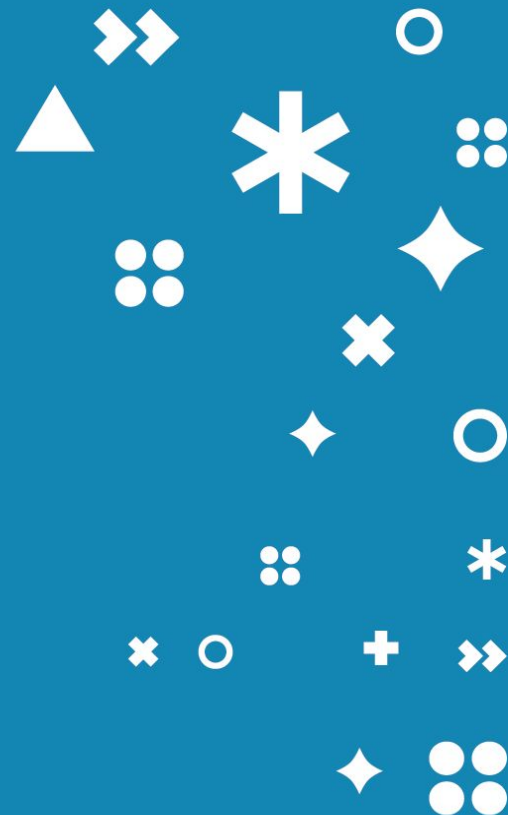


A aula começará
em breve!



Programação para Dispositivos Móveis

Aula 15 - Desenvolvimento Híbrido

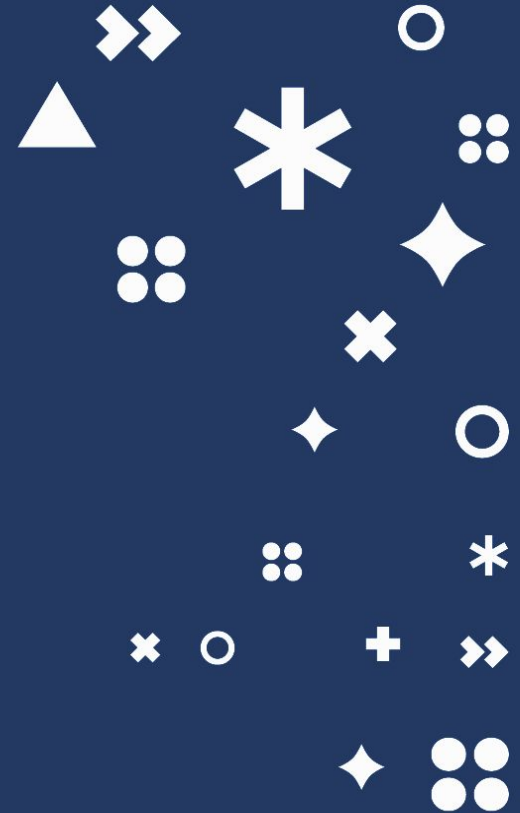
Prof. Fernando Concatto

Links úteis

- ❑ <https://reactnative.dev/docs/getting-started>
- ❑ <https://react.dev/>
- ❑ <https://docs.expo.dev/get-started/create-a-project/>
- ❑ Layouts:
 - ❑ <https://reactnative.dev/docs/flexbox>
 - ❑ <https://flexboxfroggy.com/>



Prática!



Atividade para exercitar



Construir um aplicativo que permita realizar a soma entre dois operandos. O aplicativo deve ter dois campos de texto numéricos, um botão para efetuar a soma, e deve apresentar o resultado da operação em um componente textual.

Atividade para exercitar



Construir um aplicativo que permita realizar cálculos aritméticos entre dois operandos. O aplicativo deve ter dois campos de texto numéricos, deve suportar as operações de soma, subtração, multiplicação e divisão, e deve apresentar o resultado da operação em um componente textual.

[illegible]

Por que?

- ❑ Encapsulamento de complexidade
- ❑ Reutilização de código
- ❑ Código limpo e legível

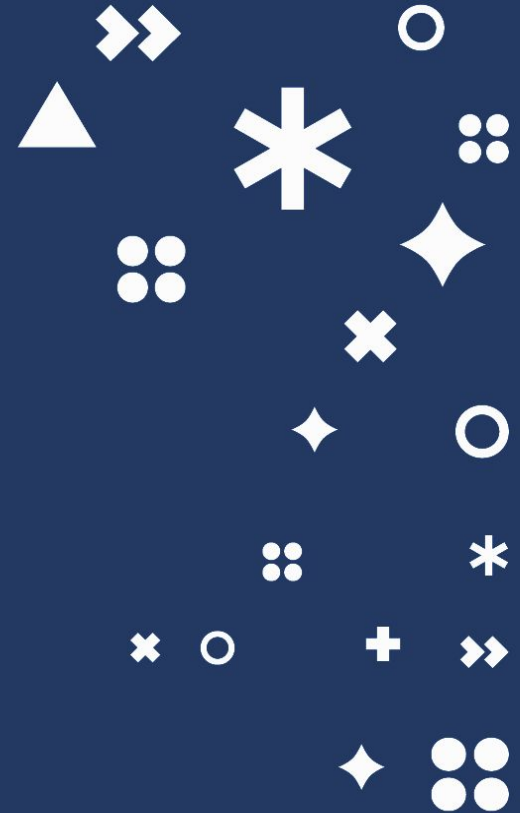


Em React



- ❑ São definidos como funções que começam com letra maiúscula
- ❑ Seus parâmetros são conhecidos como propriedades (*props*)
- ❑ Para instanciar: `<Componente propriedade="valor" />`

Prática



Atividade para exercitar



Transformar os botões de operações da calculadora em componentes, de modo a criar quatro instâncias do componente, mantendo o código o mais limpo possível.

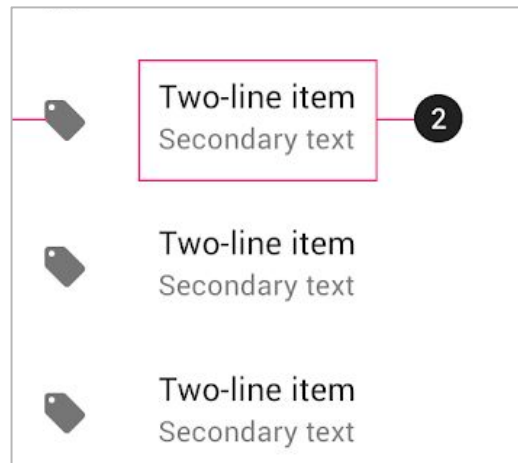
Atividade para exercitar (difícil)

Criar um componente chamado Operations, que recebe por propriedades quais as operações disponíveis e cria os botões apropriados. A calculadora deve permanecer funcionando corretamente.

Atividade para exercitar

Criar um novo projeto em React Native.
Criar um componente chamado
CoffeeDrink que recebe por *props* um
nome e descrição, e exibe-os um abaixo
do outro, conforme inspirações deste link:

<https://m2.material.io/components/lists>



Atividade para exercitar



Criar um componente chamado CoffeeList que armazena em memória um conjunto de dados de cafés em JSON, segundo o link a seguir. Armazenar apenas os 3 primeiros itens. Exibir os itens na tela criando 3 instâncias do componente CoffeeDrink.

<https://sampleapis.com/api-list/coffee>

(Clicar em “Run” para obter o JSON. Será necessário decodificá-lo no aplicativo. Pesquisar como fazer isso em JavaScript.)

Atividade para exercitar



Ampliar o conjunto de dados para 10 itens e exibir cada item na tela, sem precisar copiar e colar as instâncias de CoffeeDrink.

Atividade para exercitar (difícil)

Incluir um campo de texto no topo da lista que permite realizar a filtragem de acordo com o nome do café. Exibir apenas os cafés cujos nomes batem com o filtro. Atualizar a cada nova letra.