




# Semana 06: Lógica e Introdução ao Javascript

Aline Pereira

 line.asramos@gmail.com

 @alinespereira

 Aline Pereira

 @line.arp

18 de setembro de 2021

## Objetivo

### Conteúdo

Tópicos

Mapa mental

### Lógica

### Algoritmo

### Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

### Referências

# Objetivo

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

# O que esperar desta aula?

- Compreender conceitos fundamentais de lógica de programação;
- Entender o que é um algoritmo;
- Conhecer a sintaxe básica de Javascript;
- Ler e escrever programas simples em Javascript.

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

# O que esperar desta aula?

- Compreender conceitos fundamentais de lógica de programação;
- Entender o que é um algoritmo;
- Conhecer a sintaxe básica de Javascript;
- Ler e escrever programas simples em Javascript.

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

# O que esperar desta aula?

- Compreender conceitos fundamentais de lógica de programação;
- Entender o que é um algoritmo;
- Conhecer a sintaxe básica de Javascript;
- Ler e escrever programas simples em Javascript.

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

# O que esperar desta aula?

- Compreender conceitos fundamentais de lógica de programação;
- Entender o que é um algoritmo;
- Conhecer a sintaxe básica de Javascript;
- Ler e escrever programas simples em Javascript.

# Conteúdo

## Objetivo

## Conteúdo

### Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

- O que é lógica?
- O que é um algoritmo?
- Console
- Variáveis
- Tipos de dados primitivos
- Operadores aritméticos
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição



# Conteúdo

- O que é lógica?
- O que é um algoritmo?
- Console
- Variáveis
- Tipos de dados primitivos
- Operadores aritméticos
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição

## Objetivo

## Conteúdo

### Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

- O que é lógica?
- O que é um algoritmo?
- **Console**
- Variáveis
- Tipos de dados primitivos
- Operadores aritméticos
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição

Objetivo

Conteúdo

**Tópicos**

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# Conteúdo

- O que é lógica?
- O que é um algoritmo?
- Console
- **Variáveis**
- Tipos de dados primitivos
- Operadores aritméticos
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição

# Conteúdo

- O que é lógica?
- O que é um algoritmo?
- Console
- Variáveis
- **Tipos de dados primitivos**
- Operadores aritméticos
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição

Objetivo

Conteúdo

**Tópicos**

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# Conteúdo

- O que é lógica?
- O que é um algoritmo?
- Console
- Variáveis
- Tipos de dados primitivos
- **Operadores aritméticos**
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição

Objetivo

Conteúdo

**Tópicos**

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# Conteúdo

- O que é lógica?
- O que é um algoritmo?
- Console
- Variáveis
- Tipos de dados primitivos
- Operadores aritméticos
- **Operadores lógicos**
- Estruturas condicionais
- Estruturas de repetição

## Objetivo

## Conteúdo

### Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

- O que é lógica?
- O que é um algoritmo?
- Console
- Variáveis
- Tipos de dados primitivos
- Operadores aritméticos
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição

## Objetivo

## Conteúdo

### Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

- O que é lógica?
- O que é um algoritmo?
- Console
- Variáveis
- Tipos de dados primitivos
- Operadores aritméticos
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição



## Objetivo

## Conteúdo

### Tópicos

#### Mapa mental

## Lógica

## Algoritmo

## Javascript

### Console

### Sintaxe básica

### Variáveis

### Tipos primitivos de dados

### Operadores aritméticos

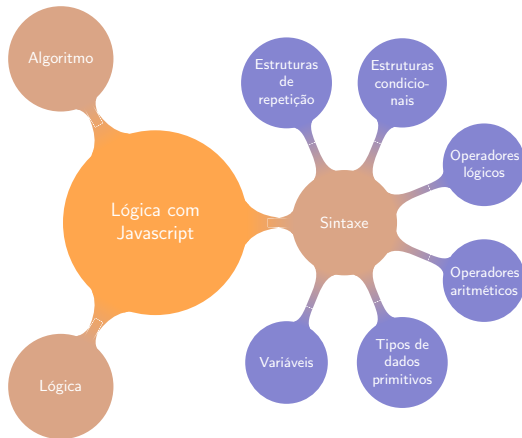
### Operadores de comparação

### Operadores lógicos

### Estruturas condicionais

### Estruturas de repetição

## Referências



Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# Lógica

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Lógica (do grego logos) tem dois significados principais: discute o uso de raciocínio em alguma atividade e é o estudo normativo, filosófico do raciocínio válido. No segundo sentido, a lógica é discutida principalmente nas disciplinas de filosofia, matemática e ciência da computação. Ambos os sentidos se baseando no foco comum referente a harmonia de raciocínio, a proporcionalidade formal entre argumentos, assim sendo, a correta e equilibrada relação entre todos os termos, a total concordância entre cada um deles dentro de um desenvolvimento.[3]*

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Síntaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Lógica (do grego logos) tem dois significados principais: discute o uso de **raciocínio** em alguma atividade e é o estudo normativo, filosófico do raciocínio válido. No segundo sentido, a lógica é discutida principalmente nas disciplinas de filosofia, matemática e ciência da computação. Ambos os sentidos se baseando no foco comum referente a harmonia de raciocínio, a proporcionalidade formal entre argumentos, assim sendo, a correta e equilibrada relação entre todos os termos, a total concordância entre cada um deles dentro de um desenvolvimento.[3]*

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Síntaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Lógica (do grego logos) tem dois significados principais: discute o uso de **raciocínio** em alguma atividade e é o estudo normativo, filosófico do **raciocínio válido**. No segundo sentido, a lógica é discutida principalmente nas disciplinas de filosofia, matemática e ciência da computação. Ambos os sentidos se baseando no foco comum referente a harmonia de raciocínio, a proporcionalidade formal entre argumentos, assim sendo, a correta e equilibrada relação entre todos os termos, a total concordância entre cada um deles dentro de um desenvolvimento.[3]*

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Síntaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Lógica (do grego logos) tem dois significados principais: discute o uso de **raciocínio** em alguma atividade e é o estudo normativo, filosófico do **raciocínio válido**. No segundo sentido, a lógica é discutida principalmente nas disciplinas de filosofia, matemática e ciência da computação. Ambos os sentidos se baseando no foco comum referente a **harmonia de raciocínio**, a proporcionalidade formal entre argumentos, assim sendo, a correta e equilibrada relação entre todos os termos, a total concordância entre cada um deles dentro de um desenvolvimento.[3]*

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Lógica (do grego logos) tem dois significados principais: discute o uso de **raciocínio** em alguma atividade e é o estudo normativo, filosófico do **raciocínio válido**. No segundo sentido, a lógica é discutida principalmente nas disciplinas de filosofia, matemática e ciência da computação. Ambos os sentidos se baseando no foco comum referente a **harmonia de raciocínio**, a proporcionalidade formal entre argumentos, assim sendo, a correta e equilibrada **relação entre todos os termos**, a total concordância entre cada um deles dentro de um desenvolvimento.[3]*

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Lógica (do grego logos) tem dois significados principais: discute o uso de **raciocínio** em alguma atividade e é o estudo normativo, filosófico do **raciocínio válido**. No segundo sentido, a lógica é discutida principalmente nas disciplinas de filosofia, matemática e ciência da computação. Ambos os sentidos se baseando no foco comum referente a **harmonia de raciocínio**, a proporcionalidade formal entre argumentos, assim sendo, a correta e equilibrada **relação entre todos os termos**, a total **concordância** entre cada um deles dentro de um desenvolvimento.[3]*



## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Síntaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Lógica (do grego logos) tem dois significados principais: discute o uso de **raciocínio** em alguma atividade e é o estudo normativo, filosófico do **raciocínio válido**. No segundo sentido, a lógica é discutida principalmente nas disciplinas de filosofia, matemática e ciência da computação. Ambos os sentidos se baseando no foco comum referente a **harmonia de raciocínio**, a proporcionalidade formal entre argumentos, assim sendo, a correta e equilibrada **relação entre todos os termos**, a total **concordância** entre cada um deles dentro de um **desenvolvimento**. [3]*

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

Lógica é:

- raciocínio;
- validade;
- concordância.

Ou seja...

Lógica é:

- raciocínio;
- validade;
- concordância.

Então, como usar a lógica *na programação*?

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

Lógica é:

- raciocínio;
- validade;
- concordância.

Então, como usar a lógica *na programação*? **Algoritmo!**

# Algoritmo

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# O que é um algoritmo?

# O que é um algoritmo?

- Sequência de passos

# O que é um algoritmo?

- Sequência de passos (mas não somente!)



# O que é um algoritmo?

- Sequência de passos (mas não somente!)
- Sequencia de ações

# O que é um algoritmo?

- Sequência de passos (mas não somente!)
- Sequencia de ações (mas não somente!)

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# O que é um algoritmo?

- Sequência de passos (mas não somente!)
- Sequencia de ações (mas não somente!)
- Precisa ser uma sequência?

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# O que é um algoritmo?

- Sequência de passos (mas não somente!)
- Sequencia de ações (mas não somente!)
- Precisa ser uma sequência? Não!

# O que é um algoritmo?

- Sequência de passos (mas não somente!)
- Sequencia de ações (mas não somente!)
- Precisa ser uma sequência? Não!
- Precisa ter coerência!

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

# O que é um algoritmo?

- Sequência de passos (mas não somente!)
- Sequencia de ações (mas não somente!)
- Precisa ser uma sequência? Não!
- Precisa ter coerência!
- Precisa ter *encadeamento lógico*.

## Algoritmo: como fazer café

- Abrir o armário
- Pegar o fervedor (?)
- Colocar 500ml de água no fervedor
- Colocar o fervedor numa boca do fogão
- Acender a boca do fogão onde o fervedor está
- Pegar a garrafa de café sobre a geladeira
- Lavar a garrafa de café
- Pegar o suporte do filtro de papel no armário e colocar na garrafa
- Pegar o filtro de papel no armário, dobrar as bordas e colocar no suporte
- Pegar o pó de café e colocar *qsp* no filtro
- Quando a água estiver quase fervendo, jogar sobre o pó
- Quando tiver coado todo o café, remover o suporte do filtro, encher uma xícara e fechar a garrafa.

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

Em grupos de até 5 alunas, escolham uma tarefa e descrevam o algoritmo para completá-la.

Sugestões:

- Como fazer brigadeiro?
- Como escolher uma calça numa loja de roupas?
- Como aprender algoritmos?
- Como dividir 5 balas entre 3 crianças?
- Como ordenar uma lista com 10 números?



# Javascript

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Síntaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

*Foi originalmente implementada como parte dos navegadores web para que scripts pudessem ser executados do lado do cliente e interagissem com o usuário sem a necessidade deste script passar pelo servidor, controlando o navegador, realizando comunicação assíncrona e alterando o conteúdo do documento exibido, porém os mecanismos JavaScript agora estão incorporados em muitos outros tipos de software host, incluindo servidores em servidores e bancos de dados da Web e em programas que não são da Web, como processadores de texto e PDF, e em tempo de execução ambientes que disponibilizam JavaScript para escrever aplicativos móveis e de desktop, incluindo widgets de área de trabalho. [2]*

Muito mais detalhes: [1]

## Objetivo

## Conteúdo

Tópicos  
Mapa mental

## Lógica

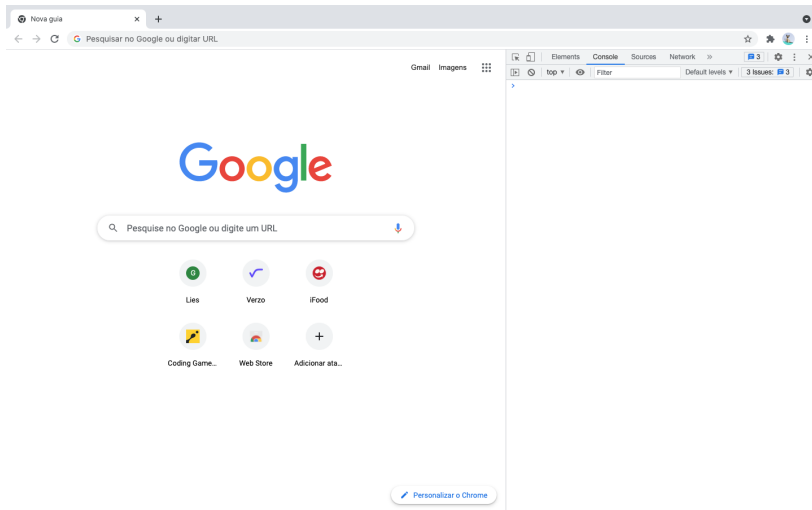
## Algoritmo

## Javascript

### Console

Sintaxe básica  
Variáveis  
Tipos primitivos  
de dados  
Operadores  
aritméticos  
Operadores de  
comparação  
Operadores  
lógicos  
Estruturas  
condicionais  
Estruturas de  
repetição

## Referências



## Algumas funções úteis

- `Number()`: transforma a entrada num número
- `alert()`: mostra um alerta na tela
- `confirm()`: mostra uma mensagem de confirmação na tela
- `prompt()`: exibe uma mensagem na tela com um espaço para a entrada de texto
- `parseInt()`: transforma uma string que representa um número inteiro num `Number`
- `parseFloat()`: transforma uma string que representa um número *float* num `Number`

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

**Sintaxe básica**

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

- **case-sensitive**: maiúsculas e minúsculas são diferenciadas;

```
1 let someVariable;  
2 let SomeVariable;  
3
```

- caracteres **unicode**: caracteres unicode podem ser usados como nomes de variáveis;

```
1 let cinquenta = 50;  
2
```

- instruções: são chamadas *declarações*, e são terminadas por ponto-e-vírgula

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

## Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

- **camelCase**: geralmente, variáveis e funções são nomeadas usando camelCase:

```
1 let aNumber = 24.956;  
2 const greeting = "Boa tarde!";  
3
```

- **PascalCase**: utilizado em classes:

```
1 class Student {  
2     ...  
3 }  
4
```

- **snake\_case**: não é comum ser utilizado em JavaScript:

```
1 var my_variable = 42 // Noooooo!  
2
```

- Comentários: servem para explicar ao leitor o que o código faz

```
1 // A hora que o café da manhã será servido
2 const breakfast = '07:00';
3
4 /*
5     Esta função recebe um horário
6     e verifica se é hora do café:
7         - em caso afirmativo, imprime uma mensagem na tela;
8         - caso contrário, ela não faz nada.
9     */
10 function serveBreakfast(time) {
11     if (time == breakfast) {
12         console.log("Tá na mesa!");
13     }
14 }
15
```

## Objetivo

## Conteúdo

Tópicos  
Mapa mental

## Lógica

## Algoritmo

## Javascript

Console  
Sintaxe básica

**Variáveis**

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

## Referências

## Variáveis cujo valor pode ser alterado posteriormente

```
1 let aVariable;  
2 let aNumber = 23;  
3 let aText = 'Só tem gente bela aqui!';  
4 let anotherNumber = 15.29348;  
5  
6 aVariable = 15;  
7 aNumber = aVariable;  
8 console.log(aNumber); // 15  
9
```



## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

**Variáveis**Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

## Variáveis cujo valor não pode ser alterado

```
1 const aVariable; // Não é permitido declarar uma const sem
   valor
2 const aNumber = 42;
3 const aText = 'Greetings, Earth People!';
4 const g = 9.81;
5 const pi = 3.141592;
6 g = 10; // Não é permitido alterar uma const.
7 const nextNumber = aNumber + 1;
8 console.log(nextNumber); // 43
9
```

## Objetivo

## Conteúdo

[Tópicos](#)  
[Mapa mental](#)

## Lógica

## Algoritmo

## Javascript

[Console](#)  
[Síntaxe básica](#)  
**Variáveis**[Tipos primitivos  
de dados](#)[Operadores  
aritméticos](#)[Operadores de  
comparação](#)[Operadores  
lógicos](#)[Estruturas  
condicionais](#)[Estruturas de  
repetição](#)

## Referências

Uma maneira mais antiga de se declarar variáveis. Podem ser alteradas

```
1 if (true) {  
2     var x = 5;  
3 }  
4 console.log(x); // 5  
5  
6 if (true) {  
7     let y = 5;  
8 }  
9 console.log(y); // ReferenceError: y não está definido  
10
```

# Tipos primitivos de dados

- **String**: representa uma sequência de caracteres (aka um texto);

# Tipos primitivos de dados

- **String**: representa uma sequência de caracteres (aka um texto);
- **Number**: representa um número (independente de ter ou não casas decimais);

# Tipos primitivos de dados

- **String**: representa uma sequência de caracteres (aka um texto);
- **Number**: representa um número (independente de ter ou não casas decimais);
- **Boolean**: os dois valores lógicos `true` (verdadeiro) e `false` (falso);

# Tipos primitivos de dados

- **String**: representa uma sequência de caracteres (aka um texto);
- **Number**: representa um número (independente de ter ou não casas decimais);
- **Boolean**: os dois valores lógicos **true** (verdadeiro) e **false** (falso);
- **null**: um valor que representa a ausência de valor (nulo);

# Tipos primitivos de dados

- **String**: representa uma sequência de caracteres (aka um texto);
- **Number**: representa um número (independente de ter ou não casas decimais);
- **Boolean**: os dois valores lógicos `true` (verdadeiro) e `false` (falso);
- `null`: um valor que representa a ausência de valor (nulo);
- `undefined`: representa um valor que não pode ser definido.

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

**Tipos primitivos  
de dados**Operadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

São delimitadas por " ou por '.

```
1 const name = "Aline";  
2 const city = 'São Carlos';  
3 let frase1 = 'Sou ' + name + '; moro em ' + city;  
4 console.log(frase1); // Sou Aline; moro em São Carlos  
5
```



## Números!

```
1 let meaningOfLife = 42;
2 const pi = 3.14192;
3
4 let g = Number('9.81'); // Number('9.81') transforma a
    string 9.81 em um number
5 console.log(g); // 9.81
6 console.log(typeof(g)); // 'string'
7 g = Number(g);
8 /* Number(g) transforma a string g, que vale '9.81'
9  * em um number, cujo valor é 9.81
10  */
11 console.log(g); // 9.81
12 console.log(typeof(g)); // 'number'
13
```

Observe que o separador decimal *deve ser* um **ponto**.

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

**Tipos primitivos  
de dados**Operadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

## Separa a verdade da mentira

```
1 let a = true;  
2 const b = false;  
3  
4 let isBig = 2 > 1000000;  
5 console.log(isBig); // false  
6  
7 const day = 'cloudy'; // nublado  
8 const isSunny = day == 'sunny'; // o dia está ensolarado?  
9 console.log(isSunny); // false  
10
```

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

**Tipos primitivos  
de dados**Operadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

Verdadeiro (**true**) ou falso (**false**)

```
1 let a = true;
2 const b = false;
3
4 let isBig = 2 > 1000000;
5 console.log(isBig); // false
6
7 const day = 'cloudy'; // nublado
8 const isSunny = day == 'sunny'; // o dia está ensolarado?
9 console.log(isSunny); // false
10
```

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

**Tipos primitivos  
de dados**

Operadores

aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências

Um valor nulo.

```
1 let aNull = null;  
2 console.log(aNull); // null  
3
```

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

**Tipos primitivos  
de dados**Operadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

## Um valor que não foi definido

```
1 let aVariable;  
2 console.log(aVariable); // undefined  
3  
4 const anotherVariable = undefined;  
5 console.log(anotherVariable); // undefined  
6
```

# Exercício: variáveis

## Operadores aritméticos

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dados**Operadores  
aritméticos**Operadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

Vamos inicialmente pensar somente em operações com números

Operador	O que faz?	Exemplo
+	soma dois números	<code>let a = 42 + 2.71;</code>
-	subtrai dois números	<code>let s = 10 - 5;</code>
*	multiplica dois números	<code>let m = 10 * 5;</code>
/	divide dois números	<code>let d = 50 / 8;</code>
%	calcula o resto da divisão	<code>let r = 5 % 3;</code>
++	incrementa o valor de uma variável	<code>i++;</code>
--	decrementa o valor de uma variável	<code>j--;</code>

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dados**Operadores  
aritméticos**Operadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

## Operadores aritméticos

```
1 let a = 42 + 2.71;
2 console.log(a); // 44.71
3
4 let s = 10 - 5;
5 console.log(s); // 5
6
7 let m = 10 * 5;
8 console.log(m); // 50
9
10 let d = 50 / 8;
11 console.log(d); // 6.25
12
```



# Operadores aritméticos

```
1 let r = 5 % 3;
2 console.log(r); // 2
3
4 let i = 0;
5 i++; // equivale a i = i + 1
6 console.log(i); // 1
7
8 let j = 10
9 j--; // equivale a j = j - 1
10 console.log(j); // 9
11
```

# Operadores aritméticos

## Outros operadores importantes

```
1 let a = 0;  
2 a += 5;  
3  
4 let b = 100  
5 b -= 15;  
6  
7 let c = 1;  
8 c *= 2;  
9  
10 let d = 100;  
11 d /= 10;  
12
```

# Operadores aritméticos

E se uma string aparecer??

```
1 const me = "Aline";  
2 const year = 1990;  
3 let message = "Me chamo " + me + ' e nasci em ' + year;  
4 console.log(message); // Me chamo Aline e nasci em 1990  
5
```

# Exercício: operadores

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticos**Operadores de  
comparação**Operadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

## Operadores de comparação

Operador	Nome
==	igual
!=	não igual
===	estritamente igual
!==	estritamente não igual
>	maior que
>=	maior que ou igual
<	menor que
<=	menor que ou igual

## Operadores de comparação

## Objetivo

## Conteúdo

## Tópicos

## Mapa mental

## Lógica

## Algoritmo

## Javascript

## Console

## Sintaxe básica

## Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticos**Operadores de  
comparação**Operadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

```
1 const a = 5;
2 const b = "8";
3
4 console.log(a == b); // false
5 console.log(a != b); // true
6 console.log(a == '5'); // true
7 console.log(a === '5'); // false
8 console.log(b != 8); // false
9 console.log(b !== 8); // true
10
11 console.log(a > 0); // true
12 console.log(a >= 5); // true
13 console.log(a > 5); // false
14 console.log(a < 10); // true
15 console.log(a <= 10); // true
16
```

# Operadores lógicos

Operador	Nome	Ação
&&	E lógico	Se o primeiro valor é verdadeiro, retorna o segundo
	Ou lógico	Retorna o primeiro valor verdadeiro
!	Não lógico	Inverte o valor de uma expressão lógica

# Operadores lógicos

## Expressões convertidas para falso

- `null`
- `undefined`
- `NaN`
- `0`
- `string vazia`



# Operadores lógicos

```
1 true && false; // false
2 true && true; // true
3 true && 5; // 5
4 "" && true; // ""
5
6 true || false; // true
7 true || true; // true
8 true || 5; // true
9 false || ""; // ""
10
11 !true; // false
12 !5; // false
13 !""; // true
14
```

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

**Estruturas  
condicionais**

Estruturas de  
repetição

Referências

# Estruturas condicionais

- Determina se uma ação deve ser executada, baseada numa condição;
- É a estrutura sintática que direciona o código por caminhos lógicos.

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

**Estruturas  
condicionais**

Estruturas de  
repetição

Referências

```
1 let sol = true;  
2  
3 if (sol) {  
4     alert('Vou viajar para a praia!');  
5 }  
6
```

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

**Estruturas  
condicionais**

Estruturas de  
repetição

## Referências

```
1 let sol = true;  
2  
3 if (sol) {  
4     alert('Vou viajar para a praia!');  
5 } else {  
6     alert('Vou ao cinema');  
7 }  
8
```

## if... else if... else

```
1 let hora = 15;
2
3 if (hora <= 12) {
4     alert('Já tomou café?');
5 } else if (hora == 15) {
6     alert('É hora do chá da tarde');
7 } else {
8     alert('Tome água');
9 }
10
```

## switch/case

## Objetivo

## Conteúdo

- Tópicos
- Mapa mental

## Lógica

## Algoritmo

## Javascript

- Console
- Sintaxe básica
- Variáveis
- Típos primitivos de dados
- Operadores aritméticos
- Operadores de comparação
- Operadores lógicos
- Estruturas condicionais**
- Estruturas de repetição

## Referências

```
1 let dia = 'Segunda';
2
3 switch ( dia ) {
4   case 'Segunda':
5     alert('Montag');
6     break;
7   case 'Terça':
8     alert('Dienstag');
9     break;
10  case 'Quarta':
11    alert('Mittwoch');
12    break;
13  case 'Quinta':
14    alert('Donnerstag');
15    break;
16  case 'Sexta':
17    alert('Freitag');
18    break;
19  case 'Sábado':
20    alert('Samstag');
21    break;
22  case 'Domingo':
23    alert('Sonntag');
24    break;
25  default:
26    alert('Não entendi o que disse');
27 }
28
```

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

**Estruturas  
condicionais**

Estruturas de  
repetição

Referências

# Operador ternário

```
1 let idade = 20;  
2 let permissao = idade > 18 ? true : false;  
3
```

# Exercício: condições



# Estruturas de repetição

Usado para executar ações repetidamente:

- **for**: geralmente, o número de execuções é conhecido;
- **while**: geralmente, o número de execuções é desconhecido;

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

**Estruturas de  
repetição**

## Referências

```
1 let passo;  
2 for (passo = 1; passo <= 5; passo++) {  
3   console.log('já dei ' + passo + ' passos');  
4 }  
5 // já dei 1 passos  
6 // já dei 2 passos  
7 // já dei 3 passos  
8 // já dei 4 passos  
9 // já dei 5 passos  
10
```

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

```
1 const passos = 5;
2 let pulos = 0;
3 for (let passo = 1; passo <= passos; passo++) {
4     console.log('já dei ' + passo + ' passos');
5     console.log('agora darei ' + passo + ' pulos');
6     pulos += passos;
7 }
8 console.log('cansei depois de ' + passos + ' passos e ' + pulos + ' pulos');
9 // já dei 1 passos
10 // agora darei 1 pulos
11 // já dei 2 passos
12 // agora darei 2 pulos
13 // já dei 3 passos
14 // agora darei 3 pulos
15 // já dei 4 passos
16 // agora darei 4 pulos
17 // já dei 5 passos
18 // agora darei 5 pulos
19 // cansei depois de 5 passos e 15 pulos
20
```

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionais**Estruturas de  
repetição**

## Referências

```
1 let indiceUVAto = true;
2
3 while (indiceUVAto) {
4   alert('Use filtro solar');
5 }
6
```

```
1 let indiceUVAto = true;
2
3 while (indiceUVAto) {
4   alert('Use filtro solar');
5   indiceUVAto = confirm('o índice UV continua alto?');
6 }
7
```

# Exercício: loops

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionaisEstruturas de  
repetição

## Referências

```
1 function isEven(num) {  
2     if (num % 2 == 0) {  
3         return true;  
4     } else {  
5         return false;  
6     }  
7 }  
8  
9 isEven(2); // true  
10 isEven(7); // false  
11  
12 function sayHello(person) {  
13     alert('Olá, ' + person + '!');  
14 }  
15  
16 sayHello('Aline'); // Mostra um popup com a mensagem "Olá, Aline!"  
17
```

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

**Estruturas de  
repetição**

## Referências

- Representa uma lista de itens
- O primeiro item está na posição 0;
- Um array pode ter qualquer tipo de dados dentro;
- Não é boa prática misturar tipos de dados dentro de um array;

## Objetivo

## Conteúdo

Tópicos

Mapa mental

## Lógica

## Algoritmo

## Javascript

Console

Sintaxe básica

Variáveis

Tipos primitivos  
de dadosOperadores  
aritméticosOperadores de  
comparaçãoOperadores  
lógicosEstruturas  
condicionais**Estruturas de  
repetição**

## Referências

```
1 let minhaSaladaDeFrutas = ['banana', 'maçã', 'uva', 'morango']
2
3 let arrayHeterogenea = ['string', true, false, 988776, [1, 1, 2,3, 44, 5678] ]
4
5 console.log(minhaSaladaDeFrutas[0]);
6 console.log(minhaSaladaDeFrutas[2]);
7 console.log(arrayHeterogenea[1]);
8
```



# Referências

Objetivo

Conteúdo

Tópicos

Mapa mental

Lógica

Algoritmo

Javascript

Console

Sintaxe básica

Variáveis

Típos primitivos  
de dados

Operadores  
aritméticos

Operadores de  
comparação

Operadores  
lógicos

Estruturas  
condicionais

Estruturas de  
repetição

Referências



The Mozilla Foundation.

Javascript.

<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>.

Acessado em: 27/08/2021.



Wikipedia.

Javascript.

<https://pt.wikipedia.org/wiki/JavaScript>.

Acessado em: 17/09/2021.



Wikipédia.

Lógica.

<https://pt.wikipedia.org/wiki/Lógica>.

Acessado em: 27/08/2021.