Roteiro

JavaScript

Tópicos

Opção 1

* DOM
* Tipos de dados
* Variáveis
* Operadores
* Estrutura de controles
* Objetos
* Arrays
* Funções
* Objetos customizados
* Clojures
* Arrow functions
* Classes (propriedades, métodos e herança e entender o "this")
* Templates literais
* let e const

Opção 2

* Let
* Const
* Arrow functions
* Imports and Exports
* Classes

O que é JavaScript

Linguagem de programação interpretada

Links

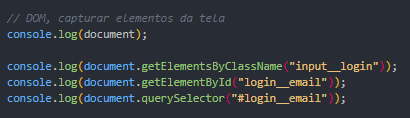
<https://www.w3schools.com/js/default.asp>

NodeJS

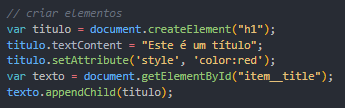
Ambiente de execução JavaScript no lado do servidor. Criar aplicações back-end.

DOM

Estruturas



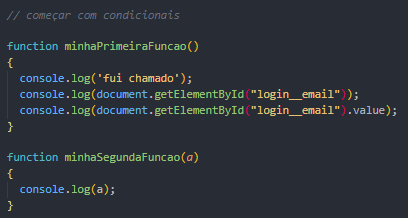
Variáveis





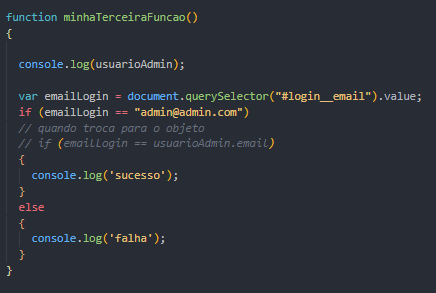
Funções



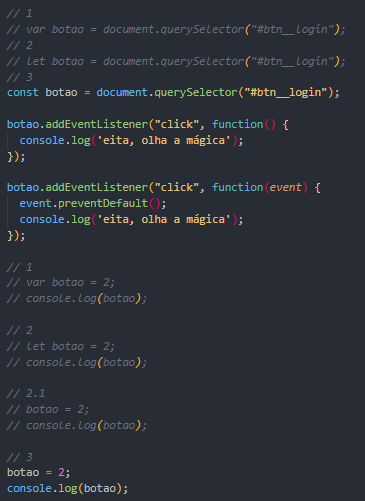


Operadores

Estrutura de controles



Let e const

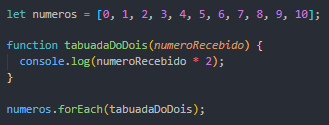


Percorrendo Arrays









PUSH/POP



Arrays



MAP



FILTER



REDUCE

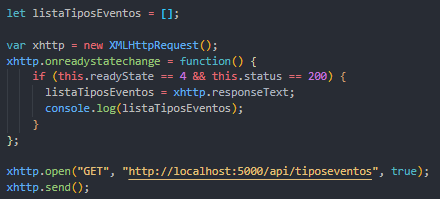


Mostrar uma lista fixa na tabela



https://medium.com/beginners-guide-to-mobile-web-development/the-fetch-api-2c962591f5c

XMLHttpRequest





NodeJS

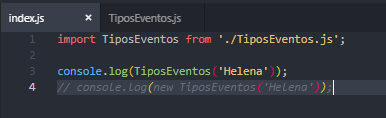
<https://www.npmjs.com/package/http-server>

npm install http-server -g

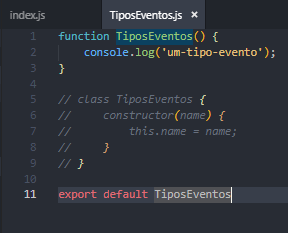
http-server ./Svigufo-js

Import/Export

index.js



TiposEventos.js



tipos-eventos.html



Instalar NodeJS

npm install -g create-react-app

create-react-app senai-svigufo-ui

cd senai-svigufo-ui

npm start

Criar projeto

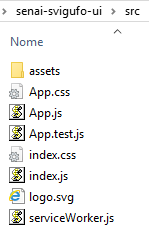
Classes

Babel

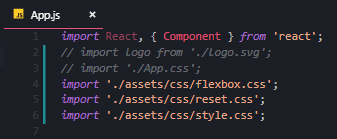
JSX

webpack

Copiar a pasta assets definida no projeto base e colar dentro de src do projeto do React.



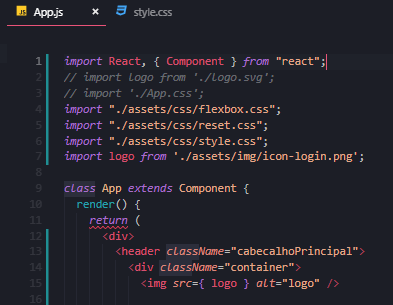
Abrir o arquivo App.js e colocar as referências para os css’s imputados.



Alterar tudo o que for class, nas tags para className.

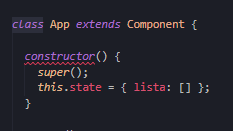


Realizar a alteração da importação do logo para que o mesmo apareça.

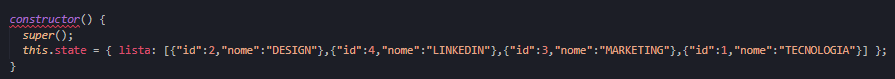


<https://fontawesome.com/how-to-use/on-the-web/using-with/react>

Começar a trabalhar com a lista



Copiar alguns valores inicialmente da API.



<tbody>{*this*.state.lista}</tbody>

<tbody>

{*this*.state.lista.map(*function*(*tipoEvento*) {

*return* <tr />;

})}

</tbody>

<tbody>

{*this*.state.lista.map(*function*(*tipoEvento*) {

*return* (

<tr>

<td>{tipoEvento.id}</td>

<td>{tipoEvento.nome}</td>

</tr>

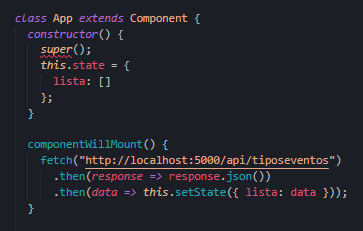
);

})}

</tbody>

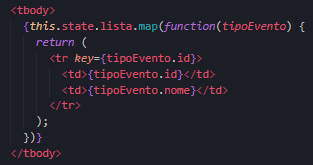
Fazendo a busca para a API. Queremos que o valor seja renderizado e buscado antes do render ser invocado.

Deixar o valor da lista em branco, realizar a alteração para realizar a chamada à API.



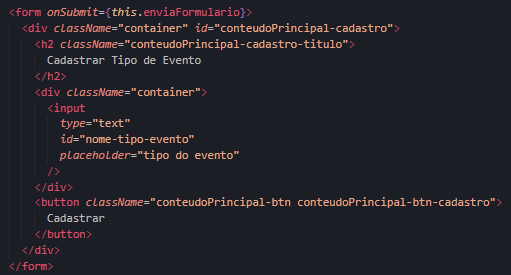
Com o setState, informamos para o React que queremos realizar a alteração no estado do componente e que o mesmo deve ser renderizado.

Para identificar a linha como única e ter sua propriedade, o react necessita ter seu próprio id.

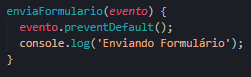


Axios – Exemplo

Englobar com um formulário e colocar o onSubmit para que uma ação seja efetuada.

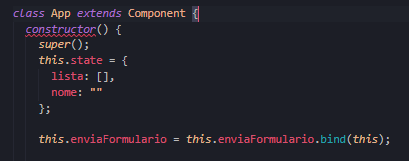


Criar uma função que seja chamada.



Para realizar um cadastro, vamos puxar a mesma chamada do outro projeto.





Porém, de onde precisamos puxar este valor?

