

Objetos e Arrays

FUTURE4

Big picture

Visão geral

- Já vimos vários tipos de valores simples, como números, booleanos e strings
- Agora, vamos ver tipos mais complexos, que nos permitem modelar o mundo de forma mais apropriada
- Permitem uma melhor legibilidade e maior abstração

Objetos

O que é

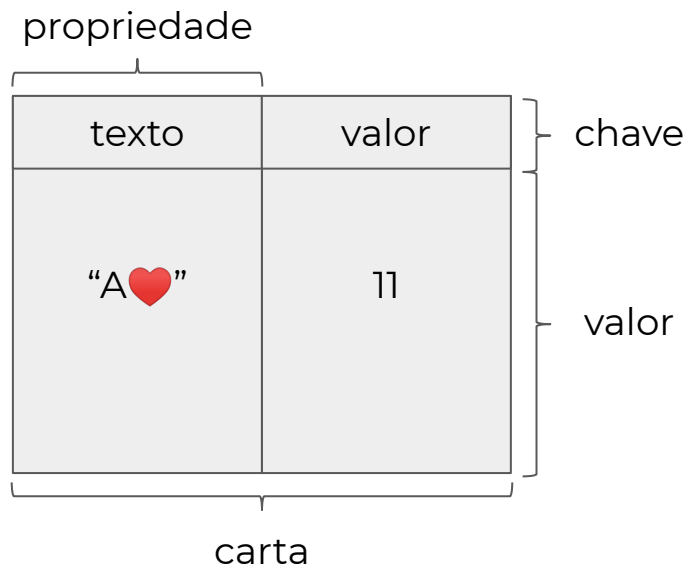
- Tipo de dados usado para **agrupar** outros dados
- Gaveta com divisórias
- Variável que guarda variáveis

Estrutura

The diagram illustrates the structure of a user object. It consists of a table with three columns: 'nome', 'idade', and 'email'. The first row contains the property names, and the second row contains their corresponding values. Annotations include a bracket above the first column labeled 'propriedade', a bracket below the entire table labeled 'usuario', a bracket to the right of the first row labeled 'chave', and a bracket to the right of the second row labeled 'valor'.

nome	idade	email
"Astrodev"	25	"astrodev@future4.com.br"

Estrutura



Criando um objeto

- objeto
- chave
- valor

```
const usuario = {  
  nome: 'Astrodev',  
  idade: 25,  
  email: 'astrodev@future4.com.br'  
}
```


Criando um objeto

```
const carta = {  
  texto: 'A♥',  
  valor: 11  
}
```

- objeto
- chave
- valor

Acessando propriedades - notação do ponto

- objeto
- chave
- valor

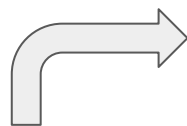
```
const usuario = {  
  nome: 'Astrodev',  
  idade: 25,  
  email: 'astrodev@future4.com.br'  
}
```

```
const nomeDoUsuario = usuario.nome;  
console.log(nomeDoUsuario) // imprime "Astrodev"
```

Acessando propriedades - notação de colchetes

- objeto
- chave
- valor

```
const usuario = {  
  nome: 'Astrodev',  
  idade: 25,  
  email: 'astrodev@future4.com.br'  
}
```



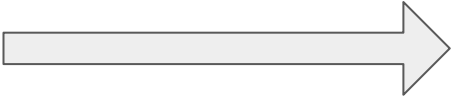
string com a
chave

```
const nomeDoUsuario = usuario['nome'];  
console.log(nomeDoUsuario) // imprime "Astrodev"
```

Acessando propriedades - notação de colchetes

- objeto
- chave
- valor

```
const usuario = {  
  nome: 'Astrodev',  
  idade: 25,  
  email: 'astrodev@future4.com.br'  
}
```

```
const chave = 'nome';  string com a  
chave  
const nomeDoUsuario = usuario[chave];  
console.log(nomeDoUsuario) // imprime "Astrodev"
```

Definindo propriedades

```
const usuario = {  
  nome: 'Astrodev',  
  idade: 25,  
  email: 'astrodev@future4.com.br'  
}
```

- objeto
- chave
- valor

```
usuario.nome = 'Pedro';
```

```
usuario['email'] = 'pedro.darvas@future4.com.br';
```

```
console.log(usuario['nome']) // imprime "Pedro"
```

```
console.log(usuario.email) // imprime "pedro.darvas@..."
```

Características extras

- Podem conter **qualquer tipo** de propriedades: outros objetos, arrays, funções...
- Propriedades **não são ordenadas**

Demo time 🐾

Pausa :)

Arrays

O que é

- Tipo de dados usado agrupar dados **sequenciais**
- Gaveta com divisórias **numeradas**
- Tipo de objeto especial: chaves são números e propriedades são **ordenadas**
- Normalmente usado para modelar **listas**

Estrutura

0	1	2	}	chave
"Astrodev"	"Pedro"	"Paula"		
}				valor
}				usuarios

Criando um array

- array
- chave
- valor

```
const usuario = [ 'Astrodev', 'Pedro', 'Paula' ];
```

Acessando propriedades

- array
- chave
- valor

```
const usuario = ['Astrodev', 'Pedro', 'Paula'];  
const primeiroUsuario = usuario[0];  
console.log(primeiroUsuario); // imprime 'Astrodev'
```

Protótipos

- Alguns tipos em JS possuem protótipos
- Métodos e propriedades **padrões**
- Quando criamos uma variável, temos acesso a eles
- Podemos criar nossos próprios protótipos

Protótipo do array

- `length`: retorna tamanho do array
- `indexOf()`: retorna index de um determinado valor
- `push()`: adiciona elemento ao array
- `pop()`: remove último elemento do array
- `splice()`: remove elementos específicos

Demo time 🐾

Pausa :)

Características comuns

Referências

- Objetos e arrays não são valores absolutos
- A variável guarda uma **referência**
- Dois objetos com as mesmas propriedades não são iguais
- Quando alteramos um item de um objeto ou array, alteramos em todos os lugares

Desestruturação

- Método para “*abrir*” objetos e arrays
- Cria uma cópia de cada propriedade
- Pode ser usado para juntar dois objetos ou arrays

Demo time 🐾

Dúvidas?

Obrigado!