

# Laboratório de Programação Web

Carolina Ribeiro Xavier ([carolina.xavier@ice.ufjf.br](mailto:carolina.xavier@ice.ufjf.br))

UFJF

# Eventos

# Javascript

## Eventos

- Com os eventos, abordamos a parte do Javascript mais específicas da linguagem.
- Os eventos Javascript, associados às funções, aos métodos e às propriedades, abrem uma grande porta para uma verdadeira interatividade das páginas.

- Em HTML clássico, há um evento bem conhecido.
- É o clique do mouse sobre um link que vai abrir outra página Web.
- Infelizmente, é praticamente o único.
- Felizmente, o Javascript vai acrescentar uma boa dezena.
- Os eventos Javascript, associados as funções, aos métodos e aos formulários, abrem uma grande porta para uma verdadeira interatividade das páginas.

- **Click:** Quando o usuário clica sobre um botão, um link ou outro elementos.
- **Load:** Quando a página é carregada pelo browser.
- **Unload:** Quando o usuário sai da página.
- **MouseOver:** Quando o usuário coloca a seta do mouse sobre um elemento.
- **MouseOut:** Quando a seta do mouse sai de um outro elemento.

- **Focus:** Quando um elemento de formulário tem o foco, isto é, está ativo.
- **Blur:** Quando um elemento de formulário perde o foco, isto é, quando deixa de estar ativo.
- **Change:** Quando o valor de um campo de formulário é modificado.
- **Select:** Quando o usuário seleciona um campo dentro de elemento de formulário.
- **Submit:** Quando o usuário clica sobre o botão Submit para enviar um formulário.

# Javascript

## Gerenciamento de Eventos

- Para o gerenciamento dos eventos ser eficaz, é necessário que a estes eventos sejam associados as ações previstas pelo programador.
- A sintaxe é:  
`onevento="function()"`
- Por exemplo:  
`onClick="alert('Hello World!!')"`
- Ao clicar no elemento, a mensagem será exibida.
- A definição do evento deve ficar posicionada como um atributo de um elemento HTML.
- A seguir serão exemplificados os eventos e essa ideia ficará mais facilmente compreendida.

# Javascript

## Click

```
1 <form>
2 <input type="button" value="Clique Aqui!" onClick="alert('Usuário clicou no botão!!')">
3 </form>
```

- Colocamos um botão na página HTML normalmente.
- A chamada onClick fica posicionada como um atributo do input.



# Javascript

## onLoad e onUnload

```
1 <html>
2 <head>
3 <script language='Javascript'>
4 function bemVindo() {
5     alert("Bem vindo a esta página");
6 }
7 function adeus() {
8     alert("Adeus");
9 }
10 </script>
11 </head>
12 <body onLoad='bemVindo()' onUnload='adeus() '>
13 Aqui vai o HTML normalmente.
14 </body>
15 </html>
```

# Javascript

## onLoad e onUnload

- Até agora só tínhamos utilizado a função `alert()` na chamada do evento.
- Nesse exemplo estamos usando duas funções definidas pelo usuário, `bemVindo()` e `adeus()`
- Quando a página é carregada `bemVindo()` é executada.
- Quando a página fecha `adeus()` é executada.
- Como o evento de carregamento é associado à página, posicionamos sua chamada na tag `body`.

# Javascript

## onFocus

- O evento onFocus refere-se a quando um objeto recebe o foco, ou seja, quando se torna ativo.
- Os objetos podem ganhar o foco com o mouse ou com a tecla tab.

```
1 <form>
2 <input type="text" onFocus="alert('Recebeu o foco ')">
3 </form>
```

# Javascript

## onFocus

- O evento onBlur refere-se a quando um objeto perde o foco.

```
1 <form>
2 <input type="text" onBlur="alert('Perdeu o foco ')">
3 </form>
```

# Javascript

## onChange

- Este evento é bastante semelhante ao do onBlur mas com uma pequena diferença.
- Não só o elemento deve ter perdido o foco mas também o seu conteúdo deve ter sido alterado pelo usuário.

```
1 <form>
2 <input type="text" value="Mude meu texto" onChange="alert('Mudou o texto ')">
3 <br>
4 <select type="text" onChange="alert('Mudou a selecao ')">
5 <option>Mude meu valor</option>
6 <option>Valor 1</option>
7 <option>Valor 2</option>
8 <option>Valor 3</option>
9 </select>
10 </form>
```

# Javascript

## onSelect

- Este evento é acionado quando o usuário seleciona uma parte do elemento.

```
1 <form>
2 <input type="text" value="Selecione meu texto" onSelect="alert('Selecionou o texto ')">
3 </form>
```

# Javascript

## onmouseover e onmouseout

- O evento onmouseover está relacionado a uma ação que será executada quando o mouse passar sobre um elemento..
- Este elemento é bastante prático para enriquecer o layout de uma página.
- O evento onmouseout, geralmente associado um onmouseover, executa-se quando o cursor sai da zona do elemento.

# Javascript

## onmouseover e onmouseout

- Utilizando onmouseover / Out em uma imagem:

```
1 
```



# Javascript

## onmouseover e onmouseout

- Utilizando onmouseover / Out em uma imagem:

```
1 
```

# Javascript

## onmouseover e onmouseout

- Uma utilização dos eventos mouseOver e mouseOut é com o objetivo de trocas de imagens.
- Podemos fazer isso facilmente com o código a seguir:

```
1 <html>
2 <head>
3 <script language="Javascript">
4 function mudaImg() {
5     document.images["copa"].src="grafite.png";
6 }
7 function voltaImg() {
8     document.images["copa"].src="adriano.png";
9 }
10 </script>
11 </head>
12 <body>
13 
14 </html>
```

# Laboratório de Programação Web

Carolina Ribeiro Xavier ([carolina.xavier@ice.ufjf.br](mailto:carolina.xavier@ice.ufjf.br))

UFJF