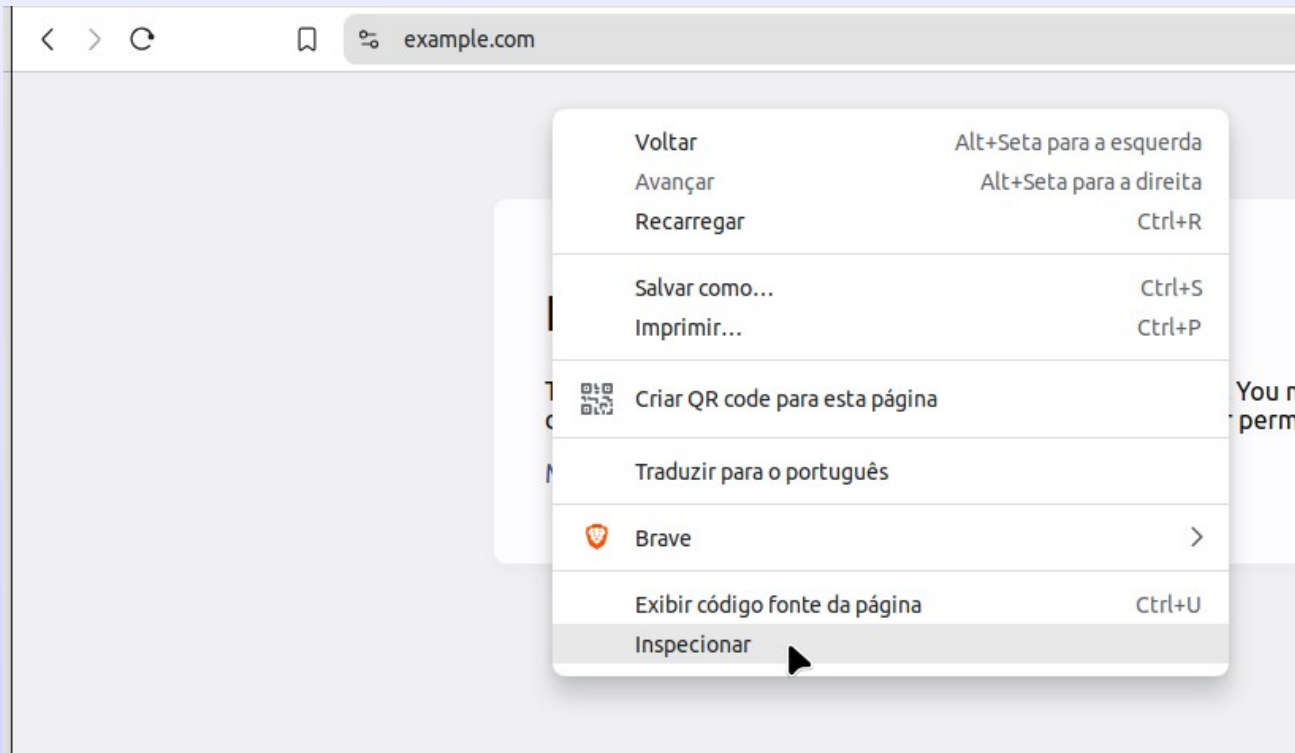




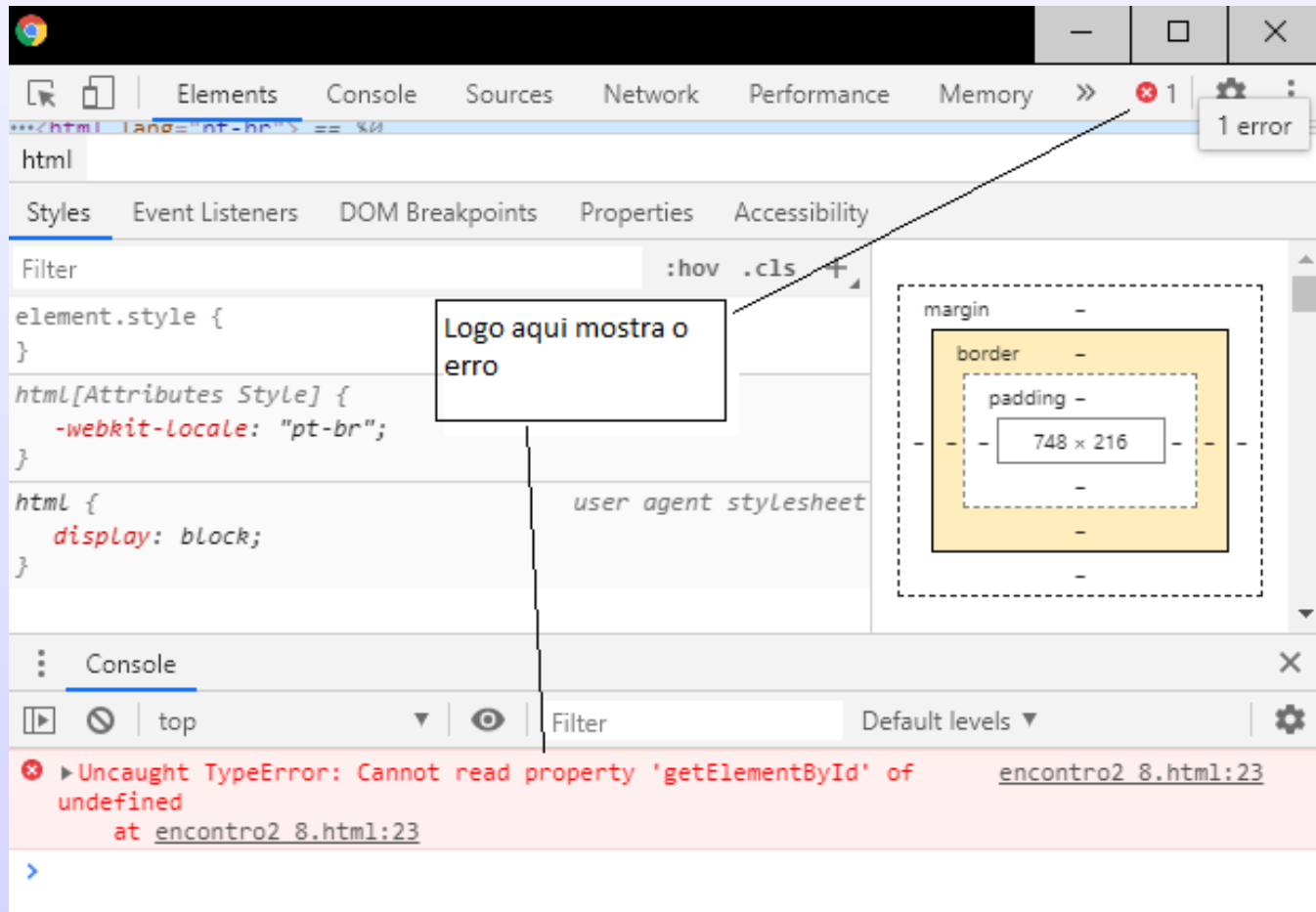
JavaScript

Como descobrir erros no JavaScript

Para verificar o que acontece em uma página do navegador e identificar um erro no JavaScript, clique com o botão direito do mouse sobre a área de visualização e selecione inspecionar no menu suspenso como ilustrado a seguir:



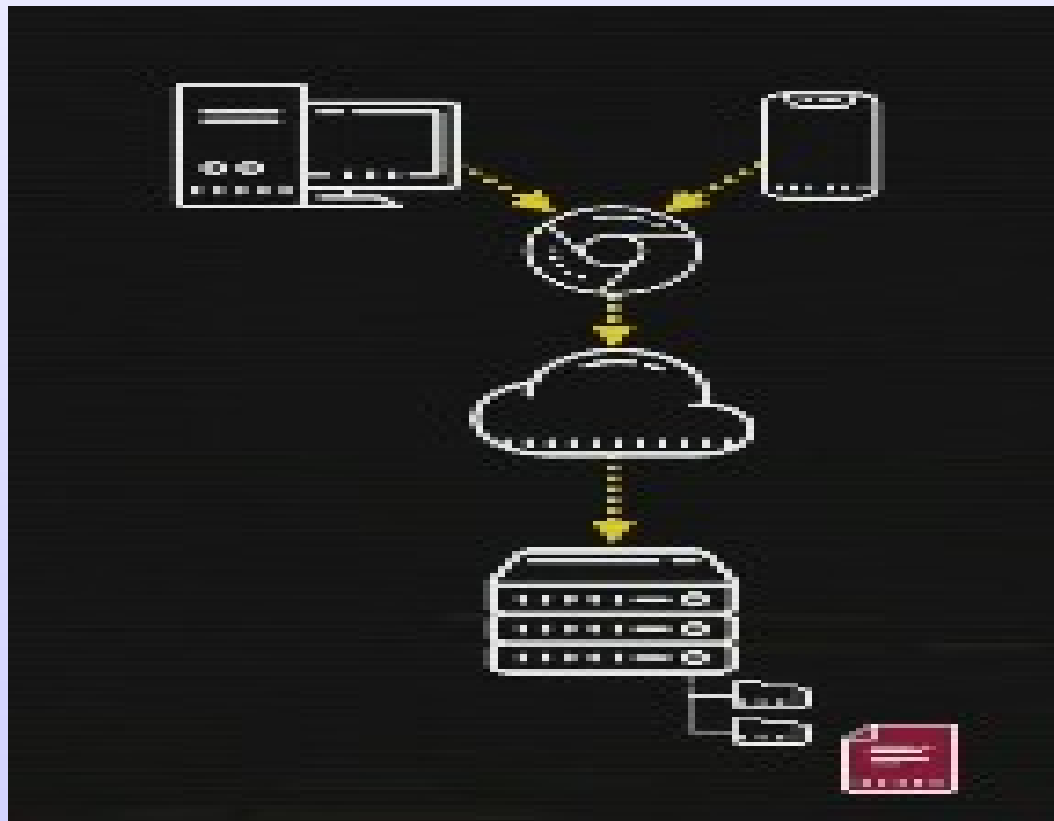
Como descobrir erros no JavaScript



Client x Server

Conceitos

Celulares e PCs (que são clientes) requisitam páginas de um servidor web, para visualizar em seus dispositivos.

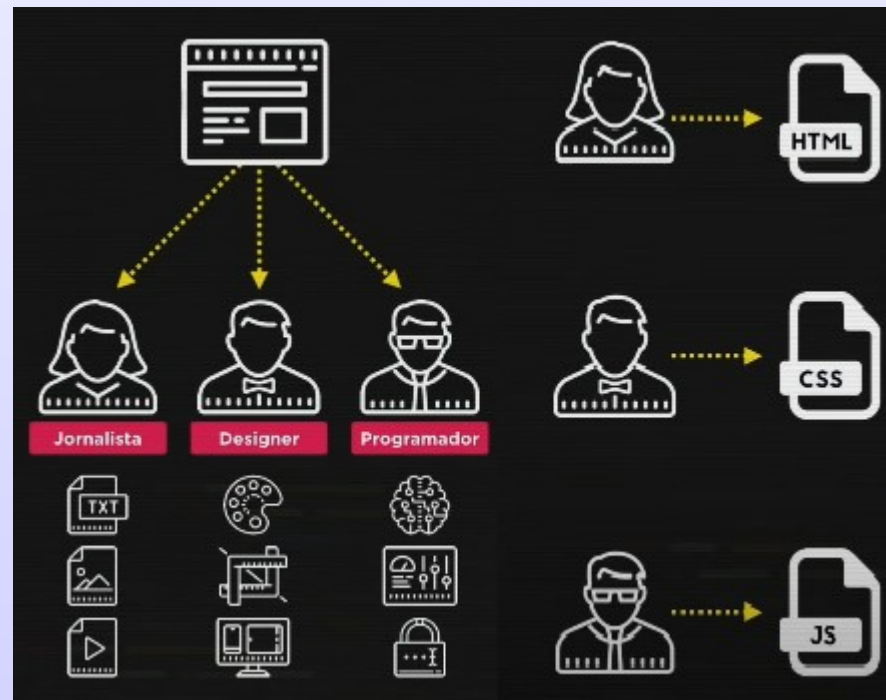


Website

Conceitos

No website existem três tecnologias focadas no cliente que são:

- Html - Conteúdo, que são imagens, arquivos, vídeos etc.
- CSS - Design que organiza os conteúdos de uma forma atraente.
- Javascript - Programação que trata da funcionalidade



Website

Conceitos

No website existem três tecnologias focadas no cliente que são:

- Html - Conteúdo, que são imagens, arquivos, vídeos etc.
- CSS - Design que organiza os conteúdos de uma forma atraente.
- Javascript - Programação que trata da funcionalidade



Website

O que é node.js

O Node.js pode ser definido como um **ambiente de execução Javascript *server-side***.

Isso significa que com o Node.js é possível criar aplicações Javascript para rodar como uma aplicação *standalone* em uma máquina, não dependendo de um browser para a execução, como estamos acostumados.



Website

Baixar NodeJS

Entrar no site e baixar no link abaixo: <https://nodejs.org/en/>

Depois de instalado entrar no terminal dele, para sair dele digitar .exit



Website

Comentarios em JavaScript

// é utilizado para uma única linha

Enquanto

/*

Mais de uma linha de código
é comentada assim...

*/

Código 1a

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <meta charset="utf-8"/>
5     <title>HTML5 Estrutura básica</title>
6     <style>
7       body {
8         background-color: rgb(105,
9         color: white;
10        font: normal 20pt Arial;
11      }
12    </style>
13  </head>
14  <body>
15    <h1>Olá, mundo!</h1>
16    <p>Agora não há mais perigo para continuar...</p>
17
18    <script>
19      //Exibir alerta na pagina
20      window.alert ('Minha primeira mensagem')
21      // Botão de sim ou não
22      window.confirm ('Está gostando de JS?')
23      // Pergunta
24      window.prompt ('Qual é o seu nome?')
25    </script>
26  </body>
27 </html>
```

← → ↺ G código1.html

Olá, mundo!

Agora não há mais perigo para continuar...

Ao executar este código no navegador, primeiro serão mostradas as mensagens em JavaScript e a seguir aparecerá o conteúdo principal.

Variáveis (identificadores)

A declaração de variáveis em JavaScript pode ser feita utilizando: **var** ou **let** enquanto para constantes, utilizamos **const**

Algumas regras precisam ser seguidas para declaração dos identificadores:

- Pode começar com letra, \$ ou underline _
- Pode conter letras e números
- NÃO pode começar com número
- NÃO pode conter espaços
- NÃO pode ser uma palavra reservada*

*confira as palavras reservadas:

https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Lexical_grammar#palavras-chave

Variáveis

ATENÇÃO! Maiúsculas e minúsculas fazem a diferença

Escolha nomes coerentes para seus identificadores. Isto facilita muito a leitura do código por você e revisores



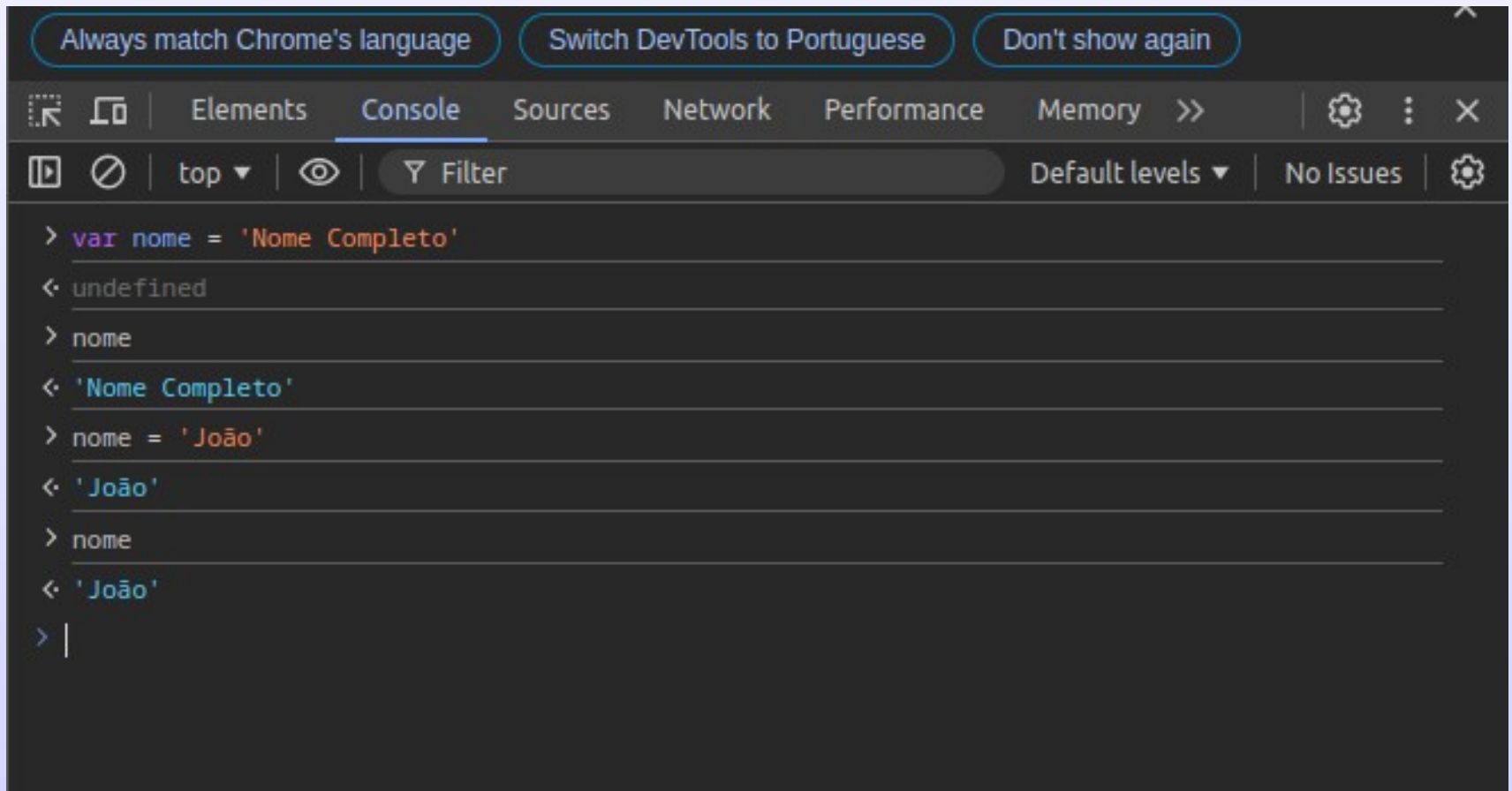
Variáveis

Tipos de dados:

- number, infinity, NaN (Not a Number);
- string
- boolean
- null
- undefined
- object, array
- function

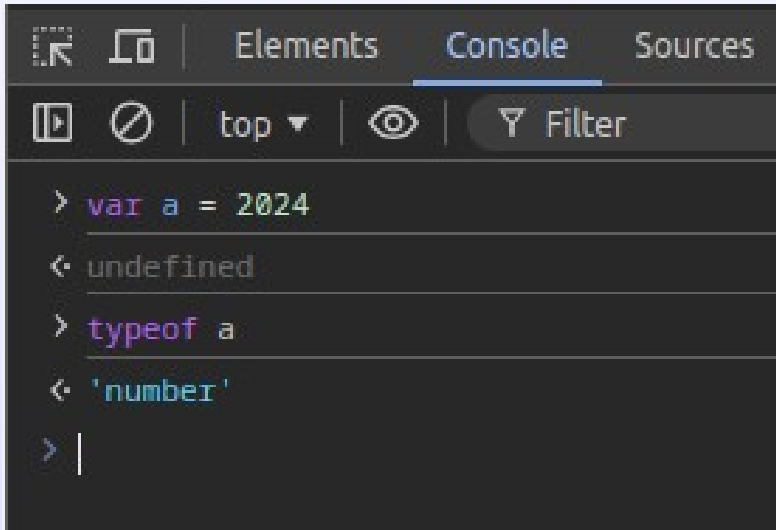
Praticando

Para testar e praticar o funcionamento do JavaScript enquanto não tiver o ambiente de estudo/trabalho disponível, basta abrir um navegador, inspecionar uma página em branco e utilizar os recursos disponíveis, como ilustrado a seguir:



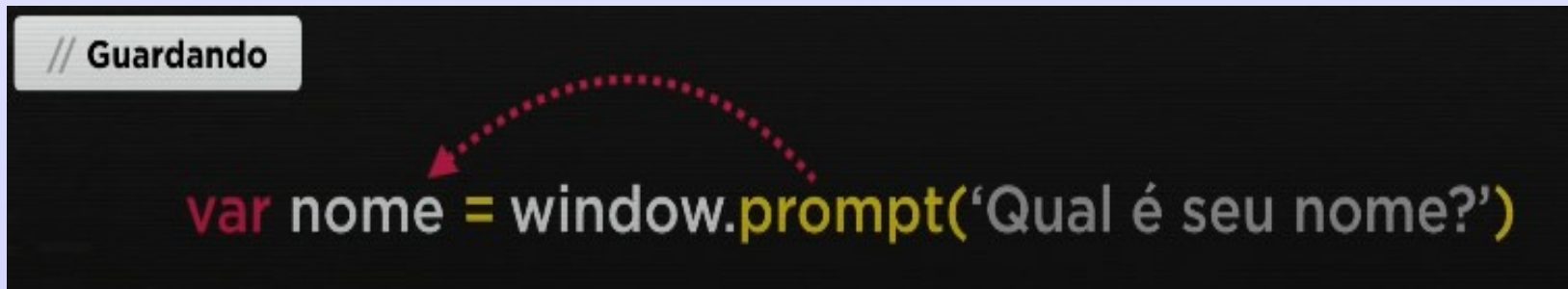
Praticando

Para verificar o tipo de um identificador, é possível consultar desta forma:



```
> var a = 2024
< undefined
> typeof a
< 'number'
> |
```

Enquanto o armazenamento de um valor pode ser feito assim:



```
// Guardando
var nome = window.prompt('Qual é seu nome?')
```

Código 1b

REV0904a

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="utf-8"/>
  <title>HTML5 - Estrutura básica</title>
  <style>
    body{
      background-color: rgb(105, 105, 173);
      color: white;
      font: normal 20pt Arial;
    }
  </style>
</head>
<body>

  <h1>Olá, mundo</h1>
  <p>Já me livrei da maldição</p>

  <script>
    <!-- Pergunta -->
    var nome = window.prompt('Qual é o seu nome?')
    window.alert('É um grande prazer em te conhecer,'+nome+'!')//concatenação
  </script>
</body>
</html>
```


JavaScript

Conversão de string para numero.

```
Number.parseInt(n)
```

```
Number.parseFloat(n)
```

```
Number(n)
```

Código 1c

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="utf-8"/>
  <title>HTML5 - Estrutura básica</title>
  <style>
    body{
      background-color: rgb(105, 105, 173);
      color: white;
      font: normal 20pt Arial;
    }
  </style>
</head>
<body>
  <h1>Olá, mundo</h1>
  <p>Já me livreii da maldiçãoo</p>

  <script>
    //Transforma string em numero inteiro
    var n1 = Number.parseInt(window.prompt('digite um numero:'))
    var n2 = Number.parseInt(window.prompt('digite outro numero:'))
    var s = n1 + n2
    window.alert('A soma dos valores é '+ s)//concatenação
    //(number + number)para adição
    //(string + string)para concatenação

  </script>
</body>
</html>
```

Código 1d

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="utf-8"/>
  <title>HTML5 - Estrutura básica</title>
  <style>
    body{
      background-color: rgb(105, 105, 173);
      color: white;
      font: normal 20pt Arial;
    }
  </style>
</head>
<body>
  <h1>Olá, mundo</h1>
  <p>Já me livrei da maldição</p>

  <script>
    //Transforma string em numero inteiro
    var n1 = Number(window.prompt('digite um numero:'))
    var n2 = Number(window.prompt('digite outro numero:'))
    var s = n1 + n2
    window.alert('A soma dos valores é '+ s)//concatenação
    //(number + number)para adição
    //(string + string)para concatenação
  </script>
</body>
</html>
```

JavaScript

Transformar de numero para string.



```
String(n)
```

```
n.toString()
```

Código 1e

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="utf-8"/>
  <title>HTML5 - Estrutura básica</title>
  <style>
    body{
      background-color: rgb(105, 105, 173);
      color: white;
      font: normal 20pt Arial;
    }
  </style>
</head>
<body>
  <h1>Olá, mundo</h1>
  <p>Já me livreii da maldição</p>

  <script>
    //Transforma string em numero inteiro
    var n1 = Number(window.prompt('digite um numero:'))
    var n2 = Number(window.prompt('digite outro numero:'))
    var s = String(n1) + String(n2)
    //Transformanda numero em strings
    window.alert('A soma dos valores é '+ s )//concatenação
    //(number + number)para adição
    //(string + string)para concatenação
  </script>
</body>
</html>
```

Variaveis em Js

```
// Formatando Strings
```

```
var s = 'JavaScript'
```

```
'Eu estou aprendendo s'
```

No exemplo acima ele vai escrever 'estou aprendendo s' e não 'estou aprendendo JavaScript' vamos ver no próximo slide com o nodejs.

Variaveis em Js

Exemplo JavaScript



Node.js

```
Welcome to Node.js v12.18.1.  
Type ".help" for more information.  
> var s = 'JavaScript'  
undefined  
> s  
'JavaScript'  
> 'Eu estou estudando s'  
'Eu estou estudando s'  
> 'Eu estou estudando JavaScript' + s  
'Eu estou estudando JavaScriptJavaScript'  
>
```

Variaveis em Js

Exemplo JavaScript

```
// Formatando Strings
```

```
var s = 'JavaScript'
```

```
'Eu estou aprendendo s' // não faz interpolação
```

```
'Eu estou aprendendo' + s // usa concatenação
```

```
`Eu estou aprendendo ${s}` // usa template string
```


JavaScript

Exemplo de concatenação

 Node.js

```
Welcome to Node.js v12.18.1.  
Type ".help" for more information.  
> nome = 'Eder'  
'Eder'  
> idade = 37  
37  
> nota = 7.8  
7.8  
> `O aluno ${nome} com ${idade} anos tirou a nota ${nota}`  
'O aluno Eder com 37 anos tirou a nota 7.8'  
>
```

Formatando Strings

```
var s = 'JavaScript'
```

```
s.length
```

```
// quantos caracteres a string tem
```

```
s.toUpperCase()
```

```
// tudo para 'MAIÚSCULAS'
```

```
s.toLowerCase()
```

```
// tudo para 'minúsculas'
```

Código 1f

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="utf-8"/>
  <title>HTML5 - Estrutura básica</title>
  <style>
    body{
      background-color: rgb(105, 105, 173);
      color: white;
      font: normal 20pt Arial;
    }
  </style>
</head>
<body>
  <script>
    var nome = window.prompt('Qual é o seu nome?')
    //O comando document.write escreve no corpo do html
    document.write(`Seu nome tem ${nome.length} letras`)
  </script>
</body>
</html>
```

Código 1g

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="utf-8"/>
  <title>HTML5 - Estrutura básica</title>
  <style>
    body{
      background-color: rgb(105, 105, 173);
      color: white;
      font: normal 20pt Arial;
    }
  </style>
</head>
<body>
  <script>
    var nome = window.prompt('Qual é o seu nome?')
    //O comando document.writeln escreve no corpo do html
    document.write(`Ola,<strong> ${nome}</strong>! Seu nome tem ${nome.length} letras,<br/>`)
    document.write(`Seu nome em maiusculas ${nome.toUpperCase()}<br/>`)
    document.write(`Seu nome em minunscula ${nome.toLowerCase()} `)
  </script>
</body>
</html>
```

Formatando números

O `toFixed(numero)` ele determina quantos caracteres vai ter depois do ponto
`toLocaleString()` determina string padrão no mundo

```
// Formatando Números
```

```
var n1 = 1543.5
```

```
n1
```

```
n1.toFixed(2)
```

```
n1.toLocaleString('pt-BR', {style: 'currency', currency: 'BRL'})
```

Formatando números



```
Welcome to Node.js v12.18.1.  
Type ".help" for more information.  
> var n1 = 1545.5  
undefined  
> n1  
1545.5  
> n1.toFixed(2)  
'1545.50'  
> n1.toFixed(2).replace('.', ',')  
'1545,50'  
> n1.toLocaleString('pt-Br',{style:'currency',currency:'BRL'})  
'R$ 1,545.50'  
> n1.toLocaleString('pt-Br',{style:'currency',currency:'USD'})  
'US$ 1,545.50'  
> n1.toLocaleString('pt-Br',{style:'currency',currency:'EUR'})  
'€ 1,545.50'
```