

MIND HUB.

An aerial, long-exposure photograph of a multi-lane highway at night. The image is dominated by a dense grid of city lights in the background and long, white and yellow light trails from vehicles on the highway. The overall color palette is a deep blue, with the light trails providing a stark contrast. The word 'OBJETOS' is overlaid in a bright yellow, sans-serif font on the left side of the image. There are also several small, stylized graphic elements: two pink L-shaped brackets at the top center, a dashed line at the top right, two horizontal pink lines on the right side, and a white L-shaped bracket at the bottom center.

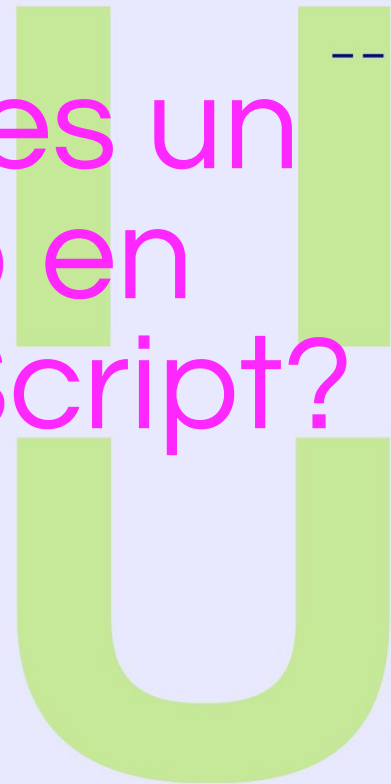
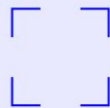
OBJETOS

Índice

- 1 Definición
- 2 Propiedades
- 3 Operando con Objetos
- 4 Constructor y métodos



¿Que es un
objeto en
JavaScript?



Objetos

Un objeto es un conjunto de **propiedades**, y una propiedad es una asociación entre un nombre (o *clave*) y un *valor*.

Los objetos en JavaScript, como en tantos otros lenguajes de programación, **se pueden comparar con objetos de la vida real**.

Los objetos de JS se definen entre llaves de la siguiente forma:

```
let pantalon = {  
  material: "jean",  
  color: "azul",  
  tipo: "oxford",  
  talla: 40,  
  bolsillo_lat: false,  
  cierre: "botones",  
  marca: "levis"  
}
```

```
let taza = {  
  color: "blanco",  
  diseño: "vengadores",  
  material: "plastico",  
  medidas: {  
    diametro: 7,  
    altura: 10,  
  }  
}
```

```
let auto = {  
  marca: "ford",  
  color: "rojo",  
  tipo: "deportivo",  
  seguro: true,  
  puertas: 3,  
  vel_max: 250,  
  distancia: (t,v) => t*v  
}
```

Propiedades de un Objeto

Las propiedades de un objeto **definen las características** del objeto.

Tanto el nombre del objeto (que puede ser una variable normal) como el nombre de la propiedad **son sensibles a mayúsculas y minúsculas**.

Las propiedades pueden ser de cualquier tipo: string, number, **boolean**, **date**, **array**, object e incluso funciones.

para invocar el objeto completo basta con nombrarlo:

nombreObjeto

para acceder a una propiedad del objeto, utilizamos el **punto o los corchetes**

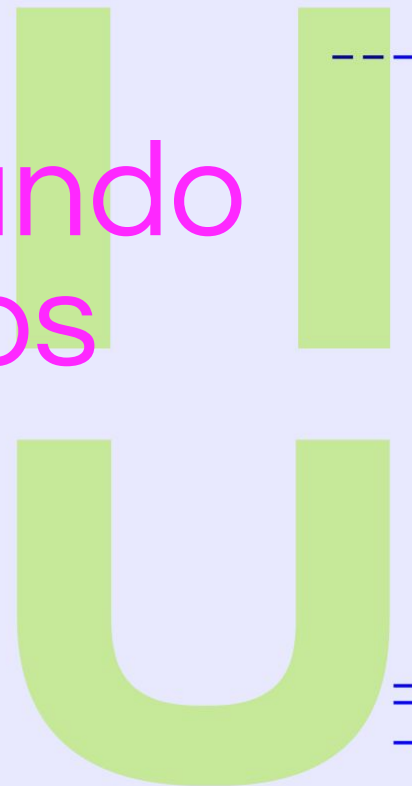
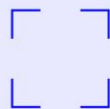
nombreObjeto.nombrePropiedad

nombreObjeto[nombrePropiedad]





Operando Objetos



└─ Modificar propiedades

Podemos modificar las propiedades de los objetos utilizando la siguiente sentencia

nombreObjeto.nombrePropiedad = nuevoValor

```
pantalon.color = "rojo"  
pantalon.talle = 42
```

```
taza.manija = false  
taza.medidas.diametro = 6  
taza.medidas.altura = 9
```

```
▼ {material: 'jean', color: 'rojo', tipo: 'oxford', talle: 42, bolsillo_lateral: false, ...} ⓘ  
  bolsillo_lateral: false  
  cierre: "botones"  
  color: "rojo"  
  marca: "levis"  
  material: "jean"  
  talle: 42  
  tipo: "oxford"  
  ► [[Prototype]]: Object
```

```
▼ {color: 'blanco', diseño: 'vengadores', material: 'plastico', medidas: {...}, manija: false} ⓘ  
  color: "blanco"  
  diseño: "vengadores"  
  manija: false  
  material: "plastico"  
  ► medidas: {diametro: 6, altura: 9}  
  ► [[Prototype]]: Object
```


└─ Agregar propiedades

Podemos agregar propiedades a los objetos declarando la nueva propiedad al objeto y su valor de la siguiente forma

nombreObjeto.nuevaPropiedad = nuevoValor

```
auto.asientos = 2
auto.baul = "sin baul"

let medidas = {
  longitud: 4.5,
  ancho: 2
}

auto.medidas = medidas
```

```
▼ {marca: 'ford', color: 'rojo', tipo: 'deportivo', seguro: true, puertas: 3, ...} ⓘ
  asientos: 2
  baul: "sin baul"
  color: "rojo"
  ▶ distancia: (t,v) => t*v
    marca: "ford"
  ▶ medidas: {longitud: 4.5, ancho: 2}
    puertas: 3
    seguro: true
    tipo: "deportivo"
    vel_max: 250
  ▶ [[Prototype]]: Object
```

Eliminar propiedades

Podemos eliminar propiedades con la palabra reservada de JS **delete** con la siguiente sentencia

delete nombreObjeto.nombrePropiedad

```
delete pantalon.tipo  
delete pantalon.bolsillo_lat  
delete pantalon.cierre
```

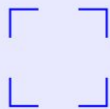
```
delete taza.manija  
delete taza.diseño
```

```
▼ {material: 'jean', color: 'rojo', talla: 42, marca: 'Levis'} ⓘ  
  color: "rojo"  
  marca: "levis"  
  material: "jean"  
  talla: 42  
  ► [[Prototype]]: Object
```

```
▼ {color: 'blanco', material: 'plastico', medidas: {...}} ⓘ  
  color: "blanco"  
  material: "plastico"  
  ► medidas: {diámetro: 6, altura: 9}  
  ► [[Prototype]]: Object
```



Constructor y Métodos



Constructor Object

Otra manera de crear objetos en JavaScript es instanciando al constructor *"Object"*

```
let spock = new Object({
  tipo: "perro",
  raza: "perro perro",
  edad: 5,
  color: "negro",
  familia: {
    hermanos: 5,
    hijos: 7
  },
  ladrar: () => console.log('guau!')
})
```

llamamos **métodos** a las funciones que solamente aplican a un objeto

ladrar() es un método **exclusivo** del objeto spock y no funcionará para los otros objetos

existen métodos disponibles para **TODOS** los objetos, algunos son:

https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Global_Objects/Object

Object.keys

Devuelve un array con el nombre de las propiedades del objeto

```
> Object.keys(pantalon)
```

```
< ▼ (7) ['material', 'color', 'tipo', 'talle', 'bolsillo_lat', 'cierre', 'marca'] ⓘ
```

```
  0: "material"
```

```
  1: "color"
```

```
  2: "tipo"
```

```
  3: "talle"
```

```
  4: "bolsillo_lat"
```

```
  5: "cierre"
```

```
  6: "marca"
```

```
length: 7
```

```
▶ [[Prototype]]: Array(0)
```

```
> Object.keys(spock)
```

```
< ▼ (6) ['tipo', 'raza', 'edad', 'color', 'familia', 'ladrar'] ⓘ
```

```
  0: "tipo"
```

```
  1: "raza"
```

```
  2: "edad"
```

```
  3: "color"
```

```
  4: "familia"
```

```
  5: "ladrar"
```

```
length: 6
```

```
▶ [[Prototype]]: Array(0)
```


Object.values

Devuelve un array con los valores de las propiedades del objeto

```
> Object.values(pantalon)
< ▼ (7) ['jean', 'rojo', 'oxford', 42, false, 'botones', 'levis'] ⓘ
  0: "jean"
  1: "rojo"
  2: "oxford"
  3: 42
  4: false
  5: "botones"
  6: "levis"
  length: 7
  ▶ [[Prototype]]: Array(0)
```

```
> Object.values(spock)
< ▼ (6) ['perro', 'perro perro', 5, 'negro', {...}, f] ⓘ
  0: "perro"
  1: "perro perro"
  2: 5
  3: "negro"
  4: {hermanos: 5, hijos: 7}
  5: () => console.log('guau!')
  length: 6
  ▶ [[Prototype]]: Array(0)
```

Object.entries

Crea un array por cada clave, valor del objeto

```
> Object.entries(pantalon)
< ▼ (7) [Array(2), Array(2), Array(2), Array(2), Array(2), Array(2), Array(2)] ⓘ
  ▶ 0: (2) ['material', 'jean']
  ▶ 1: (2) ['color', 'rojo']
  ▶ 2: (2) ['tipo', 'oxford']
  ▶ 3: (2) ['talle', 42]
  ▶ 4: (2) ['bolsillo_lat', false]
  ▶ 5: (2) ['cierre', 'botones']
  ▶ 6: (2) ['marca', 'levis']
    length: 7
  ▶ [[Prototype]]: Array(0)
```

```
> Object.entries(spock)
< ▼ (6) [Array(2), Array(2), Array(2), Array(2), Array(2), Array(2)] ⓘ
  ▶ 0: (2) ['tipo', 'perro']
  ▶ 1: (2) ['raza', 'perro perro']
  ▶ 2: (2) ['edad', 5]
  ▶ 3: (2) ['color', 'negro']
  ▶ 4: (2) ['familia', {...}]
  ▶ 5: (2) ['ladrar', f]
    length: 6
  ▶ [[Prototype]]: Array(0)
```

¡Muchas gracias!

MIND
HUB.