Curso Técnico Superior Profissional em: Tecnologias e Programação de Sistemas de Informação

2º Ano/1º Semestre

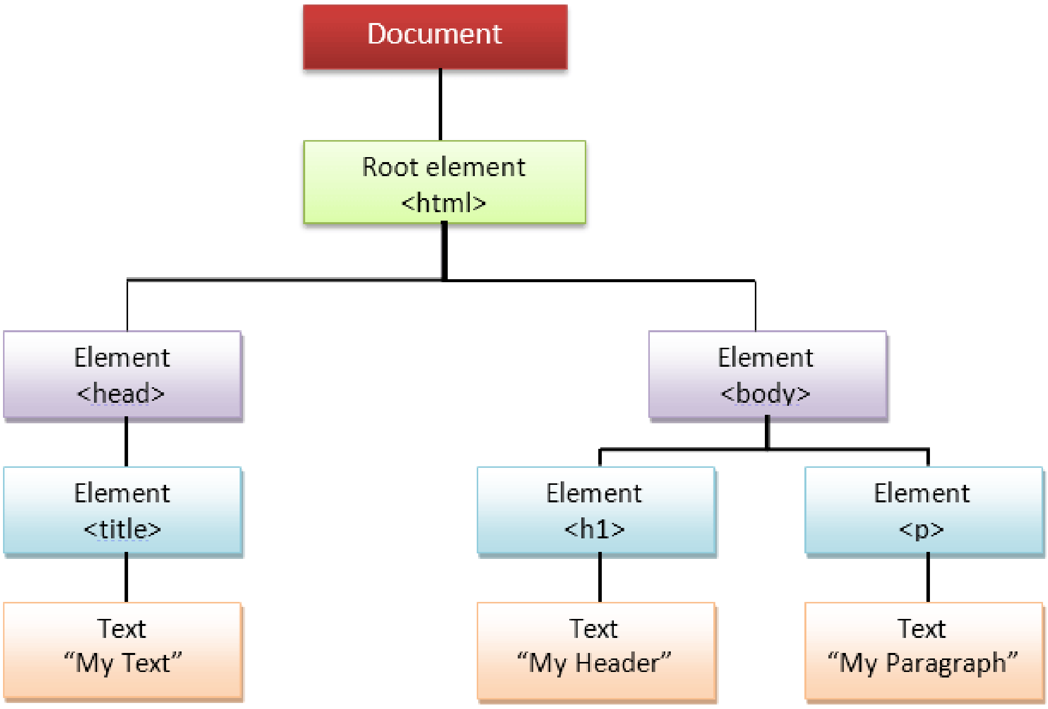
Unidade Curricular: Aplicações Centradas em Redes
Docente: Michael Silva / Hugo Perdigão

INTRODUÇÃO ÀS LINGUAGENS PARA O DESENVOLVIMENTO DE APLICAÇÕES CENTRADAS EM REDES

HTML DOM (Document Object Model)

HTML you write is parsed by the browser and turned into the DOM.

The HTML DOM model is constructed as a tree of Objects:



Each element in the DOM is also called a node.

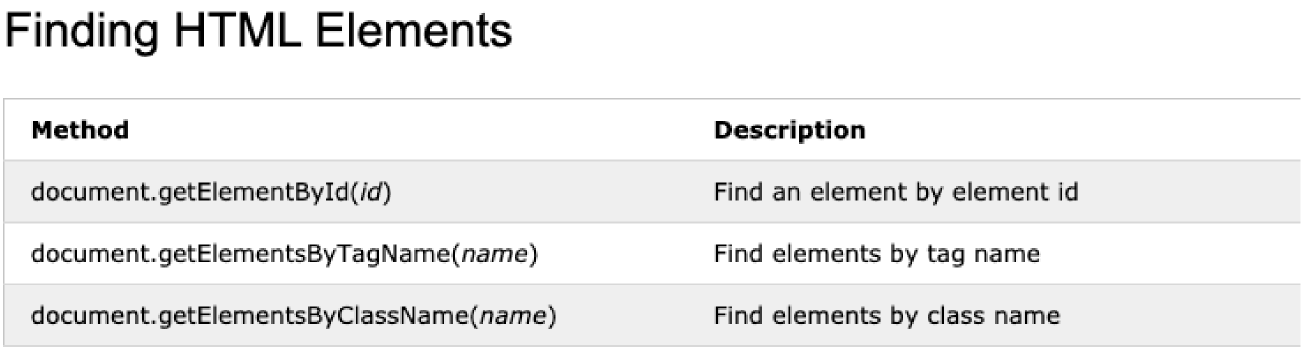
Cofinanciado por:

MADEIRA 14-20
REGIÃO AUTÓNOMA DA MADEIRA
PORTUGAL 2020
UNIÃO EUROPEIA Fundo Social Europeu

The HTML DOM is a standard for how to get, change, create, or remove HTML elements. React and create HTML events.

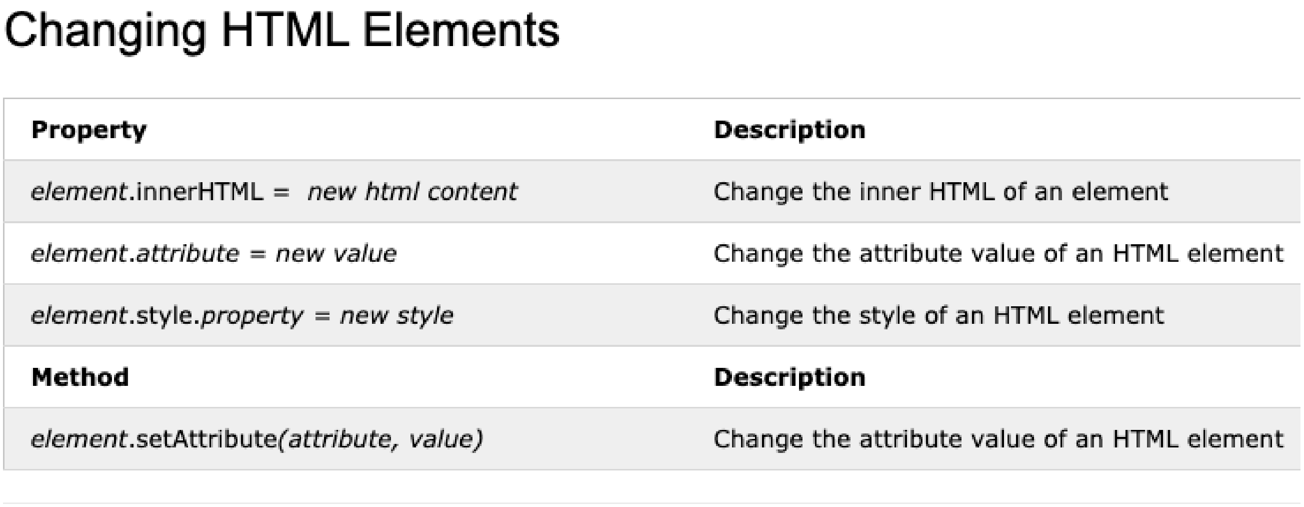
DOM manipulation in (vanilla/plain) JavaScript

JavaScript can manipulate (access and change) all the elements of an HTML document.



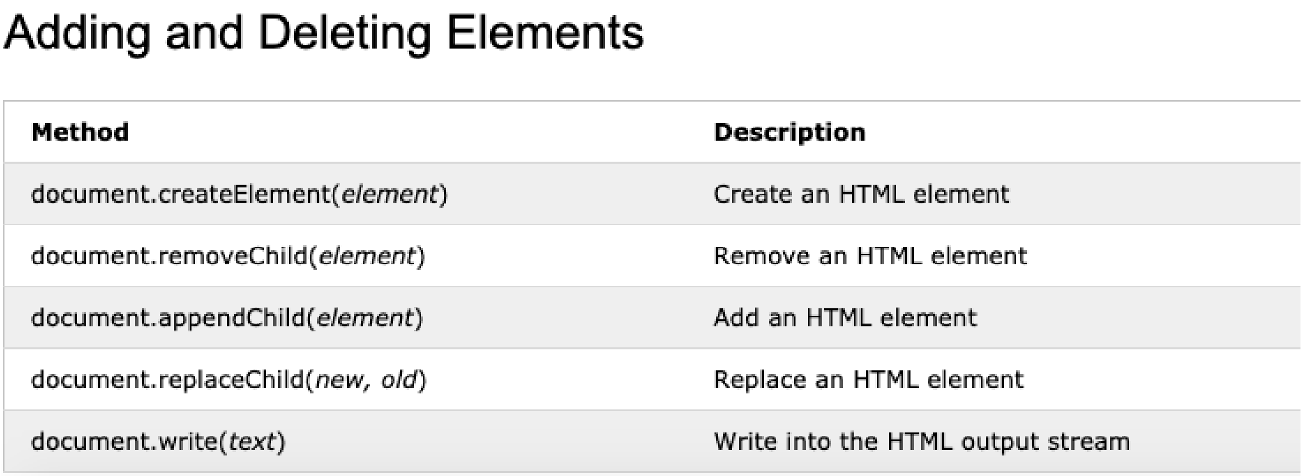
Finding HTML Elements

Method	**Description**
document.getElementById(*id*)	Find an element by element id
document.getElementsByTagName(*name*)	Find elements by tag name
document.getElementsByClassName(*name*)	Find elements by class name



Changing HTML Elements

Property	**Description**
element.innerHTML = *new html content*	Change the inner HTML of an element
element.attribute = new value	Change the attribute value of an HTML element
element.style.*property = new style*	Change the style of an HTML element
Method	**Description**
element.setAttribute*(attribute, value)*	Change the attribute value of an HTML element



Adding and Deleting Elements

Method	**Description**
document.createElement(*element*)	Create an HTML element
document.removeChild(*element*)	Remove an HTML element
document.appendChild(*element*)	Add an HTML element
document.replaceChild(*new, old*)	Replace an HTML element
document.write(*text*)	Write into the HTML output stream

Cofinanciado por:

MADEIRA 14-20

REGIÃO AUTÓNOMA DA MADEIRA

PORTUGAL 2020

UNIÃO EUROPEIA
Fundo Social Europeu

Query Selectors

```
<div id="container">
  <div class="display"></div>
  <div class="controls"></div>
</div>
const myElement = document.querySelector('#container');
```

Exemplos de argumentos:

- div.display
- .display
- #container > .display
- div#container > div.display
- input[type="submit"]

Altering Elements

Adding inline style

```
myElement.style.color = 'blue';
// adds the indicated style rule
myElement.style.cssText = 'color: blue; background: white';
// adds several style rules
myElement.setAttribute('style', 'color: blue; background: white');
// adds several style rules
```

Working with classes

```
myElement.classList.add('new');
// adds class "new" to your new div

myElement.classList.remove('new');
// remove "new" class from div

myElement.classList.toggle('active');
// if div doesn't have class "active" then add it, or if
// it does, then remove it
```

- É melhor usar o toggle

Create Element

```
const div = document.createElement('div');
```

Só cria, não adiciona ao DOM. Adicionar com os seguintes:

Append Elements

- *parentNode*.**appendChild**(*childNode*) appends *childNode* as the last child of *parentNode*
- *parentNode*.**insertBefore**(*newNode*, *referenceNode*) inserts *newNode* into *parentNode* before *referenceNode*

Remove Elements

- *parentNode*.**removeChild**(*child*) removes *child* from *parentNode* on the DOM and returns reference to *child*

Events

método 1

```
<button onclick="alert('Hello World')">Click Me</button>
```

- não é o ideal (HTML + JS)
- apenas 1 “onclick” event por elemento.

método 2

```
// the html file
<button id="btn">Click Me</button>


// the JavaScript file
var btn = document.querySelector('#btn');
btn.onclick = function() { alert("Hello World");}
```

- 1 “onclick” por elemento

método 3

```
// the html file
<button id="btn">Click Me Too</button>


// the JavaScript file
var btn = document.querySelector('#btn');
btn.addEventListener('click', function(e) {
  alert("Hello World");
});
```

- permite múltiplos events listeners

Cofinanciado por:

MADEIRA 14-20

REGIÃO AUTÓNOMA DA MADEIRA

PORTUGAL 2020

UNIÃO EUROPEIA
Fundo Social Europeu

Alguns Eventos úteis:

- click
- dblclick
- keypress
- keydown
- keyup

Mais eventos e descrição: https://www.w3schools.com/jsref/dom_obj_event.asp

Cofinanciado por:

MADEIRA 14-20

REGIÃO AUTÓNOMA
DA MADEIRA

PORTUGAL 2020

UNIÃO EUROPEIA
Fundo Social Europeu