Aqui falaremos sobre a manipulação do D.O.M, as diferenças entre o D.O.M e B.O.M e os principais métodos de manipulação de uma pagina da web.

**D.O.M** , o Document Object Model HTML é um padrão de como **acessar e modificar os elementos HTML de uma página**. São os nós que apresentam cada elemento HTML de uma página.

**B.O.M**, o Browser Object Model é tudo aquilo **dentro do objeto window**. Sempre que abrimos o browser abrimos uma janela, uma window, **aquilo que faz parte do B.O.M NÃO faz parte do D.O.M**, são coisas diferentes.

A estrutura de uma página HTML segue um esqueleto padrão formado por **tags**, as representações da parte do código, como o **body** (contendo o corpo da página**, html** (que mostra que o código se trata de um código HTMLe o **head** (que contém todas as informações de configuração da página, como stylesheets, regras de fonte, título, etc.).

**H1-H6** também são tags, são diferentes niveis de titulo que vem do h1 como titulo principal, h2 como subtítulo, e assim por diante.

**Id** é um identificador **único** que permite a manipulação de uma tag única, não se pode possuir duas ids iguais.

**Class** é um identificador que funciona como o id, permitindo a manipulação, porêm na class nós podemos utiliza-la para modificar mais de uma tag. É possível ter mais de uma classe em um elemento.

Acima vemos os métodos de selecionar os elementos deuma página pelo js. Note que o comando *getElementby* sempre **retorna as informações em array**.

O comando **query**, que também sempre retorna o elemento em array, permite selectionar arquivos por classes, tags etc.

É possivel **adicionar e deletar elementos** com os comandos abaixo.

o **css** é o que permite estilizar os elementos do HTML, utilizando como método os Ids e as classes que vimos anteriormente. É possível também adicionar e remover classes através do JS, possibilitando também acionar ou desligar um uma classe ou elemento.

Podemos acessar diretamente os elementos CSSs para modifica-los através do JS.

*Document.getElementsByTagName(“p”).style.color=”blue”;*

Isso permite modificar diversas propriedades de uma tag, class e/ou Id.

**Eventos** se refere as ações do usuário ná pagina, separado em diversos tipos:

Os **eventos de mouse** que se refere aos eventos de interação com o mouse, como o **mouseover**¸que reage quando o mouse está em cima de um elemento, o **mouseout**¸quando o mouse se retira de um elemento, **click**, que se refere quando ocorre o ‘click’ do mouse no elementos e o **dbclick**, que funciona como o click só que para **cliques duplos**.

Os **eventos de atualização** que reagem pela atualização de valores de um campo, como o **change**, que ocorre quando mudanças acontecem na página e o **load** que ocorre quando uma informação nova é carregada na página.

Para acionar esses eventos utilizamos do comando *eventListener*, diretamente pelo JS, cria-se um evento que vai ser acionado quando o usuário fazer a ação.

*Const botao=document.getElementById(“meubotao”);*

*Botao.addEventListener(“click”,outrafunção);*

Também é possível fazer essas manipulações por dentro do próprio HTML através dos seus próprios métodos. Assim, especifica a função a ser chamada diretamente no elemento HTML.

*<html><body>*

*<h1* ***onclick****=”mudaTexto(this)”Clique aqui!;</h1>*

*<script>function mudaTexto(id){id.innerHTML=”mudei!”;}>/script>*

*</body></html>*