



IN

DEV FULL STACK - JavaScript

Aula 06
Eventos

DEV FULL STACK - JavaScript

Aula 06: Eventos

Objetivos da aula:

1. Entender o conceito de eventos
2. Aprender como lidar com eventos utilizando JS
3. Exercitar o uso dos principais “event handlers”
4. Criar um pequeno projeto que manipula eventos

Introdução

O que é são **eventos**?

Eventos são ações ou ocorrências que acontecem no sistema que estamos desenvolvendo, no qual este te alerta sobre essas ações para que você possa responder de alguma forma, se desejado.

O que fazer com eles?

Qualquer evento pode ser capturado e tratado. Este tratamento pode ser definido em uma função e executar tarefas simples ou complexas.

Como Tratar Eventos

Há duas maneiras de se capturar e tratar eventos:

1) Inline

- São definições feitas em uma tag HTML que estabelecem o que a página deve fazer quando determinado evento acontecer.

2) Eventos externos

- São scripts JS adicionados à página HTML usando a tag `<script>` (com código na própria página ou em arquivos separados), que definem os eventos e como eles serão tratados.

Veja as duas opções implementadas no exemplo 1 dos materiais fornecidos pelo professor (Aula06_pagina_exemplo1.html).

Alguns Eventos Interessantes

Evento	Situação
onclick	O usuário clica em um elemento da página.
onchange	Um elemento da página HTML foi alterado (um campo de formulário, por exemplo).
onmouseover	O usuário posiciona o cursor do mouse sobre um elemento da página.
onmouseout	O usuário movimenta o mouse, afastando o mouse de cima de um elemento que antes estava sob o cursor.

OBS: Para uma lista completa de eventos, consulte:
<https://developer.mozilla.org/pt-BR/docs/Web/Events>

Exemplos

- **onclick**

- Exemplo 1 (Aula06_pagina_exemplo1.html).
- Três maneiras de tratar o evento de “click” do botão do mouse.

- **onchange**

- Exemplo 2 (Aula06_pagina_exemplo2.html).
- Tratando o evento de mudança de um elemento da página.

- **onmouseover e onmouseout**

- Exemplo 3 (Aula06_pagina_exemplo3.html).
- Tratando o evento de posicionamento do cursor do mouse sobre um elemento da página.

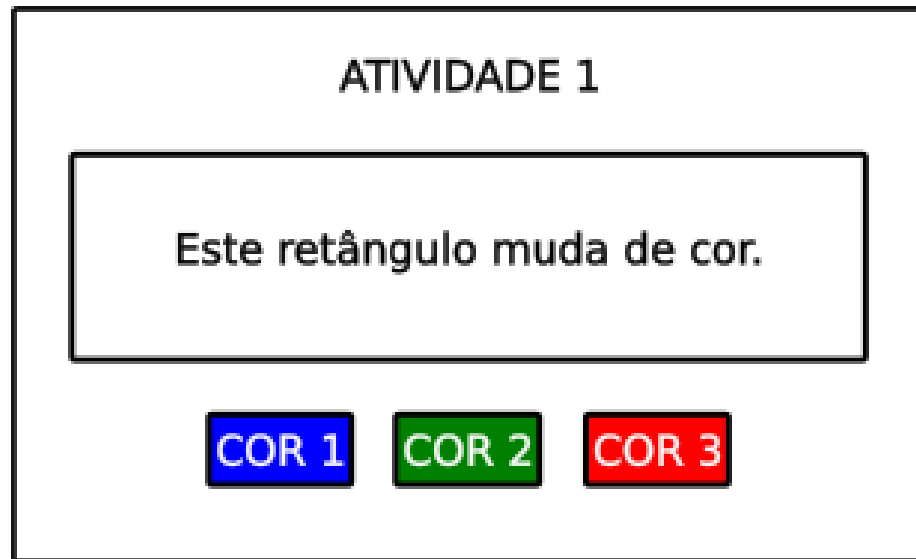
ATIVIDADE 1

Utilizando como inspiração o layout ao lado, crie uma página JS que obedeça as seguintes regras:

(1) Toda vez que um botão for apertado, troque a cor do retângulo para a cor do botão.

(2) Se o usuário passar o mouse sobre o retângulo, ele deve voltar a ser da cor branca.

OBS: os comportamentos 1 e 2 devem ser definidos e controlados usando JS



Funções Anônimas em Eventos

Funções anônimas são funções que não possuem nome e, por este motivo, para serem chamadas precisam estar hospedadas em uma variável ou evento.

Exemplo:

Alterando o exemplo 3 ([Aula06_exemplo4.html](#))



ATIVIDADE 2

Crie um formulário semelhante ao desenho ao lado e implemente comportamentos que, na medida em que cada campo seja preenchido, vá calculando e exibindo o total, a gorjeta e o valor a pagar.

Lembre-se:

Não há um botão para disparar o processo de testes, por isso você pode acrescentar um ou utilizar eventos de preenchimento (onchange).

Conta

Entrada:

Refeição:

Sobremesa:

Bebida:

Total do consumo:

Gorjeta (10%):

Valor a pagar:

Outro Exemplo

Veja um exemplo de página que atribui um comportamento a diversos elementos da mesma classe:

[Aula06_exemplo5.html](#)



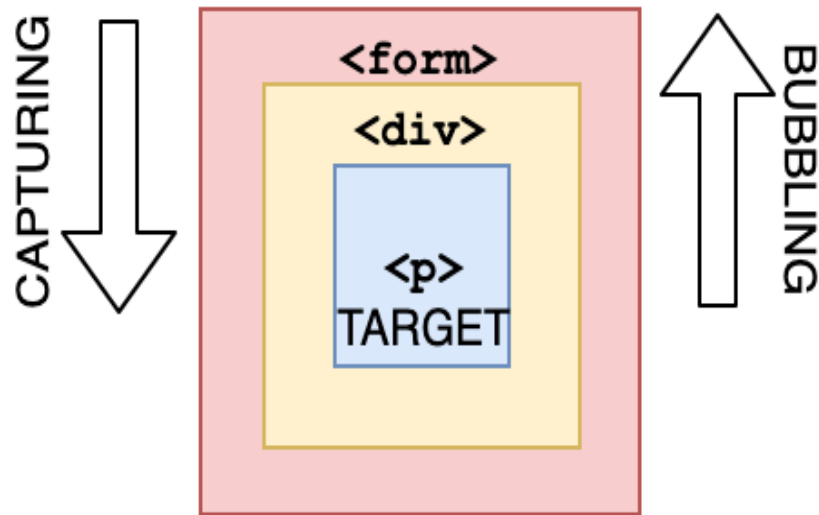
Borbulhamento e Captura

Borbulhamento e **Captura** de eventos são dois mecanismos que descrevem o que acontece quando dois manipuladores do mesmo tipo de evento são ativados em um elemento.

Exemplo: Aula06_exemplo6.html

É possível evitar isso com parando a propagação do evento.

Exemplo: Aula06_exemplo7.html



Projeto: Formulário Interativo

Observe o modelo de página fornecido pelo professor em: [Aula06_projeto.html](#)

Ele contém um formulário com uma caixa de texto solicitando um nome, 3 opções de esportes (usando radio buttons) e um botão (submit) para enviar o formulário.

O formulário também contém uma tag de imagem () que, inicialmente, está vazia.

Ao selecionar um dos esportes, deve aparecer no formulário uma imagem de 200x200 pixels com a bola utilizada no esporte escolhido (o aluno utiliza a imagem que desejar).

O botão submit, ao ser pressionado, emite um “alert” informando erro caso o nome não tenha sido preenchido (do contrário, o “alert” informa que os dados foram enviados)



ATIVIDADE Extra Casse

Investigue o efeito de manipular outros eventos em sua página.

Aqui estão algumas sugestões de eventos:

onfocus e **onblur**: monitora se o elemento está focado e fora de foco.

ondblclick: captura um duplo clique em um elemento.

onkeypress, **onkeydown**, **onkeyup**: detectam se uma tecla for pressionada no teclado. **keypress** refere-se a um pressionamento geral (botão para baixo e depois para cima), enquanto **keydown** e **keyup** refere-se apenas a parte do pressionamento da tecla para baixo e de soltar a tecla para cima, respectivamente. Podem ser associados a elementos (element), páginas (document) ou janelas (windows).

Referências

Aqui estão alguns links interessantes para leitura complementar:

Introdução a eventos:

https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/Building_blocks/Events

Lista de eventos:

<https://developer.mozilla.org/pt-BR/docs/Web/Events>