

MATERIAL EXTRA

O que é Javascript



O QUE É JAVASCRIPT

Javascript (JS), é uma das linguagens de programação mais utilizadas do mundo.

Foi criada por Brendan Eich, um programador norte-americano extremamente visionário, que teve esta ideia no longínquo ano de 1995, muito antes da internet se popularizar massivamente em todo o mundo e sem imaginarmos que a linguagem se tornaria tão popular entre os desenvolvedores.



[Brendan Eich](#)

Com ela, você pode criar páginas web dinâmicas, animações, mapas interativos, gráficos em três dimensões, aplicativos para dispositivos móveis e games para plataformas portáteis.

No desenvolvimento de páginas web controla o comportamento dos elementos de uma página, executando funções dinâmicas dos conteúdos, animações, aplicativos e interações em geral.

Vantagens de Usar JavaScript

Saber como programar em JavaScript e usar a linguagem de programação em um site traz uma série de vantagens ao desenvolvedor e ao usuário que acessa uma página dinâmica.

- Alta compatibilidade com plataformas, sistemas e navegadores web.
- É mais leve e rápida que outras linguagens de programação.
- Faz com que as páginas na internet sejam mais dinâmicas e interativas, características essenciais do UX.
- Os navegadores interpretam a linguagem por conta própria, tirando a necessidade de usar um compilador.
- Erros de programação são mais fáceis de encontrar e de corrigir.

O diferencial dela está no uso do **Engine V8**, um motor gráfico de código aberto conhecido por ter um alto desempenho nos sistemas e aplicações em que estiver instalado. O navegador Google Chrome, por exemplo, tem ele por padrão.



Foi a partir do Engine V8 que surgiu o Node.js, uma plataforma completa de desenvolvimento que permite aos programadores explorarem códigos e sistemas para baseados em JavaScript. Inclusive, pelo lado do servidor, e não apenas do usuário (cliente).

ECMAScript

Antes que o Javascript se tornasse popular, para que a linguagem evoluísse obedecendo a determinados padrões e normativas, os criadores do Javascript se associaram ao ECMA (European Computer Manufacturers Association) em 1996.

Como o nome Javascript já havia sido patenteado pela Sun Microsystems (atual Oracle), optou-se por se definir um novo nome à linguagem utilizando a junção das palavras ECMA e Javascript, surgindo então o ECMAScript.

Mas, como o nome Javascript ficou popular na comunidade, a linguagem é chamada por este nome até hoje, sendo o ECMAScript referenciado apenas para se determinar a versão da linguagem?

Atualmente, padrões e normativas referentes à linguagem são mantidas pela ECMA-262, grupo criado na ECMA para a padronização do Javascript e conta com participação de grandes empresas de tecnologia como Microsoft, Google, dentre outras.

Versões da linguagem

As versões ECMAScript foram abreviadas para ES1, ES2, ES3, ES5 e ES6.

Desde 2016, as novas versões são nomeadas por ano (ECMAScript 2016/2017/2018)

Versão	Nome oficial	Descrição
ES1	ECMAScript 1 (1997)	Primeira edição
ES2	ECMAScript 2 (1998)	Alterações editoriais
ES3	ECMAScript 3 (1999)	Esta versão possui melhorias importantes para a comunidade, permitindo implementações de laços de repetição do-while, tratamento de exceções, dentre outros recursos.
ES4	ECMAScript 4 (2008)	Concluída em 2008, esta versão teve seu desenvolvimento baseado em ML (Meta Language), uma linguagem de programação funcional mais utilizada em ambientes de pesquisa acadêmica. Devido à grande quantidade de mudanças em relação ao ECMAScript 3, o que tornaria esta versão totalmente disruptiva em relação às anteriores, a mesma foi abandonada pelo comitê técnico da ECMA-262, optando-se por dar continuidade evolutiva à versão anterior.
ES5	ECMAScript 5 (2009)	Após o impasse gerado com a versão 4, foi lançada em 2012 oficialmente a versão 5 do ECMAscript, possuindo recursos valiosos como o suporte a JSON, métodos mais avançados de manipulação de arrays, getters e setters, dentre outros.

ES6	ECMAScript 6 (2015)	Atribui recursos avançados à linguagem como reflection, collections, binary data, dentre outros.
ES2016	ECMAScript 2016 (2016)	Adicionado operador exponencial (**) Adicionado Array.includes()
ES2017	ECMAScript 2017 (2017)	Adicionado: <ul style="list-style-type: none"> • string padding • Object.entries() • Object.values() • async functions • shared memory
ES2018	ECMAScript 2018 (2018)	Adicionado: <ul style="list-style-type: none"> • rest / spread • asynchronous iteration • Promise.finally()
ES2019	ECMAScript 2019 (2019)	Adicionadas melhorias na manipulação e ordenação de arrays.
ES2020	ECMAScript 2020 (2020)	Adicionado o tipo BigInt primitivo e suporte a nullish coalescing operator.
ES2021	ECMAScript 2021 (2021)	Adicionado replaceAll para strings e Promise.any.

ES.Next

ES.Next é um nome dinâmico que se refere a qualquer que seja a próxima versão no momento da escrita. Os recursos ES.Next incluem propostas finalizadas (também conhecidas como "propostas de estágio 4") conforme listadas nas propostas finalizadas que não fazem parte de uma especificação ratificada.

O comitê de idiomas segue um modelo de "especificação viva", portanto, essas mudanças fazem parte do padrão e a ratificação é uma formalidade.

"Hello, World" no Javascript

JavaScript é uma das tecnologias da web modernas mais populares! À medida que suas habilidades em JavaScript aumentam, seus sites entrarão em uma nova dimensão de poder e criatividade.

No entanto, familiarizar-se com JavaScript é mais desafiador do que familiarizar-se com HTML e CSS. Você pode ter que começar pequeno e progredir gradualmente. Para começar, vamos examinar duas formas de como adicionar JavaScript à sua página para criar um exemplo "Hello world!". (Hello world! é [o padrão para exemplos de programação introdutória](#).)

Script interno

1. Crie um documento index.html e preencha com sua estrutura básica

```
<!DOCTYPE html>
<html lang="pt-BR">

<head>
```



```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width,
initial-scale=1.0">
<title>Hello World</title>
</head>

<body>

</body>

</html>
```

2. Em seu documento HTML, insira o código em uma nova linha, logo antes da tag de fechamento `</body>`.

```
<body>

  <script>

  </script>
</body>
```

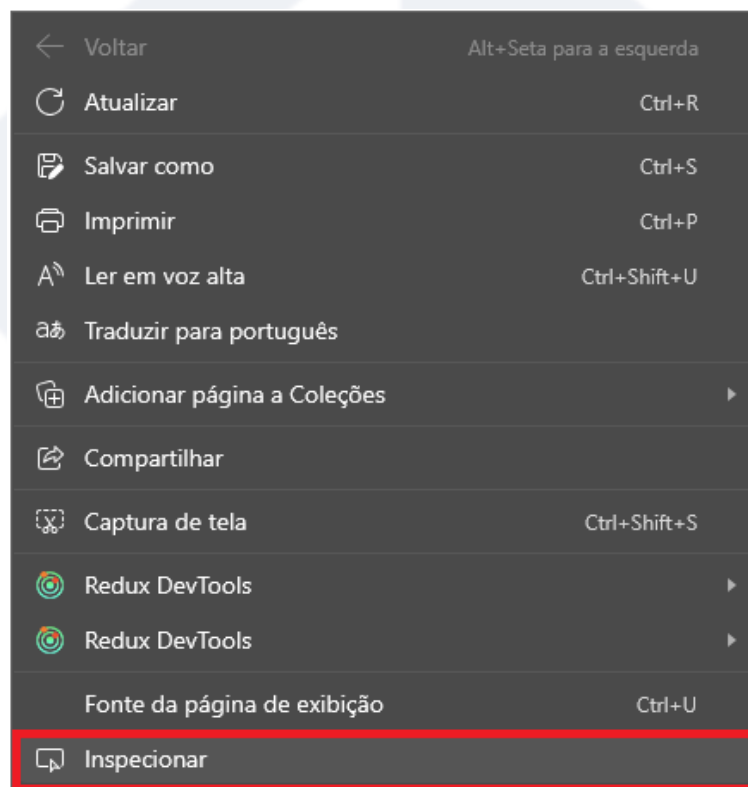
3. Isso faz o mesmo trabalho que o elemento `<link>` para CSS. Ele aplica o JavaScript à página, para que possa executar instruções dentro do seu documento HTML.
4. Adicione este código entre a abertura e fechamento da tag `<script>`.

```
<script>

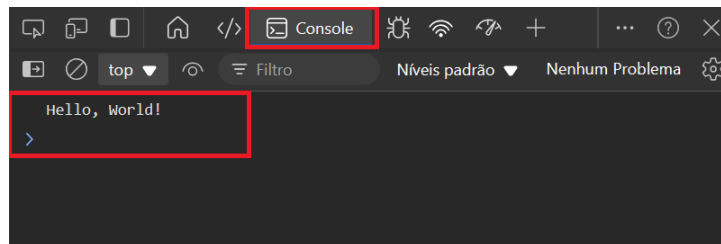
  console.log("Hello, World!");

</script>
```


5. Certifique-se de que o documento HTML foi salvo e, em seguida, abra o documento HTML utilizando o navegador. Recomendamos a utilização do navegador [Google Chrome](#), mas você pode utilizar outro de sua preferência.
6. Abra a ferramenta de desenvolvedor (DevTools), clicando com o botão direito do mouse sobre qualquer parte da página aberta e clicando em **inspecionar**.

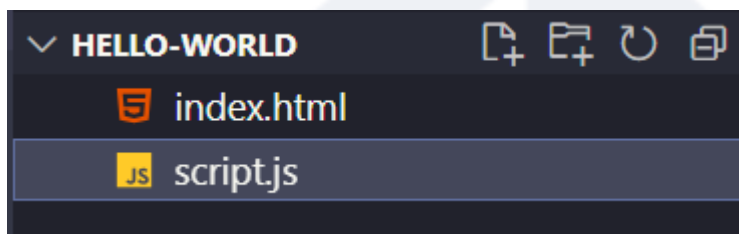


7. Acesse o recurso de console do DevTools e perceba a sua mensagem sendo mostrada.



Script externo

1. Junto do documento index.html, crie um novo documento chamado script.js



2. Realize as modificações necessárias no documento HTML para que ele se pareça com o código abaixo.

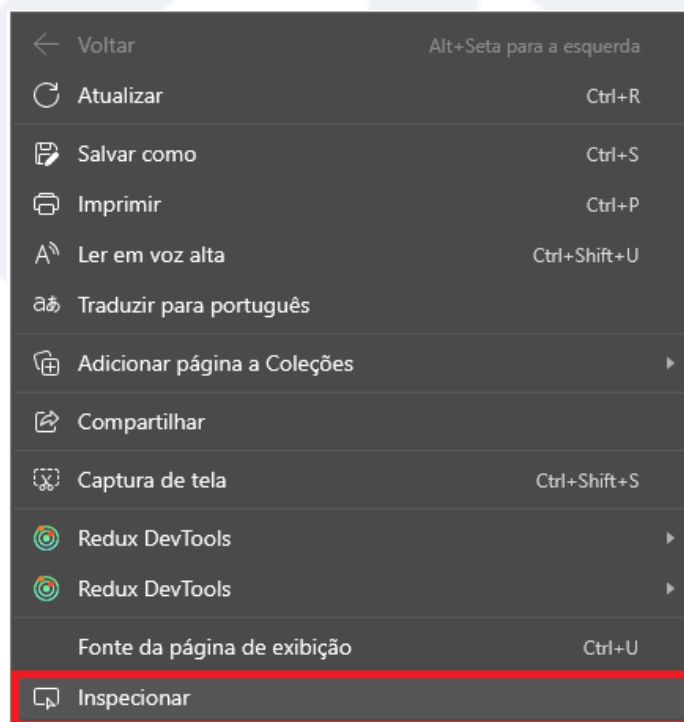
```
<!DOCTYPE html>
<html lang="pt-BR">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Hello World</title>
</head>
<body>
  <script src="./script.js"></script>
</body>
</html>
```

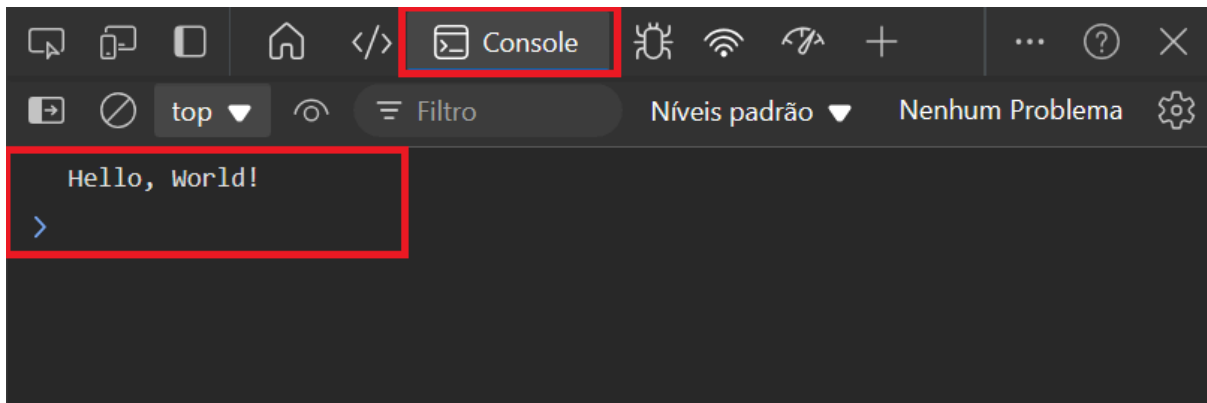
3. Adicione esse código ao arquivo script.js

```
console.log("Hello, World!");
```

4. Certifique-se de que o documento HTML e JavaScript foram salvos e, em seguida, abra o documento HTML utilizando o navegador.
5. Abra a ferramenta de desenvolvedor (DevTools), clicando com o botão direito do mouse sobre qualquer parte da página aberta e clicando em **inspecionar**



6. Acesse o recurso de console do DevTools e perceba a sua mensagem sendo mostrada



REFERÊNCIAS

1. [ECMA Internacional](#)
2. [Versões ECMAScript](#)
3. [Noções básicas de Javascript](#)

BONS ESTUDOS

