



Web

DOM: *Document
Object Model*

Catarina Oliveira



DEPARTAMENTO DE CIÊNCIA
E TECNOLOGIA



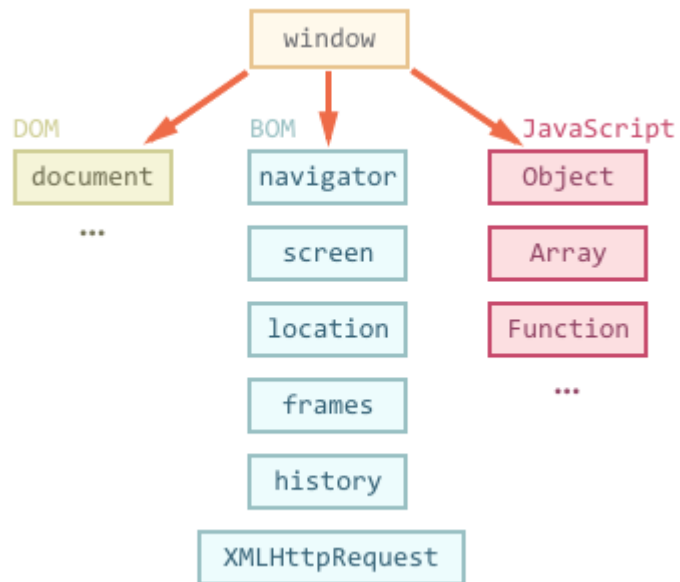
UNIVERSIDADE PORTUGALENSE

CONTENT

1. Context
2. Integration of JavaScript into HTML
3. Structure
4. DOM methods examples
 1. getElementById(id)
 2. getElementsByTagName(name)
 3. getElementsByClassName(name)
 4. querySelector(selector)
 5. querySelectorAll(selector)
 6. createElement(element)
 7. createTextNode(text)
 8. appendChild(element)
 9. replaceChild(new, old)
 10. removeChild(element)
 11. Methods to change HTML elements
5. Event types
6. Examples
 1. onclick
 2. Verify matching passwords

Contexto

- **DOM:** *Document Object Model*
- W3C standard
- Defines a uniform mean to access documents
- Divided in three parts
 - Core
 - XML
 - HTML
- JavaScript can access and alter all the elements of the HTML **document** and the *Browser Object Model (BOM)*



Integrating JavaScript into HTML

- Local

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

- External

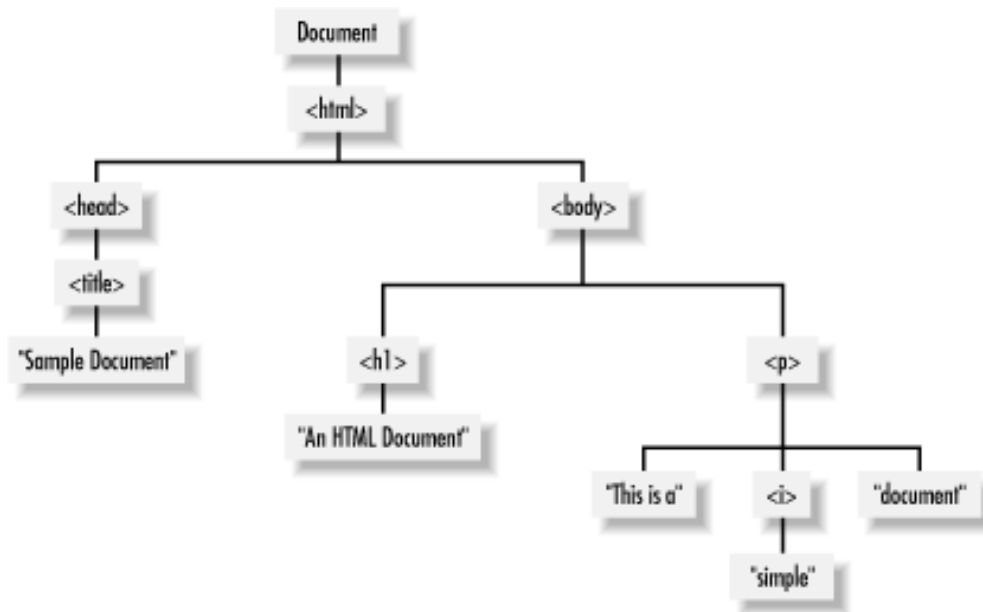
```
<html>
  <head>
    ...
  </head>
  <body>
    <script src="path_to_file.js"></script>
  </body>
</html>
```

Advantages:

- HTML / JavaScript separation
- Readability and maintenance of HTML and JavaScript code
- JavaScript files are cached → faster loading

Structure

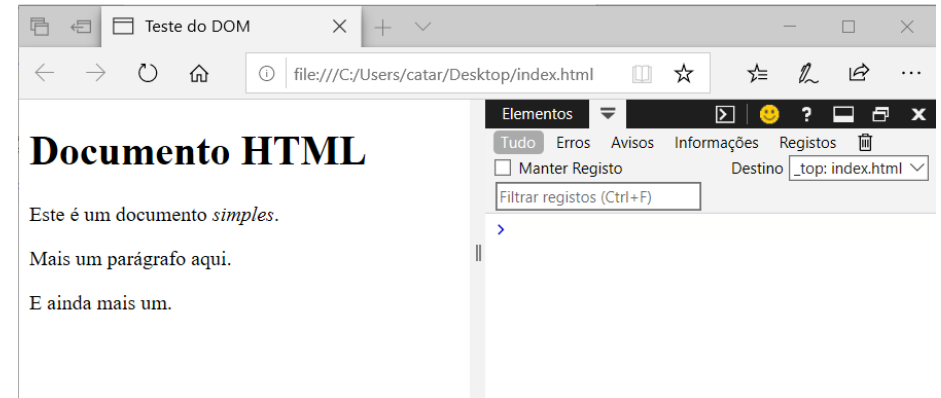
- Tree of different types of objects:
 - **Document**: the document (fig: Document)
 - **Node**: an HTML's generic node (fig: <title>, "Sample Document", <p>, ...)
 - **NodeList**: the list of the nodes
 - **Element**: elements created by HTML tags (fig: <html>, <head>, <title>, <body>, <h1>, ...)
 - **Text**: text inside elements (fig: "Sample Document", "An HTML Document", "This is a", ...)



- The tree is composed of **nodes**
- The root is the Document node (no parent)
- All the other nodes have a **parent** node
 - **Document** is parent of **<html>**
 - **<h1>** is parent of **"An HTML Document"**
- Nodes can have **children**
 - **"Sample Document"** is children of **<title>**
 - **"This is a"** is children of **<p>**

DOM examples: starting point

```
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>
```



Accessing elements: getElementById(id)

```

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)

Finds the reference to an HTML element from its id attribute

Accessing elements: `getElementsByTagName(name)`

```
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(name)`

Finds the reference to a set of HTML elements from their tags' name

Accessing elements: `getElementsByClassName(name)`

```

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>

```

The screenshot shows a web browser window with the title "Documento HTML". The page content includes the text "Este é um documento *simples*.", "Mais um parágrafo aqui.", and "E ainda mais um.". On the right side, the "Elementos" (Elements) panel is open, showing the DOM tree. It displays the result of the JavaScript code `document.getElementsByClassName("c1")`, which is an `HTMLCollection` with a length of 1, containing a single `<p class="c1">E ainda mais um.</p>` element.

`getElementsByClassName(nome)`

Finds the reference to a set of HTML elements from their class attribute

Accessing elements: `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)`

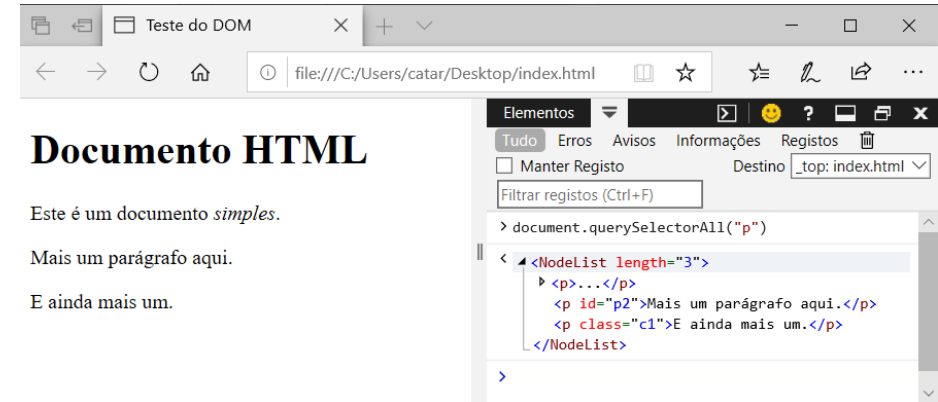
Finds the reference to an HTML element from a CSS selector

Accessing elements: `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)`

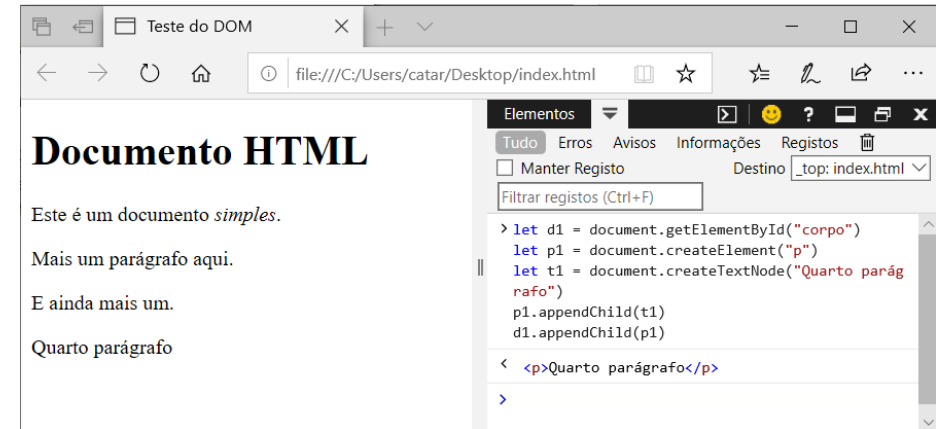
Finds the reference to a set of HTML elements from a CSS selector

Creating/modifying elements

```

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(element): creates and HTML element

createTextNode(text): creates a text node

appendChild(element): adds an HTML element

Creating/modifying elements

```

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>

```

The screenshot shows a web browser window titled 'Teste do DOM' displaying a document titled 'Documento HTML'. The document content is:

Este é um documento *simples*.

Mais um parágrafo aqui.

E ainda mais um.

Novo texto

The DOM Inspector panel on the right shows the document structure. The selected element is a text node with the value 'Novo texto'.

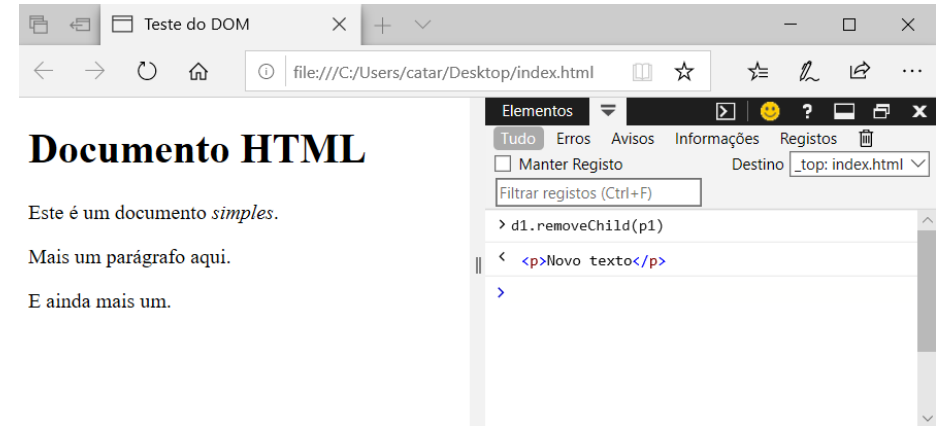
replaceChild(new, old): replaces the old HTML by the new one

Creating/modifying elements

```

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>

```



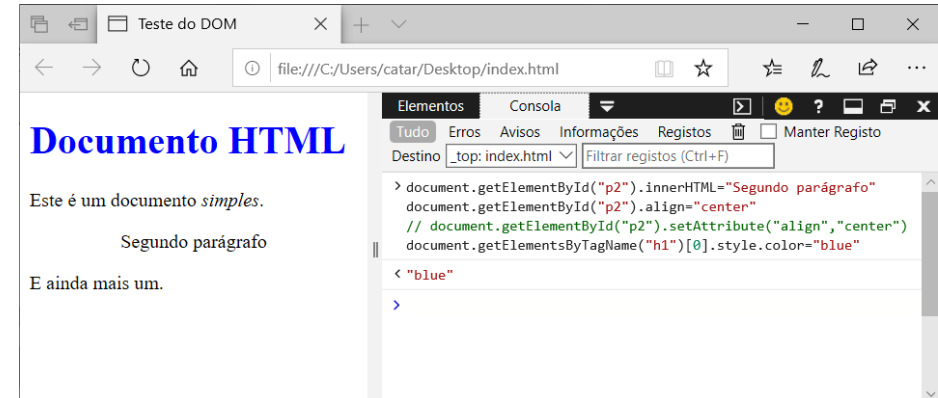
removeChild(element): removes the HTML element

Modifying HTML elements

```

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: alters the HTML element's content

...<attributeName> ou setAttribute(attribute,value): alters the HTML element's content

...style.property: alters the element's css in the property

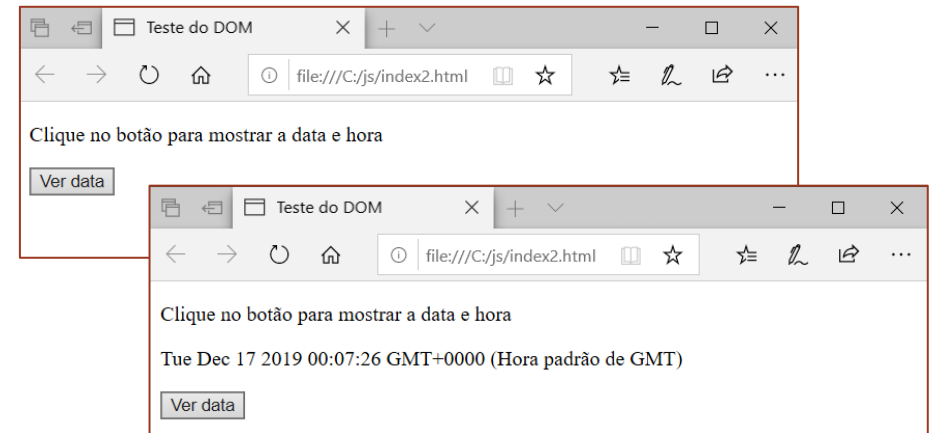
Event types

Category	Type	Description
Window	load	Fully available window
	unload	End of “window available”
	resize	Resizing the window
	scroll	Using scroll on the page
Mouse	click	Click
	doubleclick	Double click
	mousedown	Pressing the mouse button
	mouseup	Releasing the mouse button
	mousemove	Moving the mouse
Keyboard	keypress	Pressing key (full)
	keydown	Pressing key
	keyup	Releasing key
Elements	change	Value change
	focus	Receive focus
	select	Text selection
	submit	Form submission

Events: example onclick 1

- Example: onclick

```
index2.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 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>
```

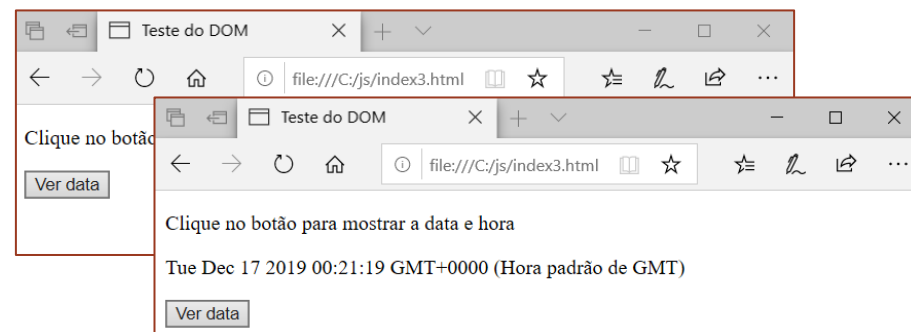


Events: example onclick 2

- Example: 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 }
```



Events: example onclick 3

- Example: onclick

```

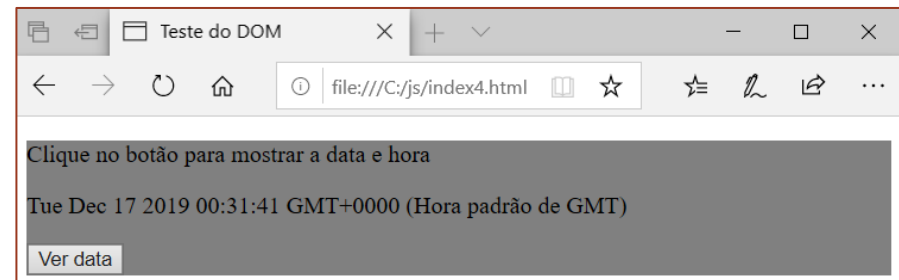
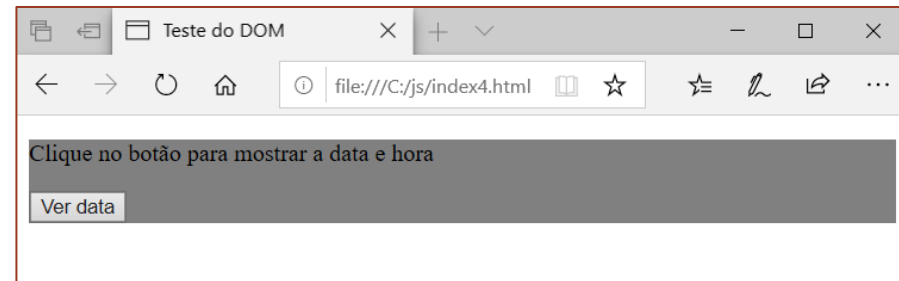
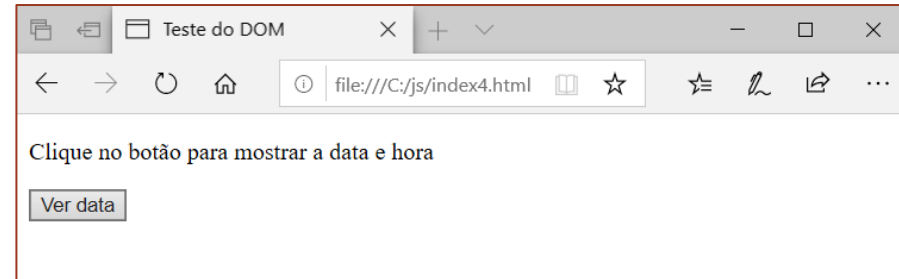
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta http-equiv="content-type"
5       content="text/html; charset=utf-8">
6     <title>Teste do DOM</title>
7   </head>
8   <body>
9     <div id="corpo">
10      <p>Clique no botão para mostrar a data e hora</p>
11      <p id="result"></p>
12      <button 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  })

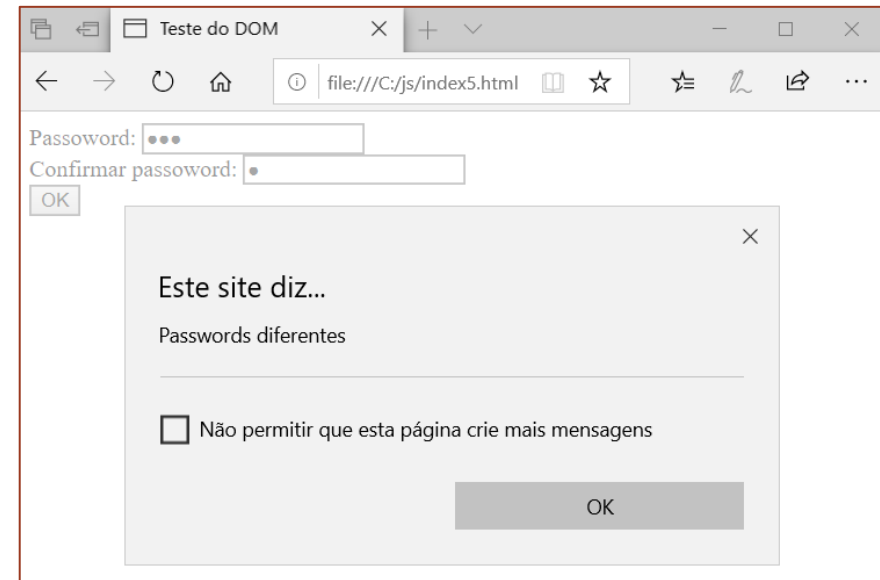
```



Events example: verify matching passwords

```
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.