

Programação Web Prof. Carlos Veríssimo

Document Object Model - DOM

Nomes: Matheus Oldoni Amaral – 1142465860

Rafael Martins Da Silva Saraiva – 1142496167

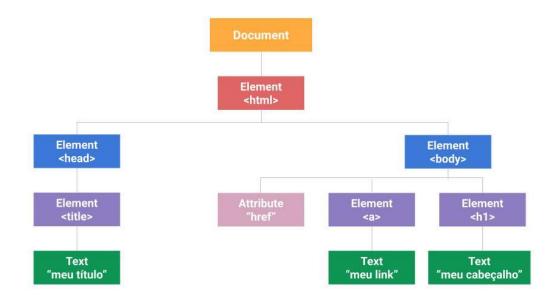
Document Object Model

Conceito

O Document Object Model (DOM) é uma API de programação para documentos HTML e XM, ele define a estrutura lógica dos documentos e a forma como um documento é acessado e manipulado. Com o Document Object Model, os programadores podem criar e construir documentos, navegar em sua estrutura e adicionar, modificar ou excluir elementos e conteúdo.

Quando uma página é carregada, o navegador cria um Document Object Model da página.

O modelo HTML DOM é construído como uma árvore de objetos, conforme vermos no exemplo abaixo:



Relação entre DOM e JavaScript

O DOM não é uma linguagem de programação, mas sem ela, a linguagem JavaScript não teria nenhum modelo ou noção de páginas da web, documentos HTML, e seus componentes (por exemplo, elementos). No início, o JavaScript e o DOM estavam interligados, mas, eventualmente, evoluíram para entidades separadas. A página é armazenada no DOM ao ser acessada é informada que pode ser manipulada por meio de JavaScript:

Exemplo: API (página HTML ou XML) = DOM + JS (linguagem de script)

O DOM foi projetado para ser independente de qualquer linguagem de programação específica, disponibilizando uma representação estrutural do documento consistente a partir de uma API única. Embora nos concentremos exclusivamente no JavaScript neste trabalho, as implementações DOM podem ser construídas em outros idiomas

Elementos de implementação JS/DOM

Alguns elementos do DOM que mais são utilizados:

getElementById()

Esse método retorna o elemento que estiver contendo o nome do ID passado. Como os IDs devem ser únicos, é um método muito útil para pegar apenas o elemento desejado.

getElementsByClassName()

Esse método retorna um objeto do tipo array de todos os elementos filhos que possuem todos os nomes de classe fornecidos.

getElementsByTagName()

Esse método retorna um HTMLCollection dos elementos com o nome da tag fornecida.

querySelector()

Esse método retorna o primeiro elemento dentro do documento que corresponde ao seletor ou grupo de seletores especificado. Se nenhuma correspondência for encontrada, é retornado um valor nulo.

querySelectorAll()

Esse método retorna uma lista de elementos presentes no documento que coincidam com o grupo de seletores especificado. O objeto retornado é uma NodeList.

InnerHTML

Esse método retorna propriedades que definem ou retornam o conteúdo HTML (HTML interno) de um elemento.

InnerText

Esse método retorna apenas o conteúdo de texto do elemento, sem espaçamento de texto oculto e tags CSS, exceto os elementos <script> e <style>.

Caso Prático

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Números com JS</title>
   <style>
       body { font: 12pt Arial; }
       button { font-size: 12pt; padding: 30px; }
   </style>
</head>
<body>
   <h1>Senac - TADS - PW - 2° Semestre </h1>
   <h2>Aula #02 - Introdução ao JS</h2>
       <h3>Cálculo da Média</h3>
   <button onclick="media()">Calcular média</button>
   <section id="situacao">
       O resultado vai aparecer aqui...
   </section>
   <script>
       function media() {
           let nom = window.prompt('Qual é o nome do aluno?')
           let n1 = Number(window.prompt(`Qual foi a primeira nota de ${nom}?`))
           let n2 = Number(window.prompt(`Além de ${n1}, qual foi a outra nota de ${nom}?`))
           med = (n1 + n2)/2
           let res = document.getElementById('situacao')
           res.innerHTML = `Calculando a média final de <mark>${nom}</mark>.`
           res.innerHTML += `As notas obtidas foram <mark>\{n1\} e \{n2\}</mark>.
           res.innerHTML += `A média final será <mark>${med}</mark>.
       }
   </script>
</body>
</html>
```

No caso prático acima, podemos ver como o DOM está sendo executado, fazendo sua função, dando uma boa forma no código e facilitando o entendimento do cálculo da média do aluno.

No começo do código JavaScript, já é iniciado abrindo uma função, com o nome de "media()", em seguida as variáveis já são declaradas e assim as informações para a realização do cálculo são coletadas, sendo assim, o calculo é realizado. Após isso, é usado um elemento de implementação (método) do DOM, sendo o **getElementById**, método que busca os elementos que estão usando um ID, então é feito o uso de mais um elemento, o **innerHTML()**, onde o mesmo traz o texto junto com as tags, trazendo o resultado final do cálculo da média.

Referências Bibliográficas

- JavaScript (Interatividade para Web) (<u>Biblioteca Digital SENAC</u>)
- IFRN DOM (Instituto Federal do Rio Grande do Norte)
- W3C Organização de Padronização (<u>W3C</u>)
- Caso Pratico GitHub (GitHub)