

SI401

Programação para a Web

2º Semestre - 2024

Programação *Front-End*: *JavaScript*

Unidade 10

Prof. Guilherme Palermo Coelho
guilherme@ft.unicamp.br

Roteiro

- Detecção e Reação a Eventos;
- Caixas de *pop-up*;
- Referências.

EVENTOS

Eventos

- **Eventos** são ocorrências que podem ser detectadas pela linguagem *JavaScript*;
 - Clique do *mouse*;
 - Uma página ou uma imagem de uma página sendo carregada;
 - O *mouse* está sobre uma determinada região da página;
 - Um determinado campo de entrada em um **formulário HTML** foi selecionado;
 - O conteúdo de um formulário HTML deve ser submetido para o servidor;
 - Uma determinada tecla foi pressionada; ...

Eventos

- Quando bem explorada, a detecção da ocorrência de eventos permite a criação de *páginas dinâmicas*;
- Em *JavaScript*, a detecção de eventos geralmente está associada à **atribuição de uma função a um atributo** do elemento HTML que se deseja monitorar;
 - Quando ocorre o evento no elemento em questão, a função é executada.
- Aqui serão vistos apenas alguns exemplos de eventos que podem ser detectados e tratados via *JavaScript*:
 - Lista completa com todos os eventos e os respectivos atributos dos elementos HTML (que devem ter funções *JavaScript* associadas):
http://w3schools.com/jsref/dom_obj_event.asp

Eventos – Clique de Mouse

- Para invocar uma dada função *JavaScript* sempre que um determinado elemento HTML for “***clicado***”, deve-se atribuir o nome da função ao atributo “**onclick**” do elemento:
 - `onclick="nome_da_funcao()";`

Eventos – Clique de Mouse

```
<html>

<head>
<script type="text/javascript">
function displayDate()
{
document.getElementById("demo").innerHTML=Date();
}
</script>
</head>

<body>

<h1>Minha Página</h1>

<p id="demo"></p>

<button type="button"
onclick="displayDate()">Mostrar Data</button>

</body>
</html>
```

Minha Página

Mostrar Data

Minha Página

Wed Mar 21 2012 15:13:02 GMT-0300 (E. South America
Standard Time)

Mostrar Data

Eventos – Entrada/Saída de página

- Também é possível invocar funções *JavaScript* quando o usuário **entra** ou **sai** de uma página web:
 - **Entrada:** atributo “**onload**” deve ter o nome da função atribuído a ele;
 - **Saída:** atributo “**onunload**” deve ter o nome da função atribuído a ele;
 - Define-se este atributo para o elemento **<body>** da página;
 - Ex.: **<body onload=“função_JavaScript()”> ... </body>**

Eventos – Mouse sobre elemento

- Basta atribuir o nome da função *JavaScript* ao atributo “**onMouseOver**” do elemento HTML:

```
<html>

<head>
<script type="text/javascript">
function displayDate()
{
document.getElementById("demo").innerHTML=Date();
}
</script>
</head>

<body>

<h1 onMouseOver="displayDate()">Passe o mouse!
</h1>

<p id="demo"></p>

</body>
</html>
```

Passe o mouse!

Passe o mouse!

Wed Mar 21 2012 15:39:22 GMT-0300 (E. South America
Standard Time)

CAIXAS DE *POP-UP*

Caixas de *Pop-Up*

- *JavaScript* possui três tipos de caixas de *pop-up*, usadas para interagir com o usuário:
 - **Caixa de alerta;**
 - **Caixa de confirmação;**
 - **Caixa para entrada de informação** (caixa de *prompt*);
- As **caixas de alerta** são usadas para garantir que o usuário recebeu a informação que se deseja passar:
 - O usuário é obrigado a clicar em “OK” para continuar;
 - Sintaxe: “**alert**("texto qualquer");”

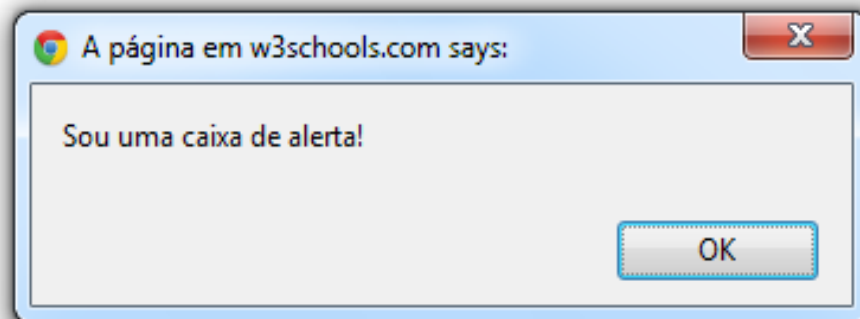
Caixas de *Pop-Up*

```
<html>
<head>
<script type="text/javascript">
function show_alert()
{
alert("Sou uma caixa de alerta!");
}
</script>
</head>
<body>

<input type="button" onclick="show_alert()"
value="Mostrar caixa de alerta" />

</body>
</html>
```

Mostrar caixa de alerta



Caixas de *Pop-Up*

- As **caixas de confirmação** são usadas quando se deseja que o usuário verifique ou aceite algo:
 - O usuário deve clicar em “**OK**” ou “**CANCELAR**” para prosseguir;
 - Caso clique em “OK”, a caixa retorna o valor **true**;
 - Caso clique em “CANCELAR”, a caixa retorna **false**;
 - Sintaxe: “**valor = confirm("algum texto");**”

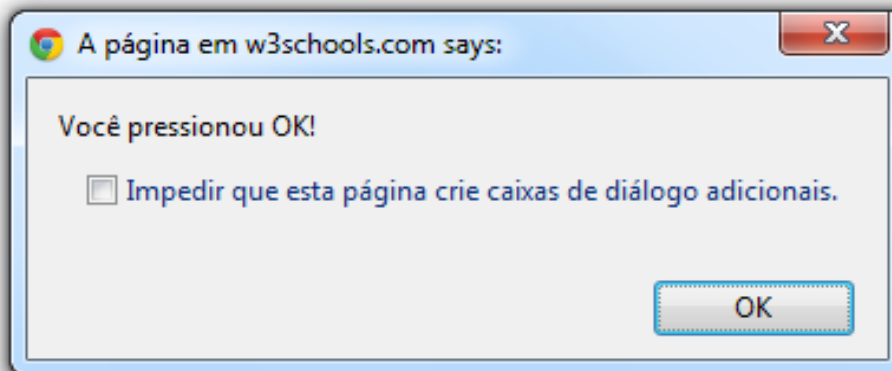
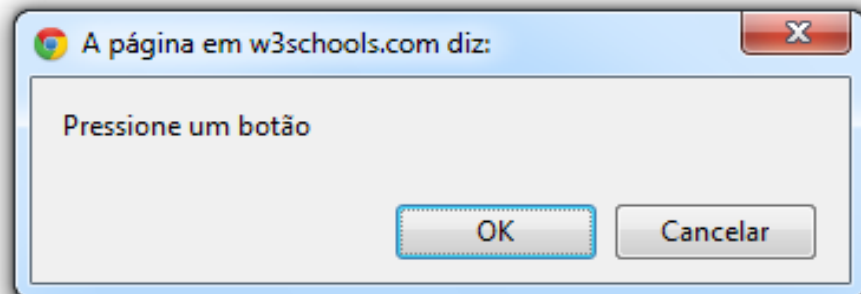
Caixas de *Pop-Up*

```
<html>
<head>
<script type="text/javascript">
function show_confirm()
{
var r=confirm("Pressione um botão");
if (r==true)
{
alert("Você pressionou OK!");
}
else
{
alert("Você pressionou Cancel!");
}
}
</script>
</head>
<body>

<input type="button" onclick="show_confirm()"
value="Caixa de confirmação" />

</body>
</html>
```

Caixa de confirmação



Caixas de *Pop-Up*

- Já as **caixas de prompt** são utilizadas para que o usuário forneça um valor;
 - O usuário deve clicar em “OK” ou “CANCELAR” para prosseguir;
 - Caso clique em “OK”, a caixa retorna o **valor passado**;
 - Caso clique em “CANCELAR”, a caixa retorna **null**;
 - Sintaxe: “**valor = prompt**(“algum texto”,“valor padrão”);”

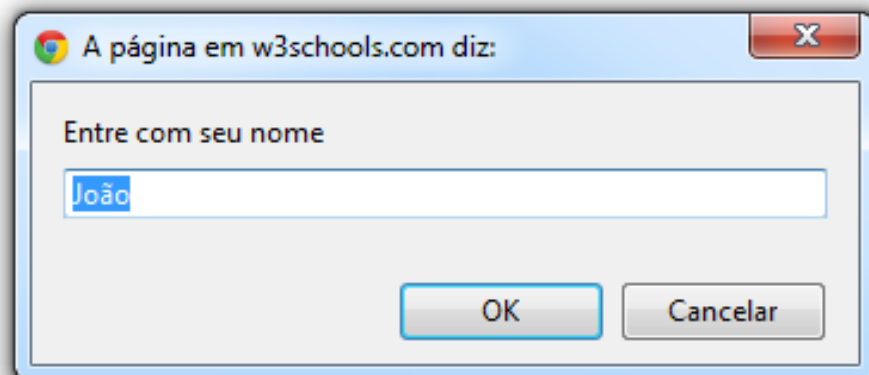
Caixas de *Pop-Up*

```
<html>
<head>
<script type="text/javascript">
function show_prompt()
{
var name=prompt("Entre com seu nome","João");
if (name!=null && name!="")
{
document.write("<p>Olá " + name + "! Tudo bem?
</p>");
}
}
</script>
</head>
<body>

<input type="button" onclick="show_prompt()"
value="Mostrar caixa de prompt" />

</body>
</html>
```

Mostrar caixa de prompt



EXERCÍCIO SUGERIDO 1

Exercício Sugerido 1

- Crie um documento HTML que exiba três imagens do “Sol” lado a lado.
 - Quando o usuário clicar em uma destas imagens, ela deve ser dinamicamente substituída por uma imagem da “Lua”. Se estiver sendo exibida a imagem da “Lua”, um clique deve restaurar a imagem do “Sol”.
 - Além disso, quando o usuário passar o mouse sobre uma destas imagens, deve ser exibido, abaixo das imagens, um texto explicativo sobre qual imagem está sendo exibida (“Sol” ou “Lua”). Armazene os scripts em arquivos externos.
- Para esta atividade será necessário utilizar o método **getElementById(“id”)** de **document**, que retorna um *elemento* do documento.
 - Para maiores informações sobre métodos que podem ser usados neste elemento, consulte: http://www.w3schools.com/jsref/dom_obj_all.asp

REFERÊNCIAS

Referências

- [1] W3Schools JavaScript Tutorial:

<http://www.w3schools.com/js/default.asp>

- [2] Standard ECMA-262: ECMAScript Language Specification

<http://www.ecma-international.org/publications/standards/Ecma-262.htm>

- [3] Mozilla Developer Network – JavaScript:

<http://developer.mozilla.org/pt-BR/JavaScript>