

MAIS JAVASCRIPT

Destructuring

Destructuring

Permite a desestruturação de Arrays e Objetos. Atribuindo suas propriedades à novas variáveis.

```
const carro = {  
  marca: 'Fiat',  
  ano: 2018,  
  portas: 4,  
}  
  
const {marca, ano} = carro;  
  
console.log(marca); // Fiat  
console.log(ano); // 2018
```

Destructuring Objects

A desestruturação irá facilitar a manipulação de dados. Principalmente quando temos uma grande profundidade de objetos.

```
const cliente = {  
  nome: 'Andre',  
  compras: {  
    digitais: {  
      livros: ['Livro 1', 'Livro 2'],  
      videos: ['Video JS', 'Video HTML']  
    },  
    fisicas: {  
      cadernos: ['Caderno 1']  
    }  
  }  
}  
  
console.log(cliente.compras.digitais.livros);  
console.log(cliente.compras.digitais.videos);  
  
const {livros, videos} = cliente.compras.digitais;
```

```
console.log(videos);
```

Nesting

É possível aninhar uma desestruturação dentro de outra.

```
const cliente = {
  nome: 'Andre',
  compras: {
    digitais: {
      livros: ['Livro 1', 'Livro 2'],
      videos: ['Video JS', 'Video HTML']
    },
    fisicas: {
      cadernos: ['Caderno 1']
    }
  }
}

const {fisicas, digitais, digitais: {livros, videos}} =
cliente.compras;

console.log(fisicas);
console.log(livros);
```

[COPIAR](#)

```
console.log(digitais);
```

Nome das Variáveis

É necessário indicar o nome da propriedade que você deseja desestruturar de um objeto. É possível mudar o nome da variável final com:

```
const cliente = {  
  nome: 'Andre',  
  compras: 10,  
}  
  
const {nome, compras} = cliente;  
// ou  
const {nome: nomeCliente, compras: comprasCliente} = cliente;
```

Valor Inicial

Caso a propriedade não exista o valor padrão dela será `undefined`. É possível modificar este valor no momento da desestruturação.

```
const cliente = {  
  nome: 'Andre',  
  compras: 10,  
}  
  
const {nome, compras, email = 'email@gmail.com', cpf} =  
cliente;  
console.log(email) // email@gmail.com  
console.log(cpf) // undefined
```


Destructuring Arrays

Para desestruturar array's você deve colocar as variáveis entre `[]` colchetes.

```
const frutas = ['Banana', 'Uva', 'Morango'];

const primeiraFruta = frutas[0];
const segundaFruta = frutas[1];
const terceiraFruta = frutas[2];

// Com destructuring
const [primeira, segunda, terceira] = frutas;
```

Declaração de Variáveis

A desestruturação pode servir para declararmos uma sequência de variáveis.

```
const primeiro = 'Item 1';  
const segundo = 'Item 2';  
const terceiro = 'Item 3';  
// ou  
const [primeiro, segundo, terceiro] = ['Item 1', 'Item 2',  
  'Item 3'];
```

Argumento Desestruturado

Se uma função espera receber como argumento um objeto, podemos desestruturar ele no momento da declaração.

```
function handleKeyboard(event) {  
  console.log(event.key);  
}  
  
// Com Destructuring  
function handleKeyboard({key}) {  
  console.log(key);  
}  
  
document.addEventListener('keyup', handleKeyboard);
```

Exercícios

```
// Extraia o backgroundColor, color e margin do btn  
const btn = document.querySelector('button');  
const btnStyles = getComputedStyle(btn);  
  
// Troque os valores das variáveis abaixo  
let cursoAtivo = 'JavaScript';  
let cursoInativo = 'HTML';  
  
// Corrija o erro abaixo  
const cachorro = {  
  nome: 'Bob',  
  raca: 'Labrador',  
  cor: 'Amarelo'  
}  
  
const {bobCor: cor} = cachorro;
```