

Vitor Amadeu Souza

Introdução ao

Tailwind CSS

Aplicações práticas – Parte IV

© 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.

Dezembro 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

*“Seja como a rocha que as ondas continuam a bater, mas que
permanece inabalável.”*

Marco Aurélio

Cerne Tecnologia

A Cerne Tecnologia tem uma equipe preparada para desenvolvimento de projetos elétricos, eletrônicos, automação 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 e iOS 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.



Sumário

Introdução	6
II. Tailwind CSS	7
1. tracking- <code>{size}</code>	7
2. flex	10
3. flex-row	13
4. flex-col	15
5. items-center	18

Introdução

A proposta desta série de livros é explorar o uso do Tailwind CSS, começando com uma introdução ao HTML e aos comandos iniciais do framework. Nas partes seguintes, são abordados diversos comandos e técnicas avançadas, permitindo ao leitor entender e aplicar de maneira eficaz as funcionalidades do Tailwind CSS em seus projetos.

Consulte o sumário de cada parte para conferir os temas detalhados abordados em cada etapa da série.

Capítulo I

Tailwind CSS

1. `tracking-{size}`

As classes utilitárias do Tailwind CSS para espaçamento entre letras são definidas pela convenção `tracking-{size}` e permitem ajustar o espaçamento entre os caracteres de um texto, proporcionando diferentes estilos de apresentação. As principais classes disponíveis incluem `tracking-tighter`, que aplica um espaçamento de letra de `-0.05em`, resultando em caracteres mais próximos; `tracking-tight`, que define um espaçamento de letra de `-0.025em`; e `tracking-normal`, que é o valor padrão, com um espaçamento de `0em`, sem ajustes.

Além disso, existem classes que aumentam o espaçamento, como `tracking-wide`, que aplica `0.025em`; `tracking-wider`, com `0.05em`; e `tracking-widest`, que define um espaçamento de `0.1em`, resultando em caracteres bem distantes. Essas classes podem ser aplicadas diretamente a elementos HTML usando a sintaxe apropriada, e o Tailwind também permite o uso de modificadores para aplicar essas classes em diferentes estados ou tamanhos de tela.

Os desenvolvedores têm a possibilidade de personalizar esses valores no arquivo de configuração do Tailwind, ajustando a seção `theme.letterSpacing`. Caso necessário, também é possível usar valores arbitrários com a sintaxe `tracking-[valor]`, oferecendo maior flexibilidade na definição do espaçamento. Essas utilidades são úteis para melhorar a legibilidade e a estética do texto em projetos web, permitindo controle preciso sobre a apresentação tipográfica.

Acompanhe um exemplo que faz uso deste comando.

```
<!DOCTYPE html>
<html lang="pt-BR">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemplo de tracking-{size} - Tailwind CSS</title>
  <script src="https://cdn.tailwindcss.com"></script>
</head>

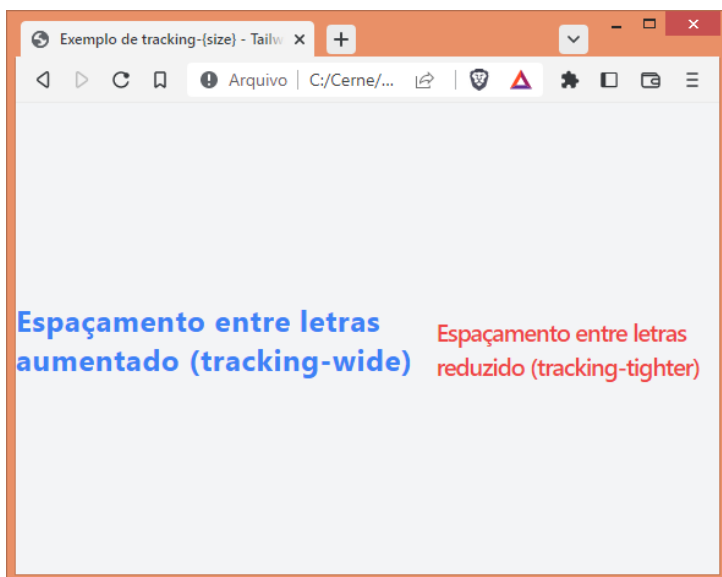
<body class="bg-gray-100 flex items-center justify-center h-screen">
  <div class="text-2xl tracking-wide text-blue-500 font-bold">
    Espaçamento entre letras aumentado (tracking-wide)
  </div>
  <div class="text-xl tracking-tighter text-red-500 font-medium mt-4">
    Espaçamento entre letras reduzido (tracking-tighter)
  </div>
</body>

</html>
```

No cabeçalho, são especificados o conjunto de caracteres, a viewport e o título da página. O Tailwind CSS é carregado através de um script, permitindo o uso de suas classes utilitárias.

No corpo do documento, há duas divs que contêm textos com diferentes espaçamentos entre letras. A primeira div usa a classe `tracking-wide`, que aumenta o espaçamento entre as letras e apresenta o texto em azul com tamanho 2xl e em negrito. A segunda div utiliza a classe `tracking-tighter`, que reduz o espaçamento entre as letras, apresentando o texto em vermelho com tamanho xl e em um estilo de fonte médio.

Esse exemplo ilustra como as classes do Tailwind CSS podem ser aplicadas para manipular o espaçamento entre letras, permitindo que os desenvolvedores ajustem a apresentação tipográfica de acordo com suas necessidades.



2. flex

O conceito de flex no Tailwind CSS é fundamental para criar layouts responsivos e dinâmicos. Para utilizar o flexbox, basta adicionar a classe flex a um contêiner, transformando seus filhos em itens flexíveis. O Tailwind oferece várias classes utilitárias para controlar o comportamento desses itens, como flex-1, flex-auto, flex-initial e flex-none. A classe flex-1 permite que um item cresça e encolha conforme necessário, ignorando seu tamanho inicial. A classe flex-auto faz com que um item cresça e encolha levando em conta seu tamanho inicial, enquanto flex-initial permite que um item encolha, mas não cresça, respeitando seu tamanho inicial. Por outro lado, flex-none impede que um item cresça ou encolha.

Além disso, o Tailwind CSS permite a aplicação condicional de classes utilizando modificadores de estado, como hover:flex-1, que aplica a classe apenas ao passar o mouse sobre o elemento. Também é possível usar modificadores para consultas de mídia, como md:flex-1, aplicando a classe apenas em tamanhos de tela médios ou maiores. O layout flexível pode ser ainda mais ajustado com classes que controlam a direção dos itens, como flex-row e flex-col, além do