

**Vitor Amadeu Souza**

Introdução ao

**React**

e

**CSS**

No ambiente Replit

Parte II

© 2024 by Cerne Tecnologia e Treinamento Ltda.

© 2024 by Vitor Amadeu Souza

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

**Agosto de 2024**

Direitos reservados por:

Cerne Tecnologia e Treinamento Ltda

*Produção: Cerne Tecnologia e Treinamento*

*E-mail da Empresa: cerne@cerne-tec.com.br*

*Home Page: www.cerne-tec.com.br.com.br*

*Atendimento ao Consumidor: sac@cerne-tec.com.br*

*Contato com o Autor: vitor@cerne-tec.com.br*



**FEITO NO BRASIL**

***“Venha também sobre mim a tua benignidade, ó Senhor, e a tua  
salvação, segundo a tua palavra.”***

**Sl 119:41**

## **Cerne Tecnologia**

A Cerne Tecnologia tem uma equipe preparada para desenvolvimento de projetos eletrônicos em diversas áreas: Médica, Entretenimento, Industrial, Robótica, Científica, Automobilística, Aeronáutica, etc. Trabalhamos com tecnologia microcontrolada usando o PIC, ARM, AVR, 8051, dsPIC, PIC24, PIC32 além do Arduino, Raspberry, Beaglebone etc. Desenvolvemos o projeto desde sua concepção até a entrega do produto final, passando pelas etapas de esquema elétrico, protótipo e desenvolvimento de circuito impresso.

Desenvolvemos aplicativos para smartphones/tablets Android, iOS, Blackberry, Windows Phone e no desenvolvimento de softwares a nível PC para plataforma Windows, usando ferramentas como o Visual Basic, C# e C++.

Atuamos na parte de montagem de placas, onde podemos fornecer ambos os serviços de desenvolvimento de projetos e produção ou apenas um destes.

Desenvolvemos esquemas elétricos e layout de PCI, tanto em tecnologia convencional como SMD.

Temos a flexibilidade de customizar um de nossos produtos, de modo a atender a uma necessidade específica do cliente, tornando o custo de desenvolvimento menor se comparado a construção de um projeto desde a sua fase inicial.

Desenvolvemos e fornecemos kits didáticos para diversos microcontroladores além de apostilas, livros e e-books.

Na hora de desenvolver um projeto ou equipar seu laboratório não hesite em nos contatar. Entre em contato conosco através do endereço [cerne-tec.com.br](http://cerne-tec.com.br) para obter mais informações.



**cerne-tec.com.br**

# Sumário

|   |          |
|---|----------|
| <b>Capítulo I – Metodologia de desenvolvimento.....</b> | <b>6</b> |
| 1. Introdução.....                                      | 6        |
| <b>Capítulo II – Programação em React e CSS.....</b>    | <b>7</b> |
| 1. Alterando a cor da div.....                          | 7        |
| 2. Alterando a cor de fundo.....                        | 10       |
| 3. Adicionando uma div.....                             | 13       |
| 4. Exemplo usando o flexbox.....                        | 17       |

# Capítulo I

## Metodologia de desenvolvimento

### 1. Introdução

Neste livro, você encontrará uma parte da série sobre o desenvolvimento de projetos utilizando React e CSS. Ao longo desta obra, diversos experimentos serão apresentados segundo o sumário da obra, permitindo o aperfeiçoamento neste framework para a construção de interfaces responsivas juntamente com o CSS.

# Capítulo II

## Programação em React e CSS

### 1. Alterando a cor da div

Baseado no último exemplo, o objetivo deste exercício é deixar a primeira div na cor vermelha e a segunda na cor azul e os textos em negrito. Para este exemplo, precisaremos modificar o arquivo App.css para que atenda a esta demanda, usando o `className` para citar a classe com as definições deita neste arquivo. Observe como fica o arquivo App.css, App.jsx e App1.jsx para deixar o exercício conforme solicitado.

#### App.css

```
.red
{
  background-color:red;
}
.azul
{
  background-color:blue;
}
h1
```

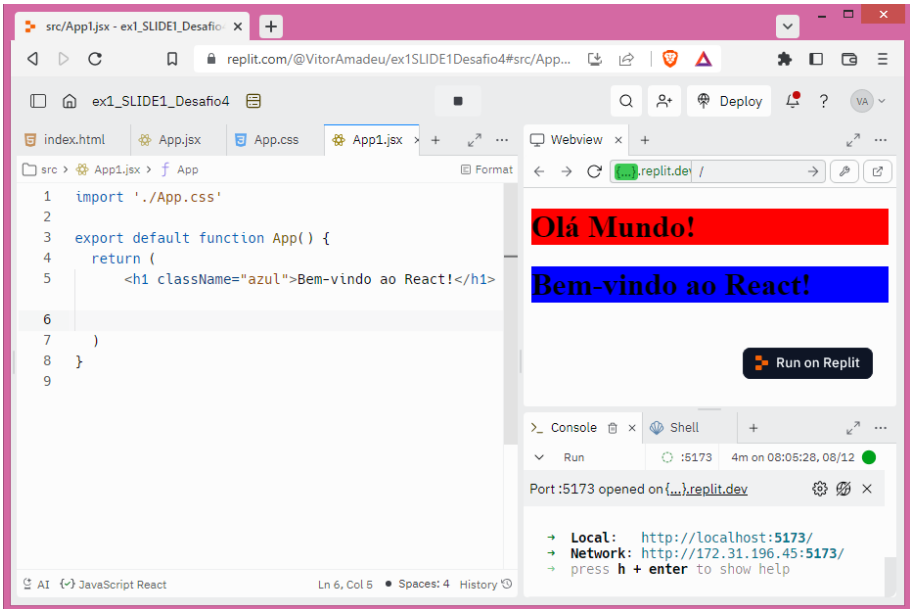
```
{  
  font-weight: bold;  
}
```

## App.jsx

```
import './App.css'  
  
export default function App() {  
  return (  
    <h1 className="red">Olá Mundo!</h1>  
  );  
}
```

## App1.jsx

```
import './App.css'  
  
export default function App() {  
  return (  
    <h1 className="azul">Bem-vindo ao React!</h1>  
  
  )  
}
```



Os códigos apresentados são dois trechos de uma aplicação React que utiliza CSS para estilizar o conteúdo. No primeiro trecho, uma classe CSS chamada `.red` é definida para aplicar um fundo vermelho (`background-color:red;`) a qualquer elemento HTML que utilize essa classe. Outra classe `.azul` é criada para aplicar um fundo azul (`background-color:blue;`). Além disso, uma regra CSS é estabelecida para que qualquer elemento `<h1>` tenha o texto em negrito, utilizando `font-weight: bold;`.

No código JavaScript que segue, há uma função chamada App, que é um componente React. Neste componente, o texto "Olá Mundo!" é envolvido por um elemento `<h1>` com a classe `red`. Quando este componente for renderizado, o texto "Olá Mundo!" aparecerá em negrito e com o fundo vermelho, devido ao uso da classe `red` e da regra CSS para o `<h1>`.

No segundo trecho, o código é similar, mas o componente App agora renderiza um texto diferente: "Bem-vindo ao React!". Desta vez, o elemento `<h1>` utiliza a classe `azul`, o que faz com que o fundo do texto seja azul. Assim, ao ser renderizado, o texto "Bem-vindo ao React!" aparecerá em negrito e com o fundo azul, devido à aplicação da classe `azul` e da regra CSS para o `<h1>`.

Ambos os trechos fazem uso do arquivo `App.css` para a estilização dos componentes.

## **2. Alterando a cor de fundo**

Baseado no último exemplo, o objetivo é deixar a tela na cor verde, mantendo o que foi feito no exemplo passado. Acompanhe os arquivos atualizados.