

CENTRO UNIVERSITÁRIO SENAC - CAMPUS SANTO AMARO

**TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
PROGRAMAÇÃO WEB**

**GUSTAVO BRAGALHA STOPPA
ITALO COSTA AMS FERNANDES
JOÃO VICTOR DA SILVA SOUSA**



**ATIVIDADE DISCENTE ORIENTADA 1 - RELAÇÃO ENTRE D.O.M E O
JAVASCRIPT**

PROFESSOR: CARLOS VERÍSSIMO - carloshvpereira@sp.senac.br

**SÃO PAULO
24/08/2022**

OBJETIVO:

O objetivo é proporcionar ao leitor conhecimentos sobre a relação entre DOM e o JavaScript, listando os seguintes tópicos: a definição de DOM, os elementos da relação entre DOM e JavaScript, elementos da implementação do JavaScript/DOM e finalizar exemplificando com um caso prático.

CONCEITOS:

- O que é DOM?

Document Object Model, ou DOM, é a API (*Application Programming Interface*, traduzido para o português como interface de programação de aplicações) fundamental para representar e manipular o conteúdo de documentos HTML e XML. A API não é especialmente complicada, mas existem vários detalhes de arquitetura que precisam ser entendidos. Primeiramente, você deve entender que os elementos aninhados de um documento HTML ou XML são representados na DOM como uma árvore de objetos.

O DOM também define tipos pouco utilizados, para representar coisas como declarações doctype e instruções de processamento XML.

- Dissertar sobre a relação entre DOM e Javascript & Elementos da implementação do JavaScript/DOM:

A maioria dos programas JavaScript do lado do cliente funciona manipulando de alguma forma um ou mais elementos de documento. Quando esses programas começam, podem utilizar a variável global “document” para se referirem ao objeto Document. Contudo, para manipular elementos do documento, eles precisam de algum modo obter ou selecionar os objetos Element que se referem a esses elementos de documento.

O DOM define várias maneiras de selecionar elementos, você pode consultar um documento quanto a um elemento (ou elementos):

- com um atributo id especificado;
- com um atributo name especificado;
- com o nome de marcação especificado;
- com a classe ou classes CSS especificadas;

As subseções a seguir explicam cada uma dessas técnicas de seleção de elementos. Qualquer elemento HTML pode ter um atributo id. O valor desse atributo deve ser único dentro do documento – dois elementos no mesmo documento não podem ter a mesma identificação. Você pode selecionar um elemento com base nessa identificação exclusiva com o método *getElementById()* do objeto Document.

Assim como o atributo id, name atribui um nome a um elemento. Ao contrário de id, contudo, o valor de um atributo name não precisa ser único: vários elementos podem ter o mesmo nome e isso é comum no caso de botões de seleção e caixas de seleção em formulários. Além disso, ao contrário de id, o atributo name é válido somente em alguns elementos HTML, incluindo formulários, elementos de formulário, iframe e elementos img.

Ao selecionar o atributo class de uma HTML é uma lista separada de zero ou mais identificadores por espaços. Ele descreve uma maneira de definir conjuntos de elementos relacionados do documento: todos os elementos que têm o mesmo identificador em seu atributo class fazem parte do mesmo conjunto. class é uma palavra reservada de JavaScript, de modo que JavaScript do lado do cliente utiliza a propriedade className para conter o valor do atributo HTML class. O atributo class normalmente é usado em conjunto com uma folha de estilos CSS para aplicar os mesmos estilos de apresentação em todos os membros de um conjunto. Vamos vê-lo outra vez, no Capítulo 16. Além disso, contudo, a HTML5 define um método getElementsByClassName() que nos permite selecionar conjuntos de elementos de documento com base nos identificadores que estão em seu atributo class.

- Caso prático:

Para exemplificar a relação entre DOM e o JavaScript utilizamos o código a seguir:

```
/**-----*
 *      SENAC - TADS - Programação Web      *
 *      Exemplo de função                    *
 *-----*/
function somar(parcela1,parcela2){
    let resultado = parcela1+parcela2;
    return resultado;
}
// vamos chamar a função e mostrar o resultado
let valor1=3;
let valor2=6
somaDosValores = somar(valor1, valor2);

console.log(`O resultado da soma de ${valor1} + ${valor2} =${somaDosValores} `);
```

Ao observar o código concluímos que a relação se estabelece em todos os navegadores atuais que implementam o DOM, de modo que tipos como Element e HTML Document 5 são classes como String e Array. Elas não são construtoras, mas têm protótipos de objetos e você pode estendê-las com métodos personalizados. Nas versões do Internet Explorer anteriores ao IE8, getElementById() faz uma correspondência que não diferencia letras maiúsculas e minúsculas nas identificações de elemento e também retorna elementos que tenham um atributo name coincidente.

BIBLIOGRAFIA:

FLANAGAN, David. JavaScript, o guia definitivo. 6º edição. Bookman Companhia Editora LTDA, 2013.