

# JavaScript DOM

## Programação para Internet I

**Prof. Msc. Felipe Leivas Teixeira**

Versão 1.0

- 1 JavaScript DOM
- 2 Métodos do Objeto window
- 3 Encontrando Elementos HTML
- 4 Alterando Elementos HTML
- 5 Adicionando e Excluindo Elementos HTML
- 6 Referências

- 1 JavaScript DOM
- 2 Métodos do Objeto window
- 3 Encontrando Elementos HTML
- 4 Alterando Elementos HTML
- 5 Adicionando e Excluindo Elementos HTML
- 6 Referências

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- DOM (*Document Object Model*) - Modelo de Objetos para Documentos

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- DOM (*Document Object Model*) - Modelo de Objetos para Documentos
- O DOM é uma interface de plataforma e linguagem neutra que permite que programas e scripts acessem e atualizem dinamicamente o conteúdo, a estrutura e o estilo de um documento

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- DOM (*Document Object Model*) - Modelo de Objetos para Documentos
- O DOM é uma interface de plataforma e linguagem neutra que permite que programas e scripts acessem e atualizem dinamicamente o conteúdo, a estrutura e o estilo de um documento
- DOM é um padrão de como obter, alterar, adicionar ou excluir elementos HTML

- DOM (*Document Object Model*) - Modelo de Objetos para Documentos
- O DOM é uma interface de plataforma e linguagem neutra que permite que programas e scripts acessem e atualizem dinamicamente o conteúdo, a estrutura e o estilo de um documento
- DOM é um padrão de como obter, alterar, adicionar ou excluir elementos HTML
- O DOM é uma estrutura hierarquica onde o objeto **window** está no topo da hierarquia

- DOM (*Document Object Model*) - Modelo de Objetos para Documentos
- O DOM é uma interface de plataforma e linguagem neutra que permite que programas e scripts acessem e atualizem dinamicamente o conteúdo, a estrutura e o estilo de um documento
- DOM é um padrão de como obter, alterar, adicionar ou excluir elementos HTML
- O DOM é uma estrutura hierarquica onde o objeto **window** está no topo da hierarquia
- É o objeto window que contém o objeto **document** que representa sua página web





- 1 JavaScript DOM
- 2 Métodos do Objeto window
- 3 Encontrando Elementos HTML
- 4 Alterando Elementos HTML
- 5 Adicionando e Excluindo Elementos HTML
- 6 Referências

- O objeto window contém vários métodos, mas os principais métodos são:

- O objeto window contém vários métodos, mas os principais métodos são:
  - **alert()** - Cria uma caixa de alerta que exibe uma mensagem para o usuário

- O objeto window contém vários métodos, mas os principais métodos são:
  - **alert()** - Cria uma caixa de alerta que exibe uma mensagem para o usuário
  - **confirm()** - Cria uma caixa de confirmação que solicita um retorno booleano para o usuário

- O objeto window contém vários métodos, mas os principais métodos são:
  - **alert()** - Cria uma caixa de alerta que exibe uma mensagem para o usuário
  - **confirm()** - Cria uma caixa de confirmação que solicita um retorno booleano para o usuário
  - **prompt()** - Cria uma caixa onde o usuário pode digitar uma informação

- **alert** é uma caixa de diálogo contendo uma informação

- **alert** é uma caixa de diálogo contendo uma informação
- Aparecerá na interface do usuário

- **alert** é uma caixa de diálogo contendo uma informação
- Aparecerá na interface do usuário
- A sintaxe do alert é a seguinte:

```
1 window.alert(<mensagem>);  
2 ou  
3 alert(<mensagem>);
```



# Métodos do Objeto window

JavaScript  
DOM

Felipe  
Teixeira

JavaScript  
DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- **alert** é uma caixa de diálogo contendo uma informação
- Aparecerá na interface do usuário
- A sintaxe do alert é a seguinte:

```
1 window.alert(<mensagem>);  
2 ou  
3 alert(<mensagem>);
```

- No lugar de <mensagem> podemos escrever uma mensagem entre " " e passar uma ou mais variáveis

- **alert** é uma caixa de diálogo contendo uma informação
- Aparecerá na interface do usuário
- A sintaxe do alert é a seguinte:

```
1 window.alert(<mensagem>);
2 ou
3 alert(<mensagem>);
```

- No lugar de <mensagem> podemos escrever uma mensagem entre "" e passar uma ou mais variáveis

```
1 var idade=50;
2 alert("A idade e: " + idade);
3 alert("Obrigado");
```

- **confirm** é uma caixa de diálogo que contém dois botões, o OK e o Cancelar

- **confirm** é uma caixa de diálogo que contém dois botões, o OK e o Cancelar
- Se clicar no OK retorna o true como valor, e Cancelar retorna false

- **confirm** é uma caixa de diálogo que contém dois botões, o OK e o Cancelar
- Se clicar no OK retorna o true como valor, e Cancelar retorna false
- A sintaxe do confirm é a seguinte:

```
1 window.confirm(<mensagem>);  
2 ou  
3 confirm(<mensagem>);
```

- **confirm** é uma caixa de diálogo que contém dois botões, o OK e o Cancelar
- Se clicar no OK retorna o true como valor, e Cancelar retorna false
- A sintaxe do confirm é a seguinte:

```
1 | window.confirm(<mensagem>);
2 | ou
3 | confirm(<mensagem>);
```

- No lugar de <mensagem>, da mesma forma que o alert, podemos escrever uma mensagem entre " e passar uma ou mais variáveis

```
1 | var resultado = confirm("Deseja sair?");
```

- **prompt** é uma caixa de entrada de uma String, onde o usuário pode digitar uma informação

- **prompt** é uma caixa de entrada de uma String, onde o usuário pode digitar uma informação
- O retorno do prompt é sempre uma String, caso o valor digitado seja um número, devemos fazer a conversão de string para número com as funções **parseInt** (para número inteiro) ou **parseFloat** (para número com casas decimais)



- **prompt** é uma caixa de entrada de uma String, onde o usuário pode digitar uma informação
- O retorno do prompt é sempre uma String, caso o valor digitado seja um número, devemos fazer a conversão de string para número com as funções **parseInt** (para número inteiro) ou **parseFloat** (para número com casas decimais)
- A sintaxe do prompt é a seguinte:

```
1 | window.prompt(<mensagem>,<valor inicial>);
2 | ou
3 | prompt(<mensagem>,<valor inicia>);
```

- **prompt** é uma caixa de entrada de uma String, onde o usuário pode digitar uma informação
- O retorno do prompt é sempre uma String, caso o valor digitado seja um número, devemos fazer a conversão de string para número com as funções **parseInt** (para número inteiro) ou **parseFloat** (para número com casas decimais)

- A sintaxe do prompt é a seguinte:

```
1 | window.prompt(<mensagem>,<valor inicial>);
2 | ou
3 | prompt(<mensagem>,<valor inicia>);
```

- No lugar de <mensagem>, da mesma forma que o alert e o confirm, podemos escrever uma mensagem entre " " e passar uma ou mais variáveis

- **prompt** é uma caixa de entrada de uma String, onde o usuário pode digitar uma informação
- O retorno do prompt é sempre uma String, caso o valor digitado seja um número, devemos fazer a conversão de string para número com as funções **parseInt** (para número inteiro) ou **parseFloat** (para número com casas decimais)
- A sintaxe do prompt é a seguinte:

```
1 | window.prompt(<mensagem>,<valor inicial>);
2 | ou
3 | prompt(<mensagem>,<valor inicia>);
```

- No lugar de <mensagem>, da mesma forma que o alert e o confirm, podemos escrever uma mensagem entre " " e passar uma ou mais variáveis
- No lugar de <valor inicia> podemos que adicionar um valor inicial, este campo é opcional

```
1 | var nome = prompt("Informe seu nome:");
2 | var cpf = prompt("Digite seu CPF","000.000.000-00");
```

# EXEMPLO MÉTODOS DO OBJETO WINDOW

- 1 JavaScript DOM
- 2 Métodos do Objeto window
- 3 Encontrando Elementos HTML**
- 4 Alterando Elementos HTML
- 5 Adicionando e Excluindo Elementos HTML
- 6 Referências

- Como dito, com o DOM podemos obter, alterar, adicionar ou excluir elementos HTML

# Encontrando Elementos HTML

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

## Encontrando Elementos HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- Como dito, com o DOM podemos obter, alterar, adicionar ou excluir elementos HTML
- Existem alguns métodos do objeto document, que permitem que acessemos um determinado elemento do HTML

- Como dito, com o DOM podemos obter, alterar, adicionar ou excluir elementos HTML
- Existem alguns métodos do objeto document, que permitem que acessemos um determinado elemento do HTML
- Os métodos seletores que permitem acessar um determinado elemento são:



- Como dito, com o DOM podemos obter, alterar, adicionar ou excluir elementos HTML
- Existem alguns métodos do objeto document, que permitem que acessemos um determinado elemento do HTML
- Os métodos seletores que permitem acessar um determinado elemento são:
  - **document.getElementById(id)** - acessa um elemento HTML através do id definido a ele

- Como dito, com o DOM podemos obter, alterar, adicionar ou excluir elementos HTML
- Existem alguns métodos do objeto document, que permitem que acessemos um determinado elemento do HTML
- Os métodos seletores que permitem acessar um determinado elemento são:
  - **document.getElementById(id)** - acessa um elemento HTML através do id definido a ele
  - **document.getElementsByTagName(name)** - acessa um elemento HTML pelo nome da tag

- Como dito, com o DOM podemos obter, alterar, adicionar ou excluir elementos HTML
- Existem alguns métodos do objeto document, que permitem que acessemos um determinado elemento do HTML
- Os métodos seletores que permitem acessar um determinado elemento são:
  - **document.getElementById(id)** - acessa um elemento HTML através do id definido a ele
  - **document.getElementsByTagName(name)** - acessa um elemento HTML pelo nome da tag
  - **document.getElementsByClassName(name)** - acessa um elemento HTML pelo nome da classe

- Como dito, com o DOM podemos obter, alterar, adicionar ou excluir elementos HTML
- Existem alguns métodos do objeto document, que permitem que acessemos um determinado elemento do HTML
- Os métodos seletores que permitem acessar um determinado elemento são:
  - **document.getElementById(id)** - acessa um elemento HTML através do id definido a ele
  - **document.getElementsByTagName(name)** - acessa um elemento HTML pelo nome da tag
  - **document.getElementsByClassName(name)** - acessa um elemento HTML pelo nome da classe
  - **document.querySelector(seletor)** - acessa um elemento HTML utilizando a sintaxe do CSS

# EXEMPLO ENCONTRANDO ELEMENTOS HTML

- 1 JavaScript DOM
- 2 Métodos do Objeto window
- 3 Encontrando Elementos HTML
- 4 Alterando Elementos HTML**
- 5 Adicionando e Excluindo Elementos HTML
- 6 Referências

# Alterando Elementos HTML

JavaScript  
DOM

Felipe  
Teixeira

JavaScript  
DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

**Alterando  
Elementos  
HTML**

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- A partir do momento que acessamos um determinado elemento podemos alterá-los

# Alterando Elementos HTML

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

**Alterando  
Elementos  
HTML**

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- A partir do momento que acessamos um determinado elemento podemos alterá-los
- Existem algumas propriedades e métodos, que permitem que alteremos um determinado elemento do HTML



# Alterando Elementos HTML

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- A partir do momento que acessamos um determinado elemento podemos alterá-los
- Existem algumas propriedades e métodos, que permitem que alteremos um determinado elemento do HTML
- As principais propriedades e métodos que permitem alterar um determinado elemento são:

# Alterando Elementos HTML

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

## Alterando Elementos HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- A partir do momento que acessamos um determinado elemento podemos alterá-los
- Existem algumas propriedades e métodos, que permitem que alteremos um determinado elemento do HTML
- As principais propriedades e métodos que permitem alterar um determinado elemento são:
  - **<elemento>.innerHTML** - utilizado para obter ou substituir o conteúdo de elementos HTML

- A partir do momento que acessamos um determinado elemento podemos alterá-los
- Existem algumas propriedades e métodos, que permitem que alteremos um determinado elemento do HTML
- As principais propriedades e métodos que permitem alterar um determinado elemento são:
  - **<elemento>.innerHTML** - utilizado para obter ou substituir o conteúdo de elementos HTML
  - **<elemento>.<attribute>** - altera o valor de um atributo, no nome do atributo deve vir no lugar de <attribute>

- A partir do momento que acessamos um determinado elemento podemos alterá-los
- Existem algumas propriedades e métodos, que permitem que alteremos um determinado elemento do HTML
- As principais propriedades e métodos que permitem alterar um determinado elemento são:
  - **<elemento>.innerHTML** - utilizado para obter ou substituir o conteúdo de elementos HTML
  - **<elemento>.<attribute>** - altera o valor de um atributo, no nome do atributo deve vir no lugar de <attribute>
  - **<elemento>.setAttribute(attribute, value)** - um método que permite alterar o valor de um atributo

- A partir do momento que acessamos um determinado elemento podemos alterá-los
- Existem algumas propriedades e métodos, que permitem que alteremos um determinado elemento do HTML
- As principais propriedades e métodos que permitem alterar um determinado elemento são:
  - **<elemento>.innerHTML** - utilizado para obter ou substituir o conteúdo de elementos HTML
  - **<elemento>.<attribute>** - altera o valor de um atributo, no nome do atributo deve vir no lugar de <attribute>
  - **<elemento>.setAttribute(attribute, value)** - um método que permite alterar o valor de um atributo
  - **<elemento>.style.<property>** - altera o valor de uma propriedade CSS, no nome da propriedade deve vir no lugar de <property>

# EXEMPLO ALTERANDO ELEMENTOS HTML

- 1 JavaScript DOM
- 2 Métodos do Objeto window
- 3 Encontrando Elementos HTML
- 4 Alterando Elementos HTML
- 5 Adicionando e Excluindo Elementos HTML**
- 6 Referências

# Adicionando e Excluindo Elementos HTML

JavaScript  
DOM

Felipe  
Teixeira

JavaScript  
DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

**Adicionando  
e Excluindo  
Elementos  
HTML**

Referências

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML



# Adicionando e Excluindo Elementos HTML

JavaScript  
DOM

Felipe  
Teixeira

JavaScript  
DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

**Adicionando  
e Excluindo  
Elementos  
HTML**

Referências

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML
- Existem alguns métodos, que permitem criar ou remover um elemento do HTML

# Adicionando e Excluindo Elementos HTML

## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

**Adicionando  
e Excluindo  
Elementos  
HTML**

Referências

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML
- Existem alguns métodos, que permitem criar ou remover um elemento do HTML
- Os principais métodos que permitem criar ou remover elementos HTML são:

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML
- Existem alguns métodos, que permitem criar ou remover um elemento do HTML
- Os principais métodos que permitem criar ou remover elementos HTML são:
  - **document.createElement(elemento)** - utilizado para criar um elemento HTML

# Adicionando e Excluindo Elementos HTML

JavaScript  
DOM

Felipe  
Teixeira

JavaScript  
DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

Referências

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML
- Existem alguns métodos, que permitem criar ou remover um elemento do HTML
- Os principais métodos que permitem criar ou remover elementos HTML são:
  - **document.createElement(elemento)** - utilizado para criar um elemento HTML
  - **document.removeChild(elemento)** - utilizado para remover um elemento HTML

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML
- Existem alguns métodos, que permitem criar ou remover um elemento do HTML
- Os principais métodos que permitem criar ou remover elementos HTML são:
  - **document.createElement(elemento)** - utilizado para criar um elemento HTML
  - **document.removeChild(elemento)** - utilizado para remover um elemento HTML
  - **document.appendChild(elemento)** - adiciona um elemento HTML

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML
- Existem alguns métodos, que permitem criar ou remover um elemento do HTML
- Os principais métodos que permitem criar ou remover elementos HTML são:
  - **document.createElement(elemento)** - utilizado para criar um elemento HTML
  - **document.removeChild(elemento)** - utilizado para remover um elemento HTML
  - **document.appendChild(elemento)** - adiciona um elemento HTML
  - **document.replaceChild(novo, velho)** - substitui um elemento HTML pelo outro

- Vimos como encontrar e alterar elementos, existem também a possibilidade de criar ou remover um elemento HTML
- Existem alguns métodos, que permitem criar ou remover um elemento do HTML
- Os principais métodos que permitem criar ou remover elementos HTML são:
  - **document.createElement(elemento)** - utilizado para criar um elemento HTML
  - **document.removeChild(elemento)** - utilizado para remover um elemento HTML
  - **document.appendChild(elemento)** - adiciona um elemento HTML
  - **document.replaceChild(novo, velho)** - substitui um elemento HTML pelo outro
  - **document.write(texto)** - escreve dentro do documento HTML

# EXEMPLO ADICIONANDO E EXCLUINDO ELEMENTOS HTML



## JavaScript DOM

Felipe  
Teixeira

## JavaScript DOM

Métodos do  
Objeto  
window

Encontrando  
Elementos  
HTML

Alterando  
Elementos  
HTML

Adicionando  
e Excluindo  
Elementos  
HTML

## Referências

- [https://www.w3schools.com/js/js\\_htmldom.asp](https://www.w3schools.com/js/js_htmldom.asp)
- [https://www.w3schools.com/js/js\\_htmldom\\_elements.asp](https://www.w3schools.com/js/js_htmldom_elements.asp)
- [https://www.w3schools.com/js/js\\_htmldom\\_html.asp](https://www.w3schools.com/js/js_htmldom_html.asp)
- [https://www.w3schools.com/js/js\\_htmldom\\_css.asp](https://www.w3schools.com/js/js_htmldom_css.asp)

# JavaScript DOM

## Programação para Internet I

**Prof. Msc. Felipe Leivas Teixeira**

Versão 1.0