



Do zero a iniciante - Javascript JSON

JSON é um formato para armazenar e transportar dados. É frequentemente usado para trafegar dados entre **backend** e **frontend**

- **JSON** significa JavaScript Object Notation
- **JSON** é um formato leve para troca de dados
- **JSON** é independente de linguagens de programação, ou seja, o código para ler e gerar dados **JSON** pode ser escrito em qualquer linguagem de programação.
- **JSON** é "auto-descritivo" e fácil de entender
- **JSON** é muito usado para criar variáveis que armazenam informações diversas e a este tipo de variável damos o nome de **Objeto**

Chegou a hora de trabalhar com formulários mais complexos. Até o momento todos os exemplos tinham apenas um campo, no máximo dois, porém no dia-a-dia os formulários contêm muito mais informações e regras de validação.

Neste post irei explicar, na prática, como e porque usar o **JSON** e **Arrays**. Este último é usando para armazenar múltiplos valores em uma só variável.

Vamos construir um formulário de cadastro onde o usuário informará *Nome*, *Idade*, *Sexo*, *Plataforma de streaming* e seus *gêneros de filme* preferidos. Cada novo usuário registrado deverá ser listado em uma tabela.

Faça o teste usando o formulário abaixo!

Survey

Nome

Idade

Sexo

☐ Masculino ☐ Feminino

Plataforma de streaming

Netflix

Gêneros preferidos

☐ Comédia ☐ Terror ☐ Ação

Salvar

Hands On

Agora que você já viu o formulário acima em funcionamento vamos analisar o código dele.

HTML

A **tag input** foi definida com **type="number"** para o campo *idade* o que significa que só permitirá digitar números

```
<input type="number" class="form-control" id="idade">
```

Para informações sobre o sexo foi usado a **tag input type="radio"**. Utilize esta **tag** quando o usuário deve selecionar apenas uma entre **poucas** opções. Para que essa restrição de seleção funcione corretamente as **tags** do tipo **radio** devem ter o mesmo valor no **atributo name**, no caso abaixo este valor é "sexo". O **atributo value** desta **tag** nos permite identificar a opção marcada pelo usuário.

```
<input class="form-check-input" type="radio" name="sexo" id="masc"
value="masculino">
<label class="form-check-label" for="masc">Masculino</label>
<input class="form-check-input" type="radio" name="sexo" id="fem"
value="feminino">
<label class="form-check-label" for="fem">Feminino</label>
```

Para o campo "Plataforma de streaming" foi usado a **tag select**. Utilize esta **tag** quando o usuário deve selecionar apenas uma entre **muitas** opções. Por exemplo, selecionar o país, o estado, a cidade etc. O atributo **value** da **tag option** é fundamental para saber a opção escolhida pelo usuário

```
<label for="plataforma_stream">Plataforma de streaming</label>
<select id="plataforma_stream" class="form-control">
  <option value="netflix">Netflix</option>
  <option value="amazon prime">Amazon Prime</option>
  <option value="globo play">Globo Play</option>
</select>
```

Para informações sobre os gêneros preferidos foi usado a **tag input type="checkbox"**. Utilize esta **tag** quando o usuário pode selecionar **mais de uma** opção entre poucas opções. O **atributo value** desta **tag** nos permite identificar as opções marcadas pelo usuário.

```
<input class="form-check-input" type="checkbox" id="comedia"
value="comédia">
<label class="form-check-label" for="comedia">Comédia</label>
<input class="form-check-input" type="checkbox" id="terror"
value="terror">
<label class="form-check-label" for="terror">Terror</label>
<input class="form-check-input" type="checkbox" id="acao"
value="ação">
<label class="form-check-label" for="acao">Ação</label>
```

O botão salvar tem um **evento onclick** que dispara uma **função Javascript** com o nome de *salvar*

```
<button type="button" class="btn btn-outline-primary"
onclick="salvar()">Salvar</button>
```

Repare que a **div** que é o **container** da tabela está usando a classe **d-none** do **Bootstrap** o que significa que a tabela está invisível na tela.

```
<div class="table-responsive mt-5 d-none">
  <table id="tabela_usuario" class="table table-striped table-
bordered table-hover">
    <thead class="thead-dark">
      <tr>
        <th>#</th>
        <th>Nome</th>
        <th>Idade</th>
        <th>Sexo</th>
        <th>Plataforma</th>
        <th>Gêneros</th>
      </tr>
    </thead>
    <tbody>
    </tbody>
  </table>
</div>
```

JavaScript

O código **JavaScript** tem algumas novidades e vamos analisá-lo linha a linha

```
/*
  A variável usuarios é um array de escopo global.
  Ela é responsável por armazenar as informações
  de todos os usuários que forem cadastrados
*/
let usuarios = [];

const salvar = () => {
```

```
/*Cria uma variável chamada sexo e a inicializa com uma string vazia*/
let sexo = "";
```

```
/*
Armazena a referência dos elementos HTML com atributo name igual a
"sexo"
*/
let elementosSexo = document.getElementsByName("sexo");

/*
Percorre os elementos de sexo no HTML para pegar o valor
marcado pelo usuário
*/
for (let i = 0; i < elementosSexo.length; i++) {
    /*
    Se o elemento estiver selecionado (checked)
    a variável sexo irá receber o valor do mesmo
    e o loop for será encerrado.
    */
    if (elementosSexo[i].checked) {
        sexo = elementosSexo[i].value;
        break;
    }
}

/*
Armazena a referência dos checkbox
*/
let elementosGenerosPreferidos =
document.querySelectorAll("input[type=checkbox]");

/*
Cria um Array que irá armazenar os
gêneros preferidos selecionados pelo usuário
*/
let generosPreferidos = [];

/*
Percorre os checkbox presentes no HTML
*/
for (let i = 0; i < elementosGenerosPreferidos.length; i++) {
```

```
/*
    Se o elemento HTML estiver marcado (checked) o valor do mesmo dever
    á
    ser adicionado ao array
*/
if (elementosGenerosPreferidos.item(i).checked) {

generosPreferidos.push(elementosGenerosPreferidos.item(i).value);
}
}

/*Cria um variável JSON quer armazena informações do usuário*/
let usuario = {
nome: document.getElementById("nome").value,
idade: document.getElementById("idade").value,
sexo: sexo,
plataformaStream: document.getElementById("plataforma_stream").value,
generosPreferidos: generosPreferidos
};

/*
    Adiciona as informações do usuário em um array de usuários
*/
usuarios.push(usuario);

let conteudoTabela = "";

/*
    Itera o array de usuários para montar dinamicamente
    as informações que serão exibidas na tabela
*/
for (let i = 0; i < usuarios.length; i++) {

    /*
        Armazena as informações de todas as linhas da tabela
    */
    conteudoTabela += `
        <tr>
            <td>${i + 1}</td>
            <td>${usuarios[i].nome}</td>

```

```

        <td>${usuarios[i].idade}</td>
        <td>${usuarios[i].sexo}</td>
        <td>${usuarios[i].plataformaStream}</td>
        <td>${usuarios[i].generosPreferidos.join()}
    </td>

    </tr>`;
}

/*
Altera o HTML da tabela com as informações dos usuários cadastrados at
é o presente momento
*/
document.getElementById("tabela_usuario").getElementsByTagName("tbody"
)[0].innerHTML = conteudoTabela;

/*
Amarzena uma referência ao HTML da tabela
*/
let divTabela = document.getElementsByClassName("table-responsive")
[0];

/*Remove a classe d-none fazendo com que a tabela fique visível*/
divTabela.classList.remove("d-none");

}

```

O código **javascript** abaixo é responsável por "descobrir" a opção de sexo marcada pelo usuário.

```

/*Cria uma variável chamada sexo e a inicializa com uma string vazia*/
let sexo = "";

/*
Armazena a referência dos elementos HTML com atributo name igual a "se
xo"
*/
let elementosSexo = document.getElementsByName("sexo");

/*
Percorre os elementos de sexo no HTML para pegar o valor

```

```
    marcado pelo usuário
*/
for (let i = 0; i < elementosSexo.length; i++) {
    /*
        Se o elemento estiver selecionado (checked)
        a variável sexo irá receber o valor do mesmo
        e o loop for será encerrado.
    */
    if (elementosSexo.item(i).checked) {
        sexo = elementosSexo.item(i).value;
        break;
    }
}
```

O código **javascript** abaixo é responsável por "descobrir" quais opções entre os gêneros preferidos o usuário marcou. O código é um pouco semelhante ao de cima exceto pelo uso do **querySelectorAll** e do array **generosPreferidos**.

```
/*Armazena a referência dos checkbox */
let elementosGenerosPreferidos =
document.querySelectorAll("input[type=checkbox]");

/*
    Cria um Array que irá armazenar os
    gêneros preferidos selecionados pelo usuário
*/
let generosPreferidos = [];

/*
    Percorre os checkbox presentes no HTML
*/
for (let i = 0; i < elementosGenerosPreferidos.length; i++) {

    /*
        Se o elemento HTML estiver marcado (checked) o valor do mesmo dever
        á
        ser adicionado ao array
    */
    if (elementosGenerosPreferidos.item(i).checked) {
```



```
    generosPreferidos.push(elementosGenerosPreferidos.item(i).value);  
  }  
}
```

O **querySelectorAll** é uma **função javascript** que retorna todas as referências de elementos **HTML** baseados em filtro informado como **parâmetro**. Neste caso o **parâmetro** é **input[type=checkbox]** que diz à função para retornar as referências de **tags input** deste tipo.

Uma vez que o usuário pode selecionar mais de um *gênero preferido* temos que armazenar isso num **array**, logo fica claro o propósito da **variável generosPreferidos** ter sido definida como um **array**.

Arrays são definidos usando colchetes **[]**

Utilize a função **push** para adicionar itens à um **array**

Itens dentro de um **array** são acessados através de índices. Estes índices sempre começam a partir do número **zero**. Se por exemplo a **variável generosPreferidos** tiver 3 elementos dentro dela e você quiser acessar **diretamente** um destes 3 elementos a sintaxe seria a seguinte

```
generosPreferidos[0]; // Pega o primeiro elemento  
generosPreferidos[1]; // Pega o segundo elemento  
generosPreferidos[2]; // Pega o terceiro elemento
```

Como todas as informações do formulário são atreladas à uma única pessoa. Temos que atribuir estas informações em uma única **variável**. É aí que entra o **JSON**. Como disse no começo do post este tipo de variável recebe o nome de **Objeto**

A sintaxe do **JSON** é simples e consiste em:

```
let nomeDaVariavel = { atributo1 : valor, atributo2 : valor }
```

No caso deste formulário, o **JSON** ficou assim

```
/*Cria um variável JSON quer armazena informações do usuário*/  
let usuario = {  
  nome: document.getElementById("nome").value,  
  idade: document.getElementById("idade").value,  
  sexo: sexo,  
  plataformaStream: document.getElementById("plataforma_stream").value,
```

```
    generosPreferidos: generosPreferidos
  };
```

Repare que cada atributo dentro do **JSON** representa um campo do formulário

A ideia deste form é coletar informações de vários usuários, logo para exibir estas informações em uma tabela precisamos armazená-las em um **array**.

A **variável usuarios** é um **array** de **objetos** de **escopo global**. Isto porque ela foi definida fora da função salvar, na raiz do **javascript**.

Varáveis globais são acessíveis por qualquer **função** presente no **javascript**, ou seja, se tivéssemos outra **função** além da salvar a **variável usuarios** estaria acessível para ela.

```
/*
  A variável usuarios é um array.
  Ela é responsável por armazenar as informações
  de todos os usuários que forem cadastrados
*/
let usuarios = [];
```

Todo usuário recém cadastrado é adicionado ao **array usuarios**

```
/*
  Adiciona as informações do usuário em um array de usuários
*/
usuarios.push(usuario);
```

É necessário percorrer o **array usuarios** pegando as informações de todos os usuários que foram cadastrados para montar as linhas da tabela dinamicamente.

```
let conteudoTabela = "";

/*
  Itera o array de usuários para montar dinamicamente
  as informações que serão exibidas na tabela
*/
for (let i = 0; i < usuarios.length; i++) {
```

```

/*
    Armazena as informações de todas as linhas da tabela
*/
conteudoTabela += `
    <tr>
        <td>${i + 1}</td>
        <td>${usuarios[i].nome}</td>
        <td>${usuarios[i].idade}</td>
        <td>${usuarios[i].sexo}</td>
        <td>${usuarios[i].plataformaStream}</td>
        <td>${usuarios[i].generosPreferidos.join()}
    </td>

    </tr>`;
}

```

Uma vez que temos as informações dinâmicas da tabela basta manipular o **HTML** da mesma para que seja exibido no **browser**

```

/*
    Altera o HTML da tabela com as informações dos usuários cadastrados at
    éo presente momento
*/
document.getElementById("tabela_usuario").getElementsByTagName("tbody"
)[0].innerHTML = conteudoTabela;

/*
    Armazena uma referência ao HTML da tabela
*/
let divTabela = document.getElementsByClassName("table-responsive")
[0];
/*
    Remove a classe d-none fazendo com que a tabela fique visível
*/
divTabela.classList.remove("d-none");

```

Observações

Com este post a parte **frontend** está encerrada. Se você chegou até aqui e conseguiu fazer todos os exercícios, meu parabéns. Você está no caminho certo!



Se você ainda tem dúvidas não deixe de perguntar nos comentários. A parte **backend** começará em uma semana então aproveite este tempo para reler os posts e fixar o conhecimento!

Deixe seus elogios, críticas e dúvidas nos comentários!