

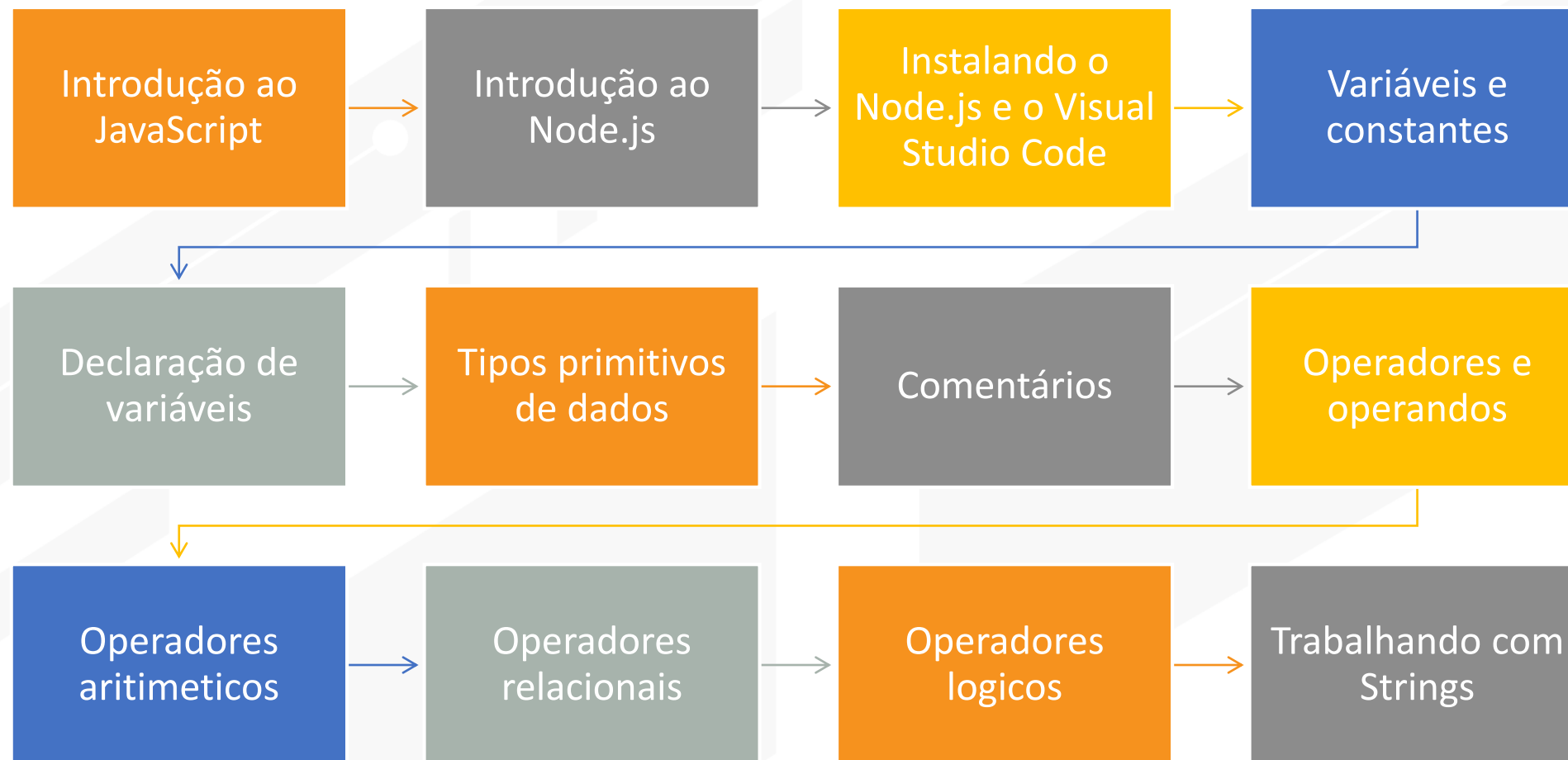


DEPARTAMENTO REGIONAL DE
PERNAMBUCO

INTRODUÇÃO AO DESENVOLVIMENTO BACK-END COM JAVASCRIPT

Agosto / 2022

Conteúdo programático:



Introdução ao Javascript:

JavaScript é uma linguagem de programação interpretada, de script em alto nível com tipagem dinâmica fraca e multiparadigma (protótipos, orientado a objeto, imperativo e, funcional).



É um interpretador Javascript que roda fora dos navegadores;

Por que usar Node.js?

- Muito leve;
- Usa Javascript;
- Muito rápido;
- Tem um dos maiores ecossistemas do mundo;
- Está sendo utilizado fortemente no mercado.



Instalando o Node.js e o Visual Studio Code:

Node.js

- <https://nodejs.org/en/>

Visual Studio Code

- <https://code.visualstudio.com/>

Variáveis e Constantes:

- A **variável** é um espaço reservado na memória do computador para guardar um tipo de informação;
- São denominados **constantes** os dados cujos valores são fixos e que não se alteram ao longo da execução de um programa.
- Tanto as variáveis quanto as constantes devem receber nomes para poder servir de referencia e serem alteradas quando necessário esses nomes são chamados de **identificadores**.

Declaração de variáveis:

- **var** nome_da_variável = valor;
- **let** nome_da_variável = valor;
- **const** nome_da_variável = valor;

Ex:

```
const numero1 = 5;
```

```
let nome = 'Bruno';
```

```
var idade = 20;
```

Tipos primitivos de dados:

- **Number:** tipos numéricos com ou sem casas decimais;

`var idade = 18;`

- **String:** tipo alfanumérico, deve ser definido entre aspas simples ou duplas;

`var nome = 'Bruno';`

- **Boolean:** aceita apenas valores verdadeiros (true) ou falsos (false);

`var teste = true;`

Comentários:

Comentários são trechos de texto, usualmente explicativos, inseridos dentro do programa de forma que não sejam considerados como parte do código.

Existem dois tipos de comentários

- `//` utilizado para comentários de uma linha;
- `/*` e `*/` utilizado para comentários de múltiplas linhas.

Operadores e operandos:

- **Operadores** representam operação que podem ser executadas;
 - **Unários:** quando atuam sobre um único operando.
 - **Binários:** quando atuam sobre dois operandos.
- **Operandos** são os valores sobre os quais o operador trabalha;

A junção de operadores e operandos formão as **expressões**.

Operadores aritméticos:

+	Adição, soma dois valores	$a + b$
-	Subtração, subtrai um valor de outro	$a - b$
*	Multiplicação, multiplica dois valores	$a * b$
/	Divisão, divide um valor pelo outro e retorna o resultado da divisão	a / b
%	Módulo, divide um valor pelo outro e retorna o resto da divisão	$a \% b$

++	Incremento unário, aumenta o valor em 1	$a++$ ou $++a$
--	Decremento unário, diminui o valor em 1	$a--$ ou $--a$
$+=$, $-=$, $*=$, $/=$	Auto-adição, auto-subtração, etc. Calcula o valor atual com o valor à direita e armazena o resultado em si mesmo.	$a += b$ (soma a e b e guarda o resultado em a , equivalente a ' $a = a + b$ ') $a -= b$ (subtrai b de a e guarda o resultado em a , equivalente a ' $a = a - b$ ') $a *= b$ (multiplica a por b e guarda o resultado em a , equivalente a ' $a = a * b$ ') $a /= b$ (divide a por b e guarda o resultado em a , equivalente a ' $a = a / b$ ') $a %= b$ (calcula o resto da divisão de a por b e guarda o resultado em a , equivalente a ' $a = a \% b$ ') $a ++$ (aumenta a em 1 e guarda o resultado em a , equivalente a ' $a = a + 1$ ') $a --$ (diminui a em 1 e guarda o resultado em a , equivalente a ' $a = a - 1$ ') $++a$ (aumenta a em 1 e retorna o valor de a , equivalente a ' $a = a + 1$ ' seguido de ' a ') $--a$ (diminui a em 1 e retorna o valor de a , equivalente a ' $a = a - 1$ ' seguido de ' a ')

Operadores relacionais:

==	Similaridade, retorna true se os valores forem iguais, independente do seu tipo	a == b
!=	Não similaridade, retorna true se os va-	a != b
===	Igualdade, retorna true se os valores e tipos forem iguais	a === b
!==	Desigualdade, retorna true se os valores ou os tipos forem diferentes	a !== b

>	Maior que, retorna true se o valor da esquerda for maior que o da direita	a > b
>=	Maior ou igual que, retorna true se o valor da esquerda for maior ou igual ao da direita	a >= b

<	Menor que, retorna true se o valor da esquerda for menor que o da direita	a < b
<=	Menor ou igual que, retorna true se o valor da esquerda for menor ou igual que o da direita	a <= b

Operadores lógicos:

&&	E lógico (AND), retorna true se ambas expressões retornarem true	a && b
	Ou lógico (OR), retorna true se ao menos uma expressão retornar true	a b
!	Negação lógica (NOT), retorna o valor oposto ao da expressão	!a

Trabalhando com Strings:

O tipo String é um dos mais utilizados dentro da programação isso porque ele possibilita o armazenamento de caracteres alfanuméricos.

O operador “+” realiza a concatenação de Strings.

Existe um serie de características e funções que podemos chamar a partir das Strings para obter variados efeitos:

- **variável.length**: retorna a quantidade de caracteres da String.
- **variável.toUpperCase()**: retorna a String toda em caracteres maiúsculos.
- **variável.toLowerCase()**: retorna a String toda em caracteres minúsculos.
- **parseInt(variável)**: retorna um número inteiro a partir da String (se for possível).
- **parseFloat(variável)**: retorna um numero decimal a partir da String (se for possível).





Bruno Souza | Instrutor