

JAVASCRIPT É UMA LINGUAGEM DE PROGRAMAÇÃO

SERVE PARA COMPOR AS TECNOLOGIAS DE DESENVOLVIMENTO DA WEB MELHORANDO A EXPERIÊNCIA DO USUÁRIO

HTML (ESQUELETO), CSS (CORPO), JS (CÉREBRO).

ELE RODA DO LADO DO CLIENTE MAS TAMBÉM PODE SER UTILIZADO DO LADO DO SERVIDOR ATRAVÉS DO NODE.JS

POR NATUREZA, O JAVASCRIPT NECESSITA DE UM MEIO PARA PROCESSAR AS INFORMAÇÕES, NORMALMENTE, UMA PÁGINA HTML, MAS COM O DESENVOLVIMENTO DE FERRAMENTAS COMO O NODE, ESTE PERMITIU A EXECUÇÃO FORA DO AMBIENTE DOS NAVEGADORES.

FOI ORIGINALMENTE DESENVOLVIDA PARA QUE OS NAVEGADORES PUDESSEM EXECUTAR SCRIPTS LOCALMENTE SEM A NECESSIDADE DE UMA CONEXÃO COM O BANCO DE DADOS ALTERANDO O CONTEÚDO DA PÁGINA.

É DIFERENTE DE JAVA E É BASEADA EM ECMA SCRIPT.

NODE É BASEADO NO MOTOR DE INTERPRETAÇÃO DE JAVASCRIPT PURO DO GOOGLE, O V8 E PERMITE QUE OS CÓDIGOS JAVASCRIPT SEJAM EXECUTADOS FORA DE UM NAVEGADOR

QUANDO COLOCAMOS **window** PODEMOS ACESSAR OS DIVERSOS ELEMENTOS DA PÁGINA COM O JAVASCRIPT MAS NÃO É MUITO UTILIZADO. PODEMOS CITAR ENTÃO AS PRINCIPAIS FORMAS DE INTERAÇÃO:

O MOTOR DE EXECUÇÃO DO JAVASCRIPT DO NAVEGADOR PARA INTERPRETAÇÃO DE SINTAXE DO CÓDIGO JAVASCRIPT COMO IF, ELSE, CONDIÇÕES, VETORES, CLASSES, OBJETOS

DOM (DOCUMENT OBJECT MODEL) - ACESSO AOS ELEMENTOS HTML QUE EXISTEM NA PÁGINA

- O DOM REPRESENTAM O CONTEÚDO DA PÁGINA NA FORMA DE OBJETOS QUE PODEMOS MODIFICAR

- O **document** É O PRINCIPAL PONTO DE ENTRADA PARA MODIFICAR O CONTEÚDO HTML

- NORMALMENTE ASSOCIADA A UMA ÁRVORE

BOM (BROWSER OBJECT MODEL) - CONTÉM OBJETOS ADICIONAIS PARA MANIPULAR O BROWSER, EXCETO O DOCUMENTO HTML

- PERMITE A OBTENÇÃO DE INFORMAÇÕES DO SISTEMA

- PERMITE LER A URL E DIRECIONAR PARA OUTRA PÁGINA

- PERMITE EXIBIR UMA MENSAGEM DE ALERTA

- FUNCIONALIDADES QUE IRÃO INTERFERIR INDIRETAMENTE COM A EXPERIÊNCIA DO USUÁRIO

#####

EXEMPLOS

#####

CRIAR PÁGINA HTML COM UM TÍTULO

CRIAR UMA FUNÇÃO DE SAUDAÇÃO QUE EXIBE UM ALERTA

*CHAMAR A FUNÇÃO ATRAVÉS DO **window.NOMEFUNÇÃO***

MOSTRAR NO ALERT O COMANDO `window.innerWidth, location.href`
CRIAR UM IF COM `confirm` E DIRECIONAR COM `location.href` PARA O GOOGLE.
ALTERAR A COR DE FUNDO `document.body.style.background`
ALTERAR A COR DO TEXTO `document.body.style.color`
PROVER UM ID PARA O TÍTULO
ALTERAR O TEXTO DE UMA TAG.
`document.getElementById('titulo').innerText`

O DOM PERMITE FAZER INTERAÇÕES COM OS SEUS ELEMENTOS E RESPECTIVOS CONTEÚDOS. ENTRETANTO, PRECISAMOS CHEGAR AO ELEMENTO QUE QUEREMOS SELECIONAR.

ANTES DE MOSTRAR ALGUMAS FUNÇÕES QUE PERMITEM ACESSAR OS ELEMENTOS DIRETAMENTE, VAMOS VER ALGUNS ITENS QUE SÃO IMPORTANTES DE CONHECER.

// GRANDES GRUPOS

<html> corresponde a document
<body> corresponde a document.body
<head> corresponde a document.head

PODEMOS NAVEGAR ATRAVÉS DOS FILHOS DESSES ELEMENTOS DOS GRANDES GRUPOS.

----- CRIAR UMA PÁGINA HTML E COLOCAR UMA TAG SCRIPT NO HEAD MOSTRANDO O CONTEÚDO DO BODY `console.log(document.body)`.

SERÁ EXIBIDO O VALOR DE null POIS COMO O ARQUIVO HTML É INTERPRETADO LINHA POR LINHA, O BODY AINDA NÃO EXISTE QUANDO O SCRIPT FOR EXECUTADO.

TROCAR A ORDEM E MOSTRAR NO CONSOLE.

VER OS CHILD NODES
`document.body.childNodes`

ALTERAR O TEXTO com `childNodes[índice].innerText = 'text'` ou `.textContent = 'texto 2'`

NÃO EH A MELHOR FORMA DE INTERAGIR COM OS ELEMENTOS

`document.documentElement.lang`

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>
    .cor {
      color: red;
    }
  </style>
</head>
<body>

  <h3 id="titulo" class="cor">TÍTULO EM HTML</h3>

  <p class="cor">PARÁGRAFO EM HTML</p>
  <p class="cor">PARÁGRAFO EM HTML 2</p>

  <div name="este_div">
    Este texto está dentro de um div
  </div>

  <form action="" method="post"><input type="text" name="text_usuario"></form>

  <script>
    /*
      vamos falar sobre as 4 funções
      # getElementById
      - PERMITE ACESSAR O ELEMENTO A PARTIR DE UM DETERMINADO ID
      - O ID DEVE SER ÚNICO
      - DEVOLVE UM ELEMENTO ÚNICO

      # getElementsByName
      - DEVOLVE UMA COLEÇÃO
      - NORMALMENTE UTILIZADO EM FORMULÁRIOS

      # getElementsByTagName
      - PERMITE BUSCAR OS ELEMENTOS PELA TAG HTML
      - DEVOLVE UMA COLEÇÃO
      - INDEPENDENTE DE UM ID OU NAME

      # getElementsByClassName
      - RETORNA UMA COLEÇÃO
      - SELECIONA TODOS OS ELEMENTOS QUE POSSUEM CLASSES

      permite acessar mais os elementos mais facilmente

    */

    // acessar o elemento
    console.log(document.getElementById('titulo'))

    // editar elemento
    document.getElementById('titulo').innerText = 'novo título'

    // acessar o elemento
    console.log(document.getElementsByName('text_usuario'))

    // editar a informação
    document.getElementsByName('text_usuario')[0].value = 'joao'

    // acessar o elemento
    console.log(document.getElementsByTagName('p'))
  </script>
</body>
</html>

```

