

# SEJAM BEM-VINDOS!



## Jornada DEV AM. CE. RR

Formação gratuita  
em Desenvolvimento  
Front-End SENAI



# Começamos em breve!

A aula vai iniciar em **5 minutos**.

Relaxe e aproveite a música enquanto aguardamos todos chegarem!



## Lógica com JavaScript #09

# Missão de hoje

- Introdução aos objetos
- Trabalhando com objetos

# Objetos em JavaScript

Agrupando informações de forma  
inteligente

**Objetos em JavaScript, assim como em muitas outras linguagens de programação, podem ser comparados com objetos na vida real.**

**O conceito de objetos em JavaScript pode ser entendido com objetos tangíveis da vida real.**

# Por que precisamos de objetos

Os objetos permitem **agrupar várias "variáveis"** em uma **única estrutura**, tornando os dados mais organizados e relacionados.



```
// Sem objeto  
let primeiroNome = "Silvio";  
let sobrenome = "Santos";  
let idade = 45;
```



```
// Com objeto  
let pessoa = {  
  primeiroNome: "Silvio",  
  sobrenome: "Santos",  
  idade: 45  
};
```

# Por que precisamos de objetos

- **Objetos:** são parecidos com vetores, porém, ao invés de estarmos limitados aos índices numéricos de 0 a N (0, 1, 2 ... N), podemos dar nomes aos índices. Inicializamos um objeto com “{ }” ao invés de “[ ]”.
- **Chaves (Keys):** Chamamos os índices de um objeto de chaves. Quando inserimos um item em um objeto criamos um par chave-valor (**key-value**).
- O acesso se dá através de um “.” (ponto) após o nome do objeto, seguido do nome da chave.

Ex.1: `pessoa.nome;`

Ex.2: `pessoa['nome'];`



Um objeto é uma **coleção de pares chave-valor**, ideal para representar entidades como pessoas, produtos, usuários, etc.

# Por que os objetos são importantes

Os objetos representam e descrevem entidades do mundo real dentro do código.

Objetos são usados em **APIs, sites e sistemas** para armazenar dados estruturados.

```
let usuario = {  
  nome: "NASA",  
  repositorios: 420,  
  site: "https://github.com/nasa"  
};  
  
console.log(usuario.nome.toLowerCase()); // "nasa"
```

# Sintaxe dos objetos

A sintaxe de um objeto é simples e flexível.

```
let pessoa = {  
  primeiroNome: "Agostinho",  
  sobrenome: "Carrara",  
  idade: 60,  
};
```

The diagram shows a code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code is as follows:

```
let pessoa = {  
  primeiroNome: "Agostinho",  
  sobrenome: "Carrara",  
  idade: 60,  
};
```

Annotations with arrows pointing to specific parts of the code:

- Declaração da variável** (orange arrow) points to the variable `pessoa`.
- Chaves para declaração literal do objeto.** (orange arrow) points to the opening curly brace `{`.
- declaração de propriedade** (green arrow) points to the property name `primeiroNome`.
- Valor da propriedade** (blue arrow) points to the property value `"Agostinho"`.
- A red arrow points to the comma `,` at the end of the `idade` property.

Pode parecer um detalhe simples, mas não subestime a importância da vírgula nos objetos! Ela é obrigatória para separar pares de chave e valor

# Regras importantes

- Cada par **chave: valor** é separado por vírgula.
- Pode-se quebrar linhas para melhor legibilidade.
- A vírgula após o último item é **opcional**.

# Acessando e alterando objetos

Use o operador ponto (.) para **acessar** ou **modificar** o valor de uma chave.

```
let pessoa = { nome: "Dani", idade: 17 };  
console.log(pessoa.nome); // "Dani"  
console.log(pessoa.idade) // 17  
  
pessoa.idade = 25;  
console.log(pessoa.idade); // 25
```

Anote aí:

`objeto.chave`  
*Acessa o valor*

`objeto.chave = novoValor`  
*altera o valor*

# Adicionando novas chaves

Podemos criar **novas propriedades** a qualquer momento.

```
let usuario = { nome: "Ana" };
usuario.idade = 30;
usuario.email = "ana@email.com";

console.log(usuario)
/**
{ nome: 'Ana', idade: 30, email: 'ana@email.com'}
**/
```

# Cuidado!

Erros de maiúsculas/minúsculas causam problemas:



```
usuario.Nome // ✗ undefined
```

## Intervalo!

Finalizamos o nosso primeiro período de hoje.

Nos vemos em 15min.

**CE**

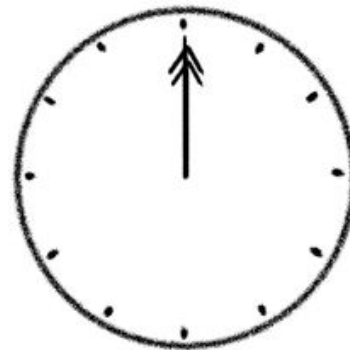
**Início:** 20:00

**Retorno:** 20:15

**AM. RR**

**Início:** 19:00

**Retorno:** 19:15





# 5 minutinhos, sem perder a amizade 😅

Crie um objeto `carro` com as seguintes chaves:

- `marca`
- `modelo`
- `ano`
- `cor`

Depois:

1. Acesse e exiba a cor no console.
2. Altere a cor.
3. Adicione uma nova chave `proprietario`.

<LAB365>

**SENAI**