

Laboratório 9

Neste laboratório vamos apresentar um pouco mais sobre JavaScript. O objetivo é criar uma página com alguns eventos.

Página HTML:

1 – Crie uma pasta chamada lab09. Nessa pasta, crie um arquivo chamado **event** com a extensão **.html (event.html)**. Neste arquivo será criada nossa página HTML.

2 – Note que já adicionamos as classes que serão utilizadas no css, além disso já colocamos algumas funções JavaScript nos eventos do HTML.

3 – Começamos nossa estrutura HTML fazendo a *linkagem* dos arquivos de css e javascript.

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

<head>
  <meta charset="utf-8">
  <title>Página Formulário</title>
  <link rel='stylesheet' type='text/css' href='style.css'>
  <script src="event.js"></script>
</head>

<body>
  <header>
    <h1>Meu Formulário</h1>
  </header>
```

4 – O primeiro evento que usamos é o evento *onclick*. Esse evento executa uma função quando um elemento HTML é clicado pelo usuário.

```
<main>
  <section id="opcoes">
    <h2>Eventos:</h2>
    <div class="box">
      <div class="box-element">
        <button onclick="alert('Clicou!')">Clique Aqui</button>
      </div>
      <div class="box-element">
        <button onclick="adicionarLivro()">Adicionar um livro</button>
      </div>
      <div class="box-element">
        <button onclick="limparLivros()">Limpar livros</button>
      </div>
```

Note que a função do botão “Clique Aqui” já foi implementada de forma in-line. Essa é uma função muito simples que irá abrir um “pop-up” com a mensagem “Clicou!” ao ser clicado pelo usuário.

```

<div class="box-element">
  <div class="group-field">
    <label>Adicionar filme:</label>
    <input type="text" id="input-filme">
    <button onclick="adicionarFilme()">Adicionar</button>
  </div>
</div>
<div class="box-element">
  <div class="group-field">
    <label>Remover filme:</label>
    <input type="text" id="input-filme-2">
    <button onclick="removerFilme()">Remover</button>
  </div>
</div>
<div class="box-element">
  <button onclick="limparFilmes()">Limpar filmes</button>
</div>

```

5 – Abaixo vamos utilizar outros eventos, como `onchange` e `onkeydown`.

```

<div class="box-element">
  <div class="group-field">
    <label>Digite uma mensagem:</label>
    <input type="text" id="input-mensagem" onchange="imprimirMensagem()"/>
  </div>
</div>
<div class="box-element">
  <div class="group-field">
    <label>Digite uma mensagem:</label>
    <input type="text" id="input-mensagem-2" onkeydown="imprimirMensagem2()"/>
  </div>
</div>
</div>
</section>

```

6 – Vamos fechar nosso HTML.

```

<section id="painel">
  <div id="livros">
    <h2>Livros:</h2>
    <p>Nenhuma informação disponível.</p>
  </div>
  <div id="filmes">
    <h2>Filmes:</h2>
    <p>Nenhuma informação disponível.</p>
  </div>
  <div id="mensagens">
    <h2>Mensagem:</h2>
    <p id="mensagem"></p>
  </div>
</section>
</main>

<footer>
  <p>© 20XX Minha Página. Todos os direitos reservados.</p>
</footer>
</body>

</html>

```

Criando o CSS:

1 – Vamos criar nosso arquivo css. Na pasta, crie um arquivo chamado **style** com a extensão **.css (style.css)**.

2 - Vamos adicionar as propriedades do CSS para estilizar nossa página.

```
html, body {  
    height: 100%;  
    margin: 0;  
    padding: 0;  
    background-color: black;  
    font-family: Arial, sans-serif;  
    line-height: 1.6;  
}
```

```
header,  
footer {  
    flex-shrink: 0;  
    text-align: center;  
    background: #89CFF0;  
    color: black;  
    padding: 20px;  
}  
  
footer p {  
    margin: 0;  
}
```

```
main {  
    height: auto;  
    overflow: hidden;  
    display: flex;  
    flex-wrap: wrap;  
    align-items: center;  
    justify-content: space-between;  
    gap: 5px;  
    padding: 20px;  
}  
  
section {  
    margin-top: 20px;  
}
```

```
.group-field {  
    display: flex;  
    align-items: center;  
    gap: 10px;  
    margin: auto;  
}  
  
label {  
    margin-left: 10px;  
    margin-bottom: 5px;  
}  
  
input {  
    padding: 8px;  
    border: 1px solid #ccc;  
    border-radius: 5px;  
    background-color: white;  
}
```

```
button {  
    display: block;  
    width: 80%;  
    padding: 10px;  
    margin: auto;  
    background-color: #89cff0;  
    color: white;  
    border: none;  
    border-radius: 5px;  
    font-size: 16px;  
    cursor: pointer;  
    text-align: center;  
}  
  
button:hover {  
    background-color: #45a049;  
}
```

```
li {  
    margin: 10px;  
}
```

```
#panel {
    height: 60vh;
    overflow-y: scroll;
    flex: 0.4;
    background: #lightcyan;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
}
```

```
#opcoes {
    height: 60vh;
    overflow-y: scroll;
    flex: 0.6;
    background: #lightcyan;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
}
```

```
.box-element {
    display: flex;
    align-items: center;
    text-align: center;
    background-color: #white;
    margin: 5px;
    padding: 10px 0px;
    border: 2px solid #ccc;
    border-radius: 5px;
}

.box {
    height: auto;
    display: grid;
    grid-template-columns: auto auto auto;
}
```

3 - A página final ficará semelhante a figura abaixo:

Meu Formulário

Eventos:

<input type="button" value="Clique Aqui"/>	<input type="button" value="Adicionar um livro"/>	<input type="button" value="Limpar livros"/>
Adicionar filme: <input type="text"/>	Remover filme: <input type="text"/>	Remover <input type="button" value="Filmes"/>
Digite uma mensagem: <input type="text"/>	Digite uma mensagem: <input type="text"/>	<input type="button" value="Limpar mensagens"/>

Livros:
Nenhuma informação disponível.

Filmes:
Nenhuma informação disponível.

Mensagem:

© 20XX Minha Página. Todos os direitos reservados.

Criando o JavaScript:

1 – Vamos criar nosso arquivo javascript com o nome **event** com a extensão **.js** (**event.js**).

2 – A primeira função que iremos criar é a função de adicionar um livro. Essa função irá ser acionada quando o usuário clicar no botão “Adicionar um livro”.

```
function adicionarLivro() {
    let livro = prompt('Digite o nome do livro:');
    let element = document.getElementById('livros');
    element.innerHTML += `<li>${livro}</li>`;
}
```

Após implementar a função, vamos para a página HTML testar o botão. Analise o que ocorreu na página e como essa funcionalidade modifica o HTML.

3 – A segunda função que iremos criar é a função de limpar a lista de livros.

```
function limparLivros() {
    let element = document.getElementById('livros');
    element.innerHTML = '<h2>Livros:</h2><p>Nenhuma informação disponível.</p>';
}
```

4 – Vamos agora implementar a função para adicionar os filmes.

```
let filmes = [];
function adicionarFilme() {
    let filme = document.getElementById('input-filme').value;
    let element = document.getElementById('filmes');
    filmes.push(filme);

    element.innerHTML = '<h2>Filmes:</h2>';
    filmes.forEach(e => {
        element.innerHTML += `<li>${e}</li>`;
    });
}
```

5 – Vamos criar uma função para remover um filme específico da lista.

```
function removerFilme() {
    let filme = document.getElementById('input-filme-2').value;
    for (let i = 0; i < filmes.length; i++) {
        if (filmes[i] === filme) {
            filmes.splice(i, 1);
            break;
        }
    }

    let element = document.getElementById('filmes');
    element.innerHTML = '<h2>Filmes:</h2>';
    filmes.forEach(e => {
        element.innerHTML += `<li>${e}</li>`;
    });
}
```

6 – Vamos criar a função para limpar a lista de filmes.

```
function limparFilmes() {
    filmes = [];

    let element = document.getElementById('filmes');
    element.innerHTML = '<h2>Filmes:</h2><p>Nenhuma informação disponível.</p>';
}
```

7 – Para finalizar, vamos criar as funções que serão utilizadas nos eventos `onchange` e `onkeydown`.

```
function imprimirMensagem() {
    let msg = document.getElementById('input-mensagem').value;
    document.getElementById('mensagem').innerHTML = msg;
}

function imprimirMensagem2() {
    let msg = document.getElementById('input-mensagem-2').value;
    document.getElementById('mensagem').innerHTML = msg;
}
```

Desafio:

1 – Faça uma refatoração no arquivo javascript para remover trechos de códigos repetidos.

2 – Corrija o bug da lista de livros onde a mensagem “Nenhuma informação disponível.” continua na tela mesmo com a adição de livros.

3 – Crie novas funcionalidades como, por exemplo, lista de música.