



Lógica de Programação - Aula 1

Santander Coders 2024

Apresentação

Apresentação

- Nome
- Curiosidade

- Conteúdo didático do LMS

▼ Lógica de Programação I (JS/TS)

Material do Aluno

📄

Sistema de Tipos (type system) da linguagem

Material do Aluno

📄

Estados

Material do Aluno

📄

Constantes e Variáveis (valores únicos) e sua relação com o conceito matemático

Material do Aluno

📄

Uso de funções nativas da linguagem

Material do Aluno

📄

Coerções (mudança de tipos)

Material do Aluno

📄

Operadores e aritmética

Material do Aluno

📄

Vetores e Matrizes (valores múltiplos) e sua relação com o conceito matemático

Santander Coders 2023 | 2º Semestre - JS (3) | #1122

Sistema de Tipos (type system) da linguagem

Existem duas vertentes de linguagens que tratam a alocação de memória de formas diferentes:

Estática: Obriga a declaração dos tipos ao criarmos uma variável. Mesmo que a variável não tenha um valor ainda. Assim a memória pode ser

Linguagens de tipagem estática não permitem que o programador coloque em uma variável já declarada um novo valor que tenha um tipo dif uma variável numérica.

```
// Aqui vai ser uma string
string nome;

// Aqui gera um erro de tipo
nome = 2
```

Dinâmica: Utiliza o valor colocado na variável para inferir que tipo utilizar. A memória só é alocada quando a variável receber um valor.

Linguagens dinâmicas, com algumas exceções, permitem que o tipo do valor mude. Ela irá realocar a memória sempre que necessário.

```
// Aqui vai ser uma string
let nome = "João"

// Agora é um número
nome = 2
```


Material de apoio

- Material da aula será compartilhado pelo drive do LMS



Pasta Geral

0 itens na pasta



Aula 1 - 31 de
jan.

0 itens na pasta



Aula 2 - 2 de
fev.

0 itens na pasta



Aula 3 - 5 de
fev.

0 itens na pasta



Aula 4 - 7 de
fev.

0 itens na pasta



Aula 5 - 9 de
fev.

0 itens na pasta



Aula 6 - 16 de
fev.

0 itens na pasta



Aula 7 - 19 de
fev.

0 itens na pasta



Aula 8 - 21 de
fev.

0 itens na pasta



Aula 9 - 23 de
fev.

0 itens na pasta

Avaliação

- Desafios
- Lista de exercícios
- Projeto*
- Participação

Sistema de Tipos

Sistema de Tipos (Type System)

JavaScript possui um Sistema de Tipos dinâmico e fraco. Isso significa que:

- **Dinâmico:** Os tipos são verificados em tempo de execução, o que permite que uma variável mude seu tipo durante a execução.
- **Fraco:** JavaScript tenta converter tipos automaticamente quando você executa operações entre diferentes tipos de dados, por exemplo, a adição de uma string e um número.

```
1  let x = 5; // x é do tipo número
2  let y = '0lá'; // y é do tipo string
3
4  x = x + y; // JavaScript tenta converter y para número
5  console.log(x); // O resultado será a string "50lá"
6
```


Estados

Estados

Em JavaScript, os estados são representados por variáveis que podem armazenar valores. Os valores podem ser alterados durante a execução do programa, refletindo as mudanças de estado.

```
let contador = 0; // Inicializa uma variável de estado
contador++;      // Altera o estado (incrementa o contador)
console.log(contador); // Mostra o novo estado (1)
```

Constantes e Variáveis

Constantes e Variáveis

Em JavaScript, você pode declarar variáveis usando **let** (variáveis mutáveis) ou **const** (constantes imutáveis).

- **Constantes:** Usadas para armazenar valores que não devem ser alterados.
- **Variáveis:** Usadas para armazenar valores que podem mudar.

Vamos a prática!

Uso de Funções Nativas da Linguagem e E/S

Uso de Funções Nativas da Linguagem e E/S

- JavaScript possui um conjunto de funções nativas para realizar várias operações, como **console.log()** para saída no console e **prompt()** para entrada a partir de uma caixa de diálogo.

```
let nome = prompt("Digite seu nome: ");  
console.log("Olá, " + nome + "!");
```

Coerções (Mudança de Tipos)

Uso de Funções Nativas da Linguagem e E/S

- JavaScript realiza coerções automaticamente ao realizar operações entre diferentes tipos de dados. No entanto, você pode forçar uma coerção usando funções como **parseInt()** e **parseFloat()**

```
let numero = 5;
let texto = "10";    Replace `"10"` with `'10'`
let resultado = numero + texto; // JavaScript converte `numero` para string e realiza a concatenação
console.log(resultado); // O resultado será a string "510"    'console' is not defined.
```

```
let numero = "5";    Replace `"5"` with `'5'`
let numeroInteiro = parseInt(numero);
console.log(numeroInteiro); // O resultado será o número 5    'console' is not defined.
```

Operadores e Aritmética

Uso de Funções Nativas da Linguagem e E/S

- JavaScript suporta uma variedade de operadores para realizar operações matemáticas e lógicas.

Operadores Aritméticos:

- **+** (adição)
- **-** (subtração)
- ***** (multiplicação)
- **/** (divisão)
- **%** (módulo)

Operadores de Comparação:

- **=** (igual a)
- **≠** (diferente de)
- **<** (menor que)
- **>** (maior que)
- **≤** (menor ou igual a)
- **≥** (maior ou igual a)

Operadores Lógicos:

- **&&** (e lógico)
- **||** (ou lógico)
- **!** (negação lógica)

Desafio 01 - Verificar Elegibilidade para Voto

Receba através do prompt:

- **idade**
- **nacionalidade**

Então deve retornar **true** se a pessoa for elegível para votar, o que ocorre quando a idade for **maior ou igual a 16** e a nacionalidade for "**brasileira**", caso contrário, retorna **false**.

Entrada:	idade: 16 - brasileira	⇒ saída: true
	idade: 15 - brasileira	⇒ saída: false
	idade: 18 - brasileira	⇒ saída: true
	idade: 16 - eua	⇒ saída: false
	idade: 15 - eua	⇒ saída: false
	idade: 18 - eua	⇒ saída: false

Desafio 02 - Aprovação de Empréstimo

Receba através do prompt:

- **salario**
- **scoreDeCredito**
- **idade**

Então deve retornar **true** se a pessoa for aprovada para um empréstimo nas seguintes condições:

- **0 salário for maior ou igual a 5000.**
- **0 score de crédito for maior ou igual a 700.**
- **A idade for maior ou igual a 18.**

Entrada:	Salário: 5001 - Score: 701 - idade: 19	⇒ Saída: true
	Salário: 5000 - Score: 700 - idade: 18	⇒ Saída: true
	Salário: 4999 - Score: 700 - idade: 18	⇒ Saída: false
	Salário: 5000 - Score: 699 - idade: 18	⇒ Saída: false
	Salário: 5000 - Score: 700 - idade: 17	⇒ Saída: false

Desafio 03 - Verificar Compatibilidade de Plataformas

Receba através do prompt dois argumentos:

- navegador
- sistemaOperacional

Deve retornar **true** se o navegador for "**Chrome**" ou "**Firefox**" e o sistema operacional for "**Windows**" ou "**Mac**".

Entrada: navegador: Chrome	- sistemaOperacional: Windows	⇒	Saída: true
navegador: Chrome	- sistemaOperacional: Mac	⇒	Saída: true
navegador: Firefox	- sistemaOperacional: Windows	⇒	Saída: true
navegador: Firefox	- sistemaOperacional: Mac	⇒	Saída: true
navegador: Firefox	- sistemaOperacional: Sony	⇒	Saída: false
navegador: Safari	- sistemaOperacional: Mac	⇒	Saída: false

Desafio 04 - Verificar Elegibilidade para Desconto em Compra

Receba através do prompt:

- **idade**
- **compraMinima.**
- **isNewClient**

Deve retornar **true** se a pessoa **for elegível** para um desconto, o que ocorre quando:

- A idade for maior ou igual a **60** anos.
- E a compra for maior ou igual a R\$ **100**.
- E se **isNewClient** for **False**

Entrada: idade: **65** - compraMinima: **100** - isNewClient: **false** ⇒ Saída: **true**
idade: **45** - compraMinima: **120** - isNewClient: **true** ⇒ Saída: **false**
idade: **45** - compraMinima: **120** - isNewClient: **false** ⇒ Saída: **false**
idade: **55** - compraMinima: **50** - isNewClient: **false** ⇒ Saída: **false**