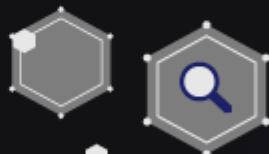


# LENGUAJES DE PROGRAMACIÓN

## JAVA SCRIPT



# LENGUAJES DE PROGRAMACIÓN

## JAVA SCRIPT

### Estructuras de Datos en JavaScript

Las estructuras de datos son una parte fundamental de la programación y nos permiten organizar y manipular datos de manera eficiente. En el contexto de JavaScript, un lenguaje de programación ampliamente utilizado, las estructuras de datos nos brindan herramientas para almacenar, acceder y manipular información de diferentes maneras. es esencial comprender los conceptos básicos de las estructuras de datos para desarrollar programas más eficientes y resolver problemas de manera efectiva.

En esta lección, exploraremos las estructuras de datos en JavaScript. Comenzaremos por definir qué son las estructuras de datos y por qué son importantes en la programación. Luego, nos sumergiremos en los diferentes tipos de estructuras de datos disponibles en JavaScript, cómo utilizarlos y en qué situaciones son más apropiados.





## Estructuras de Datos en JavaScript

### Definición de una Estructura de Datos

Una estructura de datos en JavaScript es una forma organizada de almacenar, gestionar y manipular datos. Estas estructuras nos permiten almacenar diferentes tipos de datos y realizar operaciones como agregar, eliminar, buscar y modificar los elementos almacenados. Cada estructura de datos tiene sus propias características y métodos específicos para realizar estas operaciones.

### Tipos de Estructuras de Datos en JavaScript

#### Arreglos (Arrays):

Los arreglos en JavaScript son estructuras de datos que nos permiten almacenar múltiples elementos en una sola variable. Los elementos se almacenan en posiciones numeradas llamadas índices, y se accede a ellos utilizando el índice correspondiente.

#### Por ejemplo:

```
var miArray = [1, 2, 3, 4, 5];  
console.log(miArray[0]); // Acceso al primer  
elemento del arreglo
```

Los arreglos son útiles cuando necesitamos almacenar una colección ordenada de elementos y acceder a ellos de manera secuencial.



## Objetos (Objects):

Los objetos en JavaScript son estructuras de datos que nos permiten almacenar información estructurada utilizando pares clave-valor. Cada valor está asociado a una clave única que se utiliza para acceder a él.

### Por ejemplo:

```
var miObjeto = {  
  nombre: "Juan",  
  edad: 25,  
  ciudad: "Madrid"  
};  
console.log(miObjeto.nombre); // Acceso al valor  
asociado a la clave "nombre"
```

Los objetos son útiles cuando necesitamos almacenar información estructurada y acceder a ella de forma más descriptiva.





# LENGUAJES DE PROGRAMACIÓN

## JAVA SCRIPT

### Listas Enlazadas (Linked Lists):

Las listas enlazadas son estructuras de datos compuestas por nodos, donde cada nodo contiene un valor y una referencia al siguiente nodo. Esta estructura nos permite almacenar y manipular datos de forma dinámica, ya que los nodos pueden ser agregados o eliminados en cualquier momento.

### Por ejemplo:

```
class Nodo {  
  constructor(valor) {  
    this.valor = valor;  
    this.siguiente = null;  
  }  
}  
  
var nodo1 = new Nodo(1);  
var nodo2 = new Nodo(2);  
nodo1.siguiente = nodo2;
```

Las listas enlazadas son útiles cuando necesitamos una estructura de datos flexible que pueda crecer o reducirse dinámicamente.



# LENGUAJES DE PROGRAMACIÓN

## JAVA SCRIPT

## Ejemplos y Demostraciones

Veamos algunos ejemplos y demostraciones adicionales relacionados con las estructuras de datos en JavaScript:

### Ejemplo de Uso de un Arreglo:

```
var colores = ["rojo", "verde", "azul"];

colores.push("amarillo"); // Agregar un elemento
                             al final del arreglo

colores.pop(); // Eliminar el último elemento
                del arreglo

console.log(colores); // Mostrar el contenido
                      del arreglo
```

En este ejemplo, utilizamos un arreglo para almacenar una lista de colores. Utilizamos los métodos `push()` y `pop()` para agregar y eliminar elementos del arreglo, respectivamente.



## Ejemplo de Uso de un Objeto:

```
var persona = {  
  nombre: "Ana",  
  edad: 30,  
  ciudad: "Barcelona"  
};  
console.log(persona.nombre); // Acceso al  
valor asociado a la clave "nombre"
```

En este ejemplo, utilizamos un objeto para almacenar información sobre una persona. Utilizamos la notación de punto (.) para acceder a los valores asociados a las claves del objeto.

En esta lección, hemos explorado las estructuras de datos en JavaScript. Hemos aprendido que las estructuras de datos nos permiten organizar y manipular datos de manera eficiente en nuestros programas. Hemos explorado diferentes tipos de estructuras de datos en JavaScript, como arreglos, objetos y listas enlazadas, y hemos comprendido en qué situaciones son más apropiadas.

Es fundamental comprender estos conceptos básicos de las estructuras de datos en JavaScript para desarrollar programas más eficientes y resolver problemas de manera efectiva.