A IndexPage é a página principal de um aplicativo React que lida com informações de Pokémon. Ele exibe uma coleção pessoal de Pokémon e oferece a capacidade de adicionar novos Pokémon à coleção.

**useState**

O componente IndexPage mantém o seguinte estado interno:

pokemonList: Uma lista de Pokémon que representa a coleção pessoal do usuário.

createPokemon: Um booleano que indica se o usuário está no modo de criação de Pokémon.

updateList: Um contador usado para atualizar a lista de Pokémon quando ocorrem alterações.

**useEffect**

O componente IndexPage utiliza o useEffect para buscar a lista de Pokémon do servidor quando há alterações em updateList. No entanto, esse trecho de código está atualmente comentado (linhas 13 a 19), mas pode ser descomentado para buscar dados do servidor.

**Renderização Condicional**

O componente IndexPage renderiza condicionalmente com base na variável createPokemon. Se createPokemon for verdadeira, exibe um componente PokemonCard para criar um novo Pokémon. Caso contrário, exibe a lista de Pokémon existente.

**Componentes Filhos**

O componente IndexPage inclui o componente PokemonCard para criar novos Pokémon e exibir Pokémon existentes.

Ele passa as props necessárias para o componente PokemonCard, como createPokemon, setCreatePokemon, updateList, pokemonList, id, name, imageUrl, evolution etc.

**Fluxo de Dados**

O componente IndexPage interage com uma lista de Pokémon representada por pokemonList e atualiza essa lista por meio da função setPokemonList. Quando um novo Pokémon é criado ou um Pokémon existente é alterado ou removido, a variável updateList é incrementada para acionar o efeito colateral que busca dados do servidor.

**CSS**

O componente IndexPage utiliza estilos definidos no arquivo "style.css" para controlar a aparência da página.

Sobre o componente **PokemonCard** é uma parte de um aplicativo React que lida com informações de Pokémon. Ele representa um cartão que exibe informações de um Pokémon, como nome, imagem e estágio de evolução, e fornece funcionalidade para editar e excluir Pokémon.

**Props**

O componente PokemonCard aceita as seguintes props:

id: Um identificador único para o Pokémon.

name: O nome do Pokémon.

image: A URL da imagem do Pokémon.

evolution: O estágio de evolução do Pokémon.

createPokemon: Um booleano que indica se o cartão está em modo de criação de Pokémon.

setCreatePokemon: Uma função para definir o modo de criação de Pokémon.

updateList: Uma lista de Pokémon a ser atualizada.

setUpdateList: Uma função para definir a lista de Pokémon atualizada.

Estado Interno

O componente PokemonCard mantém o seguinte estado interno:

editPokemon: Um booleano que indica se o cartão está em modo de edição de Pokémon.

nameInput: O nome do Pokémon em modo de edição.

imageUrlInput: A URL da imagem do Pokémon em modo de edição.

evolutionInput: O estágio de evolução do Pokémon em modo de edição.

**Funções**

**handleChangePokemon**

A função handleChangePokemon é chamada quando o usuário clica no botão "Confirma" ou "Cancela" no modo de edição. Se o cartão estiver em modo de criação, ele cria um novo Pokémon e atualiza a lista de Pokémon. Se o cartão estiver em modo de edição, ele atualiza o Pokémon existente e fecha o modo de edição.

**handleDeletePokemon**

A função handleDeletePokemon é chamada quando o usuário clica no botão "Remover". Ela remove o Pokémon da lista de Pokémon atualizada.

**Renderização Condicional**

O componente PokemonCard renderiza de forma condicional com base no estado de editPokemon. No modo de edição, ele exibe campos de edição para o nome, URL da imagem e estágio de evolução do Pokémon, juntamente com botões "Cancelar" e "Confirma". No modo de visualização, ele exibe informações do Pokémon, incluindo nome, imagem, estágio de evolução e botões "Alterar" e "Remover".

**Fluxo de Dados**

O componente PokemonCard interage com uma lista de Pokémon para edição, criação e remoção. Ele usa Axios para fazer solicitações HTTP para criar e atualizar Pokémon no servidor.