

# BOM (Browser Object Model)

OBS: Ele vem antes do DOM

O BOM ele possui propriedades e métodos relativo ao browser. Ele não pertence ao ECMAScript, mas é bem aceito pelos browser.

## Propriedades do BOM

**InnerWidth**: Retorna a largura interna disponível para a página;

**innerHeight**: Retorna a altura interna disponível para a página;

**outerHeight**: Retorna a altura total do navegador incluindo as barras;

**outerWidth**: Retorna a largura total do navegador incluindo as bordas;

**pageXOffset**: Retorna o quanto foi escrolado na horizontal;

**pageYOffset**: Retorna o quanto foi escrolado na vertical;

**screenX**: Retorna a posição do browser em relação ao monitor;

**screenY**: Retorna a posição do browser em relação ao monitor;

## Métodos do BOM

**alert()**: Uma caixa que o navegador apresenta com um alerta;

**confirm()**: Exibi uma janela com um pedido de confirmação ou negação;

**prompt()**: Exibi uma caixa com um campo para entrada de texto;

**open()**: Permite abrir uma página. Pode ser aberto apenas passando o link como parâmetro `open('http://google.com.br')` ou passando algumas configurações como exemplo `open('http: //google.com.br', 'janela', 'width=600, height=300')` ;

**close()**: Permite fechar uma janela, porém a função só fecha janelas que tenham sido abertas através do `open()` ;

**scrollTo()**: A função scrolla até a posição exata passada, exemplo `scrollTo(0, 150)` , onde 0 refere-se a posição X e 150 a posição Y.

**ScrollBy()**: Já o scrollBy diferente do scrollTo, ele escrola acrescentando a posição atual.

`print()`: Ela abre a janela de impressão.

## Eventos do BOM

`load`: É disparado quando a página é carregada

`error`: É utilizado em iframes. (Não sei o que faz)

`resize`: É disparado sempre que o tamanho do navegador é alterado.

`scroll`: É disparado sempre que é realizado uma scrollagem na página.

`unload`: É disparado ao encerramento do browser;

`beforeunload`: É disparado ao encerramento do browser;

## BOM - Objeto History

`history`: Retorna informações referente ao histórico.

`.length`: Retorna a quantidade de páginas navegadas;

`.go()`: Navega entre as páginas presentes no histórico, para navegar é inserido como parâmetro a quantidade de páginas que se deseja avançar, ou, passando a quantidade em negativo a quantidade de páginas que se deseja voltar.

`.back()`: Método que volta uma página no histórico;

`.forward()`: Avança uma página no histórico;

## BOM - Objeto Location

`location`: Retorna informações sobre a URL

`.hash`: Retorna no endereço se houver um hash o conteúdo desse hash, exemplo `#teste`;

`.href`: Retorna a página que está navegando;

`.protocol`: Retorna o protocolo que está sendo utilizado;

`.search`: Retorna termos de busca;

`.assign(url)`: Recebe por parâmetro uma URL e abre essa URL passada aumentando no histórico;

`.reload()`: Recarrega a página;

`.replace(url)`: Recebe por parâmetro uma URL e a carrega, porém não é acrescentado no histórico e sim substitui a página atual.

## BOM - Objeto Navigator

Navigator: Retorna informações relacionada ao navegador.

`.userAgent`: Retorna o navegador sendo utilizado;

`plugins`: Lista de plugins instalados;

`CookieEnabled`: retorna se os cookie estão habilitados;

`platform`: retorna a versão do sistema operacional;

## BOM - Objeto Screen

screen: é referente ao monitor

`.availableWidth`: Largura disponível para utilização;

`.availableHeight`: Altura disponível para utilização;

`.width`: Largura total do monitor;

`.height`: Altura total do monitor.

## Exercício 01

Mudar a configuração do menu principal se a página tiver um scroll maior do que a altura do próprio menu.

Arquivo base:

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/85b3e8a6-8bc4-41b9-9629-260139a20fca/arquivo.html>

Resolução:

```
var $body = document.querySelector('body');

window.addEventListener('scroll', function(){
    if(pageYOffset >= 150){
        $body.classList.add('fx');
    } else {
        $body.classList.remove('fx');
    }
})
```