



DOM

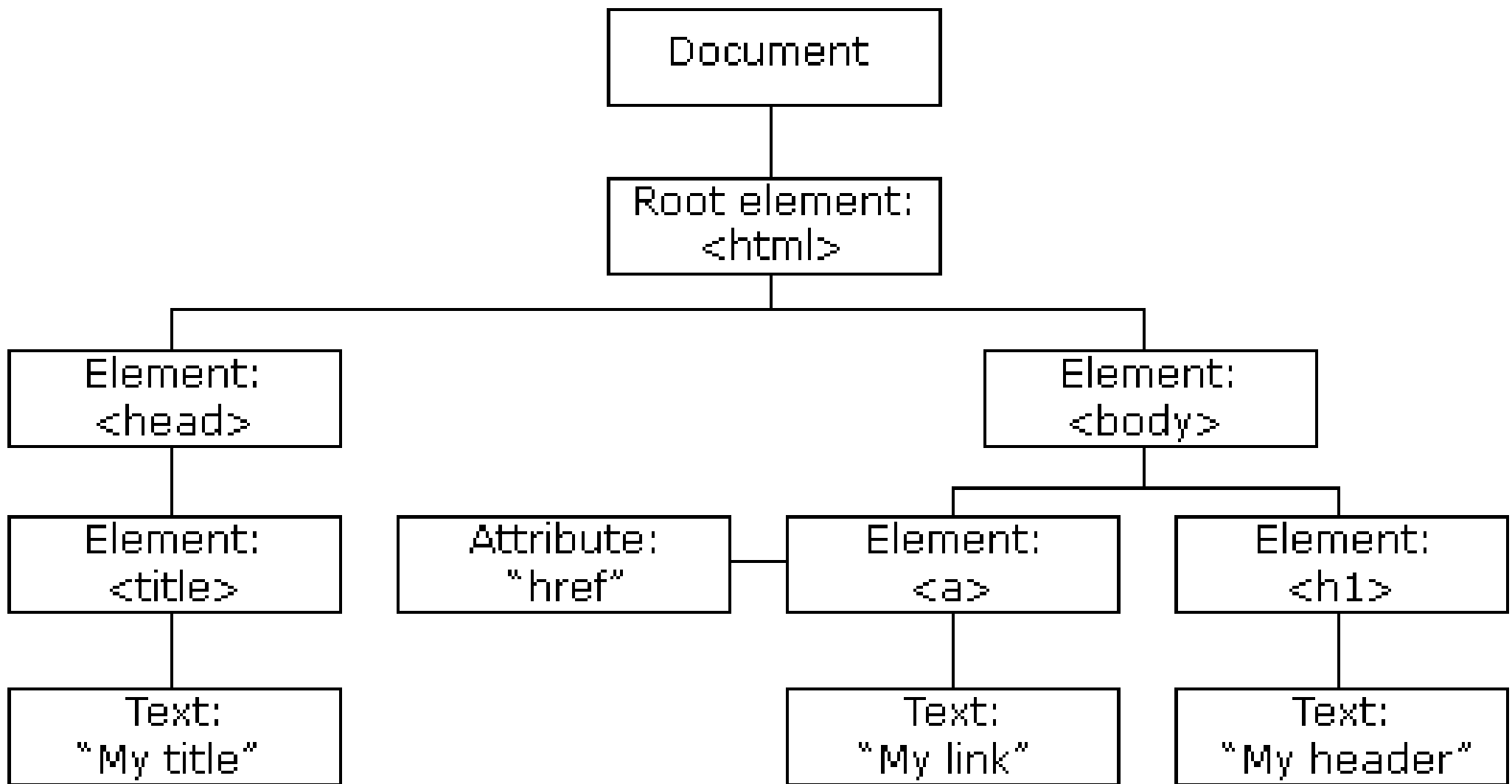
Manipulação do DOM


Gabriel Chiavegatto
T5 DRIVEN

Document Object Model (DOM)



- O DOM é a representação que compõem a estrutura e o conteúdo de um documento na Web;
- O DOM é o documento com nós e objetos, estruturando os dados;
- - Pensando em uma árvore de objetos, *document* é o nó 'raiz' do arquivo *html*.



- 
- **Element**: é tudo que está dentro da tag, um elemento podendo conter vários outros elementos dentro dele.

```
<body>  
  <div>  
    <h1> Texto </h1>  
  </div>  
</body>
```

- **Attribute**: fornecem informações adicionais sobre os elementos, são usados para especificar um element. São exemplos de atributos: href, src, width, height, alt, style, lang, class...

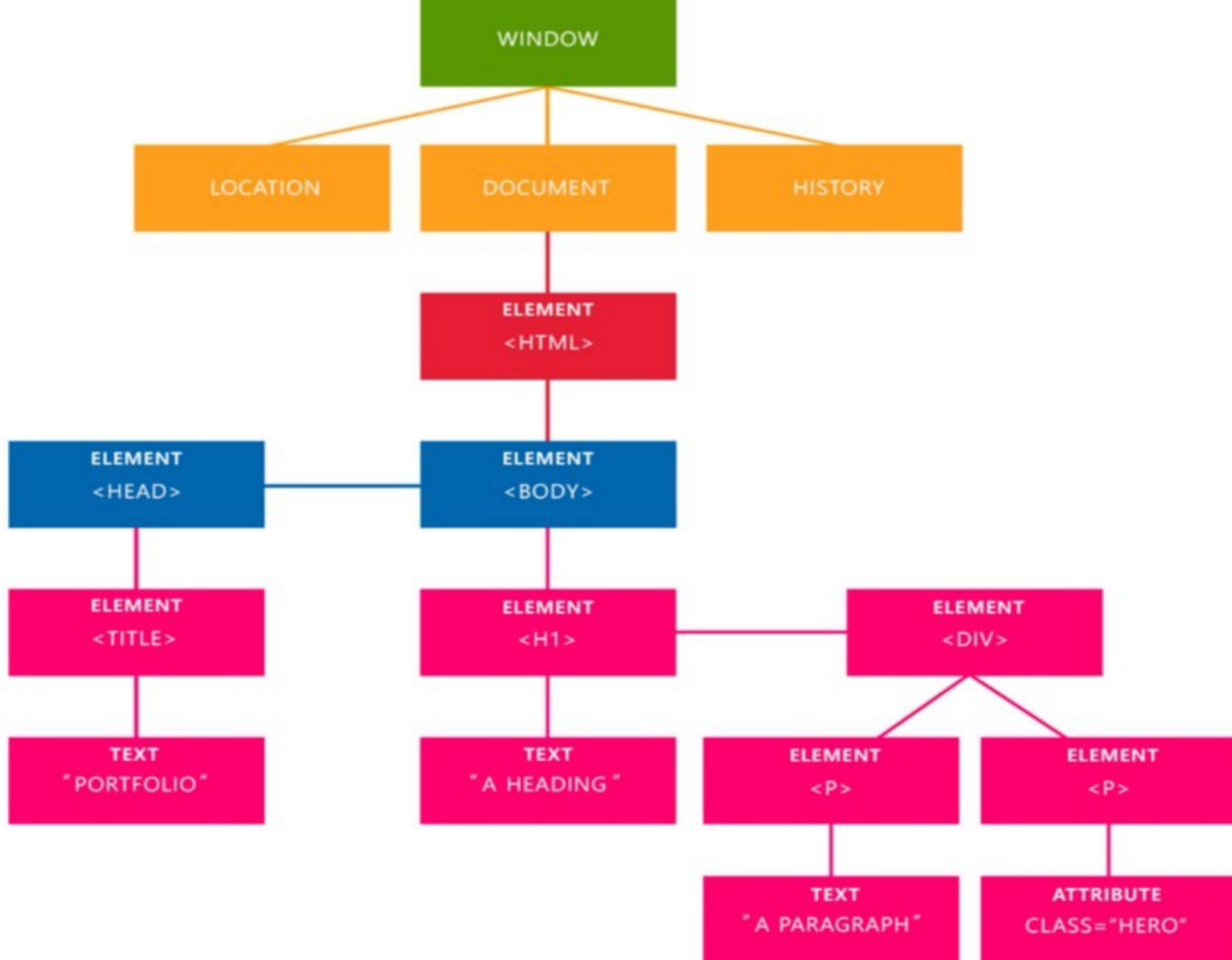
```
<section class="exemplo"> Conteúdo </section>
```

Objeto 'window'

Objeto 'document'



- - O objeto 'document' é a raiz de uma árvore de objetos, com os nós que compõem a estrutura da nossa pagina web.
- O objeto 'window' se refere à janela, o viewport do navegador que está executando o site.



Manipulação do Dom



- Para acessar o DOM possuem inúmeros métodos, são eles que fazem a ligação entre os nodes (elementos) e os eventos.

Ex.: `getElementById();`
`getElementsByTagName();`
`getElementByClassName();`
`querySelector();`
`querySelectorAll();`
`getSelection();`
`createElement();`
`setAttribute();`
`replaceChildren().....`

- <https://developer.mozilla.org/en-US/docs/Web/API/Document>



- `getElementById()`: localiza o elemento conforme o id passado, lembrando que o id deve ser único.
- `getElementsByClassName()`: localiza todos os elementos que possui a classe passada. Retorna uma lista `HTMLCollection` de elementos encontrados.
- `querySelector()`: retorna o primeiro elemento que contém o seletor passado.
- `querySelectorAll()`: retorna uma `NodeList` com todos elementos que possuem o seletor passado.



Propriedades semelhante, métodos de acesso diferentes.


- *NodeList* é um objeto que estrutura um coleção de nós.
- *HTMLCollection* é um objeto estruturado por uma coleção de elementos.
- *Array* é um objeto jQuery.

O `Array.from()` permite que você crie arrays à partir de objetos 'array-like'



```
const text = "ABCDEFGFG";  
const myArray = array.from(text);
```

```
const lista = document.querySelector(".classe");  
const viraArray = array.from(lista);  
// ou  
const viraArray = [...lista];
```

- 
- O ***innerHTML*** retorna todo o html existente, retornando também às tags, e não somente o texto. Usado para inserção de tags, textos e imagens em uma página web.
 - Já o ***innerText*** retorna apenas o texto. Ele ignora todas as tags HTML que estão dentro do elemento.
 - O ***textContent*** funciona de forma muito semelhante ao `innerText`, porém, retornando todo o conteúdo de texto, incluindo o texto oculto pelo css, bem como as quebras de linha (`\n`). Utilizamos o `textContent` quando queremos buscar todo o texto disponível no elemento.



document.createElement() cria o elemento HTML especificado

appendChild() método anexa um nó (elemento) como o último filho de um elemento.

```
// cria um novo elemento div
```

```
let divNova = document.createElement("div");  
divNova.innerText= 'ola grupo';  
document.body.appendChild(divNova);  
//adiciona a div criada no body
```