



FIAP

Shift



Aula 05

JavaScript

Parte 1



O QUE O MATERIAL COBRE

1

JAVASCRIPT PARA CLIENTE

2

SUGESTÕES DE EXERCÍCIOS



1

Javascript para **Cliente**



JAVASCRIPT: TIPOS, CONSTANTES E VARIÁVEIS

- Principais tipos: string, number, boolean;
- Outros tipos: undefined, null, symbol;
- Constantes: valores alocados em memória que não mudam;
- Variáveis: valores alocados em memória que podem mudar;

JAVASCRIPT: DOM

O **DOM**, conhecido como **Document Object Model** é a representação da estrutura HTML criada para seu site, porém no lado do Javascript.

Tudo o que é alterado no HTML pode ser acessado via Javascript com funções específicas como **querySelector** e **querySelectorAll**:

- **querySelector** - utiliza da mesma sintaxe de elementos CSS, mas retorna o elemento HTML dentro do Javascript para ser manipulado caso exista;
- **querySelectorAll** - igual o **querySelector**, mas retorna uma lista de elementos, ou uma lista vazia, enquanto o **querySelector** retorna apenas um elemento.

JAVASCRIPT: DOM

Com o DOM também é possível criar e manipular HTML. Existem algumas funções e propriedades principais que cuidam dessas ações:

- **innerHTML** – permite que obtenha ou altere o conteúdo HTML do elemento;
- **textContent** – permite que obtenha ou altere o conteúdo de texto do elemento;
- **document.createElement()** – cria um novo elemento, passando o nome da tag por parâmetro;
- **elemento.appendChild()** – insere o elemento criado via Javascript no HTML dentro de outro elemento;
- **elemento.removeChild()** – remove o elemento coletado via Javascript no HTML;
- **elemento.classList.add()** – adiciona uma classe ao elemento HTML passada via parâmetro;
- **elemento.classList.remove()** – remove uma classe do elemento HTML passada via parâmetro;

JAVASCRIPT: **OPERADORES DE COMPARAÇÃO**


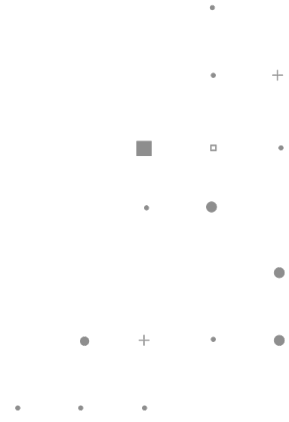
- Compara a operação e retorna um booleano;
- 8 tipos de operadores;
 - == “igual”
 - != “não igual”
 - === “estritamente igual”
 - !== “estritamente não igual”

JAVASCRIPT: **OPERADORES DE COMPARAÇÃO**

- 8 tipos de operadores;
 - > “maior que”
 - >= “maior ou igual que”
 - < “menor que”
 - <= “menor ou igual que”



JAVASCRIPT: **OPERADORES ARITIMÉTICOS**

- 
- Trabalham com valores numéricos;
 - 8 tipos de operadores;
 - + “adição”
 - - “subtração”
 - * “multiplicação”
 - / “divisão”
- 

JAVASCRIPT: **OPERADORES ARITIMÉTICOS**



- 8 tipos de operadores;
 - % “módulo”
 - ** “exponenciação”
 - ++ “incremento”
 - -- “decremento”

JAVASCRIPT: **OPERADORES LÓGICOS**

- Criam comparações compostas e retornam booleano;
- 3 tipos de operadores;
 - && “e”
 - || “ou”
 - ! “não”



JAVASCRIPT: **CONDICIONAIS**

- 
- Tomadas de decisão para executar um determinado bloco de código;
 - 3 casos de uso;
 - if “se”
 - else “se não”
 - else if “se não se”
- 




JAVASCRIPT: **LOOPS**

- 
- Executa o mesmo bloco de código por determinado número de vezes;
 - Opções:
 - for
 - while
 - doWhile
- 



JAVASCRIPT: **FUNÇÕES**


- o Blocos de códigos/instruções criados para executarem tarefas específicas em lugares diferentes;
 - o Utiliza-se muito para evitar código duplicado;
 - o Recebe parâmetros para deixar mais dinâmico;
- 

JAVASCRIPT: **ARRAYS E SUAS FUNÇÕES**

- Conhecidos como listas ou matrizes;
- Uma das estruturas de dados mais utilizada no JS
- Existem funções que facilitam a manipulação, principais:
 - map
 - forEach
 - filter
 - find
 - reduce



JAVASCRIPT: **JSON E OBJETOS**

- JSON: Formato de texto que facilita o compartilhamento de dados;
 - Objeto: tipo de dados utilizado no Javascript que é muito parecido com o JSON, utilizado para criar coleções de dados customizados;
- 

JAVASCRIPT: EVENTOS

```
<script>
  const button = document.querySelector('button');

  function buttonOnClick() {
    alert('Clicou no botão');
  }

  button.addEventListener('click', buttonOnClick);

  // Se necessário, o evento pode ser removido
  button.removeEventListener('click', buttonOnClick);
</script>

<button>
  Meu botão
</button>
```

Eventos são utilizados para gerar interação, existem vários tipos como clique (onclick), passar/retirar cursor por cima do elemento (onmouseup, onmousedown) e outros;

É possível utilizar eventos no VanillaJS com as funções “**addEventListener**” juntamente com o evento desejado;

Sempre que possível é interessante também remover o evento com “**removeEventListener**” para evitar código não utilizado.

PROJETO: **STAR WARS**



The mockup shows a web application titled "Projeto Star Wars" with the subtitle "Busque informações sobre o universo SW!". It features a search bar with a dropdown menu set to "Pessoas" and the text "luke" entered. A blue "Pesquisar" button is to the right of the search bar. Below the search bar, the section "Informações" contains a large, empty rectangular box with the placeholder text "Faça uma busca para encontrar resultados..." centered inside.

Com o conteúdo visto até agora podemos criar uma página que colete dados de um formulário para busca de informações na API pública do Star Wars.

PROJETO: **STAR WARS** – TELA INICIAL

Projeto Star Wars

Busque informações sobre o universo SW!

Pessoas



luke

Pesquisar

Informações

Faça uma busca para encontrar resultados...

PRÓJETO: STAR WARS – DADOS DO FORMULÁRIO

Projeto Star Wars

Busque informações sobre o universo SW!


Pessoas ▾

luke

Pesquisar

Informações

Faça uma busca para encontrar resultados...



127.0.0.1:5500 says

Dados do Formulário - Resource: people
e Search: luke

OK

PROJETO: STAR WARS – OUTROS DADOS

Projeto Star Wars

Busque informações sobre o universo SW!


Filmes

qualquer busca

Pesquisar

Informações

contratar resultados...



127.0.0.1:5500 says
Dados do Formulário - Resource: films e
Search: qualquer busca

OK

2

Sugestões de **Exercícios**

Exercícios para treinar Javascript

Com o site criado nas últimas aulas

- Atualize seu site para ter tema light/dark passando uma classe “dark” no body e atualizando as classes Tailwind ou o CSS para lidar com a alteração.
 - **Dica: o TailwindCSS tem um modo dark que pode ser implementado.**
- Crie um javascript que colete os dados do formulário de contato e exiba os mesmos em um alerta – utilize o evento “submit” para coletar os dados do formulário.
- Adicione validações com Javascript antes de enviar o formulário.

Bons estudos! 😊



OBRIGADO

@guscsales - gsales.io

FIAP

Copyright © 2022 | Professor Gustavo Campos Sales

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.





FIAP

