



INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 01 – INTRODUÇÃO AO JAVASCRIPT

O QUE IREMOS APRENDER

- 01** INTRODUÇÃO AO JAVASCRIPT
- 02** JAVASCRIPT
- 03** COMO E ONDE USAR O JAVASCRIPT
- 04** TIPOS DE DADOS
- 05** VARIÁVEIS
- 06** ENTRADA DE DADOS
- 07** CONSOLE
- 08** OPERAÇÕES BÁSICAS
- 09** CONCATENAÇÃO E TEMPLATE STRINGS



INTRODUÇÃO AO JAVASCRIPT

Podemos executar o JavaScript de duas maneiras distintas, sendo uma delas através do próprio navegador. Ao clicarmos com o botão direito do mouse na tela do navegador e selecionarmos "Inspecionar elemento", uma aba será aberta, geralmente na seção "Elementos". Ao lado desta seção, encontramos o Console, onde podemos executar códigos JavaScript. Basta inserir os códigos ali e pressionar "Enter". No entanto, vale ressaltar que essas alterações são temporárias, e ao atualizarmos a página, o código é resetado.

INTRODUÇÃO AO JAVASCRIPT

A segunda abordagem envolve o uso do Node.js, disponível para download em <https://nodejs.org/en/download>. Recomenda-se optar pela versão LTS e escolher a plataforma correspondente para o download. Após o download, basta executar o instalador, seguir as instruções e aguardar a conclusão. Com o Node.js instalado, torna-se possível executar JavaScript no ambiente do Visual Studio Code mas utilizaremos a extensão do vs code chamada live server.

INTRODUÇÃO AO JAVASCRIPT

O JavaScript é uma linguagem de programação de alto nível amplamente empregada no desenvolvimento de aplicações web interativas. Ela proporciona aos desenvolvedores a capacidade de adicionar funcionalidades dinâmicas às páginas web, manipular elementos da página, responder a eventos do usuário e muito mais. Por exemplo, no módulo de HTML/CSS, você aprendeu a criar um botão, mas sem funcionalidade. Para adicionar interatividade, é necessário incorporar o JavaScript.

JAVASCRIPT

Apesar do nome, o JavaScript não tem nada a ver com Java. Criado por Brendan Eich em meados de 1995, o JavaScript é uma linguagem de programação de alto nível desenvolvida com o objetivo de validar formulários HTML. Inicialmente, o JavaScript foi pensado para rodar no servidor. Com o passar do tempo, no entanto, ela se tornou uma linguagem para desenvolvimento front-end. Ao perceber o sucesso do JavaScript, a Microsoft decidiu lançar sua própria linguagem, o JScript, para rodar somente no Internet Explorer.



COMO E ONDE USAR O JAVASCRIPT

A principal maneira de se usar JavaScript é através do navegador. Podemos fazer isso de duas maneiras, a primeira é utilizando a tag html `<script>`, fazendo com que tudo que esteja dentro dessa tag seja interpretado como código JavaScript. Outra maneira é criando um arquivo coma extensão `.js` separado do nosso html e passar um atributo `src` com o caminho do nosso arquivo.

```
1 <body>
2     <script>
3         //Meu código JavaScript
4     </script>
5 </body>
```

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Document</title>
7     <script src="CaminhoDoArquivoJavaScript"></script>
8 </head>
```

TIPOS DE DADOS

Para começar a aprender JavaScript, precisamos primeiro entender os tipos de dados que existem.

O JavaScript possui 6 tipos primitivos, eles são:



```
1 'Texto qualquer' //String- São um conjunto de caracteres, sempre serão entre aspas.  
2 27 //Number- São os números  
3 True //Boolean- São valores verdadeiro ou falso  
4 null//null- São valores nulos. vazios  
5 undefined //undefined- Valores não definidos (são diferentes de null)  
6 Symbol('identificador') //Symbol É um identificador único, utilizado principalmente para definir chaves de objetos
```


VARIÁVEIS

Agora que entendemos os tipos de dados, precisamos entender como armazená-los. Para isso, usamos variáveis. Variáveis são uma forma de armazenar informações, como se fosse uma caixa, podendo reutilizá-las em diversos pontos do nosso programa.

Para criar uma variável, basta usar uma das palavras reservadas para declaração de variável e colocar o nome que quer dar a essa variável. Depois disso, precisamos atribuir um valor a ela usando o operador de atribuição =.



```
1 var minhaVariavel = 'Alguma coisa'
```

VARIÁVEIS

Devemos sempre iniciar uma variável com uma letra, underline (_) ou cifrão (\$) **NÃO** são permitidos ao iniciar uma variável. O JavaScript é case-sensitive, ou seja, ele diferencia letras maiúsculas e minúsculas. Dessa forma, a variável Bola é diferente da variável bola.

Podemos declarar uma variável de 3 formas:

- var – Declara uma variável, como visto anteriormente.
- let – Declara uma variável com escopo de bloco.
- const – Declara uma variável com escopo de bloco, que não pode ser **reatribuída**

DECLARANDO VARIÁVEIS

Declarando uma variável com **var**:

```
var minhaVariavel
```

Declarando uma variável com **let**:

```
let minhaVariavel
```

Declarando uma variável com **const**:

```
const minhaVariavel = 0
```

Note que a **const** não permite declarar uma variável **sem atribuir** um valor a ela. Isso acontece porque, como o nome diz, ela será uma **constante**, que não pode ser **reatribuída**. Portanto, precisamos definir o valor dela logo na declaração.

INICIALIZANDO VARIÁVEIS

nos permitem atribuir um valor a elas mesmo depois da declaração.

Podemos também atribuir um valor diferente, mesmo depois de um valor já ter sido atribuído a elas. Dessa forma, o valor anterior é substituído pelo novo.



```
1  var minhaVariavel
2  minhaVariavel = "alguma coisa"
3
4  let minhaVariavel2
5  minhaVariavel2 = "Teste"
```



```
1  var minhaVariavel = "Alguma coisa"
2  minhaVariavel = "Redeclaração"
3
4  let minhaVariavel2 = "Teste"
5  minhaVariavel2 = "Redeclaração"
```

ENTRADA DE DADOS

Entrada de dados são maneiras que temos para **obter um valor do usuário**.

Imagine a seguinte situação:

Você tem um programa que pega dois números e retorna a média deles, caso você queira que o usuário insira esses dois números você precisa utilizar uma entrada de dados.



ENTRADA DE DADOS



```
1 prompt("Digite alguma coisa")
```



Essa página diz

Digite alguma coisa

OK

Cancelar

Para capturar os dados que o usuário informar, uma opção é utilizar o **prompt**, que é uma caixa de diálogo com uma caixa de texto para o usuário digitar. Para utilizá-lo basta chamar a função **prompt()** e passar para ela a **mensagem** que você quer que apareça para o usuário.

ENTRADA DE DADOS

Lembre-se de **armazenar** o valor que o usuário digitar em uma **variável**.

Seguindo o exemplo anterior:



```
1 let minhaVariavel = prompt("Digite alguma coisa")
```

Dessa forma o valor que o usuário digitar vai ser armazenado dentro da variável “minhaVariavel”. Também temos outras formas de entrada de dados que serão abordadas mais pra frente no curso.

CONSOLE

O console é uma das ferramentas mais importantes que temos para poder entender o nosso código. Ele nos permite mostrar em tela diversas informações.

O que utilizaremos nessa aula é a função **console.log()**. Ela nos permite mostrar em tela a informação que passarmos para ela.

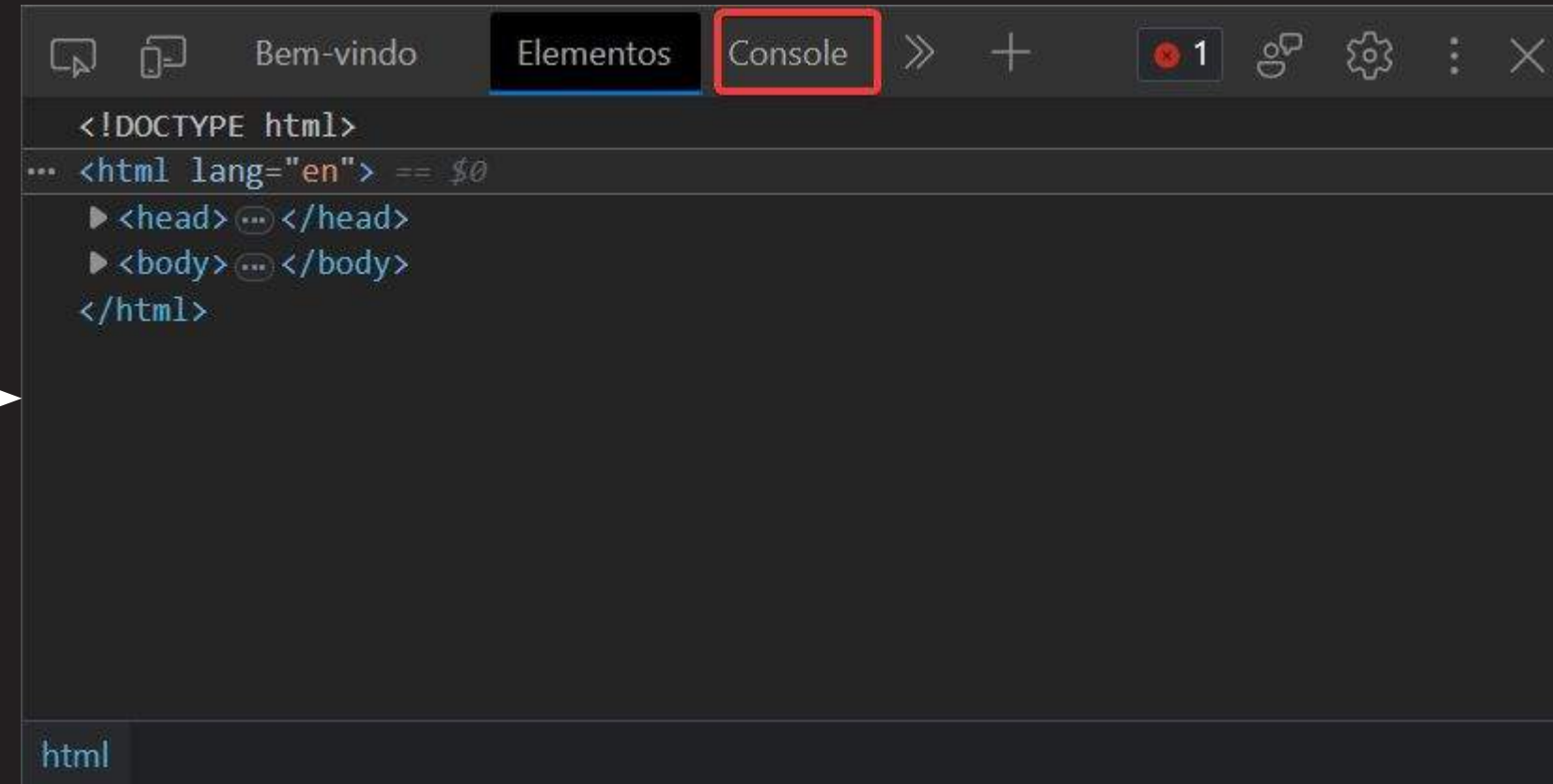
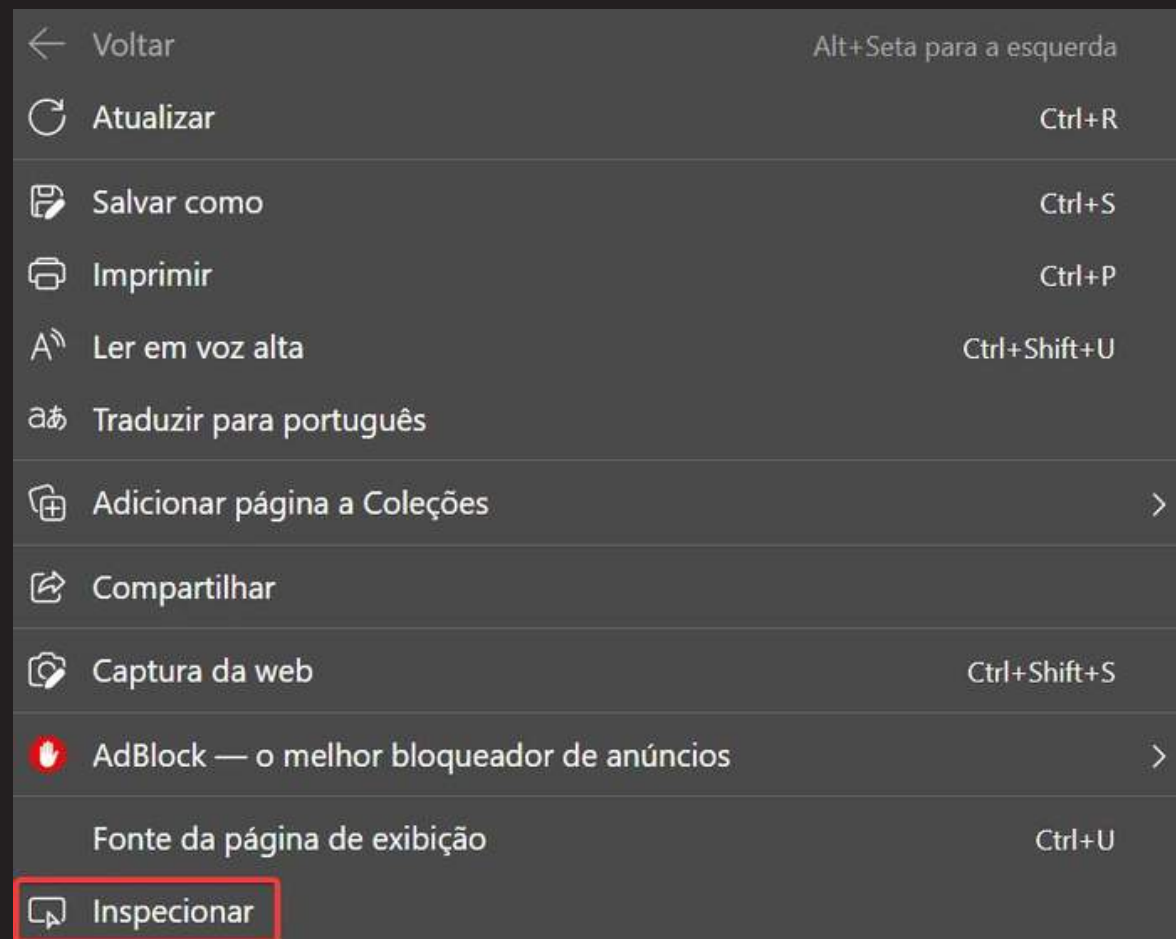
Como no exemplo:



```
1 let minhaVariavel = prompt("Digite alguma coisa")
2 console.log(minhaVariavel)
3 console.log("Alguma coisa")
```

CONSOLE

Para ter acesso ao console basta clicar com o botão direito em tela, ir em **inspecionar** e depois ir em **console**.



OPERADORES ARITMÉTICOS

Os operadores aritméticos são formas que temos para fazer **cálculos aritméticos** em nossos programas. Temos diversos operadores aritméticos no JavaScript.

```
1 + 1 // Soma - É utilizado para somar ou concatenar valores
1 - 1 // Subtração - É utilizado para subtrair valores.
1 * 1 // Multiplicação - É utilizado para multiplicar valores.
1 / 1 // Divisão - É utilizado para dividir valores.
1 ** 1 // Exponenciação - Calcula a base elevada a potência do expoente.
1 % 1 // É utilizado para pegar o resto de uma divisão.
```

OPERADORES ARITMÉTICOS

No exemplo a seguir, nós **declaramos uma variável** para guardar o resultado da nossa soma, então somamos dois valores e atribuímos o resultado dessa operação a **variável “soma”** e por fim mostramos o valor armazenado em soma no nosso console.



```
1  let soma
2  soma = 25+3
3  console.log(soma)
4
5  let multiplicacao
6  multiplicacao = 2*2
7  console.log(multiplicacao)
```


OPERADORES DE ATRIBUIÇÃO

Além do operador de atribuição "=", no JavaScript também temos acesso a mais algumas maneiras de atribuir um valor a uma variável.

Os operadores compostos uma atribuição vão realizar depois operação aritmética e atribuir o valor a variável.

```
minhaVariavel += 1 //Soma
minhaVariavel -= 1 //Subtração
minhaVariavel *= 1 //Multiplicação
minhaVariavel /= 1 //Divisão
minhaVariavel **. 1 //Exponenciação
minhaVariavel %= 1 //Módulo
```

Com eles podemos substituir o



```
1 let soma = 0
2 soma = soma + 3
```



```
1 let soma = 0
2 soma+=3
```


CONCATENAÇÃO E TEMPLATE STRINGS

CONCATENAR

A concatenação no JavaScript é a operação de combinar ou juntar duas ou mais strings. Isso cria uma nova string que contém o conteúdo das strings originais, uma após a outra. Você pode realizar a concatenação usando o operador +

Exemplo 1:

```
1 const nome = "João";  
2 const sobrenome = "Silva";  
3  
4 // Usando o operador +  
5 const nomeCompleto = nome + " " + sobrenome;  
6 console.log(nomeCompleto);
```

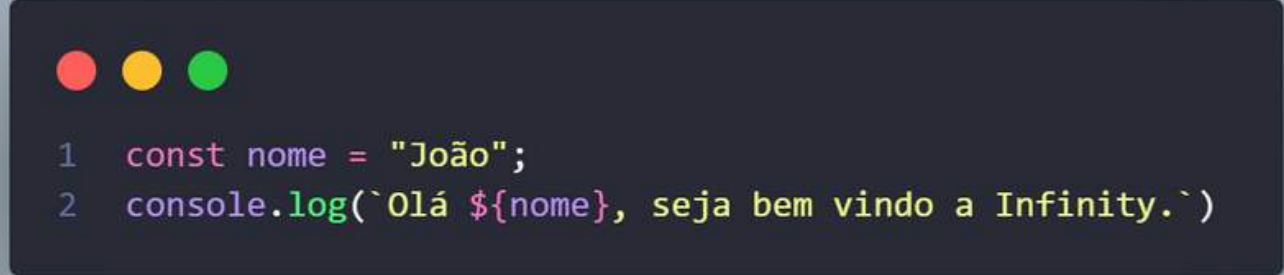
Exemplo 2:

```
1 const nome = "João";  
2 console.log("Olá " + nome + ", seja bem vindo a Infinity");
```

CONCATENAÇÃO E TEMPLATE STRINGS

TEMPLATE STRINGS

No JavaScript, os caracteres ``` ``` são conhecidos como "template literals" ou "template strings". Esse recurso foi introduzido no ECMAScript 6 (também conhecido como ES6) e fornece uma maneira mais flexível de criar strings, permitindo a incorporação de expressões e variáveis diretamente no texto, facilitando a concatenação de strings e melhorando a legibilidade do código.



```
1 const nome = "João";  
2 console.log(`Olá ${nome}, seja bem vindo a Infinity.`)
```

ATIVIDADE PRÁTICA

Atividade 01

Crie um programa que peça ao usuário para digitar seu nome usando o prompt e, em seguida, exiba o nome com uma frase personalizada com template string na tela usando o console.

Atividade 02

Solicite ao usuário que insira dois números usando o prompt, some esses números e exiba o resultado usando o console.

ATIVIDADE PRÁTICA

Atividade 03

Peça ao usuário para digitar sua idade usando o prompt e, em seguida, calcule e exiba o ano de nascimento com uma frase personalizada usando o console.

Atividade 04

Crie um programa que solicite ao usuário que insira o raio de um círculo usando o prompt e, em seguida, calcule e exiba a área do círculo usando o console.

ATIVIDADE PRÁTICA

Atividade 05

Crie um programa que solicite ao usuário que insira um valor em dólares usando o prompt. Em seguida, converta esse valor para euros (1 dólar = 0,85 euros) e exiba o resultado no console.

Atividade 06

Peça ao usuário que insira um preço de um produto usando o prompt. Em seguida, aplique um desconto de 10% a esse preço e exiba o preço com desconto no console.

ATIVIDADE PRÁTICA

Atividade 07

Solicite ao usuário que insira um valor em metros usando o prompt e, em seguida, converta esse valor para centímetros (1 metro = 100 centímetros) e exiba o resultado no console.

DESAFIO PRÁTICO

Faça um Programa que pergunte quanto você ganha por hora e o número de horas trabalhadas no mês. Calcule e mostre o total do seu salário no referido mês, sabendo-se que são descontados 11% para o Imposto de Renda, 8% para o INSS e 5% para o sindicato, faça um programa que nos dê:

- 1.salário bruto.
- 2.quanto pagou ao INSS.
- 3.quanto pagou ao sindicato.
4. salário líquido
5. calcule os descontos e o salário líquido

SE LIGA NO CONTEÚDO DA PRÓXIMA AULA!

AULA 02 DE JAVASCRIPT.
CONDICIONAIS EM JAVASCRIPT.

The logo consists of the letters 'IN' in a white, bold, sans-serif font, centered within a solid red square.

INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

CONDICIONAIS

As estruturas condicionais estão ligadas à tomada de decisão de um algoritmo. Ao utilizar expressões que retornam verdadeiro ou falso, o algoritmo executa o bloco de comandos relativos a este resultado.

Vamos aprender a utilizar as estruturas condicionais no JavaScript, começando pelo uso do if(se), else if e else(senão).



CONDICIONAIS

Exemplos de **condicionais** em JavaScript:



```
1  let idade = 16
2  if(idade >= 16 && idade <= 65){
3      console.log("Obrigado por votar")
4  }
```



INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 01 – INTRODUÇÃO AO JAVASCRIPT