

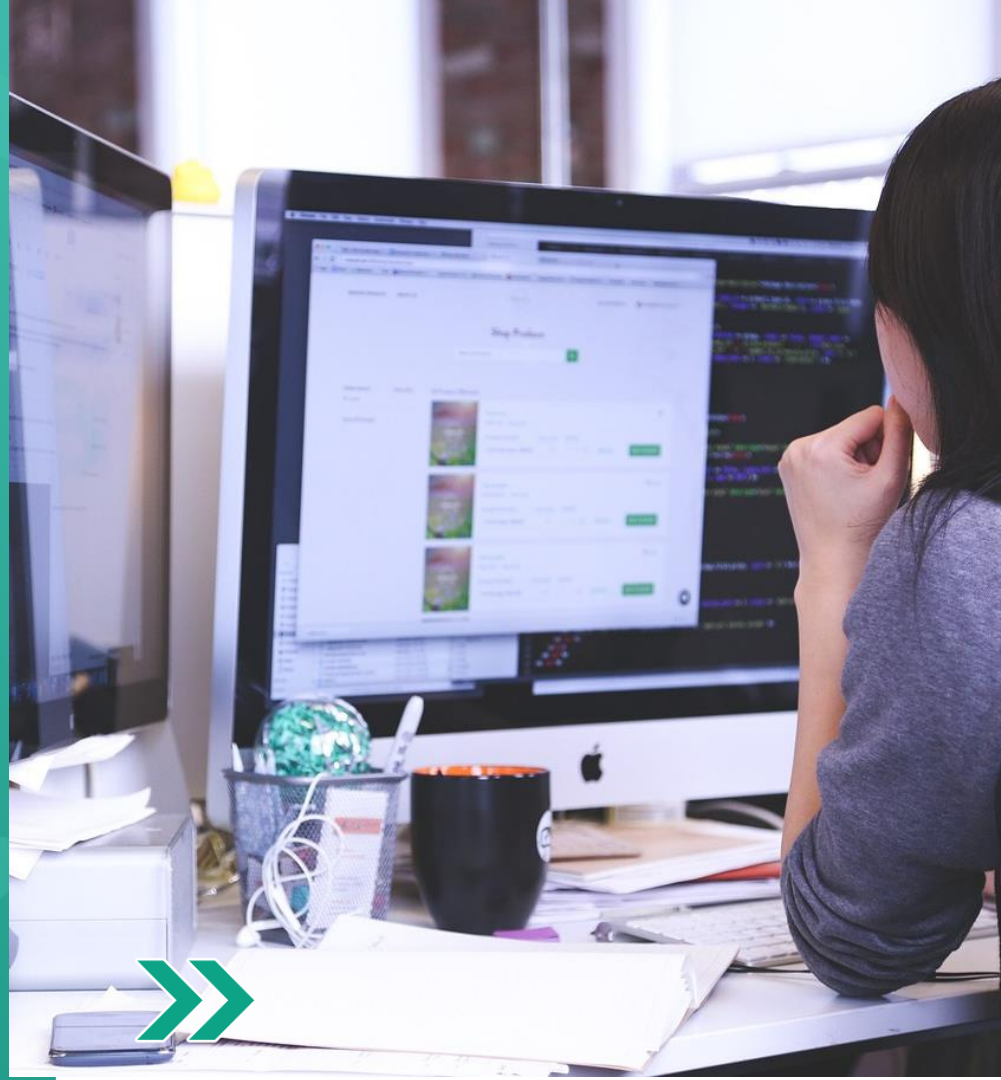


escola  
britânica de  
artes criativas  
& tecnologia

# Front-End

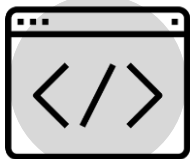


# BOAS PRÁTICAS



# Integre HTML com JavaScript

- **Documente HTML com DOM**
- **Adicione eventos a elementos HTML**



Confira boas práticas da comunidade de Front-End por assunto relacionado às aulas.



# Documente HTML com DOM



## **Location.pathname:**

O `location.pathname` retorna apenas o caminho e o nome do arquivo da URL, não incluindo o domínio, o protocolo ou outros componentes da URL. Se você precisar obter a URL completa, incluindo o domínio e o protocolo, pode usar `location.href` em vez de `location.pathname`.



# Adicione eventos a elementos HTML

Além dos eventos estudados, existem vários outros disponíveis no JavaScript que podem ser usados para manipular a interação do usuário com os elementos da página. Acompanhe!



## Eventos de mouse:

- **click:** Ocorre quando o usuário clica em um elemento.
- **mouseover:** Ocorre quando o cursor do mouse é colocado sobre um elemento.
- **mouseout:** Ocorre quando o cursor do mouse deixa um elemento.
- **mousedown:** Ocorre quando o botão do mouse é pressionado em um elemento.
- **mouseup:** Ocorre quando o botão do mouse é liberado em um elemento.



## Eventos de teclado:

- **keydown:** Ocorre quando uma tecla é pressionada.
- **keypress:** Ocorre quando uma tecla é pressionada e liberada.

# Adicione eventos a elementos HTML

Além dos eventos estudados, existem vários outros disponíveis no JavaScript que podem ser usados para manipular a interação do usuário com os elementos da página. Acompanhe!



- **Eventos de formulário:**
  - **submit:** Ocorre quando um formulário é submetido.
  - **input:** Ocorre quando o valor de um campo de entrada é alterado.
- **Eventos de janelas:**
  - **load:** Ocorre quando a página é completamente carregada.
  - **resize:** Ocorre quando a janela do navegador é redimensionada.
  - **scroll:** Ocorre quando a página é rolada.

# Adicione eventos a elementos HTML



## **Código bloqueante:**

Apesar de o JavaScript ser um código bloqueante, ele também possui recursos assíncronos que permitem contornar o bloqueio e evitar a interrupção da execução sequencial do código.



# Bons estudos!

