

JavaScript

Manipulação de eventos

Desenvolvimento WEB I

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas
Prof. Felipe Scheidt – IFPR – Campus Foz do Iguaçu
2023

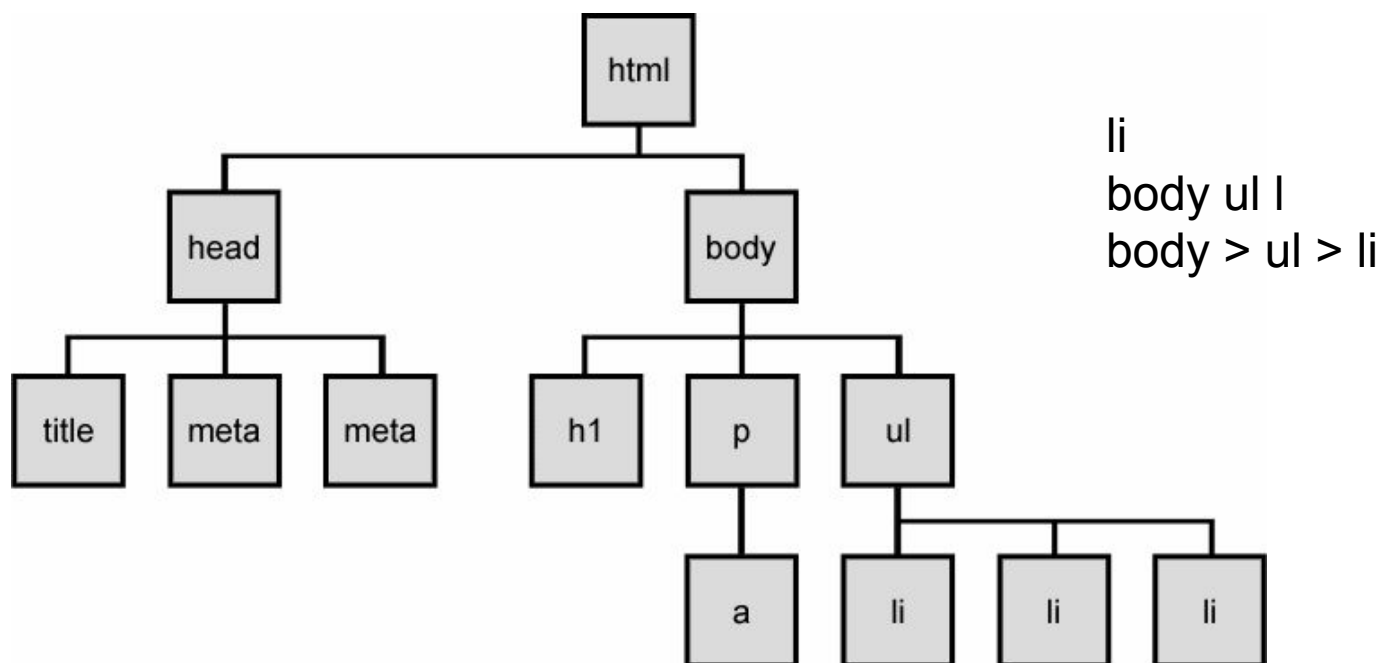
Introdução

O Javascript possui algumas funções que permitem alterar o conteúdo de um elemento HTML. Veremos alguns exemplos de uso:

- `querySelector`
- `getElementById`
- `querySelectorAll`
- `innerHTML`

DOM – Document object model

Páginas HTML são representadas através de uma árvore chamada DOM. Através do DOM pode-se manipular os elementos contidos no HTML.



Função getElementById()

Esta função permite obter uma referência a determinado elemento/tag que possua um id.

```
function calcula(){  
    var div = document.getElementById("total");  
    var input = document.getElementById("nome");  
}
```

Para isso é preciso definir o atributo id:

```
<input type="text" id="nome"/>  
<div id="total"></div>
```

Propriedade innerHTML

Permite alterar o conteúdo de um elemento HTML.

```
function alteraConteudo(){  
    var div = document.getElementById("total");  
    div.innerHTML = "novo conteúdo";  
}
```

Para a tag <input> usar a propriedade value

Propriedade display

A propriedade display permite ocultar ou exibir um elemento.

```
// função recebe uma tag e esconde a mesma  
function ocultar(tag){  
    tag.style.display = "none";  
}  
// função recebe uma tag e exibe a mesma  
function exibir(tag){  
    tag.style.display = "block";  
}
```

Eventos

- Um evento pode ser descrito como "algo" que acontece a um elemento HTML. Nesses casos é possível programar o javascript para responder de acordo com um conjunto de ações.
- Geralmente estamos interessados em eventos disparados pelo usuário, oferecendo assim mais interatividade ao site.
- Que tipos de eventos podemos tratar no javascript?

Exemplos de Eventos

Exemplos de eventos produzidos pelo usuário:

- Clique do **mouse**
- Quando o mouse passa por um elemento.
- Quando o **valor** de um elemento é alterado
- Uma tecla é pressionada
- Quando um formulário é enviado ao servidor.

Evento não relacionado ao usuário:

- Quando a página termina de carregar
- Quando uma resposta retorna do servidor

Eventos de mouse

Eventos associados a uma ação do mouse:

Função Javascript	Descrição
onclick	quando um clique é detectado
onmouseover	quando o ponteiro do mouse passa em cima de um elemento
onmouseout	quando o mouse sai de cima de um elemento
onmousedown	Aperta o botão do mouse
onmouseup	Solta o botão do mouse

Exemplo onmouseover()

Altera a cor de fundo da body para vermelho, quando o mouse passa por cima de uma div:

```
<div onmouseover="document.bgColor='red';">  
    VERMELHO  
</div>
```

Eventos de teclado

Eventos associados a uma ação do teclado:

Função Javascript	Descrição
onkeypress()	usuário aperta uma tecla
onkeydown()	quando o usuário está pressionando uma tecla
onkeyup()	quando o usuário solta a tecla

Exemplo de onkeypress()

exibe o valor digitado numa caixa alert.

```
<input type="input" onkeypress="alert(this.value);"/>
```

Evento de mudança

- **onfocus()** => elemento ganha o foco
- **onblur()** => elemento perde o foco
- **onchange()** => elemento tem seu valor alterado.

Chamando a função

Para chamar a função, podemos associar um evento qualquer e passar o nome da função que será executada. Por exemplo, para chamar a função **bemVindo**

```
<input type="button" value="Clique aqui" onclick="bemVindo()"/>
```

Funções de seleção

Javascript fornece diversas funções para seleção de elementos:

ATRIBUTO NOME DA FUNÇÃO

id getElementById()

classe getElementByClassName()

tipo da tag getElementByTagName()

nome do elemento getElementByName()

selector querySelector()

getElementsByName

Retorna um vetor de elementos de acordo com o nome da tag (atributo name).

```
var inputs = document.getElementsByName("idade");  
// percorrendo elementos cujo nome é idade  
for(i=0;i<inputs.length;i++){  
    console.log(inputs[i]);  
}
```


getElementsByClassName

retorna uma lista de elementos de acordo com o nome da classe

```
var inputs = document.getElementsByClassName("required");  
// percorrendo elementos cuja classe é required  
for(i=0;i<inputs.length;i++){  
    console.log(inputs[i]);  
}
```

getElementsByTagName

retorna uma *lista* de elementos de acordo com o nome da tag.

```
var inputs = document.getElementsByTagName("input");  
// percorrendo as inputs  
for(i=0;i<inputs.length;i++){  
    console.log(inputs[i]);  
}
```

getElementById()

Função que obtém o elemento a partir do id.
Seja o código html:

```
<div id="mensagem"></div>
```

Para obter este elemento no JS:

```
document.getElementById("mensagem");
```

querySelector

É uma função mais genérica que permite selecionar elementos usando **seletores** semelhantes ao que usamos nas regras do CSS.

Exemplos:

```
var el = document.querySelector("#mensagem")  
var el = document.querySelector(".header h1")  
var el = document.querySelector("div.card span.red")
```

Acessando atributos

Atributos agregam informações sobre os elementos e muitas vezes precisamos acessá-los para verificar seus valores. Exemplos de atributos:

- id
- class
- style
- name
- type

Ler atributos

Para obter o valor de um atributo, usamos a função `getAttribute()`, passando como parâmetro o nome do atributo desejado:

```
var div = document.getElementById("total");  
// obtem a classe da div:  
var classe = div.getAttribute("class");
```

Alterar valor do atributo

Para definir um novo atributo usamos a função `setAttribute()` passando como parâmetro o nome do atributo e qual o seu valor:

```
var input = document.getElementById("total");  
div.setAttribute("value", "200");
```

Ideia de Validação

Para implementar um validação de formulário, precisamos definir antes:

- quais campos são obrigatórios
- como notificar o usuário sobre o erro
- impedir que o formulário seja enviado ao servidor no caso de algum campo inválido.

Exercício 1

Opções de cores



Exercício: faça o script no javascript que associa o evento onmouseover a cada caixa colorida. Quando onmouseover ocorre, a cor da caixa é usada para definir a cor de fundo do respectivo parágrafo ou seja, vermelho para o parágrafo 1, azul para o segundo ...

1- Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Velit euismod in pellentesque massa placerat duis ultricies lacus sed.

2 - Orci ac auctor augue mauris augue. Habitant morbi tristique senectus et netus. Viverra orci sagittis eu volutpat odio facilisis mauris sit. Lacus luctus accumsan tortor posuere ac. Curabitur vitae nunc sed velit. Amet nisl purus in mollis nunc. Arcu dui vivamus arcu felis bibendum ut.

3 - Sed felis eget velit aliquet sagittis id consectetur. Venenatis tellus in metus vulputate eu scelerisque felis. Nulla facilisi etiam dignissim diam quis enim lobortis scelerisque. Et ligula ullamcorper malesuada proin libero.

4 - Nibh venenatis cras sed felis eget. Diam ut venenatis tellus in metus vulputate eu scelerisque felis. Cras ornare arcu dui vivamus arcu. Dignissim diam quis enim lobortis scelerisque fermentum

Exercício 2 - Valor total

Orçamento

Item: valor: quantidade: = R\$ 12.00

Exercício 3 - Orçamento total

lê os valores de cada item e a quantidade, e atualiza o valor de cada linha (caixa verde) e atualiza o total de todos items (caixa azul)

Orçamento

Item:	<input type="text" value="Leite"/>	valor:	<input type="text" value="3.00"/>	quantidade:	<input type="text" value="4"/>	=	R\$ 12.00
Item:	<input type="text" value="Sabão"/>	valor:	<input type="text" value="2.00"/>	quantidade:	<input type="text" value="10"/>	=	R\$ 20.00
Item:	<input type="text" value="Iogurte"/>	valor:	<input type="text" value="4.50"/>	quantidade:	<input type="text" value="10"/>	=	R\$ 45.00

calcular

Valor total: R\$ 77.00

Exercício 4

Faça a validação do formulário abaixo. Os 3 campos devem ser verificados quando o usuário clica no botão enviar. O campo nome deve ter no mínimo 3 caracteres, cpf tamanho 11 e email no mínimo 4. Exibir uma mensagem de sucesso (tudo válido) ou erro

Preencha o formulário:

Nome:	<input type="text"/>
CPF:	<input type="text"/>
Email:	<input type="text"/>
<input type="button" value="enviar"/>	