

	SEDUC - SP			
	PROVA: 3ª SÉRIE EM_PROVA PAULISTA 2BI_EP_DESENVOLVIMENTO DE SISTEMAS_DIA 2 - COM OE			
	DISCIPLINA: Programação Front-end			
	ALUNO:			
	Nº:	SÉRIE:	TURMA:	TURNO:
	PROFESSOR:			

Programação Front-end

Questão 41

Uma equipe de desenvolvimento está criando um painel administrativo com o React. Durante a implantação, surgiu a necessidade de exibir informações que se atualizam dinamicamente, como notificações e status dos usuários conectados. Enquanto alguns dados devem ser apenas repassados entre componentes, outros precisam ser modificados no próprio componente para refletir mudanças em tempo real.

Nessa implementação, qual é a melhor abordagem para gerenciar os dados que precisam ser atualizados dinamicamente na interface?

- A

Utilizar props, pois são valores mutáveis e permitem atualizar o estado do componente filho a partir do componente pai.
- B

É a correta

Utilizar state, pois permite armazenar e atualizar dinamicamente os dados no próprio componente.
- C

Utilizar props, pois garantem que os valores compartilhados entre os componentes sejam atualizados automaticamente sem a necessidade de intervenção do desenvolvedor.
- D

Utilizar state, pois qualquer alteração nele ocorre de forma síncrona e imediata, garantindo atualização instantânea da interface.
- E

Utilizar props e state de forma intercambiável, pois ambos permitem modificar os dados e manter a interface sempre atualizada.

Resposta comentada	Feedback do professor
<p>O item busca aferir se o estudante conhece o conceito e a aplicação correta de props e state no React. A alternativa correta é a B, pois o state permite armazenar e modificar dinamicamente os valores em um componente, essencial para interfaces que precisam atualizar informações em tempo real. Já as props são imutáveis e servem para passar informações de um componente para outro.</p>	<p>GABARITO: B) O state é uma estrutura interna do componente que pode ser alterada dinamicamente, permitindo atualizações automáticas da interface com base nas interações do usuário.</p> <p>DISTRATORES: A) Props são imutáveis no componente filho e não podem ser modificadas diretamente. O estado só pode ser alterado no próprio componente que o gerencia. C) Props não atualizam automaticamente os componentes. Se um componente pai modifica uma prop, isso pode re-renderizar o componente filho, mas essa atualização depende explicitamente da forma como o React reconstrói a árvore de componentes. D) As atualizações de state no React são assíncronas e podem ser agrupadas para otimização de desempenho. Isso significa que nem sempre um setState ou useState causa uma re-renderização imediata. E) Props e state têm funções distintas: props são usadas para comunicação entre componentes e state para gerenciamento interno de estados dinâmicos. Não podem ser usadas indistintamente.</p>