país en la primera lista, la capital correspondiente debe seleccionarse automáticamente en la segunda lista, usando JavaScript.

El comportamiento debe implementarse con JavaScript utilizando el evento `change`.

Incluir al menos 6 países y sus capitales.

Mostrar un mensaje en consola indicando qué país y capital están seleccionados.

22. Construir una página utilizando Html, Css y JavaScript que permita al usuario escribir en un campo de texto (`<input>`), y que a medida que escribe, el texto se muestre automáticamente en otro elemento de la página (por ejemplo, un `<p>`).

Usar un elemento `<input type="text">` para que el usuario pueda escribir.

Mostrar el texto ingresado en un elemento Html como un `<p>` o un `<div>`.

Aplicar estilos básicos con Css para que la interfaz sea clara y prolija.

Usar JavaScript para capturar el evento `input` y actualizar el contenido dinámicamente.

23. Construir una página web utilizando Html, Css y JavaScript que permita al usuario escribir en un campo de texto (`<input>`). A medida que el usuario escribe, el contenido debe mostrarse en otro elemento de la página (como un `<p>` o un `<div>`).

Además, cuando el texto supere los 20 caracteres, el fondo del elemento donde se muestra el texto debe cambiar de color automáticamente.

Usar un `<input type="text">` para que el usuario pueda escribir.

Mostrar el texto ingresado en tiempo real dentro de un `<p>` o `<div>`.

Aplicar estilos básicos con CSS.

Utilizar JavaScript para manejar el evento `input`.

24. Construir una página web utilizando Html, Css y JavaScript que muestre un grupo de botones de opción (radio) con diferentes opciones. Al seleccionar una opción, el

resultado debe mostrarse en tiempo real en otro elemento de la página (por ejemplo, un `<p>` o un `<div>`).

Las opciones de los botones pueden ser por ejemplo lenguajes de programación. Crear al menos 3 botones de opción (radio).

Mostrar el valor seleccionado en tiempo real en otro elemento visible.

Aplicar estilos básicos con Css para una buena presentación.

Usar JavaScript para capturar el evento `change` y actualizar el contenido dinámicamente.

Mostrar el valor seleccionado también en la consola.

25. Construir una página web utilizando Html, Css y JavaScript que incluya un botón. Al hacer clic en ese botón, el color de fondo de toda la página debe cambiar.

Usar un elemento `<button>` para permitir la interacción del usuario.

Cambiar el color de fondo del `body` al hacer clic.

Utilizar JavaScript para capturar el evento `click`.

Aplicar estilos básicos con CSS para mejorar la presentación.

Mostrar un mensaje en la consola cada vez que el color cambia.

Opcional: Cambiar el color de fondo aleatoriamente entre varios colores cada vez que se haga clic.

Entregables

- Enlace al repositorio público (github).
- Archivo **README.md** con los nombres de los integrantes y su respectivo usuario de Github.