

Guía Práctica de Ejercicios N° 9

En esta práctica estudiaremos los conceptos básicos relacionados con el lenguaje JavaScript. Comenzando con el esquema de trabajo del lenguaje, historia y evolución, características, casos de éxito, requisitos para desarrollar con JavaScript, entre otros. Aprenderemos a embeber código JS, comentar nuestros desarrollos, hacer uso de los tipos de variables, emplear variables, constantes, manejar operadores del lenguaje y estructuras de control del mismo.

Objetivo: Introducirse en los conceptos básicos de la programación en Frontend y familiarizarse con las herramientas de desarrollo que utilizaremos durante todo el resto del curso .

Introducción: Hoy en día la programación en Frontend se ha convertido en uno de los puntos ideales en el mercado para el desarrollo para aplicaciones web, desktop, e inclusive para dispositivos móviles. Para llegar a desarrollar aplicaciones de tipo Frontend, necesitamos entender el funcionamiento de HTML y sus etiquetas como así también las librerías que podríamos utilizar.

Construcciones (palabras clave) y funciones que estudiaremos en esta unidad.

Funciones con las que trabajaremos

length permite conocer el número de elementos de un arreglo o cadena.

toString() permite mostrar el contenido de un arreglo como string.

indexOf(valor) permite encontrar el índice de un elemento (valor) del arreglo.

push(valor) añade un elemento al final del arreglo. Devuelve la longitud del arreglo.

pop() elimina el último elemento del arreglo. Devuelve el elemento eliminado.

shift() obtiene el primer elemento del arreglo.

unshift(valor) agrega elementos al inicio del arreglo.

reverse() devuelve los elementos de un arreglo en orden inverso.

sort() ordena alfabéticamente los elementos de un arreglo.

charAt(valor) accede al carácter definido por *valor* en una cadena de texto.

Number(cadena) convierte una cadena en número.

toLowerCase() convierte una cadena a minúscula.

toUpperCase() convierte una cadena a mayúscula.

replace('texto_a_reemplazar','nueva_cadena') reemplaza una subcadena por otra.



substring(inicio, fin) devuelve una subcadena que comienza en **inicio** y termina en **fin**.

console.log() nos permite dar salida por consola terminal del navegador.

typeof permite obtener el tipo de una variable



Ejercicios

Preparar y verificar nuestro entorno de desarrollo PHP

Utilizar los conceptos básicos del lenguaje, búsqueda de elementos.

- 1) - Realice un script JavaScript que verifique la existencia del elemento **id** "div-1". En caso de no existir puede mostrar un `alert()` o `console.log` con un mensaje informándolo y en caso de éxito crear una variable auxiliar de nombre **existe** el cual ira incrementando su valor en 1 en los demás ejercicios. Opcional puede crear una función la cual sea la encargada de controlar la existencia y aumentar la variable (Recomendado para reutilizar código).
- 2) – Edite el script que verifique la existencia del elemento que contiene como **className** "parrafo". En caso de no existir puede mostrar un `alert()` o `console.log` con un mensaje y en caso de éxito aumente la variable **existe** en 1.
- 3) - Edite el script que verifique la existencia del elemento que contiene como **name** "apellido". En caso de no existir puede mostrar un `alert()` o `console.log` con un mensaje y en caso de éxito aumente la variable **existe** en 1.
- 4) - Edite el script que verifique la existencia del elemento que contiene como **name** "edad". En caso de no existir puede mostrar un `alert()` o `console.log` con un mensaje y en caso de éxito aumente la variable **existe** en 1.
- 5) – Edite el script que verifique la existencia del elemento que contiene como **TagName** "select". En caso de no existir puede mostrar un `alert()` o `console.log` con un mensaje y en caso de éxito aumente la variable **existe** en 1.
- 6) – Edite el script que verifique la existencia del elemento que contiene como **id** "comentario" utilizando el `querySelector`. En caso de no existir puede mostrar un `alert()` o `console.log` con un mensaje y en caso de éxito aumente la variable **existe** en 1.

Creacion de elementos en el DOM en JavaScript

- 7) – Realice una función JavaScript que verifique la existencia de un elemento pasando por parámetro un id.



En caso de no existir ningún elemento cree un `<div>` con el atributo de selector correspondiente al parámetro de la función y retorne dicho elemento.

En caso de existir el elemento, defina un nodo de texto y otro de comentario. Agregue ambos nodos al elemento. Puede utilizar cualquiera de los métodos disponibles para asignar un nodo al elemento. Por último defina al elemento `<div>` un atributo de estilo con tamaño de letra de 16px.

8) – Edite la función del punto anterior donde en el caso de existir el elemento y que se encuentre conectado al documento, el mismo sea eliminado (o desconectado) del DOM. Devuelva true si se eliminó el elemento, false en otro caso.

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

