Vitor Amadeu Souza

Introdução ao

React

e

CSS

No ambiente Replit

Parte XI

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 para obter mais informações.

Cerne
Conhecimento para o Desenvolvimento

cerne-tec.com.br

Sumário

Capítulo I
Metodologia de desenvolvimento

1. Introdução

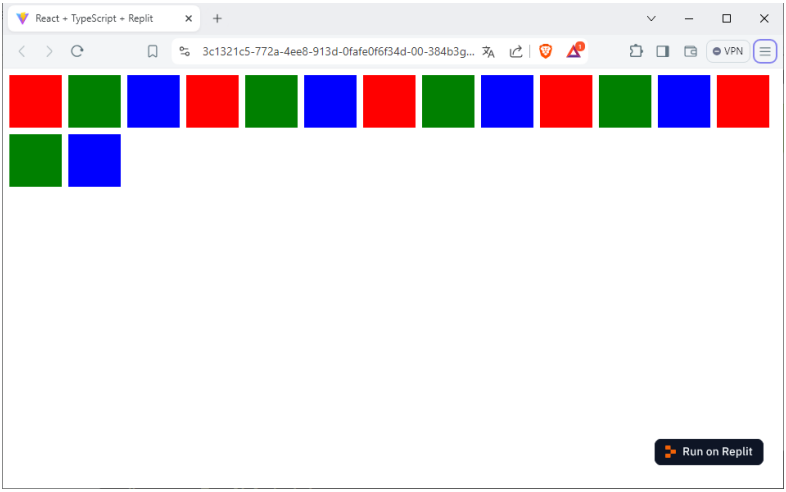
Neste livro, você encontrará uma parte da série sobre o desenvolvimento de projetos utilizando React e CSS. Ao longo desta obra, diversos experimentos serão apresentados segundo o sumário da obra, permitindo o aperfeiçoamento neste framework para a construção de interfaces responsivas juntamente com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

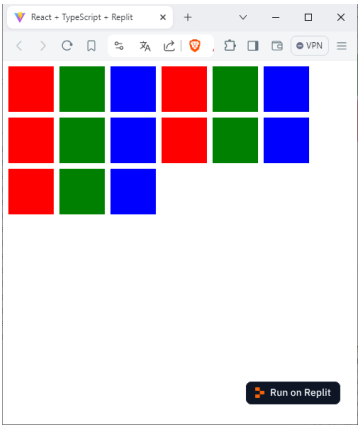
Capítulo II

Programação em React e CSS

1. Testando flex-wrap através de box

O objetivo deste exemplo é testar o flex-wrap usando para isso diversos boxes, que ao aumentar e diminuir o espaço lateral fornecem o efeito visual visto abaixo.





Acompanhe como ficam os arquivos App.css e App.jsx para este exercício.

App.css

```
.container
{
    display: flex;
    flex-direction: row;
    gap: 8px;
    flex-wrap: wrap;
}

.box
{
  width: 64px;
  height: 64px;
}

.green
{
  background-color: green;
}

.red
{
  background-color: red;
```

```
}

.blue
{
  background-color: blue;
}
```

App.jsx

```
import './App.css'

export default function App() {
  return (
    <main>

      <div className="container">
        <div className='box red'></div>
        <div className='box green'></div>
        <div className='box blue'></div>
        <div className='box red'></div>
        <div className='box green'></div>
        <div className='box blue'></div>
        <div className='box red'></div>
        <div className='box green'></div>
        <div className='box blue'></div>
        <div className='box red'></div>
        <div className='box green'></div>
        <div className='box blue'></div>
        <div className='box red'></div>
        <div className='box green'></div>
        <div className='box blue'></div>

      </div>
    </main>
  )
}
```

Este código React define um componente chamado App, que utiliza um arquivo de estilo chamado App.css para definir a aparência dos elementos na tela. No componente App, há uma estrutura que retorna um elemento main, que contém uma div com a classe container. Dentro dessa div, há várias outras divs menores, cada uma

com a classe box e uma classe adicional que define sua cor, como red, green ou blue.

No arquivo de estilo App.css, a classe container usa a propriedade display: flex para organizar os elementos box em uma linha horizontal (flex-direction: row), com um espaçamento de 8 pixels entre eles (gap: 8px). A propriedade flex-wrap: wrap permite que as caixas sejam distribuídas em múltiplas linhas, caso o espaço disponível na tela não seja suficiente para colocá-las todas em uma única linha. As divs com a classe box têm largura e altura fixas de 64 pixels, e as classes green, red e blue definem a cor de fundo dessas caixas, respectivamente verde, vermelho e azul. Como resultado, o componente App renderiza uma série de caixas coloridas organizadas em linhas e espaçadas uniformemente.

2. Carregando imagens em boxes

A proposta do próximo exemplo é carregar quatro imagens na mesma quantidade de boxes na tela, como apresentado a seguir.