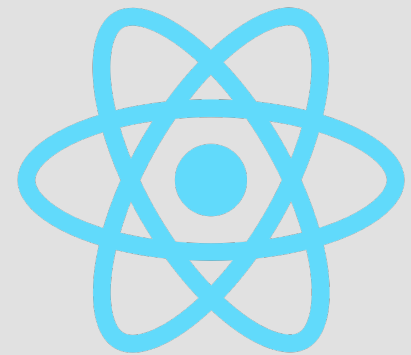


Components e Estado (*state*)

Curso de React Native



Component

- Encapsulamento de código
 - Evita repetição
- Composição de lógica, visualização e estilização
- Possui estado próprio
- É escrito de maneira **declarativa**

Component

```
...  
<View>  
  <Text>Olá mundo!</Text>  
  <Button title="Enviar mensagem"  
    onPress={() => null}/>  
</View>  
...
```

Importando componentes

- Você deve importar um componente no arquivo para que seja possível utilizá-lo
 - Similar ao *#include* em .C ou *import* em Java

Importando componentes

```
import {View,  
        Text,  
        Button} from 'react-native'  
  
import axios from 'axios'  
  
import MeuBotao from './MeuBotao.js'
```

Components nativos do React Native

- `<View>` - criação de uma área de visualização
- `<Text>` - componente para exibição de texto
- `<Image>` - componente para exibição de imagem
- `<Button>` - componente para criação de botão
- `<TouchableOpacity>` - componente para criação de botão com retorno visual

Entre outros!

Components nativos do React Native

- Ler a documentação é **essencial** para entender como cada componente funciona

Documentação dos componentes nativos do react-native

<https://facebook.github.io/react-native/>

API -> Components

Components nativos do React Native

```
...  
<View>  
  <Text>Olá mundo!</Text>  
  <Button title="Enviar mensagem"  
    onPress={() => null}/>  
</View>  
...
```

- PRÁTICA:

Ver como o código acima fica no React Native

Criando seu próprio componente

- Ao longo do desenvolvimento de um app, você pode precisar um componente que ainda não existe, ou então encapsular um código que está sendo utilizado várias vezes
- Para isso você pode criar seu próprio componente

Criando seu próprio componente

Class Component

```
import React from 'react'

class MeuComponente extends React.Component {

  render(){
    return(
      //Seu componente
    )
  }
}

export default MeuComponente
```

- OBS: o *return()* do componente **precisa** retornar apenas **uma tag** `<> ... </>`