



# Curso de JavaScript para Iniciantes: Fundamentos e Prática

EXERCÍCIOS

MÓDULO 3

LEANDRO OLIVEIRA

LECONNECT

<https://leconnect.online/>

# Exercícios do Módulo 3 - Manipulação de Elementos HTML

## Conteúdo base

1. Manipulação do DOM (Document Object Model)
2. Selecionando elementos por ID, classe e tag
3. Modificando conteúdo e estilos dos elementos
4. Manipulando eventos do usuário

## Preparado para começar?

Nesta lista, você encontrará 20 exercícios com diversos tipos de abordagem. São elas: verdadeiro ou falso, única escolha, múltipla escolha e abertos (resposta livre). Ao final da lista, você encontrará as respostas.

Exercício 1:

Assunto: Manipulação do DOM (Document Object Model)

Tipo: Verdadeiro ou Falso

Pergunta: O DOM (Document Object Model) é uma representação hierárquica dos elementos HTML de uma página, permitindo a manipulação dinâmica desses elementos.

Exercício 2:

Assunto: Manipulação do DOM (Document Object Model)

Tipo: Única escolha

Pergunta: Qual método é utilizado para adicionar um novo elemento HTML a um documento através do JavaScript?

- a) append()
- b) create()
- c) add()
- d) insert()

Exercício 3:

Assunto: Manipulação do DOM (Document Object Model)

Tipo: Preencha as lacunas

Pergunta: Complete o código JavaScript abaixo para selecionar um elemento pela sua ID e alterar o seu texto para "Olá, Mundo!":

```
const elemento = document._____('elementoID');  
elemento.______ = "Olá, Mundo!";
```

Exercício 4:

Assunto: Manipulação do DOM (Document Object Model)

Tipo: Múltiplas escolhas

Pergunta: Quando o código JavaScript é carregado no navegador?

- a) Antes do carregamento do HTML
- b) Após o carregamento do HTML
- c) Durante o carregamento do HTML
- d) O código JavaScript é carregado junto com o HTML

Exercício 5:

Assunto: Selecionando elementos por ID, classe e tag

Tipo: Resposta livre

Pergunta: Descreva três métodos diferentes para selecionar elementos HTML no DOM usando JavaScript.

Exercício 6:

Assunto: Selecionando elementos por ID, classe e tag

Tipo: Verdadeiro ou Falso

Pergunta: O método `querySelector()` retorna uma lista de todos os elementos que correspondem ao seletor especificado no DOM.

- a) Verdadeiro
- b) Falso

Exercício 7:

Assunto: Selecionando elementos por ID, classe e tag

Tipo: Única escolha

Pergunta: Qual seletor CSS é usado para selecionar elementos por sua classe em JavaScript?

- a) #
- b) .
- c) @
- d) :

Exercício 8:

Assunto: Selecionando elementos por ID, classe e tag

Tipo: Preencha as lacunas

Pergunta: Complete o código JavaScript abaixo para selecionar todos os elementos com a classe "item" e armazená-los em uma variável:

```
const elementos = document._____('_____');
```

Exercício 9:

Assunto: Modificando conteúdo e estilos dos elementos

Tipo: Múltiplas escolhas

Pergunta: Qual método pode ser usado para alterar o conteúdo HTML de um elemento em JavaScript?

- a) setInnerHTML
- b) innerHTML
- c) innerText
- d) setText

Exercício 10:

Assunto: Modificando conteúdo e estilos dos elementos

Tipo: Resposta livre

Pergunta: Explique a diferença entre os métodos innerText e innerHTML em JavaScript.

Exercício 11:

Assunto: Modificando conteúdo e estilos dos elementos

Tipo: Verdadeiro ou Falso

Pergunta: O método `setAttribute()` pode ser usado para adicionar ou modificar atributos em elementos HTML.

- a) Verdadeiro
- b) Falso

Exercício 12:

Assunto: Modificando conteúdo e estilos dos elementos

Tipo: Única escolha

Pergunta: Qual método é utilizado para alterar o estilo de um elemento em JavaScript?

- a) `setStyle()`
- b) `changeStyle()`
- c) `style()`
- d) `setAttribute()`

Exercício 13:

Assunto: Modificando conteúdo e estilos dos elementos

Tipo: Preencha as lacunas

Pergunta: Complete o código JavaScript abaixo para alterar a cor de fundo de um elemento com a ID "meuElemento" para vermelho:

```
const elemento = document.getElementById('_____');  
elemento._____ = '_____';
```

Exercício 14:

Assunto: Manipulando eventos do usuário

Tipo: Verdadeiro ou Falso

Pergunta: Os eventos do usuário, como cliques de mouse e pressionamento de teclas, podem ser tratados em JavaScript através do uso de funções de callback.

- a) Verdadeiro
- b) Falso

Exercício 15:

Assunto: Manipulando eventos do usuário

Tipo: Única escolha

Pergunta: Qual método é usado para adicionar um ouvinte de evento a um elemento em JavaScript?

- a) addEventListener()
- b) addEvent()
- c) attachEvent()
- d) setEventListener()

Exercício 16:

Assunto: Manipulando eventos do usuário

Tipo: Múltiplas escolhas

Pergunta: É possível adicionar mais de um ouvinte de evento ao mesmo elemento em JavaScript?

- a) Sim, apenas com o método addEventListener()
- b) Sim, apenas com o método attachEvent()
- c) Não, apenas um ouvinte por elemento
- d) Sim, com ambos os métodos addEventListener() e attachEvent()

Exercício 17:

Assunto: Manipulando eventos do usuário

Tipo: Preencha as lacunas

Pergunta: Complete o código JavaScript abaixo para adicionar um ouvinte de evento para o clique do mouse em um elemento com a ID "botao":

```
document.getElementById('_____')._____('_____', function() {  
    // Código a ser executado quando o botão for clicado  
});
```

Exercício 18:

Assunto: Manipulando eventos do usuário

Tipo: Resposta livre

Pergunta: Descreva a diferença entre os eventos "click" e "dblclick" em JavaScript.

Exercício 19:

Assunto: Modificando conteúdo e estilos dos elementos

Tipo: Resposta livre

Pergunta: Crie um código JavaScript que selecione um elemento com a classe "destaque" e adicione a classe "ativo" a ele.

Exercício 20:

Assunto: Manipulação do DOM (Document Object Model)

Tipo: Preencha as lacunas

Pergunta: Complete o código JavaScript abaixo para criar um novo elemento <p> com o texto "Novo parágrafo" e adicionar esse elemento ao final do elemento com a ID "conteudo":

```
const novoParagrafo = document._____('p');  
novoParagrafo.______ = "Novo parágrafo";  
document.getElementById('____').______(novoParagrafo);
```

# Respostas

Exercício 1:

Resposta: Verdadeiro

Exercício 2:

Resposta: b) create()

Exercício 3:

Resposta:

```
const elemento = document.getElementById('elementoID');  
elemento.textContent = "Olá, Mundo!";
```

Exercício 4:

Resposta: b) Após o carregamento do HTML

Exercício 5:

Resposta:

- getElementById() para selecionar elementos por ID.
- getElementsByClassName() para selecionar elementos por classe.
- getElementsByTagName() para selecionar elementos por tag.

Exercício 6:

Resposta: Falso. O método querySelector() retorna apenas o primeiro elemento que corresponde ao seletor especificado.

Exercício 7:

Resposta: b) .



Exercício 8:

Resposta:

```
const elementos = document.getElementsByClassName('item');
```

Exercício 9:

Resposta: b) innerHTML

Exercício 10:

Resposta: innerText é usado para definir ou obter o texto visível de um elemento, enquanto innerHTML é usado para definir ou obter o conteúdo HTML do elemento, incluindo todas as tags.

Exercício 11:

Resposta: Verdadeiro

Exercício 12:

Resposta: c) style()

Exercício 13:

Resposta:

```
const elemento = document.getElementById('meuElemento');  
elemento.style.backgroundColor = 'red';
```

Exercício 14:

Resposta: Verdadeiro

Exercício 15:

Resposta: a) addEventListener()

Exercício 16:

Resposta: d) Sim, com ambos os métodos addEventListener() e attachEvent()

Exercício 17:

Resposta:

```
document.getElementById('botao').addEventListener('click',  
function() {  
    // Código a ser executado quando o botão for clicado  
});
```

Exercício 18:

Resposta: O evento "click" ocorre quando um elemento é clicado uma vez, enquanto o evento "dblclick" ocorre quando um elemento é clicado duas vezes rapidamente.

Exercício 19:

Resposta:

```
const elementoDestaque = document.querySelector('.destaque');  
elementoDestaque.classList.add('ativo');
```

Exercício 20:

Resposta:

```
const novoParagrafo = document.createElement('p');  
novoParagrafo.textContent = "Novo parágrafo";  
document.getElementById('conteudo').appendChild(novoParagrafo);
```

## Considerações finais

Para cada módulo do curso, serão realizados exercícios com o objetivo de consolidar o conteúdo aprendido. Dessa forma, disponibilizarei um PDF semelhante a este, para que você possa praticar.

É importante lembrar que os exercícios serão constantemente atualizados e novos serão adicionados. Portanto, sempre verifique os módulos que você já concluiu para encontrar novos exercícios.

Agradeço imensamente por dedicar seu tempo para realizar essas atividades no [Curso de JavaScript para Iniciantes: Fundamentos e Prática](#).

Recomendo que, para avançar para o próximo módulo, você responda aos exercícios propostos. Quando se sentir confortável, sinta-se à vontade para prosseguir com os próximos módulos do curso.

Não deixe de compartilhar com seus amigos e em suas redes sociais, isso nos ajudará a melhorar e criar novos cursos.

Bons estudos!

Próximo módulo

