



Web

DOM: *Document
Object Model*

Catarina Oliveira

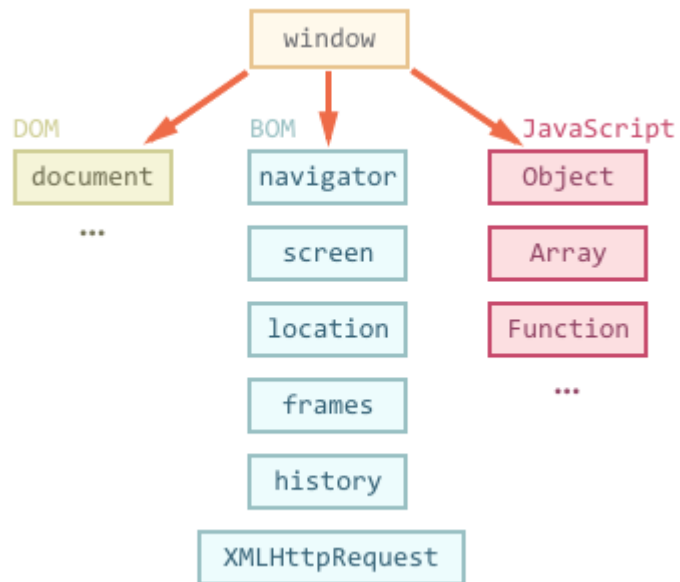
DCT DEPARTAMENTO CIÊNCIA
E TECNOLOGIA

CONTEÚDO

1. Contexto
2. Integração de JavaScript em HTML
3. Estrutura
4. Exemplos métodos DOM
 1. getElementById(id)
 2. getElementsByTagName(nome)
 3. getElementsByClassName(nome)
 4. querySelector(selector)
 5. querySelectorAll(selector)
 6. createElement(elemento)
 7. createTextNode(texto)
 8. appendChild(elemento)
 9. replaceChild(novo, antigo)
 10. removeChild(novo, antigo)
 11. Métodos de alteração de elementos HTML
5. Tipos de eventos
6. Exemplos
 1. onclick
 2. Verificar passwords iguais

Contexto

- **DOM:** *Document Object Model*
- Standard W3C
- Define um meio uniforme para aceder a documentos
- Dividido em três partes
 - Core
 - XML
 - HTML
- JavaScript pode aceder e alterar todos os elementos do **documento** HTML e do *Browser Object Model (BOM)*



Integração de JavaScript em HTML

- Local

```
<html>
  <head>
    ...
  </head>
  <body>
    <script>
      alert("Ola")
    </script>
  </body>
</html>
```

- Externa

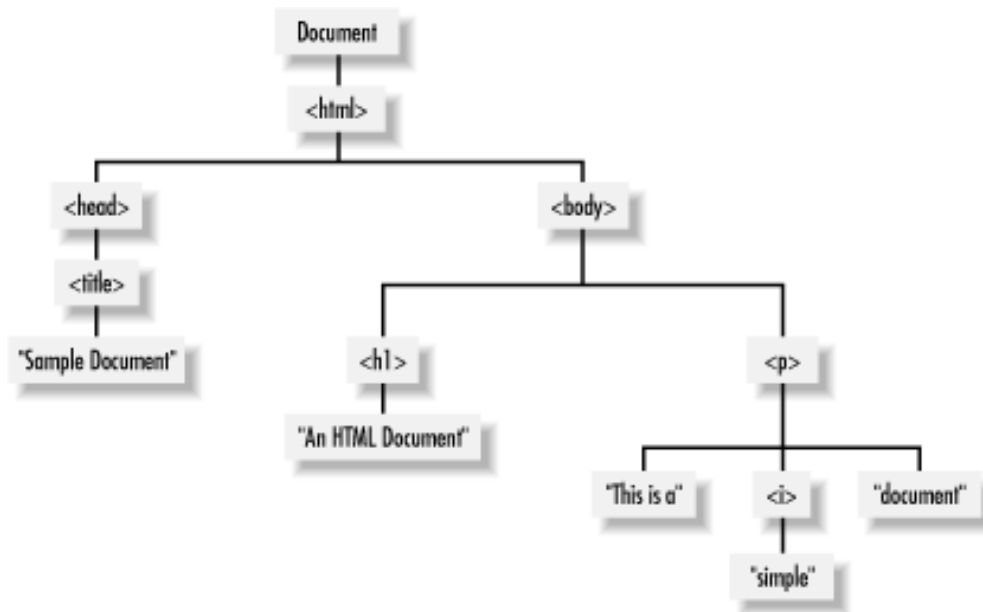
```
<html>
  <head>
    ...
  </head>
  <body>
    <script src="caminho_para_ficheiro.js"></script>
  </body>
</html>
```

Benefícios:

- Separação HTML / JavaScript
- Legibilidade e manutenção de código HTML e JavaScript
- Ficheiros JavaScript guardados em cache tornam mais rápido o carregamento das páginas

Estrutura

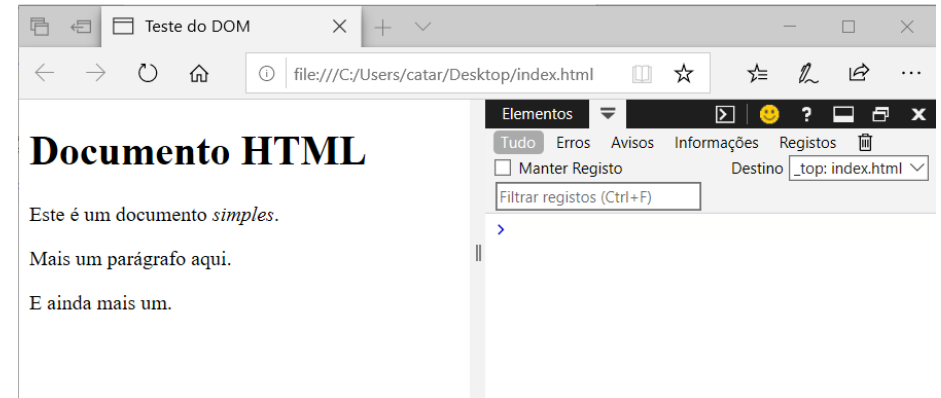
- Árvore de objetos de diferentes tipos:
 - Document**: documento (fig: Document)
 - Node**: nó genérico do documento HTML (fig: <title>, “Sample Document”, <p>, ...)
 - NodeList**: lista dos nós genéricos
 - Element**: elementos criados por anotações HTML (fig: <html>, <head>, <title>, <body>, <h1>, ...)
 - Text**: textos em elementos (fig: “Sample Document”, “An HTML Document”, “This is a”, ...)



- A árvore é composta por **nós**
- A raiz é o nó Document (não tem pai)
- Todos os outros nós têm um nó **pai**
 - Document** é pai de <html>
 - <h1>** é pai de “An HTML Document”
- Os nós podem ter **filhos**
 - “Sample Document” é filho de <title>
 - “This is a” é filho de <p>

Exemplos DOM: ponto de partida

```
index.html x
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10       <h1>Documento HTML</h1>
11       <p>Este é um documento <i>simples</i>.</p>
12       <p id="p2">Mais um parágrafo aqui.</p>
13       <p class="cl">E ainda mais um.</p>
14     </div>
15     <script type="text/javascript">
16       alert("Ola")
17     </script>
18   </body>
19 </html>
```



Métodos de acesso a elementos: getElementById(id)

```
index.html x
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10       <h1>Documento HTML</h1>
11       <p>Este é um documento <i>simples</i>.</p>
12       <p id="p2">Mais um parágrafo aqui.</p>
13       <p class="cl">E ainda mais um.</p>
14     </div>
15     <script type="text/javascript">
16       alert("Ola")
17     </script>
18   </body>
19 </html>
```



getElementById(id)

obtém referência para um elemento HTML a partir do seu atributo id

Métodos de acesso a elementos: `getElementsByTagName(nome)`

```
index.html x
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <h1>Documento HTML</h1>
11      <p>Este é um documento <i>simples</i>.</p>
12      <p id="p2">Mais um parágrafo aqui.</p>
13      <p class="c1">E ainda mais um.</p>
14    </div>
15    <script type="text/javascript">
16      alert("Ola")
17    </script>
18  </body>
19 </html>
```



`getElementsByTagName(nome)`

obtém referência para um conjunto de elementos HTML a partir do nome da sua tag

Métodos de acesso a elementos: `getElementsByClassName(nome)`

```

1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta http-equiv="content-type"
5        content="text/html; charset=utf-8">
6      <title>Teste do DOM</title>
7    </head>
8    <body>
9      <div id="corpo">
10        <h1>Documento HTML</h1>
11        <p>Este é um documento <i>simples</i>.</p>
12        <p id="p2">Mais um parágrafo aqui.</p>
13        <p class="c1">E ainda mais um.</p>
14      </div>
15      <script type="text/javascript">
16        alert("Ola")
17      </script>
18    </body>
19  </html>

```



`getElementsByClassName(nome)`

obtém referência para um conjunto de elementos HTML a partir do seu atributo class

Métodos de acesso a elementos: `querySelector(selector)`

```
index.html x
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <h1>Documento HTML</h1>
11      <p>Este é um documento <i>simples</i>.</p>
12      <p id="p2">Mais um parágrafo aqui.</p>
13      <p class="cl">E ainda mais um.</p>
14    </div>
15    <script type="text/javascript">
16      alert("Ola")
17    </script>
18  </body>
19 </html>
```



`querySelector(selector)`

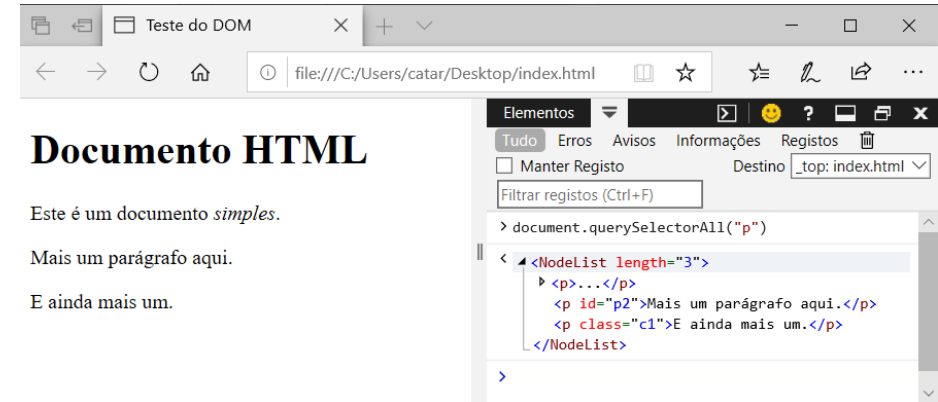
obtém referência para um elemento HTML a partir de um seletor CSS

Métodos de acesso a elementos: `querySelectorAll(selector)`

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <h1>Documento HTML</h1>
11      <p>Este é um documento <i>simples</i>.</p>
12      <p id="p2">Mais um parágrafo aqui.</p>
13      <p class="cl">E ainda mais um.</p>
14    </div>
15    <script type="text/javascript">
16      alert("Ola")
17    </script>
18  </body>
19 </html>

```



`querySelectorAll(selector)`

obtém referência para um conjunto de elementos HTML a partir de um seletor CSS

Métodos de adição/alteração de elementos

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10       <h1>Documento HTML</h1>
11       <p>Este é um documento <i>simples</i>.</p>
12       <p id="p2">Mais um parágrafo aqui.</p>
13       <p class="cl">E ainda mais um.</p>
14     </div>
15     <script type="text/javascript">
16       alert("Ola")
17     </script>
18   </body>
19 </html>

```



createElement(elemento): cria um elemento HTML

createTextNode(texto): cria um nó do tipo texto

appendChild(elemento): adiciona um elemento HTML

Métodos de adição/alteração de elementos

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <h1>Documento HTML</h1>
11      <p>Este é um documento <i>simples</i>.</p>
12      <p id="p2">Mais um parágrafo aqui.</p>
13      <p class="cl">E ainda mais um.</p>
14    </div>
15    <script type="text/javascript">
16      alert("Ola")
17    </script>
18  </body>
19 </html>

```

Documento HTML

Este é um documento *simples*.

Mais um parágrafo aqui.

E ainda mais um.

Novo texto

Elementos

Tudo Erros Avisos Informações Registos

☐ Manter Registro Destino: _top: index.html

Filtrar registros (Ctrl+F)

```

> let t2 = document.createTextNode("Novo texto")
  p1.replaceChild(t2,t1)
< [object Text]: {baseURI: null, childNodes: N
odeList, data: "Quarto parágrafo", firstChil
d: null, lastChild: null...}

```

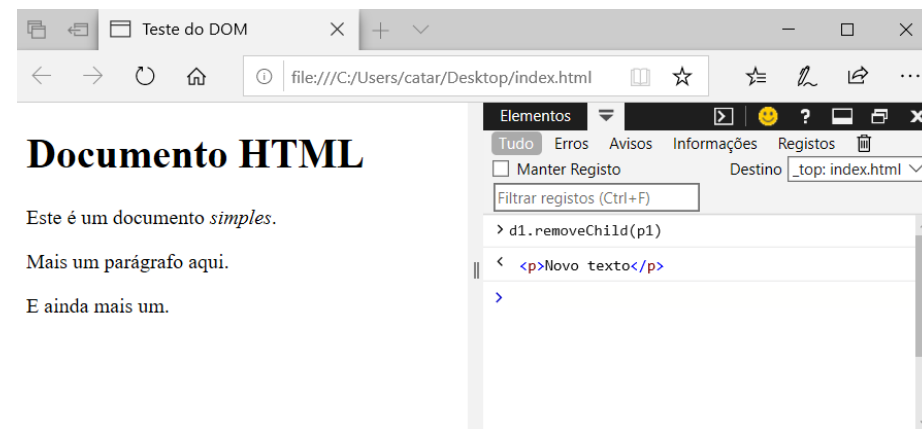
replaceChild(novo, antigo): substitui um elemento HTML por outro

Métodos de adição/alteração de elementos

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <h1>Documento HTML</h1>
11      <p>Este é um documento <i>simples</i>.</p>
12      <p id="p2">Mais um parágrafo aqui.</p>
13      <p class="cl">E ainda mais um.</p>
14    </div>
15    <script type="text/javascript">
16      alert("Ola")
17    </script>
18  </body>
19 </html>

```



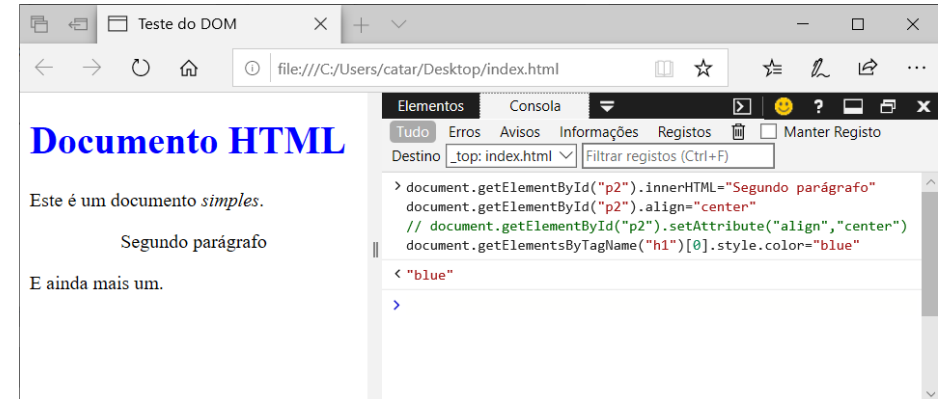
removeChild(novo, antigo): elimina um elemento HTML

Métodos de alteração de elementos HTML

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <h1>Documento HTML</h1>
11      <p>Este é um documento <i>simples</i>.</p>
12      <p id="p2">Mais um parágrafo aqui.</p>
13      <p class="cl">E ainda mais um.</p>
14    </div>
15    <script type="text/javascript">
16      alert("Ola")
17    </script>
18  </body>
19 </html>

```



...innerHTML: altera o conteúdo HTML do elemento

...nomeDoAtributo ou setAttribute(atributo,valor): altera conteúdo HTML do elemento

...style.propriedade: altera o estilo css do elemento no que diz respeito à propriedade

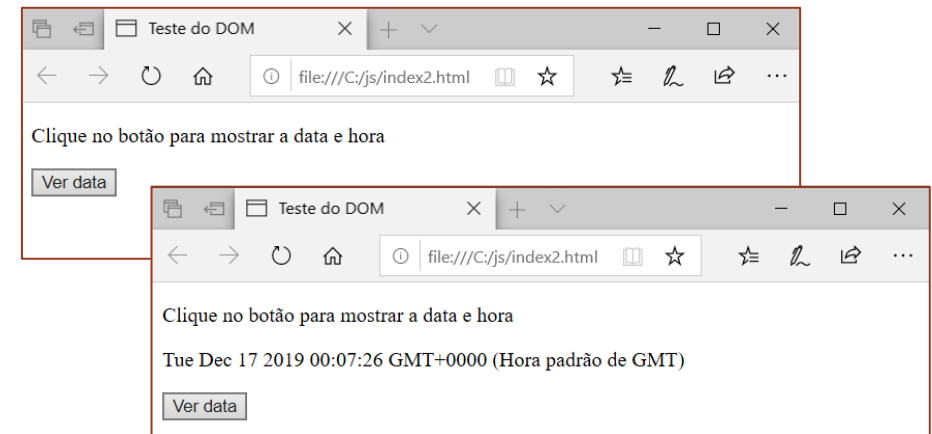
Tipos de eventos

Categoria	Tipo	Descrição
Janela	load	Janela completamente disponível
	unload	Fim de janela disponível
	resize	Alteração do tamanho da janela
	scroll	Utilização de scroll na página
Rato	click	clique
	doubleclick	duplo clique
	mousedown	início do clique (carregar)
	mouseup	fim do clique (largar)
	mousemove	mover o rato
Teclado	keypress	Pressionar tecla
	keydown	início da pressão numa tecla (carregar)
	keyup	fim da pressão numa tecla (largar)
Elementos	change	alteração de valores
	focus	receber foco
	select	seleção de texto
	submit	submissão de formulário

Eventos: exemplo onclick 1

- Exemplo: onclick

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5           content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <p>Clique no botão para mostrar a data e hora</p>
11      <p id="result"></p>
12      <button onclick="mostraData()">Ver data</button>
13    </div>
14    <script type="text/javascript">
15      function mostraData() {
16        document.getElementById("result").innerHTML = new Date()
17      }
18    </script>
19  </body>
20 </html>
```

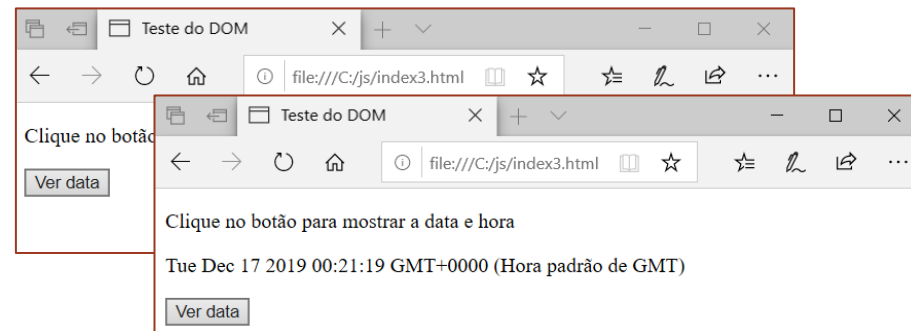


Eventos: exemplo onclick 2

- Exemplo: onclick

```
index3.html
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5           content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <p>Clique no botão para mostrar a data e hora</p>
11      <p id="result"></p>
12      <button id="btn">Ver data</button>
13    </div>
14    <script src="script3.js" type="text/javascript"></script>
15  </body>
16 </html>

script3.js
1 window.onload = function(){
2   let resultado = document.getElementById("result");
3   let botao = document.getElementById("btn");
4   botao.onclick = function(){
5     resultado.innerHTML = new Date();
6   }
7 }
```



Eventos: exemplo onclick 3

- Exemplo: onclick

```

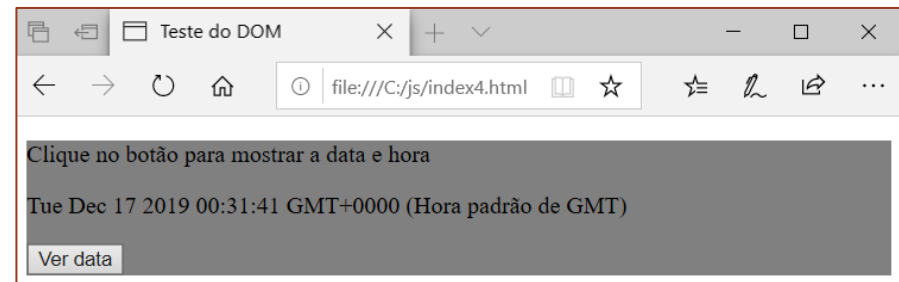
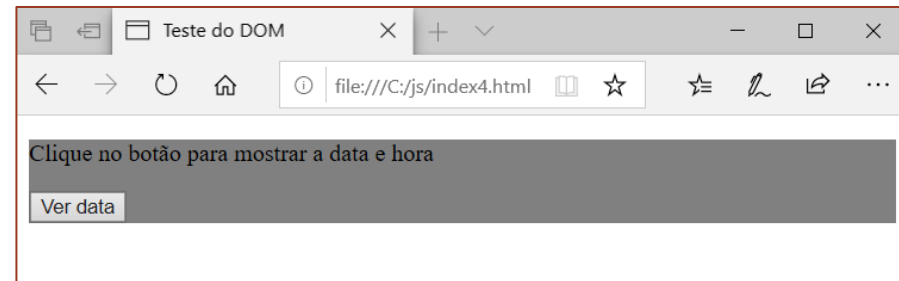
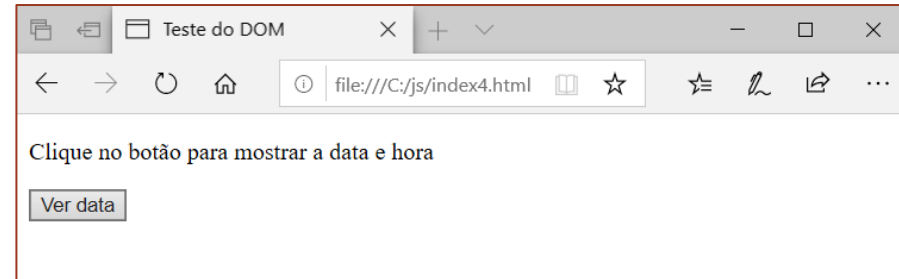
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <p>Clique no botão para mostrar a data e hora</p>
11      <p id="result"></p>
12      <button id="btn">Ver data</button>
13    </div>
14    <script src="script4.js" type="text/javascript"></script>
15  </body>
16 </html>

```

```

1 document.getElementById("corpo").addEventListener(
2   "click",
3   function() {
4     document.getElementById("corpo").style.backgroundColor="grey";
5   })
6
7 document.getElementById("btn").addEventListener(
8   "click",
9   function() {
10    document.getElementById("result").innerHTML = new Date();
11  })

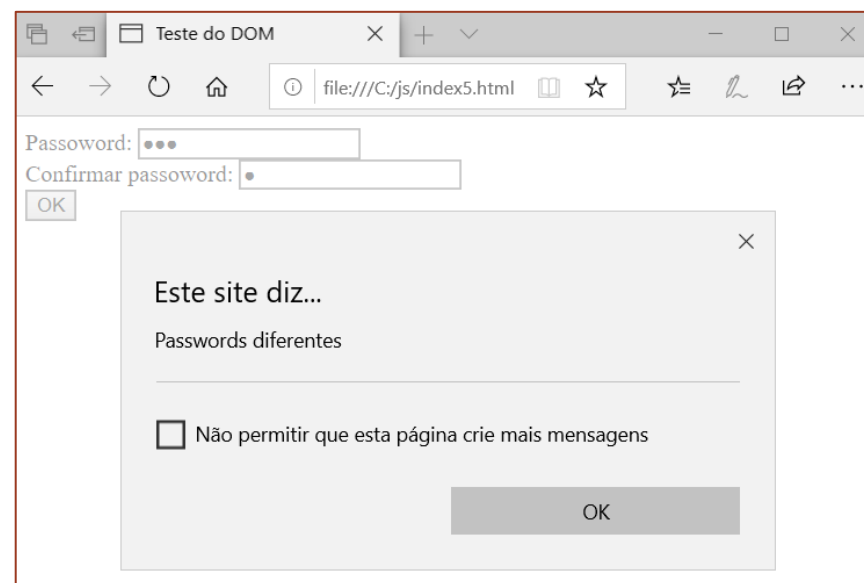
```



Exemplo com eventos: verificar passwords iguais

```
index5.html x
1 <!DOCTYPE html>
2
3 <html>
4   <head>
5     <meta http-equiv="content-type"
6       content="text/html; charset=utf-8">
7     <title>Teste do DOM</title>
8   </head>
9   <body>
10    <div id="corpo">
11      <form id="frm" action="index5.html">
12        Password: <input id="pass1" type="password" name="pass1"><br/>
13        Confirmar password: <input id="pass2" type="password" name="pass2"><br/>
14        <input type="submit" value="OK"><br/>
15      </div>
16      <script src="script5.js" type="text/javascript"></script>
17    </body>
18  </html>
```

```
script5.js x
1 let formulario = document.getElementById("frm");
2 frm.addEventListener("submit", function(event){
3   let paswd1 = document.getElementById("pass1");
4   let paswd2 = document.getElementById("pass2");
5   if(paswd1.value != paswd2.value) {
6     alert("Passwords diferentes");
7     event.preventDefault();
8   }
9 })
```





UNIVERSIDADE
PORTUCALENSE

Do conhecimento à prática.