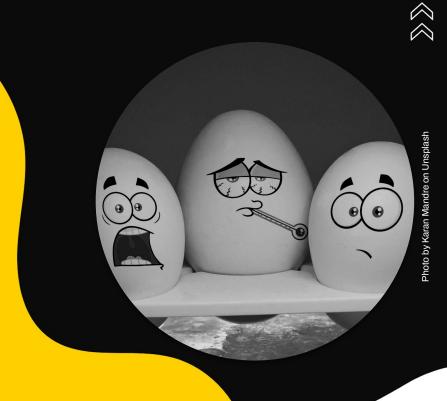


Módulo 3 - Aula 15 - Hard





 \mathbb{Z}







Vite é um build tool, serve para facilitar a configuração e instalação de vários projetos, com diversas tecnologias, como React e Vue.

Tecnologia que tem entrado em ascensão, por ser uma ferramenta de build (build tool) para diversas outras tecnologias e oferecer configurações iniciais melhores que outras do mercado.



Componentes

Podemos pensar em "funções", as quais podemos chamar e reutilizar diversas vezes ,desde que sejam passados os argumentos necessários. Isso torna o React poderoso e otimizado.

O ideal é dividir sua aplicação React em diversos componentes.

Props

As props são propriedades que podemos passar para componentes React, elas são somente de leitura e são passadas via atributo.

Props (2)

Outra forma de desestruturar as props é diretamente nos parênteses dos parâmetros.

Mão na Massa: Criar componente de botão

- Crie um novo projeto usando o vite
 - Dentro da pasta components criar um arquivo Button.jsx
 - Criar um objeto com 10 cores escolhidas pelo grupo
 - O componente Button terá uma prop chamada color
 - A cor de fundo deve ser alterada de acordo com a propriedade escrita em color
- Insira o componente Button no app.jsx





Hooks são funções que permitem a conexão dos recursos de estado e ciclo de vida do React, a partir de componentes funcionais.

Eles não substituem conceitos básicos no React

São retrocompatíveis, ou seja, não possuem nenhuma quebra de compatibilidade.

useState

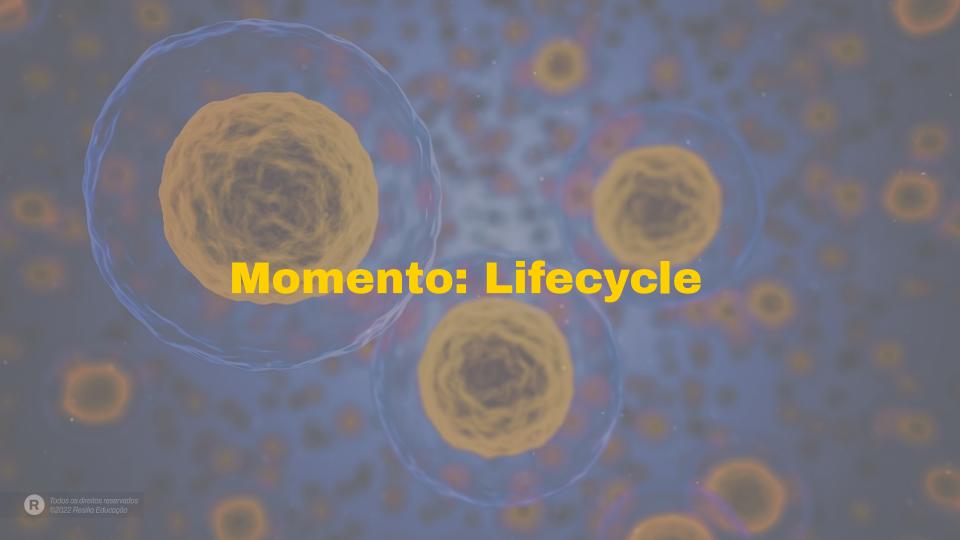
O Estado (useState) é todo dado que varia com o tempo, sendo essa alteração pode ser causada ou não pela interação do usuário.

Os estados são utilizados para valores dinâmicos, já as props são utilizadas para passar valores entre componentes (somente leitura).

useEffect

É um Hook para lidar com efeitos colaterais do componente e, portanto, está diretamente atrelado ao ciclo de vida.

Com ele, podemos lidar com todas as fases do ciclo de vida.



React Router DOM



Todos os direitos reservados ©2022 Resilia Educação

react-router-dom - Como usar?

São 3 componentes principais:

- <BrowserRouter/>
- <Routes/>
- <Route/>

Podemos declarar no arquivo main.jsx (ou index.js, se tiver usando CRA) já que ele que faz a injeção do código no HTML.

Ou no próprio componente App, transformando-o em um grande gerenciador de rotas

BrowserRouter

Componente pai que deve envolver tudo que o react-router for fazer, ou seja, tudo que for relacionado a rotas, deve estar declarado dentro dele.

Routes

Componente que envolve todas as rotas da aplicação, responsável por fazer a troca de componentes de acordo com a rota.

Route

o path (caminho). E se esse caminho for o mesmo da URL ele irá renderizar o componente contido no atributo element

Link

Para navegação podemos usar o componente Link, ele irá renderizar no HTML uma tag <a> com atributo href, o link/URL para outra rota é posto no atributo to, formando a expressão "link to", que significa "link para".