

Object (Objeto)

Web Development

JavaScript

Los objetos son una de las estructuras de datos más importantes en JavaScript.

Se usan para almacenar pares de clave–valor (key–value).

Las claves (keys) suelen ser cadenas de texto, aunque también pueden ser símbolos,

y los valores pueden ser de cualquier tipo (número, texto, función, otro objeto, etc.).

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Data_structures#objects

JavaScript

Los objetos se crean normalmente usando la sintaxis literal de objeto, por ejemplo:

```
const website = {  
    name: 'Cetis155',  
    domain: 'edu.mx'  
};
```

"name" y "domain" son las claves.

'Cetis155' y 'edu.mx' son los valores.

Symbol (Símbolo)

Un Symbol es un tipo de dato primitivo en JavaScript que se usa para crear valores únicos.

Cada símbolo es único, incluso si tienen la misma descripción.

<https://developer.mozilla.org/en-US/docs/Glossary/Symbol>

Symbol (Símbolo)

Se crean con la función Symbol(descripción):

```
const id = Symbol("id");
```

```
const otroId = Symbol("id");
```

```
console.log(id === otroId); // false
```

Aunque ambos símbolos tienen la misma descripción, no son iguales.

Esto los hace útiles para definir propiedades únicas en objetos y evitar conflictos.

Symbol (Símbolo)

También se pueden crear símbolos globales compartidos usando `Symbol.for("clave")`,
lo que permite que dos símbolos con la misma clave sean iguales.

Symbol (Símbolo)

// Los Symbols son valores únicos en JavaScript, usados para crear propiedades de objeto que no se repiten.

// Aunque dos Symbols tengan la misma descripción, siempre son diferentes.

// Pueden usarse como claves en objetos sin riesgo de sobrescribir otras propiedades.

```
const id = Symbol('id');
const id2 = Symbol('id2');

console.log(id == id2); // false
```

```
const obj = {
  [id]: 1234,
  [id2]: 0,
  id: 'hello'
};
```

```
console.log(obj);
```

Acceso a propiedades de un objeto

```
const persona = {  
    nombre: "Luis",  
    edad: 30  
};  
  
// Notación con punto  
console.log(persona.nombre); // "Luis"  
  
// Notación con corchetes  
console.log(persona["edad"]); // 30
```

Agregar, modificar y eliminar propiedades

```
const coche = {  
    marca: "Toyota",  
    modelo: "Corolla"  
};  
  
// Agregar una nueva propiedad  
coche.color = "Rojo";  
  
// Modificar una existente  
coche.modelo = "Yaris";  
  
// Eliminar una propiedad  
delete coche.marca;  
  
console.log(coche);
```

Verificar si una propiedad existe

// Podemos verificar si una propiedad existe en un objeto de dos formas:

// 1. Con el operador "in" → revisa si la propiedad existe en el objeto o su prototipo.

// 2. Con "hasOwnProperty()" → revisa solo si la propiedad pertenece directamente al objeto.

```
console.log('name' in website);
```

```
console.log(website.hasOwnProperty('name'));
```

Recorrer un objeto

```
const alumno = { nombre: "Ana", grupo: "6B", promedio: 9.4 };
```

```
for (let clave in alumno) {  
    console.log(clave + ": " + alumno[clave]);  
}
```

Salida:

nombre: Ana

grupo: 6B

promedio: 9.4

Objetos dentro de arreglos

```
const productos = [  
  { nombre: "Laptop", precio: 15000 },  
  { nombre: "Mouse", precio: 300 },  
  { nombre: "Teclado", precio: 500 }  
];  
  
for (let p of productos) {  
  console.log(p.nombre + " cuesta $" + p.precio);  
}
```

Objetos dentro de arreglos

Salida:

Laptop cuesta \$15000

Mouse cuesta \$300

Teclado cuesta \$500

Métodos

// También podemos definir funciones dentro de los objetos, llamadas "métodos".

// Los métodos permiten que el objeto tenga comportamientos o acciones propias.

```
const website = {  
    name: 'AlgoExpert',  
    rating: 5,  
    founders: ['Clement', 'Antoine'],  
    sayHello() {  
        console.log('hello');  
    }  
};  
  
website.sayHello();
```

ACTIVIDAD

Resuelva los siguientes ejercicios en JavaScript sin usar inteligencia artificial o google o cualquier otro recurso para encontrar la solución

1. Crea un objeto llamado persona con las propiedades: nombre, edad, y ocupación y Muestra en consola una frase con esa información.
2. Usando el objeto persona del ejercicio anterior:

Agrega una nueva propiedad llamada ciudad.

Cambia la propiedad ocupación a otro valor.

Muestra el objeto actualizado.

ACTIVIDAD

3. Crea un objeto auto con propiedades marca, modelo y año.

Usa un ciclo for...in para mostrar cada propiedad y su valor.

4. Crea un objeto rectángulo con las propiedades ancho y alto, y un método area() que calcule y devuelva el área.

5. Crea un arreglo alumnos que contenga tres objetos, cada uno con las propiedades nombre y calificación.

Luego recorre el arreglo e imprime solo los alumnos con calificación mayor o igual a 8.

ACTIVIDAD

6. Crea un programa que reciba o defina un objeto con varias propiedades (por ejemplo, nombre, edad, ocupación, etc.).

Tu tarea es determinar cuántas propiedades tiene ese objeto y mostrar el resultado en consola con un mensaje claro.

 Pista: piensa en una forma de obtener todas las claves (nombres de las propiedades) y luego contar cuántas hay.

7. Crea un arreglo que contenga tres objetos, cada uno representando a una persona con las propiedades nombre, edad y ciudad.

El programa debe recorrer el arreglo y mostrar en consola una frase con la información de cada persona.