

EVENT LISTENERS

(AULA 15)

CURSO BÁSICO DE PROGRAMAÇÃO COM PYTHON

MAYARA MARQUES

mmrosatab@gmail.com

SUMÁRIO

- Event Listeners
- Mão na massa

EVENT LISTENERS

Event listeners são usados para "***escutar***" eventos que ocorrem em elementos HTML, como **cliques, submissões de formulários, teclas pressionadas**, entre outros. Eles permitem que você responda a esses eventos executando código quando ocorrem.

EXEMPLO

```
// recupera elemento caixa de input
const inputName = document.getElementById("name")

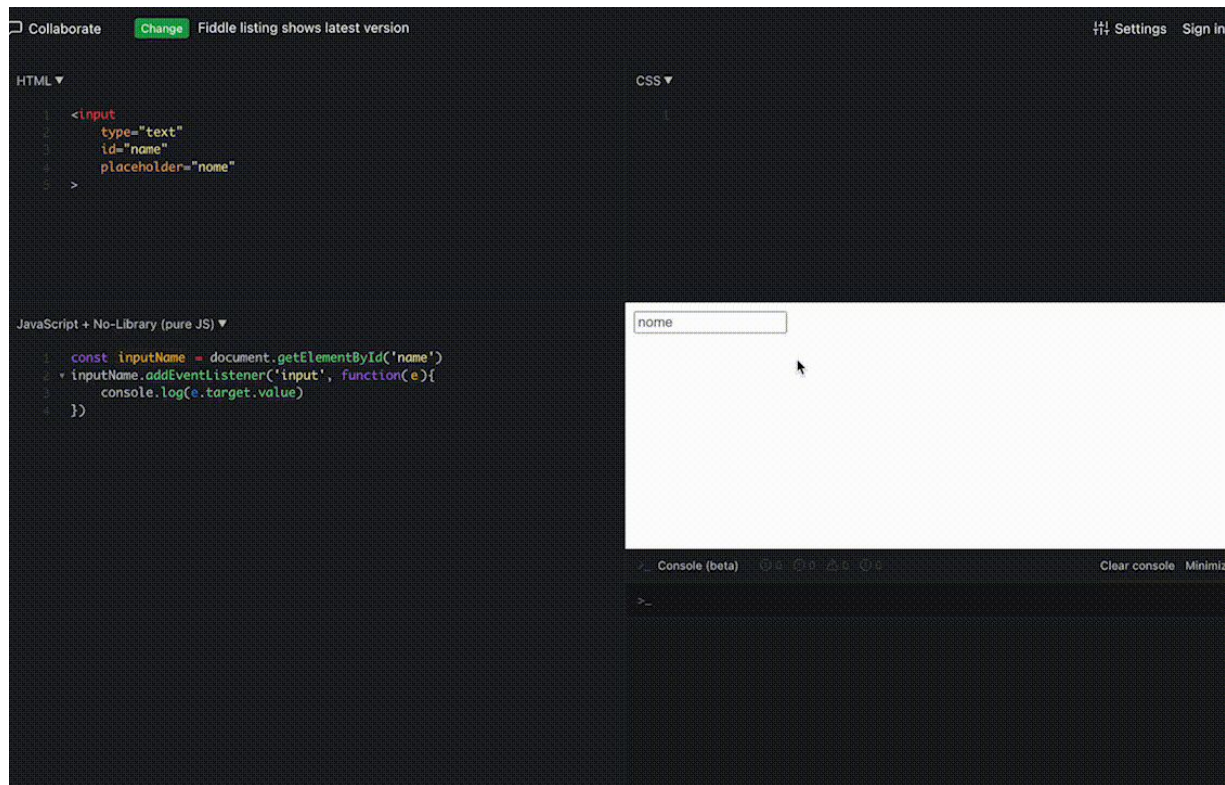
// adiciona evento de escuta para mudança de valor na caixa de input
inputName.addEventListener("input", function(e) {
    // imprime valor contido na caixa de input
    console.log(e.target.value)
})
```



Função de callback

EXEMPLO

JS



EXEMPLO

Atribuindo uma função a uma variável

```
// recupera elemento da caixa de input  
const inputName = document.getElementById("name")
```

Função anônima

```
const printInputValue = function(e) {  
    // imprime valor contido na caixa de input  
    console.log(e.target.value)  
}
```

Função de callback

```
// adiciona evento de escuta para mudança de valor na caixa de input  
inputName.addEventListener("input", printInputValue)
```

EVENT LISTENERS

Eventos de Mouse:

- click: Disparado quando o elemento é clicado com o mouse.

```
<button id="myButton">Clique aqui</button>
```

```
// Adicionando um event listener ao botão
const button = document.getElementById('myButton')

button.addEventListener('click', function() {
  console.log('Botão clicado!')
})
```

Função de callback executada assim que o evento é disparado.

EVENT LISTENERS

- Eventos de Mouse:
 - click: Disparado quando o elemento é clicado com o mouse.
 - mouseover: Disparado quando o cursor do mouse entra no elemento.
 - mouseout: Disparado quando o cursor do mouse deixa o elemento.
 - mousedown: Disparado quando um botão do mouse é pressionado sobre o elemento.
 - mouseup: Disparado quando um botão do mouse é solto após ter sido pressionado sobre o elemento.
 - mousemove: Disparado quando o cursor do mouse se move sobre o elemento.
- Eventos de Teclado:
 - keydown: Disparado quando uma tecla é pressionada.
 - keyup: Disparado quando uma tecla é solta após ter sido pressionada.
 - keypress: Disparado quando uma tecla que produz um caractere é pressionada.
- Eventos de Formulário:
 - submit: Disparado quando um formulário é submetido.
 - input: Disparado quando o valor de um elemento de formulário é alterado (input, textarea, etc.).
 - change: Disparado quando o valor de um elemento de formulário é alterado e o foco é perdido.

EVENT LISTENERS

- Eventos de Janela e Documento:
 - load: Disparado quando a página é completamente carregada.
 - DOMContentLoaded: Disparado quando o HTML da página é completamente carregado e analisado, sem esperar por folhas de estilo, imagens e outros recursos externos.
 - resize: Disparado quando a janela do navegador é redimensionada.
 - scroll: Disparado quando a barra de rolagem é movida em um elemento.
- Eventos de Focus:
 - focus: Disparado quando um elemento recebe o foco.
 - blur: Disparado quando um elemento perde o foco.



MÃO NA MASSA

MÃO NA MASSA



Baixe os arquivos script.js, index.html e styles.css contidos no link a seguir para realizar os exercícios:

<https://drive.google.com/drive/folders/1Ho8mRFRACM9S1blavFrjE2-U1AcgQCkw?usp=sharing>

MÃO NA MASSA



1. Adicione um event listener no elemento com o ID "exemplo" para que, quando ele for clicado, o conteúdo seja alterado para "Elemento clicado!".
2. Adicione um event listener no elemento com o ID "caixa" para que, quando o mouse passar sobre o elemento, a cor de fundo seja alterada para vermelho.
3. Adicione um event listener a todos os itens da lista (com a classe "item") para que, quando qualquer item for clicado, um alerta apareça com o texto "Você clicou no item X" (onde X é o número do item).
4. Adicione um event listener no campo de entrada com o ID "campo" para que, quando o campo perder o foco (evento blur), o valor do campo seja alterado para "Campo alterado por foco".
5. Adicione um event listener no botão com o ID "remove" para que, quando o botão for clicado, o elemento com o ID "caixa" seja removido do DOM.
6. Adicione um event listener no parágrafo com o ID "textoSubstituir" para que, quando ele for clicado, o texto seja alterado para "Texto alterado por clique".
7. Adicione um event listener no parágrafo com a classe "paragrafo" para que, quando o mouse passar sobre o parágrafo, a cor do texto seja alterada para azul.
8. Crie um novo botão com o texto "Adicionar Item" e adicione um event listener a ele para que, quando o botão for clicado, um novo item seja adicionado à lista "lista" com o texto "Novo Item".
9. Adicione um formulário com o campo de entrada "campo". Adicione um event listener ao formulário para que, ao ser enviado, um alerta seja exibido com o valor atual do campo de entrada.
10. Adicione um event listener a todos os itens da lista. Quando um item for clicado, altere o estilo do elemento com o ID "caixa", alterando sua cor de fundo para verde.

MÃO NA MASSA



11. Adicione um event listener ao parágrafo com o ID "textoSubstituir" para que, enquanto o mouse estiver sobre o parágrafo (evento mousemove), o texto seja alterado para "Texto alterado por movimento".
12. Adicione um event listener no campo de entrada com o ID "campo" para que, enquanto o usuário digitar algo no campo, o valor seja alterado para "Texto digitado: X", onde X é o valor digitado.
13. Adicione um event listener ao elemento com o ID "caixa2" para que, quando o mouse passar sobre ele, uma mensagem de alerta seja exibida com o texto "Você está passando o mouse sobre a caixa 2!".
14. Adicione um event listener para cada item da lista. Quando um item for clicado, adicione um novo item ao final da lista com o texto "Novo item adicionado!".
15. Adicione um event listener a todos os itens da lista "lista". Quando um item for clicado, altere o estilo do item clicado para que a cor de fundo seja amarela.
16. Adicione um event listener no campo de entrada para detectar quando uma tecla for pressionada. Quando uma tecla for pressionada, mostre uma mensagem de alerta com o código da tecla pressionada.
17. Adicione um event listener ao parágrafo com o ID "textoSubstituir". Quando o parágrafo for clicado, altere o texto para "Texto substituído por clique".
18. Adicione um event listener no botão de ID "remover" para que, ao ser clicado, a cor de fundo do botão seja alterada para vermelho.
19. Adicione um event listener para cada parágrafo com a classe "paragrafo". Quando qualquer parágrafo for clicado, altere a cor de fundo do parágrafo para cinza.
20. Adicione um event listener para todos os parágrafos. Quando o mouse passar sobre qualquer parágrafo, altere o tamanho da fonte para 20px.

DESAFIO



Como programadores, alguns dos nossos desafios são pesquisar, entender e aplicar novos conceitos quase que diariamente. Para nos acostumarmos com esse ritmo, vamos fazer uma pesquisa antes de iniciar o desafio.

Pesquise e estude sobre **posicionamento relativo** e **posicionamento absoluto** com css.

Após a pesquisa e entendimento, você deve elaborar um programa que resolva o problema elucidado no próximo slide.

DESAFIO



Neste desafio, organizamos botões dentro de uma div e modificamos seus rótulos após cada evento de clique em um dos botões.

Tarefa

Queremos criar nove botões dentro de uma div, dispostos de forma a formar uma grade. Cada botão possui um rótulo distinto de 1 a 9, e os rótulos dos botões externos devem girar no sentido horário sempre que clicarmos no botão do meio.

Complete o código no editor para que ele atenda aos seguintes critérios:

Estado inicial

O layout inicial se parece com isto:

1	2	3
4	5	6
7	8	9

*Desafio extraído de
RackerRanck*

DESAFIO



IDs dos Elementos

Cada elemento no documento deve ter um ID, conforme especificado abaixo:

- O ID da div que contém os botões deve ser **btns**.
- Cada botão deve ter o **innerHTML** e **id** como o listado abaixo:

innerHTML	id
1	btn1
2	btn2
3	btn3
4	btn4
5	btn5
6	btn6
7	btn7
8	btn8
9	btn9

DESAFIO



Estilização

- Os elementos do documento devem ter os seguintes estilos:
- A largura da div de id **btns** deve ser de **75% relativa** à largura do corpo (body) do documento.
- Cada botão (ou seja, btn1 até btn9) deve atender aos seguintes requisitos:
 - A largura deve ser de **30% relativa** à largura do seu contêiner (**btns**).
 - A altura deve ser de **48px**.
 - O tamanho da fonte deve ser de 24px.

Comportamento

- Cada vez que btn5 for clicado, o texto do innerHTML nos botões externos da grade (ou seja, btn1, btn2, btn3, btn4, btn6, btn7, btn8, btn9) deve girar no sentido horário. Não altere os IDs dos botões.

DESAFIO



Os arquivos .js e .css devem estar em diretórios diferentes, então use a tag <link> para fornecer o caminho do arquivo CSS e a tag <script> para fornecer o caminho do arquivo JS. Siga o modelo abaixo:

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="css/buttonsGrid.css" type="text/css">
  </head>

  <body>
    <script src="js/buttonsGrid.js" type="text/javascript"></script>
  </body>
</html>
```

DESAFIO



Comportamento esperado

1	2	3
4	5	6
7	8	9

Estado inicial



4	1	2
7	5	3
8	9	6

Estado após **primeiro**
clique no botão 5



7	4	1
8	5	2
9	6	3

Estado após **segundo**
clique no botão 5

REFERÊNCIAS

- <https://developer.mozilla.org/pt-BR/docs/Web/API/EventTarget/addEventListener>
- https://www.w3schools.com/js/js_htmlDOM_eventlistener.asp