

JavaScript

Prof. Daniel Di Domenico



Introdução

Revisão...

- **O que já sabemos:**

- HTML para criar o conteúdo das páginas Web
- CSS para adicionar estilos às páginas Web
 - Cores, tamanhos, posicionamento, bordas, efeitos...
 - Seletores: elemento, ID, classe, pseudo-classe e pseudo-elemento



O que é JavaScript?

- JavaScript (JS):
 - **Linguagem de programação**
 - Linguagem de script e interpretada
 - Também conhecida como linguagem de script para páginas Web
 - É suportada por todos os navegadores modernos (Chrome, Firefox, Safari...)
 - Comandos são executados pelo navegador
 - **ATENÇÃO:** não se deve confundir com Java
 - Ambas são linguagens de programação, mas possuem sintaxe, semânticas e usos muito diferentes

Por que utilizar JavaScript?

- JS é uma linguagem largamente utilizada no mundo
 - Forma a **tríade do desenvolvimento Web** junto com HTML e CSS
 - Padrão W3C (World Wide Web Consortium)
 - Tutorial oficial do JS: <https://www.w3schools.com/js/>
- Permite a implementação de **elementos dinâmicos** em páginas Web
 - Utilizando apenas HTML e CSS, as páginas são estáticas
 - JS possibilita manipular, calcular e validar informações
 - Atualização e aperfeiçoamento dos códigos HTML e CSS



Funcionalidades do JS

- **Detectar eventos** (interações do usuário) que ocorrem em uma página Web, como:
 - Clique em um elemento (botão ou link)
 - Digitação em um campo texto
 - Seleção de uma opção de seleção
 - Foco em um elemento
- **Responder aos eventos** através de ações, como:
 - Exibir uma mensagem
 - Mudar o conteúdo de um elemento
 - Alterar o estilo dos elementos
 - Validar dados
 - Exibir e ocultar elementos

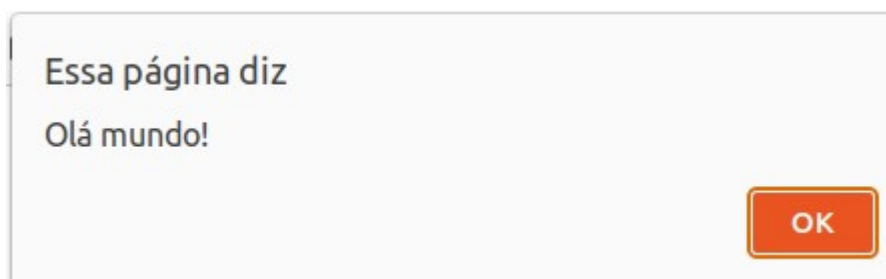
Características do JS

- Código fonte JS é incluído junto ao HTML
 - Pode ser feito de diferentes formas
- Programação é dirigida por eventos e orientada a objetos
- Tipagem do JS
 - Dinâmica: a variável assume o tipo do valor atribuído
 - Fraca: pode-se alterar o tipo de uma variável
- JS é **case-sensitive**
 - **Diferencia letras maiúsculas de minúsculas**
 - Utilize o padrão camelCase

Primeiro exemplo com JS

- Exibir mensagem com JS

```
<button type="button" onclick="alert('Olá mundo!');">Clique aqui!</button>
```



Formas de exibir dados

- Com JS, podemos exibir dados de diferentes formas:

<!--Utilizando uma caixa de alerta-->

```
<button type="button" onclick="alert('Olá mundo!');">Alerta</button>
```

<!--Escrevendo no HTML-->

```
<button type="button" onclick="document.write('Olá mundo!');">Html</button>
```

<!--Escrevendo em um elemento da página HTML-->

```
<p id="paragrafo"></p>
```

```
<button type="button"  
  onclick="document.getElementById('paragrafo').innerHTML = 'Olá mundo!';">  
  Elemento Html</button>
```

<!--Escrevendo no console do navegador-->

```
<button type="button" onclick="console.log('Olá mundo!');">Console</button>
```


Adicionar JS na página HTML

- Forma 1:

- Junto ao HTML usando a tag **<script>**
- **<script type="text/javascript">codigo JS</script>**
 - Deve ser declarado antes do fim da tag <body> (melhor desempenho)
 - Utilize funções para poder executar o código

type é opcional

```
<p id="paragrafo">Este texto vai ser alterado</p>
<button type="button" onclick="evento();">Executa Script</button>

<script>
  function evento() {
    document.getElementById('paragrafo').innerHTML = 'A mágica aconteceu!';
    document.getElementById('paragrafo').style.color = 'blue';
  }
</script>
```

Adicionar JS na página HTML

- Forma 2:
 - Scripts externos
 - Arquivo externo (.js), sendo adicionado através da tag **<script>**
 - Também deve ser declarado antes do fim da tag <body> (melhor desempenho)

```
<p id="paragrafo">Este texto vai ser alterado</p>  
<button type="button" onclick="evento();">Executa Script</button>  
  
<script src="ola_mundo.js"></script>
```

```
//Arquivo ola_mundo.js  
function evento() {  
    document.getElementById('paragrafo').innerHTML = 'Abracadabra!';  
    document.getElementById('paragrafo').style.color = 'green';  
}
```