

# Objetos JS

Link Notion: <https://cherry-client-b8f.notion.site/Objetos-JS-24c911d84e0d80058582f49147cfa7e6?pvs=73>

## O que são objetos?

Em JavaScript, objetos são estruturas que permitem armazenar e organizar dados em formato de **pares chave-valor**.

São como uma coleção de variáveis e funções, mas com nomes (chaves) que descrevem o que cada dado representa.

Como declarar:

```
let pessoa = {};
```

Dentro das chaves, você pode colocar propriedades e funções e seus respectivos valores, separados por vírgula.

```
const pessoa = {  
  nome: "Ana",  
  idade: 25,  
  saudacao: function() {  
    console.log("Olá, meu nome é " + this.nome);  
  }  
};
```

Também é possível utilizar propriedades com múltiplas palavras, porém devem utilizar as aspas, como no exemplo abaixo:

```
const pessoa = {  
  nome: "Ana",  
  idade: 25,  
  "Gosta de cachorros": true,  
  saudacao: function() {  
    console.log("Olá, meu nome é " + this.nome);  
  }  
};
```

## Como acessar as propriedades de um objeto?

Para acessar as propriedades de um objeto, basta utilizar o nome do objeto e o ponto(.) com o nome da propriedade.

```
alert( pessoa.nome ); // Ana  
alert( pessoa.idade ); // 25
```

Porém essa maneira de acessar não funciona com propriedades que possuem mais de uma palavra. Neste caso existe uma alternativa, que é utilizar a mesma forma que acessamos posições de um array.

```
alert(pessoa["Gosta de cachorros"]); // true
```

## Como adicionar uma propriedade à um objeto?

Assim como acessamos valores de um objeto, podemos utilizar o ponto(.) ou as chaves para criar novas propriedades em um objeto.

```
const pessoa = { nome: "Ana" };  
pessoa.idade = 25;  
  
console.log(pessoa); // { nome: "Ana", idade: 25 }  
  
//Outro exemplo usando as chaves
```

```
const pessoa = { nome: "Ana" };
pessoa["cidade"] = "São Paulo";

console.log(pessoa);
// { nome: "Ana", cidade: "São Paulo" }
```

## Como remover uma propriedade?

Para remover uma propriedade, utilize o comando DELETE.

```
delete pessoa.idade;
```

## Forma curta de declaração de propriedade

Quando estamos trabalhando dentro de uma aplicação, é muito comum utilizarmos variáveis como valores, como no exemplo abaixo:

```
function makeUser(name, age) {
  return {
    name: name,
    age: age,
    // ...outras propriedades
  };
}

let user = makeUser("John", 30);
alert(user.name); // John
```

Isso se tornou tão comum, que há uma maneira mais curta que evita a repetição de nomes, escrevendo uma única vez.

```
function makeUser(name, age) {
  return {
    name, // mesmo que name: name
    age,  // mesmo que age: age
    // ...
  };
}
```

## Loops em objetos

Como podemos passar por todas as propriedades de um objeto? Temos uma forma de fazer isso que é o **for...in**. Neste caso assim como os loops de array, é criada uma variável que irá passar por cada propriedade executando algum tipo de ação.

```
let user = {
  name: "John",
  age: 30,
  isAdmin: true
};

for (let key in user) {
  // propriedades
  alert( key ); // name, age, isAdmin
  // valores das propriedades
}
```

```
alert( user[key] ); // John, 30, true  
}
```

## Referências

<https://javascript.info/object>