



BOOTCAMP

Generación T

 streambe



Objetos En JavaScript

¿Qué son los Objetos en JavaScript?

Los objetos en JavaScript son entidades independientes con propiedades y tipos específicos. Funcionan de manera similar a los objetos del mundo real.

Actúan como colecciones de pares clave-valor. Pueden contener cualquier tipo de datos válido en JavaScript.



¿Qué son los Objetos en JavaScript?



Los objetos en JavaScript son estructuras versátiles que agrupan datos relacionados. Permiten organizar código de manera lógica e intuitiva.

Son la base de la POO en JavaScript. Facilitan el desarrollo de sistemas complejos mediante encapsulación y reutilización.

¿Qué Es La POO?

La **Programación Orientada a Objetos (POO)** es una manera de programar que facilita el desarrollo de sistemas complejos y de gran tamaño.

Esta técnica nació a fines de los años 80 y se volvió popular rápidamente porque es más cercana a cómo expresaríamos las cosas en la vida real: dividiendo los componentes de un programa en Objetos que poseen datos (**Propiedades**) y comportamientos (**Métodos**) que se comunican entre sí.

- Dentro de sus ventajas, podemos destacar:
- Fácil de modificar.
- Escalabilidad.
- Reusabilidad.
- Mantenibilidad.

Creación de Objetos

JavaScript ofrece múltiples formas de crear objetos, cada una con ventajas específicas.

[]

Notación Literal

La forma más directa usando llaves: `let persona = {}`

</>

Constructor Object()

Usando el constructor nativo: `new Object()`

f(x)

Funciones Constructoras

Creación de múltiples objetos con estructura similar

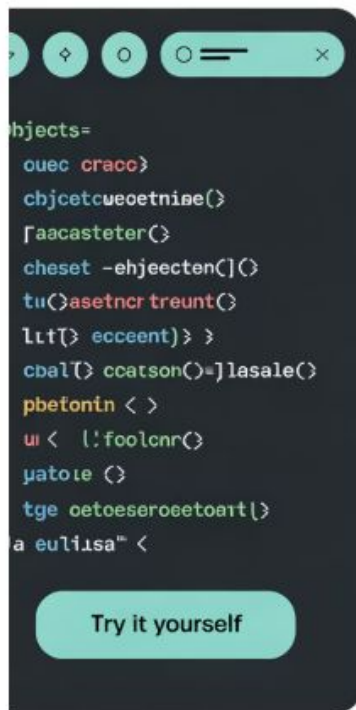


Object.create()

Herencia prototípica directa desde otro objeto

Master Javascript Objects

Build interactive web experiences



Wespe Copr
wey young



Propiedades de Objetos

Pares clave-valor

Almacenan datos dentro del objeto de forma estructurada.

Notación de punto

Acceso mediante sintaxis ``objeto.propiedad``.

Notación de corchetes

Acceso alternativo usando ``objeto["propiedad"]``.

Tipos flexibles

Las propiedades aceptan cualquier tipo de dato válido.

Objetos

Por ejemplo, si quisiéramos modelar en JavaScript el nombre, el apellido y la ocupación de Mario, podríamos expresarlo así:

```
let persona = {  
  nombre: "Mario",  
  apellido: "Mario",  
  trabajo: "Plomero"  
}
```

¿Sabías que...
... Aunque Nintendo inicialmente no le dio un apellido, Shigeru Miyamoto, el creador de Mario, reveló en una entrevista que el apellido de Mario y Luigi sería “Mario”.



Métodos de Objetos

Funciones asociadas

Permiten definir comportamientos específicos del objeto

Acceso por notación

Se invocan mediante
``objeto.metodo()``



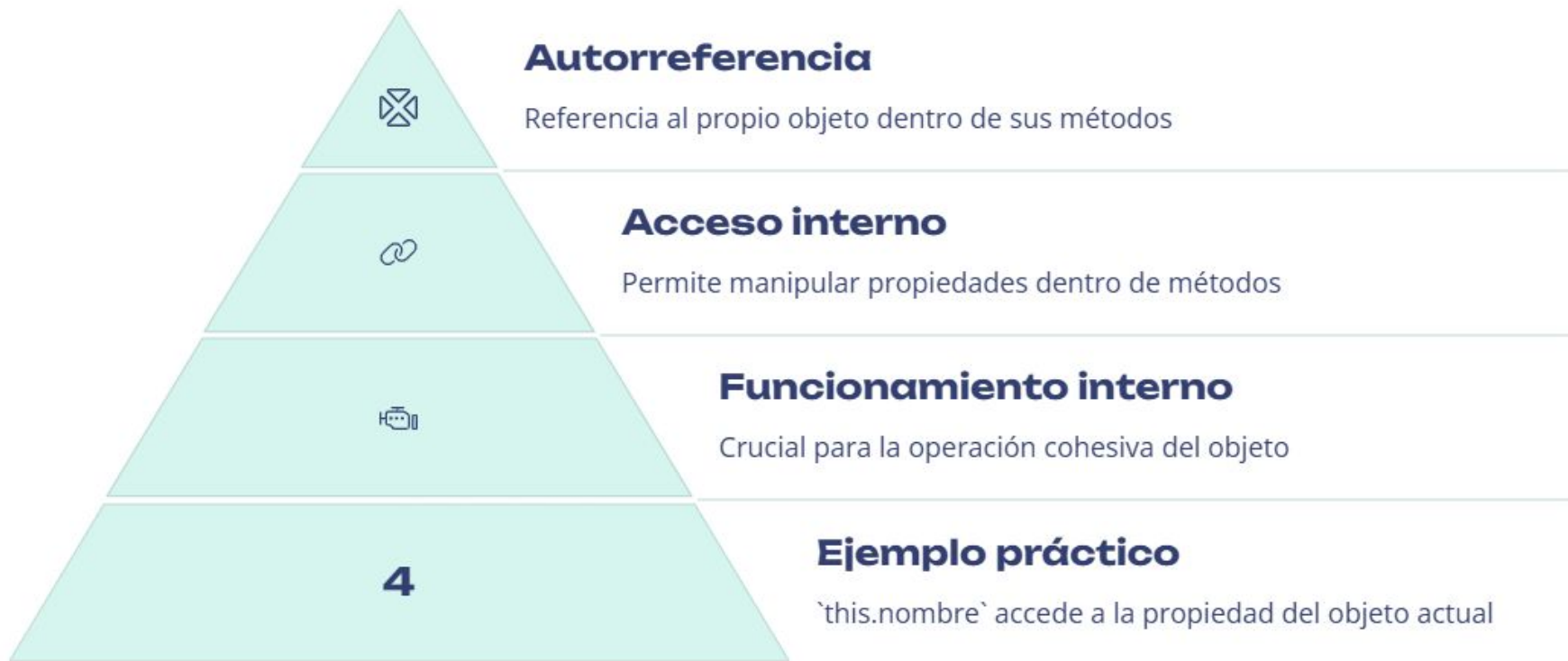
Manipulación de datos

Facilitan operaciones con los datos internos

Definición interna

Se declaran como funciones dentro del objeto

El contexto "this"



Métodos incorporados de Object

Método	Funcionalidad
Object.keys()	Obtiene todas las claves del objeto
Object.values()	Obtiene todos los valores del objeto
Object.entries()	Devuelve pares clave-valor como arrays
Object.freeze()	Previene modificaciones al objeto

Comparación Entre Arreglos Y Objetos

Arreglos

Colecciones ordenadas de elementos por índice numérico.

- Acceso secuencial mediante posiciones numéricas
- Ideal para listas de elementos similares
- Métodos específicos: push(), pop(), map()

Los arreglos mantienen un orden específico. Son excelentes para operaciones en lote.

Objetos

Colecciones de pares clave-valor sin orden específico.

- Acceso directo mediante nombres de propiedades
- Ideal para entidades con atributos relacionados
- Métodos personalizables para comportamientos específicos

Los objetos representan conceptos completos. Permiten encapsular lógica y datos juntos.

Uso Práctico

La elección depende del problema a resolver.

- Arreglos: para colecciones homogéneas
- Objetos: para entidades con propiedades heterogéneas
- Combinables: objetos dentro de arreglos o viceversa

JavaScript permite la flexibilidad de estructuras de datos anidadas.

Cómo Acceder A Los Valores De Un Objeto

[



⁶⁶₉₉



Notación de Corchetes

Accede a propiedades usando corchetes: `objeto['propiedad']`

Nombres Dinámicos

Permite usar variables como nombres de propiedades.

Caracteres Especiales

Útil para propiedades con espacios o caracteres especiales.

Expresiones

Permite usar expresiones para determinar la propiedad a acceder.

La notación de corchetes ofrece mayor flexibilidad que la notación de punto. Es indispensable cuando trabajamos con propiedades dinámicas o generadas en tiempo de ejecución.

Modificar Objetos

Los objetos en JavaScript son dinámicos y flexibles, permitiéndonos modificarlos en cualquier momento.



Añadir Propiedades

`objeto.nuevaPropiedad = valor;`



Actualizar Valores

`objeto.propiedadExistente = nuevoValor;`



Eliminar Propiedades

`delete objeto.propiedad;`



Proteger Objetos

`Object.freeze(objeto);`



GENERACIÓN T

Agregando Propiedades A Un Objeto

Los objetos en JavaScript son entidades dinámicas que pueden expandirse fácilmente después de su creación.

Asignación Directa

```
objeto.nuevaPropiedad = valor;
```

La forma más sencilla y común de agregar propiedades.

Notación de Corchetes

```
objeto['nuevaPropiedad'] = valor;
```

Ideal para nombres de propiedades dinámicos o con caracteres especiales.

Object.assign()

```
Object.assign(objeto, {prop1: valor1, prop2: valor2});
```

Permite agregar múltiples propiedades simultáneamente.

Spread Operator

```
objeto = {...objeto, nuevaPropiedad: valor};
```

Crea un nuevo objeto con todas las propiedades originales más las nuevas.



Objetos Dentro De Arreglos

Una estructura de datos poderosa que combina las ventajas de ambos elementos.

Estructura

Los arreglos pueden contener múltiples objetos como elementos, creando colecciones organizadas.

Acceso

Combinamos notación de índices y propiedades:
`miArreglo[0].propiedad`

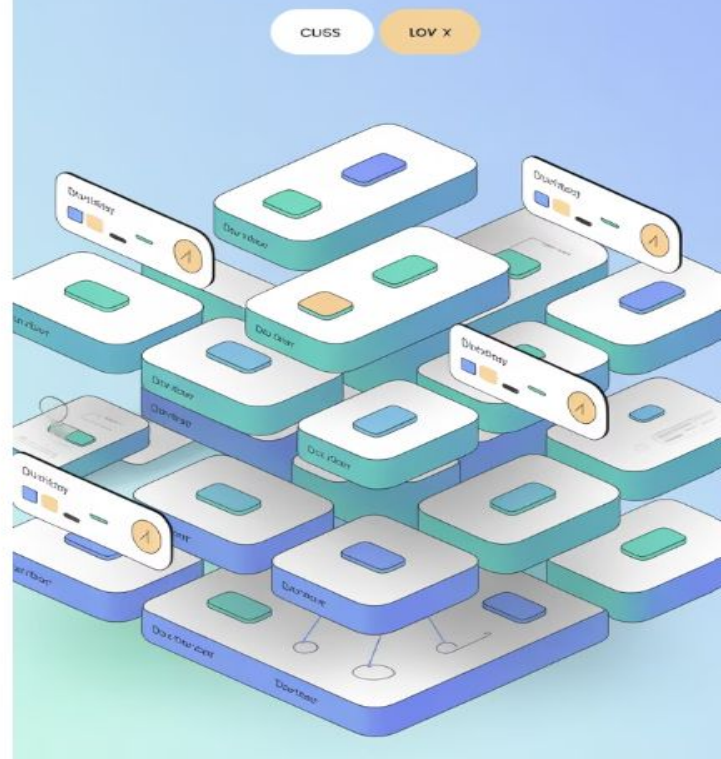
Iteración

Podemos usar bucles para recorrer y manipular todos los objetos del arreglo.

Aplicaciones

Ideal para listas de usuarios, productos o cualquier colección de entidades similares.

JavaScript / Array of objects



Existen múltiples formas de iterar a través de las propiedades de un objeto en JavaScript.

Existen múltiples formas de iterar a través de las propiedades de un objeto en JavaScript.



Itera sobre todas las propiedades enumerables, incluyendo las heredadas.



Devuelve un arreglo con todas las claves propias del objeto.



Devuelve un arreglo con todos los valores propios del objeto.



Devuelve un arreglo de pares [clave, valor] de las propiedades.



Start learning



Muchas gracias.

Nuestras Redes

www.generaciont.org

www.streambe.com

www.instagram.com/generaciont_ar

www.tiktok.com/@generaciont

generaciont@generaciont.org

Cel: [11 61331747](tel:1161331747)



[11 6133-1747](tel:1161331747)



GENERACIÓN T

generaciont@generaciont.org



GENERACIÓN T