

Vamos um App, com que pode ou não ficar oculta.

Crie um App, chamado: **Renderizacao-Condicional**

```
1  import { useState } from 'react';
2  import { View, Text, Button, SafeAreaView, StyleSheet, StatusBar } from 'react-native';
3
4  function App(){
5    const [areaShow, setAreaShow] = useState(false);
6
7    const handleClick = () => {
8      setAreaShow(!areaShow);
9    }
10
11    return (
12      <SafeAreaView style={styles.container}>
13        <Button title="Apareça / Desapareça" onPress={handleClick} />
14
15        { areaShow &&
16          <View style={styles.area}>
17            <Text style={styles.areaText}>Área secreta</Text>
18          </View>
19        }
20      </SafeAreaView>
21    );
22  }
23
24  const styles = StyleSheet.create({
25    container: {
26      marginTop: StatusBar.currentHeight || 0,
27    },
28    area: {
29      margin: 20,
30      padding: 20,
31      backgroundColor: '#333333',
32      borderRadius: 10,
33    },
34    areaText:{
35      textAlign: 'center',
36      color: '#FFFFFF'
37    },
38  });
39
40  export default App;
```

Crie um novo App, **componenteTextInput**

```
1  import { useState } from 'react';
2  import { View, Text, Button, SafeAreaView, StyleSheet, StatusBar, TextInput } from 'react-native';
3
4  function App(){
5
6      const [campo, setCampo] = useState("Teste");
7
8      const mostrar = () => {
9          alert(campo);
10     }
11
12     return (
13         <SafeAreaView style={styles.container}>
14             <TextInput
15                 style={styles.input}
16                 value={campo}
17                 onChangeText={t => setCampo(t)}
18             />
19             //<Button title="O que foi digitado" onPress={mostrar} />
20             <Text>Você digitou: {campo}</Text>
21         </SafeAreaView>
22     );
23 }
24
25 const styles = StyleSheet.create({
26     container: {
27         marginTop: StatusBar.currentHeight || 0,
28     },
29     input: {
30         width: 200,
31         height: 40,
32         backgroundColor: '#DDDDDD',
33         borderRadius: 5,
34         padding: 10,
35     },
36 });
37
38 export default App;
```

Crie um novo App, **componenteScrollView**

```
1  import { useState } from 'react';
2  import { View, Text, Button, SafeAreaView, StyleSheet, StatusBar, ScrollView } from 'react-native';
3
4  function App(){
5    return (
6      <SafeAreaView style={styles.container}>
7        <ScrollView style={styles.scroll}>
8          <View style={styles.azul}></View>
9          <View style={styles.verde}></View>
10         <View style={styles.amarelo}></View>
11        </ScrollView>
12      </SafeAreaView>
13    );
14  }
15
16  const styles = StyleSheet.create([
17    container: {
18      flex: 1,
19      marginTop: StatusBar.currentHeight || 0,
20    },
21    scroll: {
22      flex: 1,
23    },
24    azul: {
25      width: '100px',
26      height: 500,
27      backgroundColor: '#0000FF',
28    },
29    verde: {
30      width: '100px',
31      height: 500,
32      backgroundColor: '#00FF00',
33    },
34    amarelo: {
35      width: '100px',
36      height: 500,
37      backgroundColor: '#FFFF00',
38    }
39  ]);
40
41  export default App;
```

Agora vamos para o uso do VsCode:

Verifique se o Node.Js existe na máquina, usando o cmd, com comando: `node -v`

Se não existir vá no <https://nodejs.org/pt/download> e baixe a versão mais atualizada e LTS.

Para criar o projeto utilize o comando: **`npx create-expo-app --template blank-typescript`**

Em seguida dê um nome a ele. Ex: **primeiro**

Verifique se seu projeto está OK: **`npx expo-doctor`**

Instale o módulo web, para executar: **`npx expo install react-dom react-native-web @expo/metro-runtime`**

Agora vamos ao nosso App.

```
import { Button, Pressable, StyleSheet, Text, View } from 'react-native';

export default function App() {

  const handleClick = () => {
    alert("Funcionou");
  }
  return (
    <View style={styles.container}>
      <Text>Olá, tudo bem?</Text>

      { /* <Button title='Clique aqui' /> */ }

      <Pressable onPress={handleClick} style={styles.btn}>
        <Text style={styles.btnText}>Clique aqui!</Text>
      </Pressable>

    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: 'white',
    alignItems: 'center',
    justifyContent: 'center',
  },
  btn: {
    width: 200,
    height: 40,
    backgroundColor: 'blue',
    borderRadius: 10,
    justifyContent: 'center',
    alignItems: 'center',
  },
});
```

```
btnText: {  
  color: '#FFFFFF',  
  fontWeight: 'bold',  
  fontSize: 16  
}  
});
```