

## Trilha de aprendizagem 04

**Objetivo: Atribuindo escutadores de eventos DOM a elementos HTML com JavaScript.**

### Referências

[https://www.w3schools.com/jsref/dom\\_obj\\_event.asp](https://www.w3schools.com/jsref/dom_obj_event.asp)  
[https://www.w3schools.com/js/js\\_htmldom\\_events.asp](https://www.w3schools.com/js/js_htmldom_events.asp)  
[https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/Building\\_blocks/Events](https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/Building_blocks/Events)

### Introdução

O HTML DOM permite que sejam atribuídos eventos a elementos HTML usando JavaScript. Isso pode ser feito diretamente no JavaScript obtendo o objeto que corresponde ao elemento:

```
document.getElementById("myBtn").onclick = displayDate;
```

Ou através da adição de um ouvinte de eventos:

```
document.getElementById("myBtn").addEventListener("click", displayDate);
```

O método `addEventListener()` anexa um manipulador de eventos ao elemento especificado, sem substituir os manipuladores de eventos existentes. É possível adicionar vários manipuladores de eventos a um elemento, ou seja, dois eventos de "clique" a um mesmo elemento. Ao usar o método `addEventListener()`, o JavaScript é separado da marcação HTML, para melhor legibilidade, e permite que você adicione ouvintes de eventos mesmo quando não controla a marcação HTML.

Sintaxe:

```
element.addEventListener(event, function);
```

Também é possível remover um ouvinte de eventos usando o método `removeEventListener()`.

### // Exemplo de adição de evento via JavaScript

```
<!DOCTYPE html>
<html>
  <body>
    <h1> Eventos HTML JavaScript</h1>
    <h2>Evento de click </h2>

    <p>Click "aqui" para executar a função displayDate() function.</p>
    <button id="myBtn">Clique</button>

    <p id="demo"></p>

  </script>
    document.getElementById("myBtn").onclick = displayDate;

    function displayDate() {
      document.getElementById("demo").innerHTML = Date();
    }
  </script>

</body>
</html>
```

### // Adicionando ouvintes de eventos

1 – No arquivo script.js adicione um evento de click para o objeto “myBtn”. Associe a ele uma função que mostre uma mensagem de alerta.

```
let elemento = document.getElementById("myBtn");
elemento.addEventListener("click", alerta);

function alerta() {
```

## Interatividade em Páginas Web

```
alert("Olá, mundo!");
}
```

2 – Crie um título e insira três eventos: onclick, onmouseover, onmouseout.

// Em html

```
<h1 id="titulo">Clique ou passe o mouse aqui!</h1>
```

// Em javascript

```
// Seleciona o elemento
const titulo = document.getElementById('titulo');

// Adiciona um evento de clique
titulo.addEventListener('click', function() {
    alert('O título foi clicado!');
});

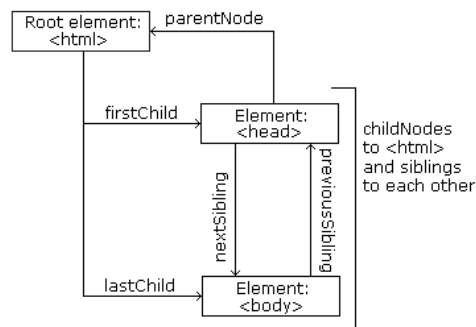
// Adiciona um evento de mouseover (quando o mouse passa sobre o elemento)
titulo.addEventListener('mouseover', function() {
    titulo.style.color = 'blue';
});

// Adiciona um evento de mouseout (quando o mouse sai do elemento)
titulo.addEventListener('mouseout', function() {
    titulo.style.color = 'black';
});
```

## Criação dinâmica de elementos HTML

A criação dinâmica de novos elementos HTML usando JavaScript, usa os métodos: **document.createElement**, **element.appendChild**, e **element.insertBefore**

### Hierarquia de relacionamento entre nós DOM.



3 – Faça um código em javascript que crie um elemento do tipo Parágrafo, em seguida crie um nó, e depois adicione ao parágrafo. Considere que na página HTML já existe uma div:

```
<div id="div1">
  <p id="p1">Este é um parágrafo.</p>
</div>
```

Agora insira o código para criação dinâmica.

```
const para = document.createElement("p");
const node = document.createTextNode("Este é o novo parágrafo.");
para.appendChild(node);

const element = document.getElementById("div1");
element.appendChild(para);
```

// appendChild insere um novo elemento como último filho do pai.

## Interatividade em Páginas Web

4 – Na sequência, crie uma inserção de um novo parágrafo como primeiro filho do nó pai. Para isso pode utilizar o *insertBefore*.

```
const child = document.getElementById("p1");
element.insertBefore(para, child);
```

5 – O próximo passo é remover um nó filho via javascript.

```
const parent = document.getElementById("div1");
parent.removeChild(child);
```

## Adicionando eventos e manipulando elementos.

6 – Seguindo coloque esses dois elementos em sua página HTML.

```
<div id="container"> <p>Este é um parágrafo inicial.</p> </div>

<button id="adicionar">Adicionar Novo Parágrafo</button>
```

7 – Agora, em javascript adicione um evento click ao botão.

```
// Seleciona o botão e o contêiner onde os novos elementos serão adicionados
const botao = document.getElementById('adicionar');
const container = document.getElementById('container');

// Adiciona um evento de clique ao botão
botao.addEventListener('click', adicionaParagrafo)
```

8 – Próximo passo é em javascript criar um elemento parágrafo e adicionar este elemento a div “container” quando clicado no botão.

```
function adicionaParagrafo() {
  // Cria um novo elemento <p>
  const novoParagrafo = document.createElement('p');

  // Adiciona algum texto ao novo parágrafo
  novoParagrafo.textContent = 'Este é um novo parágrafo adicionado dinamicamente!';

  // Adiciona o novo parágrafo ao contêiner
  container.appendChild(novoParagrafo);
}
```

9 – Próxima tarefa tem o objetivo de atualizar o conteúdo de um elemento já existente em HTML. Vamos criar dois inputs e uma tabela. Em seguida um botão, ao clicar neste botão, o conteúdo dos inputs deverá ser inserido na tabela.

Coloque os inputs no html:

```
<label>Primeiro Valor:</label>
<input type="text" id="input1">
<br><br>

<label>Segundo Valor:</label>
<input type="text" id="input2">
<br><br>
```

10 – Agora insira o botão e a tabela.

```
<button id="adicionarTabela">Adicionar à Tabela</button>

<table id="tabela">
  <thead> <tr> <th>Valor 1</th> <th>Valor 2</th> </tr> </thead>
  <tbody> </tbody>
</table>
```

11 – Crie o código em javascript que adicione o evento que insira os dados dos inputs nas tabelas.

```
const bAdicionar = document.getElementById('adicionarTabela');
const tabela = document.getElementById('tabela').getElementsByTagName('tbody')[0];

bAdicionar.addEventListener('click', adicionarDados);

function adicionarDados() {
```



## Interatividade em Páginas Web

```
const valor1 = document.getElementById('input1').value;
const valor2 = document.getElementById('input2').value;
// Verifica se os valores não estão vazios
if (valor1.trim() !== "" && valor2.trim() !== "") {
  // Cria uma nova linha
  const novaLinha = tabela.insertRow();

  // Cria e insere novas células com os valores
  const celula1 = novaLinha.insertCell(0);
  const celula2 = novaLinha.insertCell(1);
  celula1.textContent = valor1;
  celula2.textContent = valor2;

  // Limpa os inputs após adicionar os valores à tabela
  document.getElementById('input1').value = '';
  document.getElementById('input2').value = '';
} else {
  alert("Por favor, preencha ambos os campos antes de adicionar à tabela.");
}
}
```