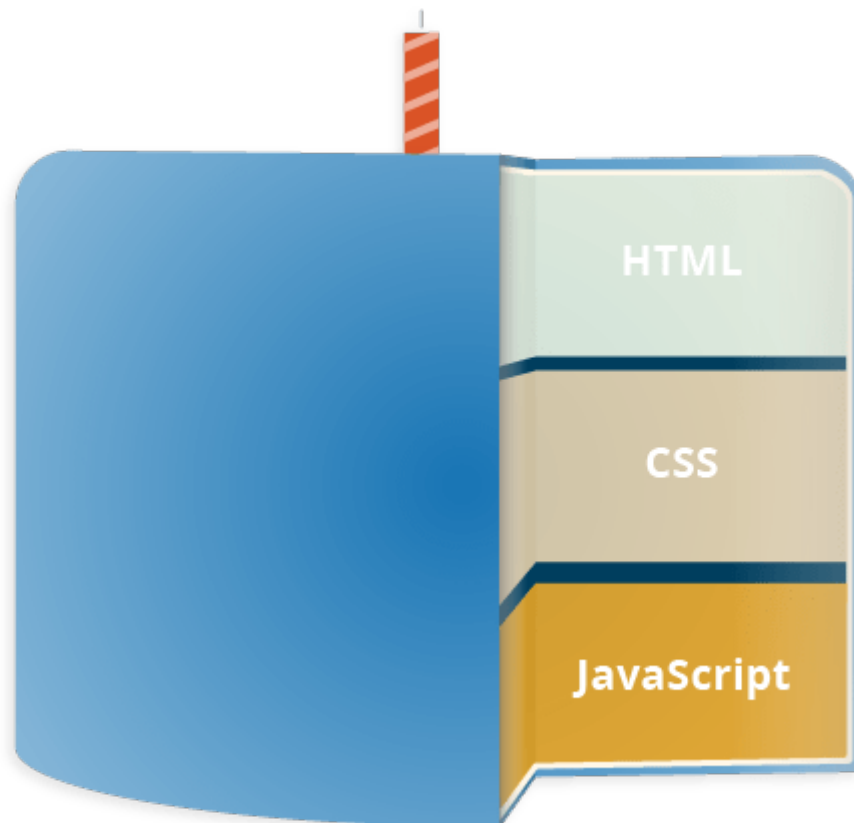


Introdução ao Javascript

JavaScript é uma linguagem de programação que permite a você implementar itens complexos em páginas web — toda vez que uma página da web faz mais do que simplesmente mostrar a você informação estática — mostrando conteúdo que se atualiza em um intervalo de tempo, mapas interativos ou gráficos 2D/3D animados, etc. É a terceira camada do bolo das tecnologias padrões da web ([HTML](#), [CSS](#) e [JavaScript](#)).



- [HTML](#) é a linguagem de marcação que nós usamos para estruturar e dar significado para o nosso conteúdo web.
- [CSS](#) é uma linguagem de regras de estilo que nós usamos para aplicar estilo ao nosso conteúdo HTML.
- [JavaScript](#) é uma linguagem de programação que permite a você criar conteúdo que se atualiza dinamicamente, controlar mídias, imagens animadas, e tudo o mais que há de interessante.

Como chamar ou utilizar o Javascript

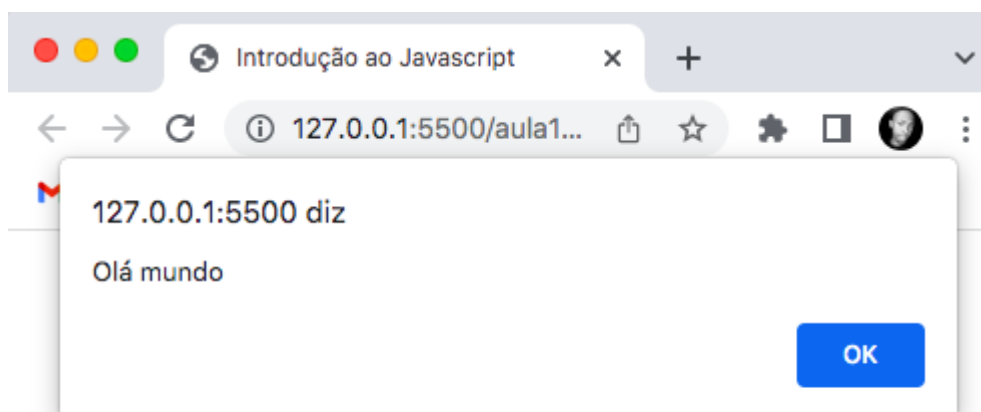
Nós podemos incorporar o javascript no nosso site de duas formas:

1. Internamente

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Introdução ao Javascript</title>
8  </head>
9  <body>
10
11      <!-- Incorporando o JavaScript internamente -->
12      <script>
13          alert("Olá mundo");
14      </script>
15  </body>
16  </html>
```

Como vimos no exemplo acima nós usamos a tag **<script>** isso indica que tudo o que estiver dentro será código javascript, e dentro usamos a função **alert()** que é uma função que permite imprimir uma caixa de diálogo como mostra a imagem abaixo.

Resultado:



2. Externamente

Para incorporar externamente no nosso site vamos primeiro criar um arquivo index.js e neste arquivo vamos colocar o seguinte código:

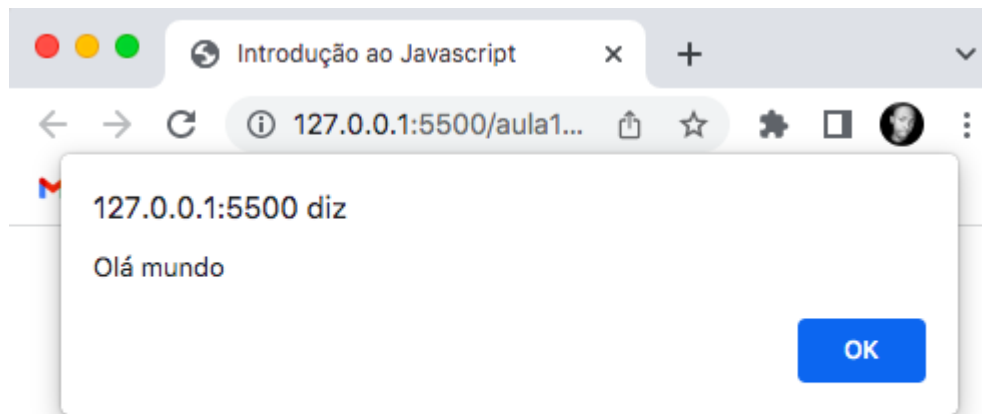
```
JS index.js
1  alert("Olá Mundo");
2
```

No arquivo index.html vamos chamar o index.js:

```
aula1 > index.html > html > body
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Introdução ao Javascript</title>
8  </head>
9  <body>
10
11      <!-- Incorporando o JavaScript externamente -->
12      <script src="./index.js"></script>
13  </body>
14  </html>
```

No exemplo acima nós voltamos a usar a tag **<script>** mas desta vez com o atributo **src** que indica a localização do nosso arquivo.

Resultado:



Variáveis

Variável é um espaço reservado na memória do computador com o objectivo de armazenar informações. Por exemplo variável para armazenar um nome, idade, cidade, ano.

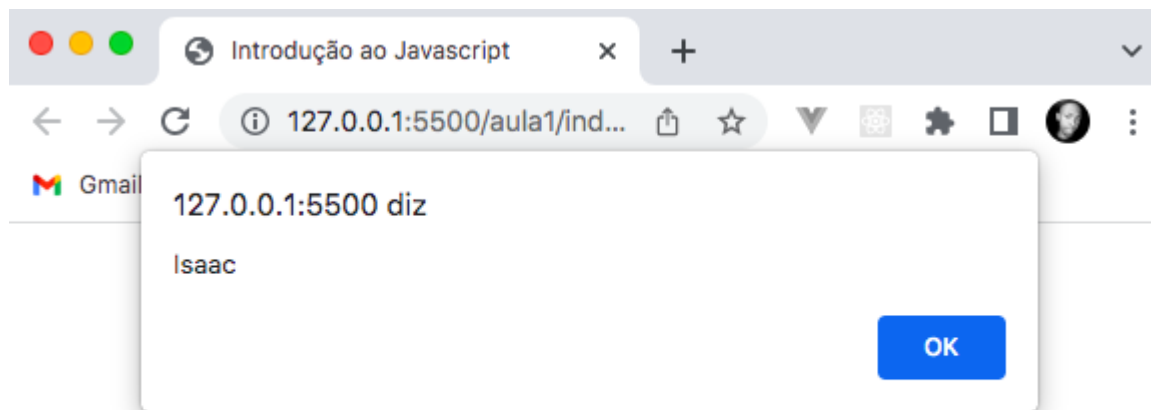
Formas de declarar variáveis:

var

Declara uma variável, opcionalmente, inicializando-a com um valor;

```
<script>
  var nome = "Isaac";
  alert(nome);
</script>
```

O código acima declara uma variável chamada nome e mostra o seu valor na caixa de diálogo como mostra a imagem a seguir:

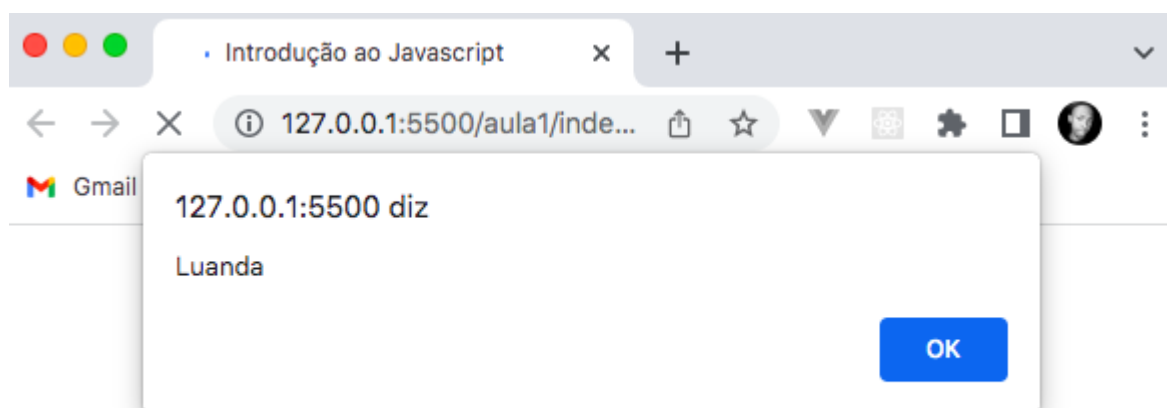


let

Declara uma variável local de escopo do bloco, opcionalmente, inicializando-a com um valor;

```
<!-- Declara uma variavel chamada cidade -->  
<script>  
  let cidade = "Luanda";  
  alert(cidade);  
</script>
```

Resultado:

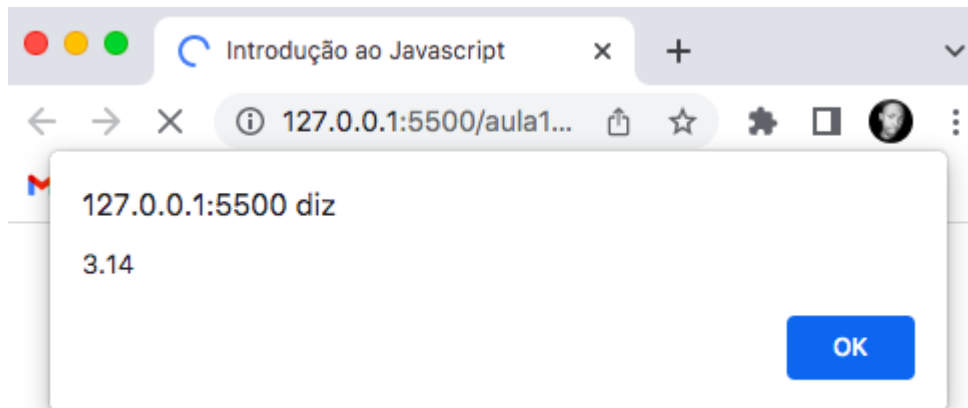


const

Declara uma constante, o que significa que o seu valor não muda após declarada;

```
<!-- Declara uma constante chamada PI -->
<script>
  const PI = 3.14;
  alert(PI);
</script>
```

Resultado:



Tipos de Dados

Os tipos de dados em JavaScript indicam os tipos de informações contidas nas variáveis. Apesar de o JavaScript ser uma linguagem de tipologia fraca mas possui os seus tipos de dados que são:

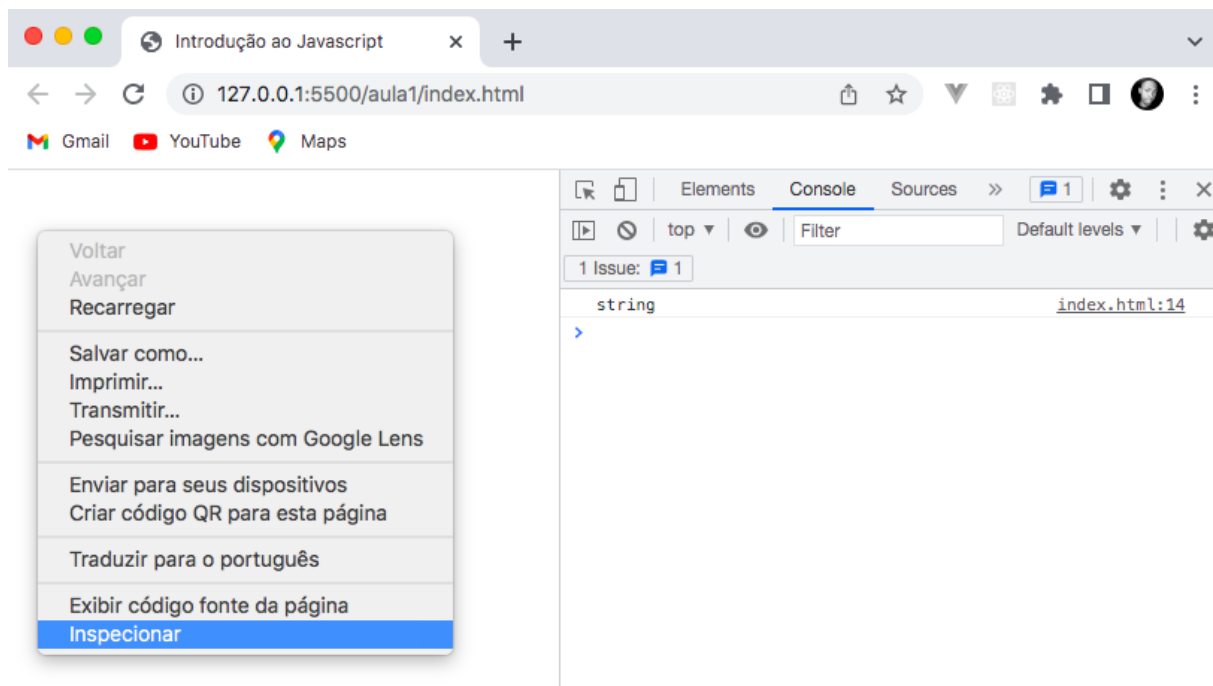
String: este tipo é usado para armazenar texto.

```
<script>
  var nome = "Isaac";
  console.log(typeof(nome));
</script>
```

As variáveis do tipo string podem ser identificadas pelo uso de vírgulas altas ("" ou "). E neste exemplo usamos a função `console.log()` que é uma função que permite imprimir as informações na consola do nosso navegador. Também usamos a função `typeof()` que é uma função que retorna ou apresenta o tipo de variável.

Para vermos o resultado vamos precisar de abrir a consola do nosso navegador. Com botão direito do mouse posicionado em dentro do corpo do nosso site clicamos em `inspecionar>Console` e na consola

vemos o nosso resultado m string como mostra a imagem.

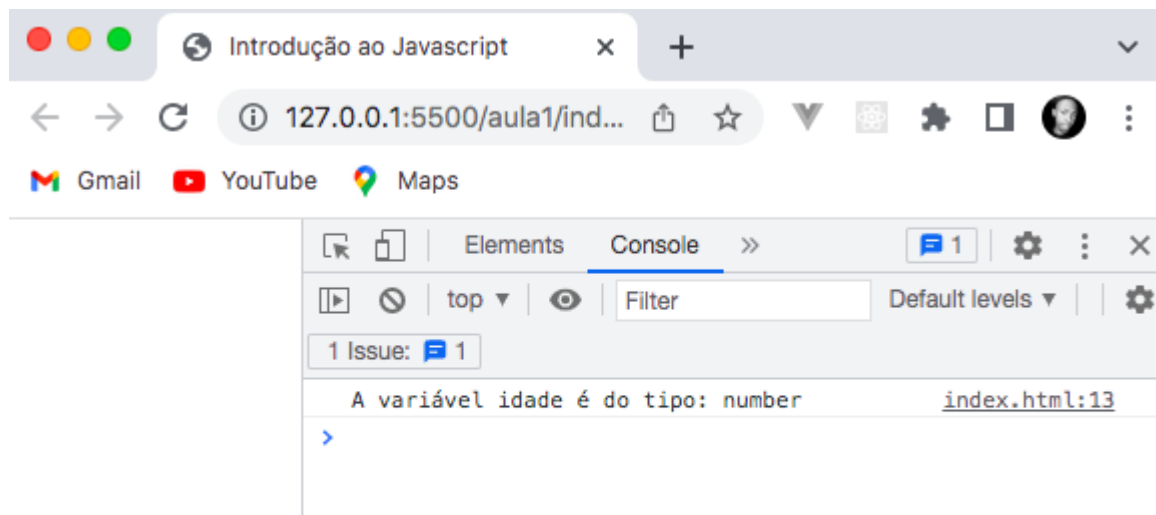


Number: este tipo é usado para armazenar número.

```
<script>
  var idade = 15;
  console.log("A variável idade é do tipo: " + typeof(idade));
</script>
```

Neste exemplo novamente imprimimos o valor de uma variável na consola. Desta vez nós concatenamos (juntamos) o texto com o resultado da função `typeof()` através do uso do sinal de mais (+).

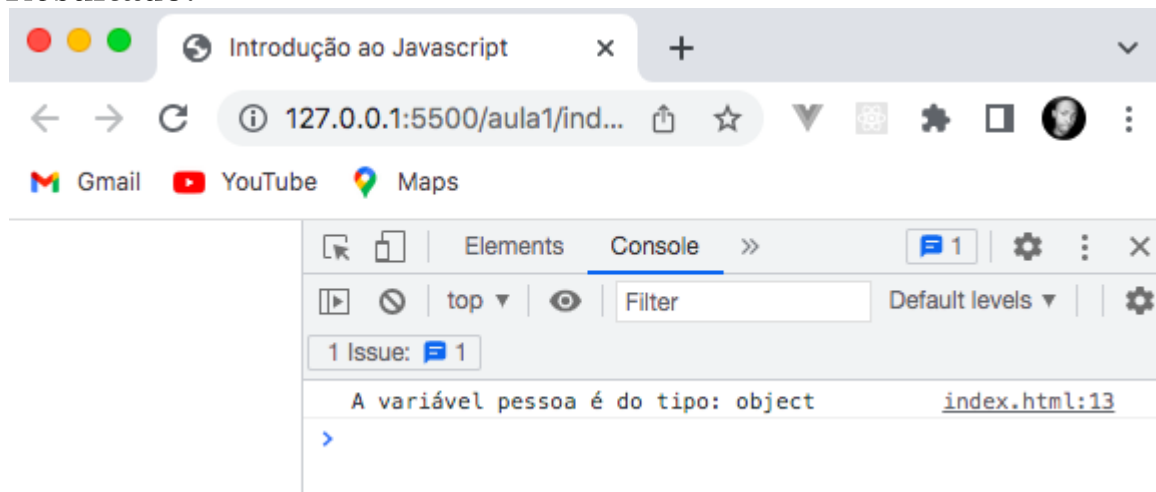
Resultado:



Object: este tipo é usado para representar objectos.

```
<script>
  var pessoa = new Object();
  console.log("A variável pessoa é do tipo: " + typeof(pessoa));
</script>
```

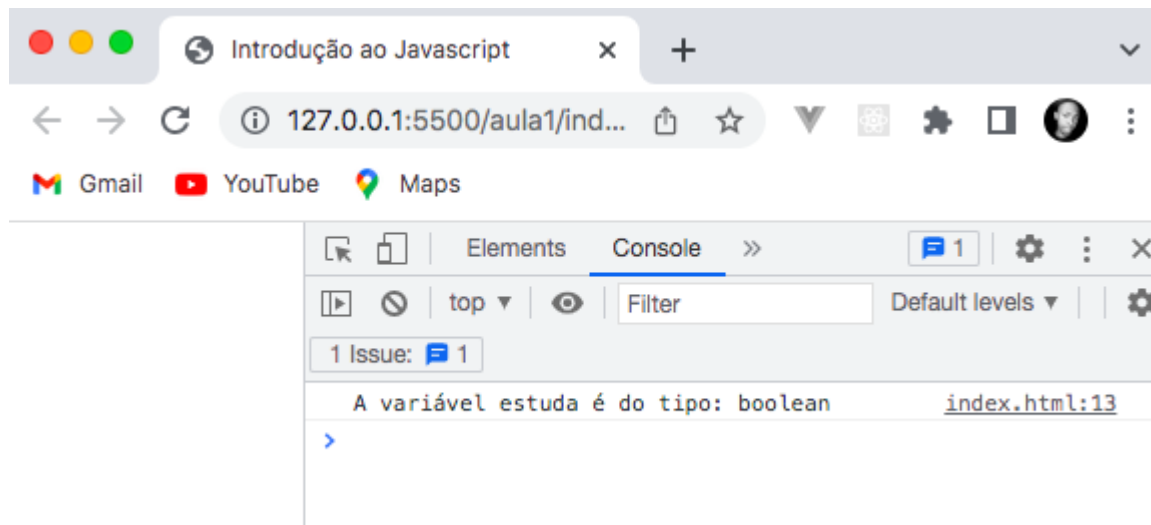
Resultado:



Boolean: este tipo é usado para armazenar valores booleanos (verdadeiro e falso).

```
<script>
  var estuda = true;
  console.log("A variável estuda é do tipo: " + typeof(estuda));
</script>
```


Resultado:



Operadores de Aritméticos

São os operadores que utilizamos para fazer as operações aritméticas básicas:

Operador	Nome	Propósito	Exemplo
<code>+</code>	Adição	Adiciona um número a outro.	<code>6 + 9</code>
<code>-</code>	Subtração	Subtrai o número da direita do número da esquerda.	<code>20 - 15</code>
<code>*</code>	Multiplicação	Multiplica um número pelo outro.	<code>3 * 7</code>
<code>/</code>	Divisão	Divide o número da esquerda pelo número da direita.	<code>10 / 5</code>
<code>%</code>	Restante (<i>Remainder</i> - as vezes chamado de modulo)	Retorna o resto da divisão em números inteiros do número da esquerda pelo número da direita.	<code>8 % 3</code> (retorna 2; como três cabe duas vezes em 8, deixando 2 como resto.)

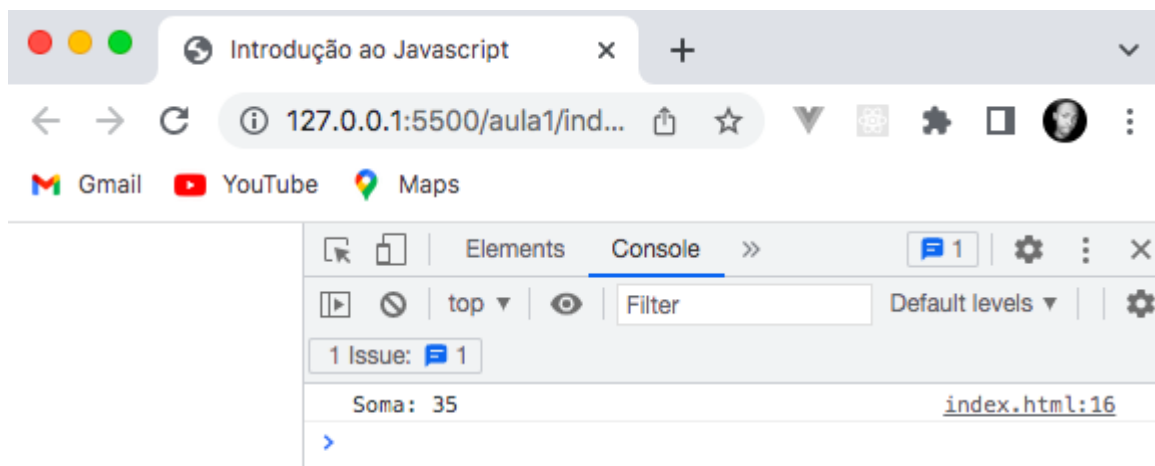
Exemplo:

Soma de Dois números:

```
<script>
  var numero1 = 12;
  var numero2 = 23;

  var soma = numero1 + numero2;
  console.log("Soma: " + soma);
</script>
```

Resultado:



Exercício:

- 1) Usando o JavaScript imprima na consola o resultado da multiplicação de 3 números.
- 2) Imprima a média de 4 números.
- 3) Imprima o resto da divisão de 5 por 2.

Estruturas Condicionais (if...else)

A condicional **if** é uma estrutura condicional que executa a afirmação, dentro do bloco, se determinada condição for verdadeira. Se for falsa, executa as afirmações dentro de **else**.

Sintaxe:

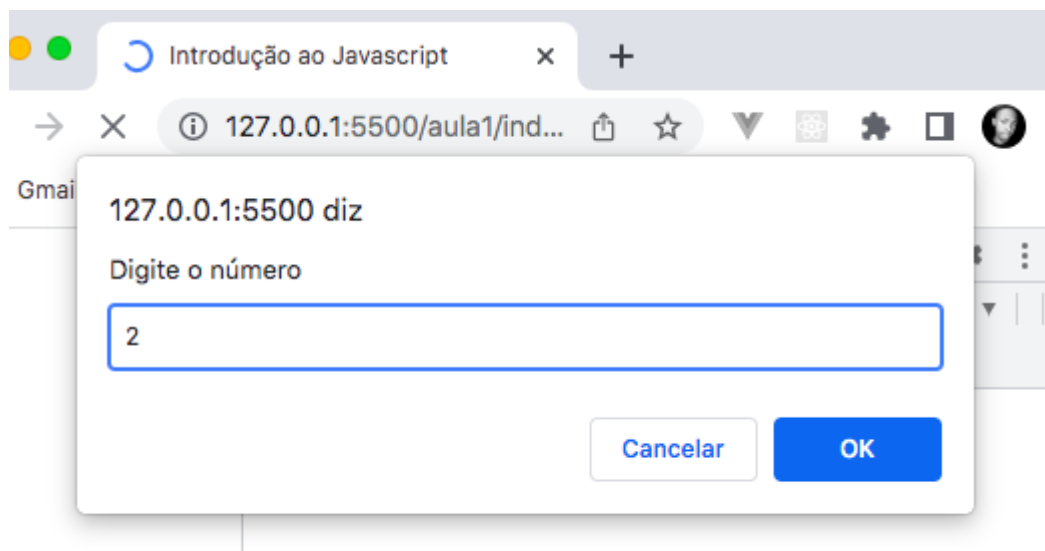
```
if (condição) {
  instrução1
} else {
  instrução2
}
```

Exemplo:

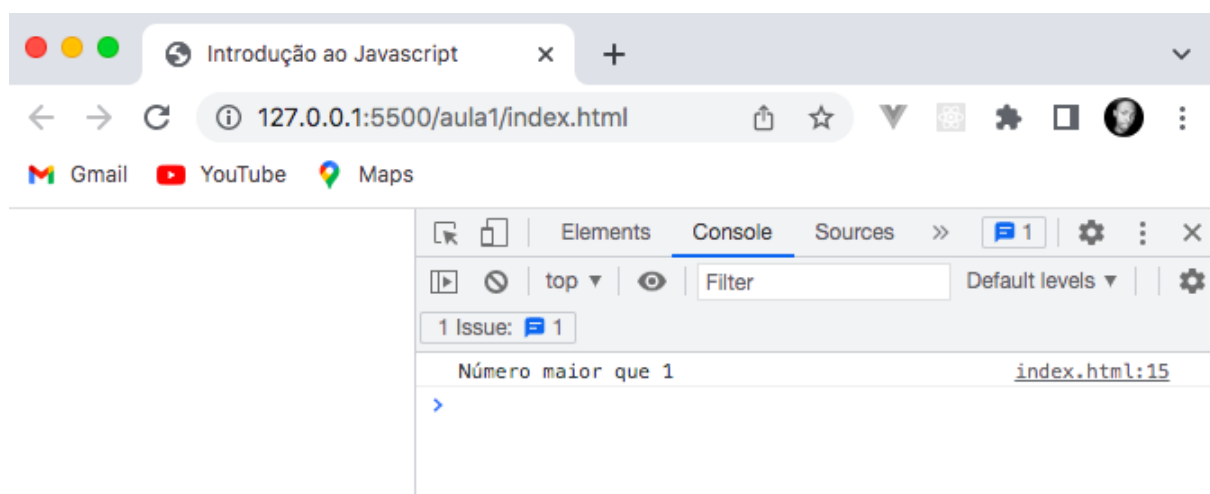
```
<script>
  var numero = prompt('Digite o número');

  if(numero > 1) {
    console.log('Número maior que 1');
  } else {
    console.log("Número não maior que 1")
  }
</script>
```

No exemplo acima usamos a variável usamos a função `prompt()` que permite abrir input do navegador, e com isso a variável `numero` receberá o valor de inserido pelo utilizador como mostra a imagem.



Resultado:



Manipulação do DOM

O HTML DOM pode ser acessado com JavaScript (e com outras linguagens de programação). No DOM, todos os elementos HTML são definidos como objetos. A interface de programação é as propriedades e métodos de cada objeto. Uma propriedade é um valor que você pode obter ou definir (como alterar o conteúdo de um elemento HTML). Um método é uma ação que você pode fazer (como adicionar ou excluir um elemento HTML).

O método **getElementById** A maneira mais comum de acessar um elemento HTML é usar o id do elemento.

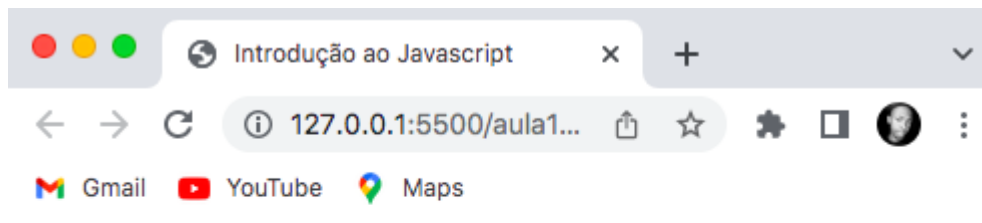
A propriedade **innerHTML** pode ser usada para obter ou alterar qualquer elemento HTML, incluindo <html> e <body>

```
aula1 > index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Introdução ao Javascript</title>
8  </head>
9  <body>
10     <h1>Javascript</h1>
11     <p id="demo"></p>
12
13     <script>
14         document.getElementById("demo").innerHTML = "Olá mundo";
15     </script>
16 </body>
17 </html>
```

O objeto do document representa sua página da web. Se você quiser acessar qualquer elemento em uma página HTML, você sempre começa a acessar o objeto do document.

Neste exemplo acessamos o elemento <p> que possui um id demo e adicionamos o texto “Olá mundo”.

Resultado:



Javascript

Olá mundo

Função

Funções são blocos de construção fundamentais em JavaScript. Uma função é um procedimento de JavaScript - um conjunto de instruções que executa uma tarefa ou calcula um valor. Para usar uma função, você deve defini-la em algum lugar no escopo do qual você quiser chamá-la.

sintaxe

function identificador (parametro1....parametron){

Bloco de intruções

Retorno;

}

Exemplo:

```
<script>
  function soma(n1, n2) {
    return n1 + n2;
  }

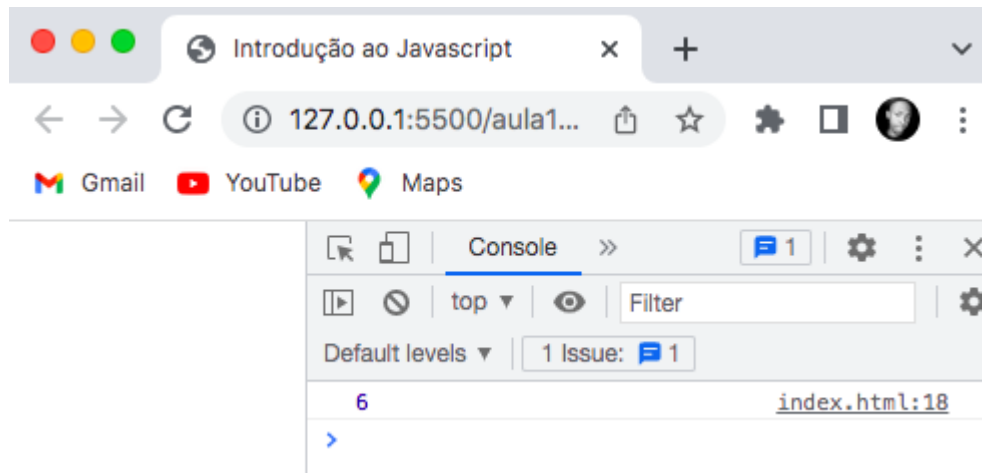
  var resultado = soma(2, 4);

  console.log(resultado);
</script>
```

Esta é uma função que soma 2 números e o resultado é armazenado na variável resultado e printado através da função console.log que o

resultado será 6 como se vê na imagem abaixo.

Resultado:



Alguns Eventos Em Javascript

- **onclick**: clica com o mouse
- **ondblclick**: clica duas vezes com o mouse
- **onmousemove**: mexe o mouse
- **onmousedown**: aperta o botão do mouse
- **onmouseup**: solta o botão do mouse (útil com os dois acima para gerenciar drag'n'drop)
- **onkeypress**: ao pressionar e soltar uma tecla
- **onkeydown**: ao pressionar uma tecla.
- **onkeyup**: ao soltar uma tecla. Mesmo acima.
- **onfocus**: quando um elemento ganha foco
- **onchange**: quando um input, select ou textarea tem seu valor alterado
- **onsubmit**: disparado antes de submeter o formulário. Útil para

realizar validações

Exemplo:

```
<h1 id="title">Javascript</h1>

<button id="btn">Clica</button>

<script>
  function tornaVermelho(n1, n2) {
    document.getElementById("title").style.color = 'red'
  }

  document.getElementById('btn').onclick = tornaVermelho;
</script>
```

Neste exemplo quando é clicado em button com o id btn é executada a função tornaVermelho(), esta função torna o texto de h1 vermelho, isto acontece por causa da propriedade style que permite aplicar CSS nos elementos selecionados.

Links Uteis:

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

https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/What_is_JavaScript