

JavaScript

Programa do curso

- ◆ Aula 1: Introdução e sintaxe
- ◆ Aula 2: Funções, arrays e objetos
- ◆ Aula 3: JavaScript integrado a HTML
- ◆ Aula 4: DOM, seletores e elementos
- ◆ **Aula 5: Eventos**
- ◆ Aula 6: Formulários
- ◆ Aula 7: Ajax
- ◆ Aula 8: Exercício integrador

1. Formulários

Formulários

Um dos principais motivos pelos quais a linguagem de programação JavaScript foi criada foi a necessidade de validar os dados dos formulários diretamente no navegador do usuário.

Usando JavaScript, é possível gerenciar os valores e eventos do formulário e dos elementos dele.

É possível selecionar um formulário utilizando o método `querySelector()`.

```
let form = document.querySelector(idFormulario);
```


Formulários - document.forms

A propriedade **forms** do objeto document retorna uma coleção de objetos tipo **array** contendo todos os formulários existentes no documento HTML, com índice 0 para o primeiro formulário por ordem de exibição, índice 1 para o seguinte e assim sucessivamente.

```
let formulario = document.forms[0];
```

Formulários - document.forms.elements

A propriedade **elements** de cada objeto **form** obtido através de **document.form** retorna uma coleção de objetos tipo array contendo todos os campos existentes no formulário concreto ao qual fazemos referência, com índice 0 para o primeiro elemento por ordem de exibição, índice 1 para o seguinte e assim sucessivamente.

```
// retorna todos os elementos do form
```

```
let elementos = form.elements;
```

```
// retorna o elemento do formulário que está no índice indicado
```

```
let elemento = form.elements[indice];
```

Formulários - Atributos

O objeto formulário tem atributos como **action**, **method**. Como são objetos de JavaScript, eles podem ser acessados da mesma forma que qualquer outro objeto.

Action: Estabelece a URL do documento que vai processar as informações enviadas pelo formulário. **Exemplo:** `form.action;`

Formulários - Atributos

Method: Estabelece o método de HTTP que será usado para enviar os dados. Pode ser get ou post. Exemplo: form.**method**;

Name: Estabelece o nome do formulário. Exemplo: form.**name**;

Formulários - Eventos

Por meio do **evento submit do formulário**, podemos **enviar os dados** carregados no formulário para o documento estabelecido no atributo **action** do formulário.

```
form.onsubmit = function() {  
    // o formulário foi enviado  
}
```

Formulários - Eventos

Com JavaScript, é possível gerenciar o evento **onsubmit()** ou **submit**. Assim como o restante dos eventos, é possível controlar que o evento não seja executado por padrão com o método do evento **preventDefault()**

```
form.onsubmit = function(evento) {  
  evento.preventDefault();  
}
```

Formulários - Elementos - Input

Os elementos **input** têm uma **propriedade** chamada **value** que permite **obter** ou **estabelecer** o **valor** de um campo de texto.

```
// retorna o valor do elemento  
elemento.value;
```

```
// estabelece um novo valor  
elemento.value = valor;
```

Formulários - Elementos - Checkbox

Outro dos elementos do formulário é **checkbox**. Esse objeto tem um **atributo checked** que retorna um **valor booleano** (true/false) através do qual é **possível saber se ele foi selecionado**.

O atributo checked pode ser utilizado tanto para obter o valor do objeto quanto para estabelecer um novo valor.

```
checkbox.checked; // retorna o valor do objeto checked
```

```
checkbox.checked = true/false; // estabelece um novo valor para o elemento.
```


Formulários - Elementos - Hidden

Outro dos elementos do formulário é **input type=hidden**. Esse elemento permite incluir um **valor oculto** no formulário. O usuário não verá esse elemento, mas como faz parte do formulário, ele será enviado junto com o restante dos dados.

Assim como o restante dos elementos do formulário, o input type=hidden tem um atributo **value** que **retorna** ou permite **estabelecer** o valor do campo oculto.

```
elementoOculto.value;
```

```
elementoOculto.value = valorOcultoComoString;
```

Formulários - Elementos - Textarea

Outro dos elementos do formulário é **textarea**. Esse elemento permite **escrever várias linhas**.

Com JavaScript, é possível **obter** e **estabelecer** o **valor** de um objeto textarea por meio do atributo **value**.

```
textarea.value; // obter o valor de textarea
```

```
textarea.value = novoValor; // estabelecer o novo valor de textarea
```

Formulários - Elementos - Eventos

Os elementos do formulário podem gerenciar eventos por meio dos métodos: **onfocus**, **onblur**, **onchange**, **oninput**. Eles também podem ser escritos usando o método **addEventListener(callback)**.

Focus: Este evento é disparado quando um elemento é enfocado

Blur: Este evento é disparado o foco é removido de um elemento

Change: Este evento é disparado quando o valor de um elemento muda

Input: Este evento é disparado quando inserimos dados em um elemento

Formulários - Elementos - Eventos

```
elemento.onfocus = function() {  
  // código que gerencia o focus do elemento  
}
```

```
elemento.onblur = function() {  
  // código que gerencia o blur do elemento  
}
```

```
elemento.onchange = function() {  
  // código que gerencia a mudança de valor  
}
```

```
elemento.oninput = function() {  
  // código que gerencia a entrada de dados em um elemento  
}
```


Formulários - Elementos - Eventos

Por meio do evento **change()**, é possível gerenciar a **mudança de seleção de um elemento select**.

```
elementoSelect. onchange = function() {  
  
  let valor = elementoSelect.value;  
  
  console.log(valor);  
}
```

Formulários - Validação

Por meio do atributo **value**, podemos obter o valor de um elemento. Além disso, é possível utilizar o atributo **length** do objeto string para saber qual é o **comprimento** dele. Dessa forma, é possível **validar se o elemento está vazio**.

```
if (elemento.length === 0) {  
    // elemento vazio  
}
```



Vamos praticar!

Vamos praticar!
Prática 1 - **Formulários**



Obrigado!

Perguntas?
