

Exercícios Módulo 3 - Web & React

 Escreva um componente Profile que represente um resumo de um perfil com uma foto, nome e descrição passados por props.
Como no exemplo a seguir:



Alan Turing

Description: Alan Mathison Turing OBE FRS was an English mathematician, computer scientist, logician, cryptanalyst, philosopher, and theoretical biologist.

O URL da imagem e nome podem ser passados como props simples e a descripção como children:

<Profile

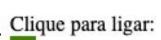
photoURL="https://upload.wikimedia.org/wikipedia/commons/a/a1/Alan_Turing_Aged_16.jpg" name="Alan Turing">

Alan Mathison Turing OBE FRS was an English mathematician, computer scientist, logician,

cryptanalyst, philosopher, and theoretical biologist.

</Profile>

2. Construa um componente que represente um interruptor que possua um estado booleano ligado ou desligado. Faça com que o Clique para ligar: 10 ser clicado inverta de estado:



Para atualizar o valor do estado, crie um método handleClick que inverta o estado do componente e passe o método usando o

3. Construa um componente que renderize um botão que ao ser clicado revela um conteúdo textual. Exemplo antes de ser clicado:

Show content

Após ser clicado:

It seems that you have found me!

Hide content

Ao clicar no botão "Hide content", voltamos a renderização anterior.

4. Construa um componente que represente uma calculadora simples:

			(
C	0	=	+
7	8	9	-
4	5	6	*
1	2	3	1

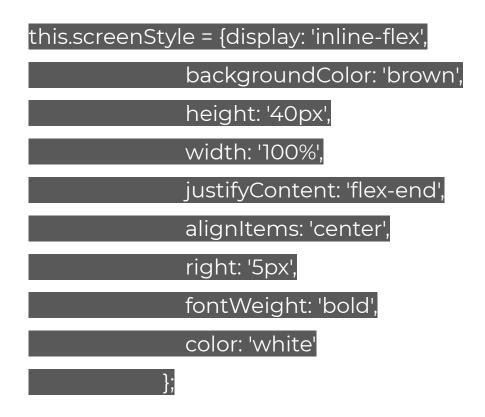
Dicas:

Você pode guardar a informação do 'visor' da calculado como um estado do componente como uma string. Para avaliar a informação é possível utilizar a função eval(). Exemplo:

let operacao = '2*15'; let resultado = eval(operacao);

Será necessário criar métodos para responderem aos cliques do usuário.

Além disso pode ser útil representar os botões como um vetor para poder utilizar o map como feito na aula de Renderização de Listas. Não é necessário adicionar estilo porém o resultado fica mais agradável, para isso é possível criar um objeto que contenha a informações. Lembrando que o nomes são bem parecidos com de CSS normal, só não usar o hífen '-' e capitalizar a palavras (exceto a primeira):



No JSX:

<div className="screen" style={this.screenStyle}>