## **Calculadora de IMC em React**

### **Visão Geral**

A Calculadora de IMC (Índice de Massa Corporal) é uma aplicação React desenvolvida para calcular o IMC de uma pessoa com base na altura (em centímetros) e no peso (em quilogramas). A aplicação exibe o resultado do IMC e sua respectiva categoria (como "Normal", "Sobrepeso", "Obesidade", etc.).

### **Tecnologias Utilizadas**

* **React**: Biblioteca JavaScript para construção da interface.
* **JavaScript (ES6+)**: Utilização de recursos modernos como classes, arrow functions, desestruturação, módulos, spread operator, entre outros.

### **Componentes e Arquivos**

#### **1. App.js**

Este é o componente principal da aplicação, responsável por renderizar a interface e lidar com a lógica do cálculo do IMC.

* **Classes**: O componente App é uma **classe** estendida de React.Component.
* **Estado (State)**: Utiliza o this.state para armazenar os valores de entrada (altura e peso), além do valor calculado do IMC e sua categoria.
* **Funções**:
  + handleInputChange: Função para atualizar o estado conforme o usuário preenche os campos de altura e peso. Implementada com **arrow function** e **desestruturação** dos eventos de input.
  + calculateBMI: Função responsável por calcular o IMC e encontrar a categoria correspondente com base no valor calculado. Também utiliza **spread operator** para atualizar o estado de forma imutável.
* **Template Literals**: Utilizado na exibição do resultado de IMC e da categoria.
* **Operador Ternário**: Usado para condicionalmente renderizar o resultado ou a mensagem de aviso.

Código usado em App.js:

import React from 'react';

import { calculateBMI, bmiCategories} from './bmiUtils';

import './App.css';

class App extends *React*.*Component* {

constructor() {

*super*();

*this*.state = {

height: '',

weight: '',

bmi: null,

category: '',

};

}

handleInputChange = ({ target: { *name*, *value*} }) => {

*this*.setState({ [name]: value});

};

calculateBMI = () => {

const { height, weight } = *this*.state;

if (height && weight) {

const bmiValue = calculateBMI(height, weight);

const category = bmiCategories.find(({ *min*, *max* }) => bmiValue >= min && bmiValue <= max);

*this*.setState({

...*this*.state,

bmi: bmiValue.toFixed(2),

category: category ? category.label : 'Fora da faixa',

});

}

};

render() {

const { height, weight, bmi, category } = *this*.state;

return (

<section>

<div *style*={{ padding: '20px'}}>

<h1>Calculadora IMC</h1>

<input

*type*="number"

*name*="height"

*value*={height}

*onChange*={*this*.handleInputChange}

*placeholder*="Altura (em cm)"

/>

<input

*type*="number"

*name*="weight"

*value*={weight}

*onChange*={*this*.handleInputChange}

*placeholder*="Peso (em kg)"

/>

<button *onClick*={*this*.calculateBMI}>Calcular IMC</button>

{bmi ? (

<p>{`Seu IMC é`} <strong>{bmi}</strong> {`e você está na categoria: `} <strong>{category}</strong></p>

) : (

<p>Preencha os campos para calcular o seu IMC</p>

)}

</div>

</section>

);

};

}

export default App;

#### **2. bmiUtils.js**

Este arquivo contém funções utilitárias e os dados das categorias de IMC.

* **Funções**:
  + calculateBMI: Função que realiza o cálculo do IMC com base na fórmula: peso / (altura / 100) ^ 2.
* **Arrays e Objetos**: Define um array bmiCategories, onde cada item do array é um objeto que contém a categoria de IMC e os limites de valor mínimo e máximo.

Código usado em bmiUtils.js:

export const calculateBMI = (*height*, *weight*) => {

return *weight* / ((*height* / 100) \*\* 2);

};

export const bmiCategories = [

{ label: 'Magreza', min: 0, max: 18.5},

{ label: 'Normal', min: 18.5, max: 24.9},

{ label: 'Sobrepeso', min: 25, max: 29.9},

{ label: 'Obesidade Grau 1', min: 30, max: 34.9},

{ label: 'Obesidade Grau 2', min:35, max: 39.9},

{ label: 'Obesidade Grau 3', min: 40, max: 100},

];

### **Conceitos Usados**

Abaixo estão os conceitos de **ES6/React** usados no projeto:

* **var, let e const**: const e let foram usados para definir variáveis e garantir imutabilidade onde necessário.
* **Arrow Function**: Utilizada para a definição das funções handleInputChange e calculateBMI.
* **Objetos**: O estado (this.state) e o array bmiCategories são manipulados como objetos.
* **Classes**: O componente App é um exemplo de um componente de classe.
* **Arrays**: O array bmiCategories armazena as categorias de IMC e seus valores.
* **Desestruturação**: Utilizada para acessar rapidamente propriedades de objetos e eventos.
* **Template Literals**: Utilizada para construir strings dinâmicas para exibir o resultado do IMC.
* **ES6 Modules**: O projeto usa módulos ES6 para separar a lógica de cálculo (bmiUtils.js) e a interface (App.js).
* **Operador Ternário**: Utilizado para condicionalmente renderizar o resultado ou uma mensagem de aviso.
* **Spread Operator**: Utilizado para garantir uma cópia imutável do estado quando ele é atualizado.

### **Como Rodar o Projeto**

1. **Iniciar o Projeto**:
   * Execute npm start no terminal para iniciar o servidor de desenvolvimento.
2. **Interface**:
   * A interface contém dois campos para entrada de altura e peso e um botão para calcular o IMC.
   * O resultado é exibido na forma de uma frase mostrando o valor do IMC e a categoria correspondente.