

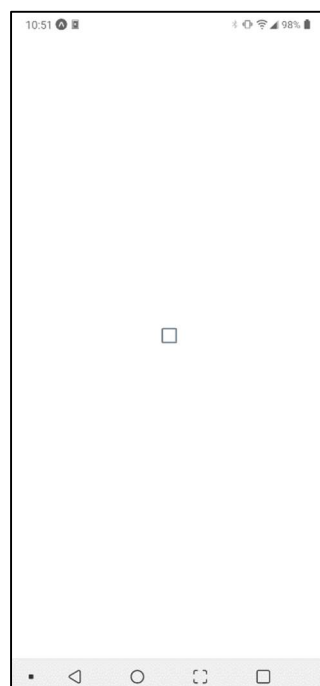
# Utilizando Checkbox em Aplicações Expo

Instale a biblioteca expo-checkbox:

```
→ checkboxexample git:(master) expo install expo-checkbox
```

Edite o arquivo "App.js" para testar a inclusão da biblioteca:

```
App.js > ...
1 | import { useState } from 'react';
2 | import { StatusBar } from 'expo-status-bar';
3 | import { StyleSheet, Text, View } from 'react-native';
4 |
5 | import Checkbox from 'expo-checkbox';
6 |
7 | export default function App() {
8 |   const [isChecked, setChecked] = useState(false);
9 |
10 |   return (
11 |     <View style={styles.container}>
12 |       <Checkbox value={isChecked} onValueChange={setChecked} />
13 |       <StatusBar style="auto" />
14 |     </View>
15 |   );
16 | }
17 |
18 | const styles = StyleSheet.create({
19 |   container: {
20 |     flex: 1,
21 |     backgroundColor: '#fff',
22 |     alignItems: 'center',
23 |     justifyContent: 'center',
24 |   },
25 | });
```



Vejamos um exemplo um pouco mais complexo:

```
App.js > [e] styles
1 | import { useState } from 'react';
2 | import { StatusBar } from 'expo-status-bar';
3 | import { StyleSheet, Text, View, FlatList } from 'react-native';
4 |
5 | import Checkbox from 'expo-checkbox';
6 |
7 | export default function App() {
8 |   const [optionList, setOptionList] = useState([ {id: 1, text: 'Option 1', checked: false}, {id: 2, text: 'Option 2', checked: false} ])
9 |
10 |   function setCheckbox(id) {
11 |     let list = []
12 |     optionList.forEach((item) => {
13 |       if(item.id === id) {
14 |         list.push({id: item.id, text: item.text, checked: !item.checked})
15 |       } else {
16 |         list.push(item)
17 |       }
18 |     })
19 |     setOptionList(list)
20 |   }
21 |
22 |   return (
23 |     <View style={styles.container}>
24 |       <FlatList style={styles.list} data={optionList} renderItem={({item}) =>
25 |         <View style={styles.optionSegment}>
26 |           <Checkbox value={item.checked} onValueChange={() => setCheckbox(item.id)} />
27 |           <Text>{item.text}</Text>
28 |         </View>
29 |       </FlatList>
30 |       <StatusBar style="auto" />
31 |     </View>
32 |   );
33 | }
34 |
35 | const styles = StyleSheet.create({
36 |   container: {
37 |     flex: 1,
38 |     backgroundColor: '#fff',
39 |     alignItems: 'center',
40 |     justifyContent: 'center',
41 |   },
42 |   list: {
43 |     padding: 30,
44 |   },
45 |   optionSegment: {
46 |     padding: 15,
47 |     flexDirection: 'row'
48 |   }
49 | });
```

