

Questão 1a:

```
App.js X # styles.css
src > App.js > OlaMundo
1 import './styles.css';
2
3 function OlaMundo() {
4   return <h1>Olá Mundo</h1>;
5 }
6
7 export default OlaMundo;
8
```

Questão 1b:

```
App.jsx X
meu-projeto-react > src > App.jsx > default
1
2 function OlaMundo() {
3   return (<h1>Olá Mundo !</h1>
4   );
5 }
6 export default OlaMundo
```

Questão 1c:

```
function DataHoraAtual(){
  const agora = new Date();
  const diasSemana = ['domingo', 'segunda', 'terça', 'quarta', 'quinta', 'sexta', 'sabado'];
  const meses = ['janeiro', 'fevereiro', 'março', 'abril', 'maio', 'junho', 'julho', 'agosto', 'setembro', 'outubro', 'novembro', 'Dezembro', ];
  const diaSemana = diasSemana[agora.getDay()];
  const dia = agora.getDay();
  const mes = meses[agora.getMonth()];
  const ano = agora.getFullYear();
  return (
    <h1>Hoje é {diaSemana}, {dia} de {mes} de {ano}</h1>
  );
}
export default DataHoraAtual
```

Questão 2a:

```
App.jsx ×
meu-projeto-react > src > App.jsx > ...
1  function Saudacao({ nome }) {
2    |   return <h1>Olá, {nome}</h1>;
3    |
4  }
5  export default function MyApp() {
6    |   return <Saudacao nome="Matheus" />;
7    |
8  }
```

Questão 2b:

```
App.jsx ×
meu-projeto-react > src > App.jsx > ...
1  function BoasVindas({ usuario }) {
2    |   return <h1>Bem vindo de volta, {usuario}</h1>
3    |
4  }
5  export default function MyApp() {
6    |   return <BoasVindas usuario="Lindesay" />
7    |
8  }
```

Questão 2c:

```
App.js ×
src > App.js > ...
1  function StatusJogo({ nivel, pontos }) {
2    const maxPontos = nivel * 1000;
3    const progresso = Math.min((pontos / maxPontos) * 100, 100);
4    const corBarra = progresso < 30 ? "red" : progresso < 70 ? "orange" : "green";
5
6    return (
7      <div>
8        <h2>Nível {nivel}</h2>
9        <p>
10          Pontos: {pontos}/{maxPontos}
11        </p>
12        <div className="progress-container">
13          <div
14            className={`progress-bar ${corBarra}`}
15            data-width={progresso}
16          ></div>
17        </div>
18        <p>{progresso.toFixed(1)}% completo</p>
19      </div>
20    );
21  }
22
23  export default function App() {
24    return <StatusJogo nivel={5} pontos={2500} />;
25  }
26
```

```
App.js ×
src > App.js > App
1  function StatusJogo({ nivel, pontos }) {
2    const maxPontos = nivel * 1000;
3    const progresso = Math.min((pontos / maxPontos) * 100, 100);
4    const corBarra = progresso < 30 ? "red" : progresso < 70 ? "orange" : "green";
5
6    return (
7      <div>
8        <h2>Nível {nivel}</h2>
9        <p>
10          Pontos: {pontos}/{maxPontos}
11        </p>
12        <div
13          style={{
14            width: "200px",
15            height: "20px",
16            backgroundColor: "#ddd",
17            borderRadius: "10px",
18            overflow: "hidden",
19          }}
20        >
21          <div
22            style={{
23              width: `${progresso}%`,
24              height: "100%",
25              backgroundColor: corBarra,
26              transition: "all 0.3s ease",
27            }}
28          ></div>
29        </div>
30        <p>{progresso.toFixed(1)}% completo</p>
31      </div>
32    );
33  }
34
35  export default function App() {
36    return <StatusJogo nivel={5} pontos={2500} />;
37  }
38
```

Questão 3a:

```
App.js ×
src > App.js > ...
1  function ListaHobbies() {
2    const hobbies = ["Leitura", "Natação", "Programação", "Culinária"];
3
4    return (
5      <div>
6        <h2> Meus Hobbies Favoritos</h2>
7        <ul>
8          {hobbies.map((hobby, index) => (
9            <li key={index}>{hobby}</li>
10          ))}
11        </ul>
12      </div>
13    );
14  }
15  export default ListaHobbies;
16  |
```

3b)

```
App.js ×
src > App.js > ListaFood
1  function ListaFood() {
2    const comidas = ["Risoto", "Macarronada", "Stronogoff", "Salmão"];
3
4    return (
5      <div>
6        <h2> Minhas Comidas Preferidas</h2>
7        <ul>
8          {comidas.map((hobby, index) => (
9            <li key={index}>{hobby}</li>
10          ))}
11        </ul>
12      </div>
13    );
14  }
15  export default ListaFood;
16
```

3c)

```
App.jsx X
vite-project > src > App.jsx > MenuRestaurante
1 import { useState } from 'react'
2 import reactLogo from './assets/react.svg'
3 import viteLogo from '/vite.svg'
4 import './App.css'
5
6 function MenuRestaurante(){
7   const pratos = [
8     {nome: "Lasanha", preco: 25.00, descricao: "Lasanha a Milanese"},
9     {nome: "Coxinha de Frango", preco: 5.00, descricao: "Coxinha de Frango com Catupiry"},
10    {nome: "Mini Pizza de Calabresa", preco: 7.50, descricao: "Mini Pizza"},
11    {nome: "Risoto de Camarão", preco: 30.00, descricao: "Risoto de Camarão"}
12  ]
13
14  };
15
16  return (
17    <div>
18      <h1>Cardápio do Restaurante</h1>
19      <div className = "menu-grid">
20        {pratos.map((prato,index) => (
21          <div key ={index} className = "prato-card">
22            <h3>{prato.nome}</h3>
23            <p className = "preco">R$ {prato.preco.toFixed(2)}</p>
24            <p className = "descricao">{prato.descricao}</p>
25          </div>
26        )
27      )}
28    </div>
29  </div>
30  )
31
32  }
33
34  export default MenuRestaurante
35
```

```
.prato-card{
  color: ■ rgb(255, 250, 250);
  border: 3px solid □ rgb(2, 2, 2);
  background: □ black;
  margin: 20px;
}
```

4a)

```
App.jsx X # App.css
vite-project > src > App.jsx > ...
1 import { useState } from 'react'
2 import reactLogo from './assets/react.svg'
3 import viteLogo from '/vite.svg'
4 import './App.css'
5
6 function CartaoPessoa({nome, idade, profissao}) {
7   return (
8     <div className = "cartao-pessoa">
9       <h2>{nome}</h2>
10      <p><strong>Idade</strong>: {idade} anos</p>
11      <p><strong>Profissão</strong>: {profissao}</p>
12    </div>
13  );
14 }
15
16 function App(){
17   return (
18     <div>
19       <CartaoPessoa nome = "Matheus" idade = {18} profissao = "Programador"/>
20     </div>
21   );
22 }
23
24 export default App
25
26
```

```
.cartao-pessoa{
  border: 3px solid black;
  padding: (0 0 10px 0);
  color: rgb(63, 136, 165);
}
```

4b)

```
App.jsx x # App.css
vite-project > src > App.jsx > CartaoLivre
1  import { useState } from 'react'
2  import reactLogo from './assets/react.svg'
3  import viteLogo from '/vite.svg'
4  import './App.css'
5
6  function CartaoLivre({titulo, autor, ano, gênero}) {
7    return (
8      <div className = "cartao-livre">
9        <h2>{titulo}</h2>
10       <p><strong>Autor</strong>: {autor}</p>
11       <p><strong>Ano</strong>: {ano}</p>
12       <p><strong>Gênero</strong>:{gênero}</p>
13     </div>
14   );
15 }
16
17 function App(){
18   return (
19     <div>
20       <CartaoLivre titulo = "Demon Slayer" autor = "Matheus" ano = {2017} gênero = "Ação"/>
21     </div>
22   );
23 }
24
25 export default App
26
27
```

4c)

App.jsx X # App.css

vite-project > src > App.jsx > App

```
1  import './App.css'
2
3  function PrevisaoTempo({temperatura,clima,umidade, cidade}) {
4    const getIcone = (clima) => {
5      const climas = {
6        'ensolarado': '☀️',
7        'nublado': '☁️',
8        'chuvoso': '🌧️',
9        'tempestuoso': '🌩️',
10       'nevando': '❄️',
11     };
12     return climas[clima.toLowerCase()] || '☀️';
13   }
14   const getCorTemperatura = (temp) => {
15     if (temp < 15) return '❄️#4a90e2'; //frio
16     if (temp < 25) return '🌞#f5a623'; //ameno
17     return '🔥#d0021b'; //quente
18   };
19
20   return (
21     <div style = {{
22       border: '2px solid 🟡#ddd',
23       borderRadius: '15px',
24       padding: '25px',
25       textAlign: 'center',
26       background: 'linear-gradient(135deg, 🟡#2062a3ff, 🟡#1e79beff',
27       color: 'white',
28       maxWidth: '200px',
29       minWidth: '200px'
30     }}>
31       <h2 style = {{margin: '0 0 15px 0'}}>{cidade}</h2>
32       <div style = {{fontSize: '60px', margin: '10px 0'}}>
33         {getIcone(clima)}
34       </div>
35       <div style={{
```

```

36     fontSize: '48px',
37     fontWeight: 'bold',
38     color: getCorTemperatura(temperatura),
39     textShadow: '2px 2px 4px rgba(0,0,0,0.3)'
40   })
41   {temperatura}°C
42 </div>
43 <p style = {{fontSize: '18px', margin: '10px 0'}}>{clima}</p>
44 <p style = {{fontSize: '14px', opacity: 0.9}}>{umidade}%</p>
45 </div>
46 );
47 }
48
49 function App() {
50   const containerStyle = {
51     display: 'flex',
52     justifyContent: 'center',
53     alignItems: 'center',
54     gap: '20px',
55     marginTop: '50px',
56     flexWrap: 'wrap',
57     width: '100%',
58   };
59   return(
60     <div style = {containerStyle}>
61       <PrevisaoTempo temperatura = {31} clima = 'ensolarado' cidade = "RJ" umidade={60}/>
62       <PrevisaoTempo temperatura = {20} clima = 'nublado' cidade = "SP" umidade={30}/>
63       <PrevisaoTempo temperatura = {12} clima = 'nevando' cidade = "Curitiba" umidade={60}/>
64     </div>
65   )
66   export default App

```

5a)

```

App.jsx  X  # App.css
vite-project > src > App.jsx > [default]
1  import { useState } from 'react'
2  import './App.css'
3
4  function Contador () {
5    const [count, setCount] = useState(0);
6
7    return (
8      <div>
9        <h2>Contador: {count}</h2>
10       <button onClick = {(()) => setCount(count + 1)}>+1</button>
11       <button onClick = {(()) => setCount(count - 1)}>-1</button>
12     </div>
13   )
14 }
15 export default Contador;

```

5b)

```
App.jsx  X  # App.css
vite-project > src > App.jsx > [🔍] default
1  import { useState } from 'react'
2  import './App.css'
3
4  function Contador () {
5    const [count, setCount] = useState(10);
6
7    return (
8      <div>
9        <h2>Contador: {count}</h2>
10       <button onClick = {() => setCount(count + 5)}>+5</button>
11       <button onClick = {() => setCount(count - 5)}>-5</button>
12     </div>
13   )
14 }
15 export default Contador;
```

5c)

```
App.jsx  X  # App.css
vite-project > src > App.jsx > Termometro
1  import { useState } from 'react'
2  import './App.css'
3
4  function Termometro () {
5    const [count, setCount] = useState(20);
6    const getCorTemperatura = (count) =>
7    {
8      if (count <= 19) return '■ #4a90e2';
9      if (count > 20) return '■ #ee2828ff';
10   }
11   return (
12     <div>
13       <div>
14         <h1 style = {{
15           color: 'Grey',
16           textAlign: 'center'
17         }}>Termômetro Digital</h1>
18       </div>
19
20       <div style={{
21         color: getCorTemperatura(count)
22       }}>
23         <h2>🌡 Temperatura: {count}°C</h2>
24         <button className = "aquecer-botao" onClick = {() => setCount(count + 2)}>🔥 (+2°C)</button>
25         <button className = "esfriar-botao" onClick = {() => setCount(count - 2)}>❄ (-2°C)</button>
26       </div>
27     </div>
28   )
29 }
30
31 export default Termometro;
32
```

6a)

```
App.jsx X # App.css
vite-project > src > App.jsx > Mostrar_Esconder
1  import { useState } from 'react'
2  import './App.css'
3
4  function Mostrar_Esconder(){
5      const [mostrar, setMostrar] = useState(true);
6      return(
7          <div>
8              <button onClick = {() => setMostrar(!mostrar)}>
9                  {mostrar ? 'Esconder' : 'Mostrar'}
10             </button>
11             {mostrar && <p>Hello</p>}
12         </div>
13     );
14 }
15
16 export default Mostrar_Esconder;
```

6b)

```
EXPLORER ... App.jsx X download.png # App.css
vite-project > src > App.jsx > default
1  import { useState } from 'react'
2  import './App.css'
3
4  function Mostrar_Esconder(){
5      const [mostrar, setMostrar] = useState(true);
6      return(
7          <div>
8              <button onClick = {() => setMostrar(!mostrar)}>
9                  {mostrar ? 'Esconder' : 'Mostrar'}
10             </button>
11             {mostrar && <img src='src/assets/download.png' alt="Trailer" />}
12         </div>
13     );
14 }
15
16 export default Mostrar_Esconder;
```

NOVA PASTA

- vite-project
 - node_modules
 - public
 - vite.svg
 - src
 - assets
 - download.png
 - react.svg
 - App.css
 - App.jsx
 - index.css
 - main.jsx
 - .gitignore
 - eslint.config.js
 - index.html
 - package-lock.json
 - package.json
 - README.md
 - vite.config.js

6c)

```
App.jsx  X  download.png  # App.css
vite-project > src > App.jsx > ...
1  import { useState } from 'react'
2  import './App.css'
3
4  function Lista_Itens(){
5      const itens = ["Diamante", "Ouro", "Ferro", "Esmeralda"];
6      return(
7          <div>
8              <ul>
9                  {itens.map((item, index) => (
10                      <li key = {index}>{item}</li>
11                  ))}
12              </ul>
13          </div>
14      )
15  }
16
17  function Cofre_Digital(){
18      const [abrir, setAbrir] = useState(true);
19      return(
20          <div>
21              <button onClick = {(()) => setAbrir(!abrir)}>
22                  {abrir ? 'Fechar Cofre 🔒' : 'Abrir Cofre 🔓'}
23              </button>
24              {abrir && <Lista_Itens/>}
25          </div>
26      );
27  }
28
29
30  export default Cofre_Digital;
31
```

7a)

```
vite-project > src > App.jsx > InputTempoReal
1  import { useState } from 'react'
2  import './App.css'
3
4  function InputTempoReal () {
5    const [texto, setTexto] = useState('')
6
7    return(
8      <div>
9        <input
10         type="text"
11         value = {texto}
12         onChange={ (e) => setTexto(e.target.value)}
13         placeholder='Digite algo: ' />
14        <p>Você digitou: {texto}</p>
15      </div>
16    )
17  }
18
19
20  export default InputTempoReal ;
21
```

7b)

```
App.jsx × download.png # App.css
vite-project > src > App.jsx > InputTempoReal
1  import { useState } from 'react'
2  import './App.css'
3
4  function InputTempoReal () {
5    const [texto, setTexto] = useState('')
6    const texto_maiusculo = texto.toUpperCase()
7
8    return(
9      <div>
10        <input
11          type="text"
12          value = {texto}
13          onChange={e => setTexto(e.target.value)}
14          placeholder='Digite algo: ' />
15        <h2>Você digitou: {texto_maiusculo}</h2>
16      </div>
17    )
18  }
19
20
21  export default InputTempoReal ;
22
```

7c)

App.jsx X download.png # App.css

vite-project > src > App.jsx > ...

```
1  import { useState } from 'react'
2  import './App.css'
3
4  function InputTempoReal () {
5    const [texto, setTexto] = useState('')
6    const textom = texto.length
7
8    return(
9      <div>
10        <input
11          type="text"
12          value = {texto}
13          onChange={ (e) => setTexto(e.target.value)}
14          placeholder='Digite algo: ' />
15        <h2>Você digitou: {texto}</h2>
16        {textom > 0 && textom < 3 && (
17          <p className = 'vermelho'>Mínimo de 3 caracteres</p>
18        )}
19        {textom >= 3 && (
20          <p className = 'verde'>Mais de 3 caracteres</p>
21        )}
22      </div>
23    )
24  }
25
26
27
28  export default InputTempoReal;
29
```


7d)

```
App.jsx × download.png # App.css
vite-project > src > App.jsx > InputTempoReal
1  import { useState } from 'react'
2  import './App.css'
3
4  function InputTempoReal () {
5    const [texto, setTexto] = useState('')
6    const textom = texto.length
7
8    return (
9      <div>
10        <input
11          type="text" maxLength={50}
12          value = {texto}
13          onChange={(e) => setTexto(e.target.value)}
14          placeholder='Digite algo: ' />
15        <h2>Você digitou: {texto} Total de Caracteres: {textom}/50</h2>
16      </div>
17    )
18
19  }
20
21
22  export default InputTempoReal;
23
```

7e)

```
App.jsx X download.png # App.css
vite-project > src > App.jsx > ...
1 import { useState } from 'react'
2 import './App.css'
3
4 function GeradorSenha (){
5   const [texto, setTexto] = useState('');
6   const textom = texto.length;
7   const texto_anagrama = texto.split('').reverse('').join('');
8   const caracteres_esp = '&*@#';
9
10  return(
11    <div>
12      <h1>🔒 Gerador de Senha Mágica </h1>
13      <input
14        type="text"
15        value = {texto}
16        onChange={(e) => setTexto(e.target.value)}
17        placeholder='Digite algo: ' />
18      <h2>Você digitou: {texto} </h2>
19      {textom > 0 && (
20        <h2>Senha Criptografada: {texto_anagrama}{textom}{textom}{caracteres_esp} </h2>)}
21    </div>
22  )
23
24 }
25
26 export default GeradorSenha;
27
```

```

vite-project > src > App.jsx > GeradorSenha
1  import { useState } from 'react'
2  import './App.css'
3
4  function GeradorSenha () {
5    const [texto, setTexto] = useState('');
6
7    const textom = texto.length;
8    const texto_anagrama = texto.split('').reverse().join('');
9    const caracteres_esp = '&*@#';
10
11    return(
12      <div>
13        <h1>🔑 Gerador de Senha Mágica </h1>
14        <input
15          type="text"
16          value = {texto}
17          onChange={e => setTexto(e.target.value)}
18          placeholder='Digite algo: '/>
19        <h2>Você digitou: {texto} </h2>
20        {textom > 0 && (
21          <h2>Senha Criptografada: {texto_anagrama}{textom}{textom}{caracteres_esp} </h2>)}
22
23        {textom > 0 && textom <= 6 && (<p className = 'fraca'>Senha: Fraca</p>)}
24        {textom > 6 && textom <= 10 && (<p className = 'media'>Senha: Média</p>)}
25        {textom > 10 && (<p className = 'forte'>Senha: Forte</p>)}
26      </div>
27    )}
28
29
30  export default GeradorSenha;
31

```

```

.fraca{
  color: ■ red
}
.media{
  color: ■ rgb(255, 187, 0)
}
.forte{
  color: ■ rgb(43, 255, 0)
}

```

8a)

```
App.jsx X # App.css
vite-project > src > App.jsx > AdicionarLista
1 import { useState } from 'react'
2 import './App.css'
3
4 function AdicionarLista(){
5   const [item, setItem] = useState('');
6   const [lista, setLista] = useState([]);
7
8   const adicionaritem = () => {
9     if (item.trim()) {
10       setLista([...lista, item]);
11       setItem('');
12     }
13   };
14
15   return (
16     <div>
17       <input type="text" value={item} onChange={(e) => setItem(e.target.value)} placeholder="Digite um item"/>
18       <button onClick={adicionaritem}>Adicionar</button>
19       <ul>
20         {lista.map((itemLista, index) => (
21           <li key={index}>{itemLista}</li>
22         ))}
23       </ul>
24     </div>
25   )
26 }
27
28 export default AdicionarLista;
29
```

8b)

```
App.jsx X # App.css
vite-project > src > App.jsx > AdicionarTarefas
1 import { useState } from 'react'
2 import './App.css'
3
4 function AdicionarTarefas(){
5   const [tarefa, setTarefa] = useState('');
6   const [lista, setLista] = useState([]);
7
8   const adicionarTarefa = () => {
9     if (tarefa.trim()) {
10       setLista([...lista, tarefa]);
11       setTarefa('');
12     }
13   };
14
15   return (
16     <div>
17       <input type="text" value={tarefa} onChange={(e) => setTarefa(e.target.value)} placeholder="Digite um item"/>
18       <button onClick={adicionarTarefa}>Nova Tarefa</button>
19       <ul>
20         {lista.map((tarefaLista, index) => (
21           <li key={index}>{tarefaLista}</li>
22         ))}
23       </ul>
24     </div>
25   )
26 }
27
28 export default AdicionarTarefas;
29
```

8c)

```
App.jsx X # App.css
vite-project > src > App.jsx > PlacarDeJogos > lista.map() callback
4 function PlacarDeJogos() {
5   const [time, setTime] = useState('')
6   const [pontos, setPontos] = useState('')
7   const [lista, setLista] = useState([])
8
9   const adicionarTime = () => {
10     const nome = time.trim()
11     const valor = parseInt(pontos, 10)
12
13     if (!nome) return
14     if (Number.isNaN(valor)) return
15
16     const novo = { nome_time: nome, pontos: valor }
17
18     const novaLista = [...lista, novo].sort((a, b) => b.pontos - a.pontos)
19
20     setLista(novaLista)
21     setTime('')
22     setPontos('')
23   }
24
25   return (
26     <div>
27       <h2>Placar de Jogos</h2>
28
29       <div>
30         <input
31           type="text"
32           value={time}
33           onChange={(e) => setTime(e.target.value)}
34           placeholder="Digite um time"
35         />
36         <input
37           type="number"
38           value={pontos}
39           onChange={(e) => setPontos(e.target.value)}
40           placeholder="Pontos"
41         />
42         <button onClick={adicionarTime}>Novo Time</button>
43       </div>
44
45       <ul>
46         {lista.map((t, index) => (
47           <li key={` ${t.nome_time}-${index}`} >{t.nome_time} - {t.pontos} pts</li>
48         ))}
49       </ul>
50     </div>
51   )
52 }
```

8d)

9A

```
meu-projeto-react > src > App.jsx > default
1  import React, {useState} from 'react';
2  import './App.css'
3
4  function AlterarCorFundo(){
5    const [cor, setCor] = useState('white');
6
7    const mudarCor = (novaCor) => {
8      setCor(novaCor);
9      document.body.style.backgroundColor = novaCor;
10   };
11
12   return (
13     <div>
14       <h2>Alterar cor de fundo</h2>
15       <button onClick={() => mudarCor('lightblue')}>Azul</button>
16       <button onClick={() => mudarCor('lightgreen')}>Verde</button>
17       <button onClick={() => mudarCor('lightcoral')}>Coral</button>
18       <button onClick={() => mudarCor('White')}>Branco</button>
19     </div>
20   );
21 }
22
23 export default AlterarCorFundo;
```

9B

```
meu-projeto-react > src > App.jsx > default
1 import React, { useState } from 'react';
2 import './App.css';
3
4 function AlterarTamanhoFonte() {
5   const [tamanho, setTamanho] = useState('16px');
6
7   const mudarTamanho = (novoTamanho) => {
8     setTamanho(novoTamanho);
9     document.body.style.fontSize = novoTamanho;
10  };
11
12  return (
13    <div>
14      <h2>Alterar tamanho de fonte</h2>
15      <button onClick={() => mudarTamanho('12px')}>Pequeno</button>
16      <button onClick={() => mudarTamanho('16px')}>Médio</button>
17      <button onClick={() => mudarTamanho('20px')}>Grande</button>
18      <p style={{ fontSize: tamanho }}>
19        Este texto muda de tamanho conforme você clica nos botões acima.
20      </p>
21    </div>
22  );
23 }
24
25 export default AlterarTamanhoFonte;
```

9C

```
meu-projeto-react > src > App.jsx > ...
1 import React, { useState } from 'react';
2 import './App.css';
3
4 function SimuladorDeHumor() {
5   const [humor, setHumor] = useState({emoji: '😄', cor: '#FFD700', fala: 'Haha!'});
6 }
7
8 const mudarHumor = (novoHumor) => {
9   setHumor(novoHumor);
10  document.body.style.backgroundColor = novoHumor.cor;
11  alert(novoHumor.fala);
12 };
13
14
15 return (
16   <div>
17     <h2>Simulador de Humor</h2>
18     <div style={{ fontSize: '100px', textAlign: 'center' }}>
19       {humor.emoji}
20     </div>
21     <button onClick={() => mudarHumor({emoji: '😄', cor: '#FFD700', fala: 'Haha!'}}>Feliz</button>
22     <button onClick={() => mudarHumor({emoji: '😞', cor: '#4169E1', fala: 'Sniff...'})>Triste</button>
23     <button onClick={() => mudarHumor({emoji: '😡', cor: '#DC143C', fala: 'Grrrr!'}>Raivoso</button>
24     <button onClick={() => mudarHumor({emoji: '😴', cor: '#98FB98', fala: 'Ahhhh...'})>Calmo</button>
25   </div>
26 );
27 }
28
29 export default SimuladorDeHumor;
30
```

10a)

```
import { useState } from 'react';
import './App.css';

function Calculadora() {
  const [num1, setNum1] = useState('');
  const [num2, setNum2] = useState('');
  const [resultado, setResultado] = useState(null);
  const handleOperacao = (operador) => {
    const n1 = parseFloat(num1);
    const n2 = parseFloat(num2);

    if (isNaN(n1) || isNaN(n2)) {
      setResultado('Erro: Insira números válidos');
      return;
    }

    let res;
    switch (operador) {
      case '+':
        res = n1 + n2;
        break;
      case '-':
        res = n1 - n2;
        break;
      case '*':
        res = n1 * n2;
        break;
      case '/':
        if (n2 === 0) {
          setResultado('Erro: Divisão por zero');
          return;
        }
        res = n1 / n2;
        break;
      default:
        return;
    }
    setResultado(res);
  };

  const limpar = () => {
    setNum1('');
  };
}
```



```

    setNum2('');
    setResultado(null);
  };
  return (
    <div className="calculadora-container">
      <h1>Calculadora Simples</h1>
      <div className="inputs-container">
        <input
          type="number"
          value={num1}
          onChange={ (e) => setNum1(e.target.value) }
          placeholder="Primeiro número"
        />
        <input
          type="number"
          value={num2}
          onChange={ (e) => setNum2(e.target.value) }
          placeholder="Segundo número"
        />
      </div>
      <div className="botoes-container">
        <button onClick={ () => handleOperacao('+') }>+</button>
        <button onClick={ () => handleOperacao('-') }>-</button>
        <button onClick={ () => handleOperacao('*') }>*</button>
        <button onClick={ () => handleOperacao('/') }>/</button>
        <button onClick={limpar}
className="botao-limpar">Limpar</button>

      </div>

      {resultado !== null && (
        <h3 className="resultado">
          Resultado: {resultado}
        </h3>
      )}

    </div>

  );
}

```

```
export default Calculadora;
```

10b)

```
import { useState } from 'react';
import './App.css';
function Calculadora() {
  const [num1, setNum1] = useState('');
  const [num2, setNum2] = useState('');
  const [resultado, setResultado] = useState(null);
  const handleOperacao = (operador) => {
    const n1 = parseFloat(num1);
    const n2 = parseFloat(num2);
    if (isNaN(n1) || isNaN(n2)) {
      setResultado('Erro: Insira números válidos');
      return;
    }
    let res;
    switch (operador) {
      case 'Calcular a area':
        res = n1 * n2;
        break;
      case '/':
        if (n2 === 0) {
          setResultado('Erro: Divisão por zero');
          return;
        }
    }
    setResultado(res);
  };
  const limpar = () => {
    setNum1('');
    setNum2('');
    setResultado(null);
  };
  return (
    <div >
```

```

<h1>Calculadora Simples</h1>

<div>
  <input
    type="number"
    value={num1}
    onChange={(e) => setNum1(e.target.value)}
    placeholder="Digite o Comprimento"
  />
  <input
    type="number"
    value={num2}
    onChange={(e) => setNum2(e.target.value)}
    placeholder="Digite sua Largura"
  />
</div>

<div>
  <button onClick={() => handleOperacao('Calcular a
area')}>Calcular a area</button>
  <button onClick={limpar}
className="botao-limpar">Limpar</button>
</div>

{resultado !== null && (
  <h3 className="resultado">
    Resultado: {resultado}
  </h3>
)}
</div>
);
}

export default Calculadora;

```

10c)

```

import {useState} from 'react';
import './App.css';

const taxasParaBRL = {

```

```

    BRL: 1.00,
    USD: 5.39,
    EUR: 6.25,
  };

function ConversorDeMoedas() {
  const [valor, setValor] = useState('');
  const [moedaOrigem, setMoedaOrigem] = useState('BRL');
  const [moedaDestino, setMoedaDestino] = useState('USD');
  const [resultado, setResultado] = useState(null);

  const handleConversao = () => {
    const valorNum = parseFloat(valor);
    if (isNaN(valorNum) || valorNum <= 0) {
      setResultado('Erro: Insira um valor válido');
      return;
    }

    if (moedaOrigem === moedaDestino) {
      setResultado(`Resultado: ${valorNum.toFixed(2)}
${moedaDestino}`);
      return;
    }

    const valorEmBRL = valorNum * taxasParaBRL[moedaOrigem];
    const valorFinal = valorEmBRL / taxasParaBRL[moedaDestino];
    const resultadoFormatado = `${valorNum.toFixed(2)} ${moedaOrigem} =
${valorFinal.toFixed(2)} ${moedaDestino}`;
    setResultado(resultadoFormatado);
  };

  const limpar = () => {
    setValor('');
    setResultado(null);
    setMoedaOrigem('BRL');
    setMoedaDestino('USD');
  };

  return (
    <div className="calculadora-container">
      <h1>Conversor de Moedas</h1>
      <div className="conversor-body">

```

```

<div className="input-group">
  <label htmlFor="valor">Valor:</label>
  <input
    id="valor"
    type="number"
    value={valor}
    onChange={(e) => setValor(e.target.value)}
    placeholder="Digite o valor"

  />
</div>

<div className="input-group">
  <label htmlFor="origem">De:</label>
  <select
    id="origem"
    value={moedaOrigem}
    onChange={(e) => setMoedaOrigem(e.target.value)}

  >
    <option value="BRL">Real (BRL)</option>
    <option value="USD">Dólar (USD)</option>
    <option value="EUR">Euro (EUR)</option>
  </select>
</div>

<div className="input-group">
  <label htmlFor="destino">Para:</label>
  <select
    id="destino"
    value={moedaDestino}
    onChange={(e) => setMoedaDestino(e.target.value)}

  >

    <option value="BRL">Real (BRL)</option>
    <option value="USD">Dólar (USD)</option>
    <option value="EUR">Euro (EUR)</option>

  </select>
</div>
</div>

<div className="botoes-container">

```

```

        <button onClick={handleConversao}
className="botao-principal">Converter</button>
        <button onClick={limpar}
className="botao-limpar">Limpar</button>
      </div>
      {resultado !== null && (
        <h3 className="resultado">
          {resultado}
        </h3>
      )}
    </div>
  );
}

export default ConversorDeMoedas;

```

11a)

```

import { useState } from 'react';
import './App.css'

function FormularioTempoReal() {
  const [dados, setDados] = useState({
    nome: '',
    email: '',
    idade: ''
  });

  const handleChange = (e) => {
    const {name, value} = e.target;
    setDados({
      ...dados,
      [name]: value
    });
  };

  return(
    <div>
      <h2>Formulário</h2>
      <form>
        <div>
          <label> Nome: </label>

```

```

        <input type="text" name ="nome" value={dados.nome} onChange =
{handleChange}/>
      </div>
      <div>
        <label> E-mail: </label>
        <input type="email" name ="email" value={dados.email} onChange
= {handleChange}/>
      </div>
      <div>
        <label> Idade: </label>
        <input type="number" name='idade' value={dados.idade} onChange
= {handleChange}/>
      </div>
    </form>

    <div>
      <h3>Dados preenchidos:</h3>
      <h4>Seu nome é: {dados.nome}</h4>
      <h4>Seu e-mail é: {dados.email}</h4>
      <h4>Sua idade é: {dados.idade}</h4>
    </div>
  </div>
)
}
export default FormularioTempoReal;

```

11c)

```

import { useState } from 'react';
import './App.css'

function FormularioTempoReal() {
  const [dados, setDados] = useState({
    cidade: '',
    estado: '',
    cep: ''
  });

  const handleChange = (e) => {
    const {name, value} = e.target;
    setDados({
      ...dados,
      [name]: value
    });
  };

```

```

};

return(
<div>
  <h2>Formulário</h2>
  <form>
    <div>
      <label> Cidade: </label>
      <input type="text" name="cidade" value={dados.cidade} onChange
= {handleChange}/>
    </div>
    <div>
      <label> Estado: </label>
      <input type="text" name="estado" value={dados.estado} onChange
= {handleChange}/>
    </div>
    <div>
      <label> CEP: </label>
      <input type="number" name='cep' value={dados.cep} onChange =
{handleChange}/>
    </div>
  </form>

  <div>
    <h3>Dados preenchidos:</h3>
    <h4>Seu cidade é: {dados.cidade}</h4>
    <h4>Seu estado é: {dados.estado}</h4>
    <h4>Sua cep é: {dados.cep}</h4>
  </div>
</div>
)
}
export default FormularioTempoReal;

```

11d)

```

import { useState } from 'react';
import './App.css'

function Avatar(){
  const [avatar, setAvatar] = useState({
    cabelo: '',
    olhos: '',
    acessorios: []

```



```

});

const handleChange = (e) => {
  const { name, value, type, checked } = e.target;

  if (type === 'checkbox') {
    setAvatar(prevState => {
      const novosAcessorios = checked
        ? [...prevState.acessorios, value]
        : prevState.acessorios.filter(item => item !== value);
      return { ...prevState, acessorios: novosAcessorios };
    });
  } else {
    setAvatar(prevState => ({
      ...prevState,
      [name]: value
    }));
  }
};

return (
  <div>
    <h2>Criador de Avatar</h2>
    <form>
      <div>
        <label>Cor do Cabelo: </label>
        <select name="cabelo" value={avatar.cabelo}
onChange={handleChange}>
          <option value="">Selecione uma cor</option>
          <option value="Loiro">Loiro</option>
          <option value="Castanho">Castanho</option>
          <option value="Preto">Preto</option>
          <option value="Ruivo">Ruivo</option>
          <option value="Colorido">Colorido</option>
        </select>
      </div>

      <div>
        <label>Cor dos Olhos: </label>
        <label>
          <input type="radio" name="olhos" value="Azuis"
checked={avatar.olhos === 'Azuis'} onChange={handleChange} /> Azuis
        </label>

```

```

        <label>
            <input type="radio" name="olhos" value="Verdes"
checked={avatar.olhos === 'Verdes'} onChange={handleChange} /> Verdes
        </label>
        <label>
            <input type="radio" name="olhos" value="Castanhos"
checked={avatar.olhos === 'Castanhos'} onChange={handleChange} />
Castanhos
        </label>
    </div>

    <div>
        <label>Acessórios: </label>
        <label>
            <input type="checkbox" name="acessorios" value="Óculos"
checked={avatar.acessorios.includes('Óculos')} onChange={handleChange}
/> Óculos
        </label>
        <label>
            <input type="checkbox" name="acessorios" value="Chapéu"
checked={avatar.acessorios.includes('Chapéu')} onChange={handleChange}
/> Chapéu
        </label>
        <label>
            <input type="checkbox" name="acessorios" value="Brinco"
checked={avatar.acessorios.includes('Brinco')} onChange={handleChange}
/> Brinco
        </label>
    </div>
</form>

<div>
    <h3>Avatar Montado:</h3>
    <p>
        <strong>Cabelo:</strong> {avatar.cabelo || 'Não definido'} |
        <strong> Olhos:</strong> {avatar.olhos || 'Não definido'} |
        <strong> Acessórios:</strong> {avatar.acessorios.length > 0 ?
avatar.acessorios.join(', ') : 'Nenhum'}
    </p>
</div>
</div>
);
}

```

```
export default Avatar;
```