

# Hooks - useState

JS App.js > App

```
1  import React, { useState } from 'react';
2  import {Button, View, Text} from 'react-native';
3
4  export default function App() {
5    // Declare uma nova variável de state, a qual chamaremos de "count"
6    /*FAZ 3 COISAS
7    1- DECLARA A VARIÁVEL
8    2- DEFINIMOS UM METODO QUE ALTERA A VARIÁVEL
9    3- INICIALIZAMOS A VARIÁVEL
10   */
11   const [contador, setContador] = useState(0);
12
13   return (
14     <View>
15       <Text>Você clicou {contador} vezes</Text>
16       <Button onPress={() => setContador(contador + 1)} title="Clique aqui!"/>
17     </View>
18   );
19 }
```

# TextInput

<https://reactnative.dev/docs/textinput>

```
1  import React, { Component, useState } from 'react';
2  import { View, Text, TextInput } from 'react-native';
3
4  function App() {
5
6
7    const [nome, setNome] = useState("ValorInicial");
8
9    return (
10     <View style={{ justifyContent: 'center', flex: 1, alignItems: 'center' }}>
11       <TextInput placeholder="Informe seu nome!"
12         underlineColorAndroid="transparent" onChangeText={textoDigitado => setNome(textoDigitado)} />
13       <Text style={{ color: '#73d5bc' }}>vc digitou {nome}</Text>
14     </View>
15   );
16 }
17
18 export default App;
```

# App – Gasolina/Etanol

<https://github.com/halleygondim/react-native/tree/main/App%20-%20Calculadora%20Gasolina%20x%20Etanol>



/\*

## Adição de bibliotecas

```
yarn add react-native-svg
```

```
yarn add @fortawesome/fontawesome-svg-core
```

```
yarn add @fortawesome/free-solid-svg-icons
```

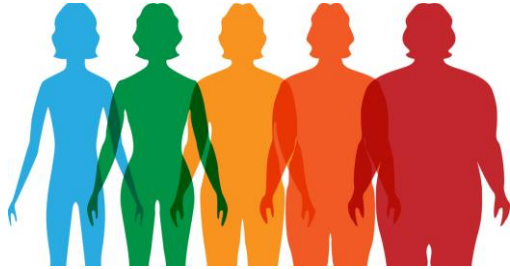
```
yarn add @fortawesome/react-native-fontawesome
```

## Escolher ícones

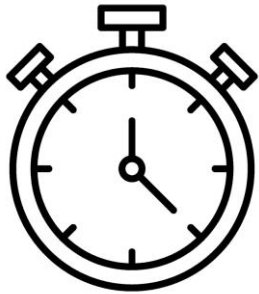
<https://fontawesome.com/v5.15/icons?d=gallery&p=2>

\*/

# Desafio I



Com as entradas necessárias calcular O IMC do indivíduo.



Criar um cronometro com opções de início, pausa e stop.

**Dica:** setInterval e clearInterval  
1000 ms = 1 segundo. Use 100ms



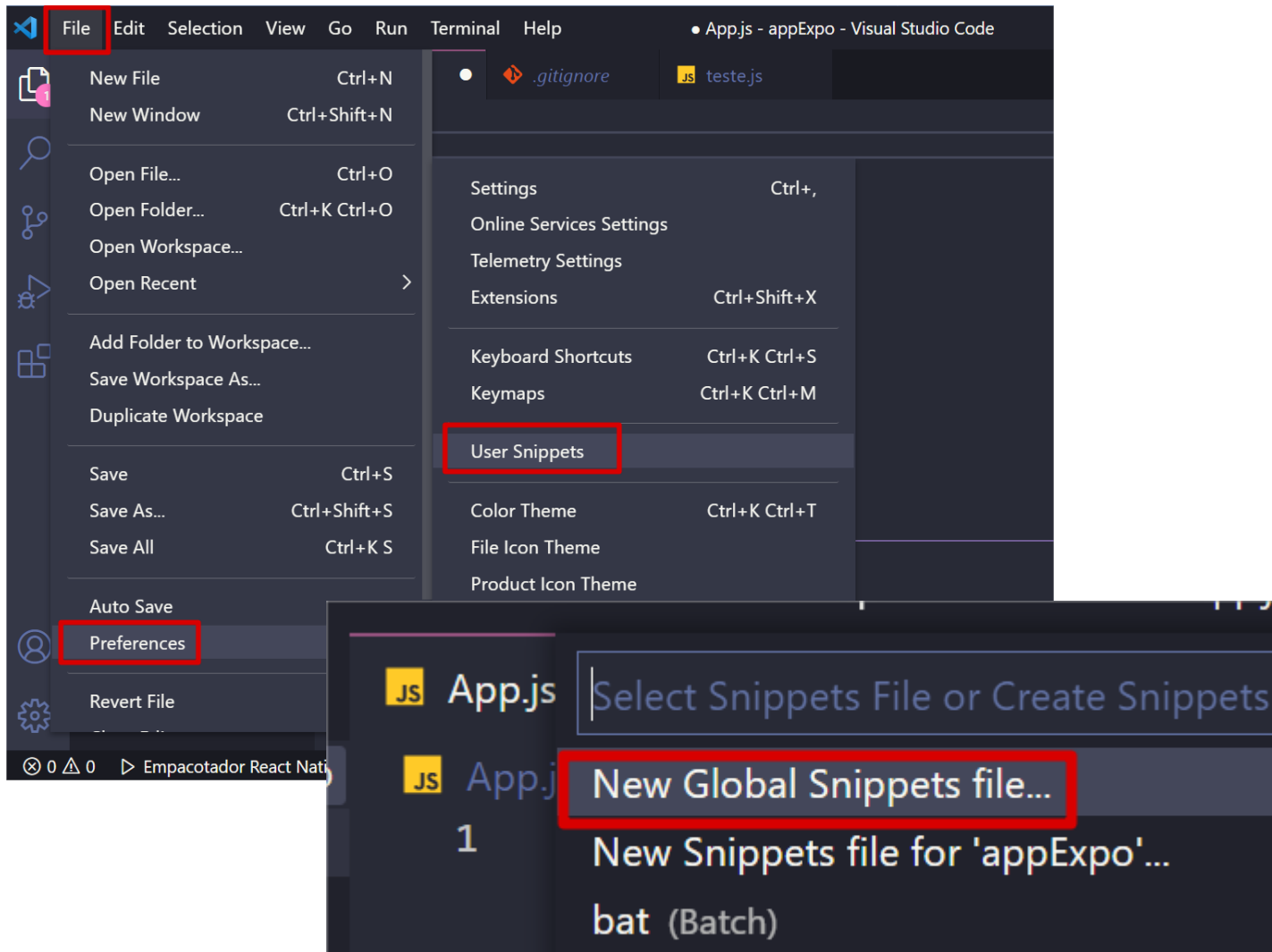
Com base em uma lista de frases positivas escolher uma frase aleatoriamente ao se clicar em um botão (quebrar o biscoito).

**Dica:** usar array e Math.random()

Em grupo (grupo da fábrica) – entregar link Github e apresentar (20/08).

**Obs.** Usem useState, imagens e estilização.

# Snippets I



# Snippets I

```
{  
  "Função": {  
    "scope": "javascript,typescript",  
    "prefix": "fnc",  
    "body": [  
      "import React from 'react'",  
      "import {View} from 'react-native'",  
      "export default function App() {",  
      "  return (  
        "<View>",  
        "$1",  
        "</View>",  
        ");",  
      "}"  
    ],  
    "description": "Log output to console"  
  },  
}
```

# Snippets I

```
},  
"Função2": {  
  "scope": "javascript,typescript",  
  "prefix": "fnc1",  
  "body": [  
    "import React from 'react'",  
    "import {View, Text} from 'react-native'",  
    "export default function App() {",  
    "  return ("  
    "    <View>",  
    "      $1",  
    "    </View>",  
    "  );",  
    "}"  
  ],  
  "description": "Log output to console"  
}
```

# Renderização condicional

```
1  import React from 'react'
2  import {View, Text, StatusBar} from 'react-native'
3  export default function App() {
4
5      const num = 11;
6
7      return (
8          <View>
9              <StatusBar hidden={true}></StatusBar>
10             {
11                 num === 10 ? <Text>O número é igual a 10</Text>
12                 : <Text>O número é diferente de 10</Text>
13             }
14             {/*PODE-SE USAR O FALSE*/}
15         </View>
16     );
17 }
```

<https://github.com/halleygondim/react-native/tree/main/Renderiza%C3%A7%C3%A3o%20Condicional>



# Objeto

```
1  //OBJETO
2  var pessoa = {
3      nome: "Jose",
4      idade: 45
5  }
6  console.log(pessoa.nome)
7
8  //VETOR DE OBJETOS
9  var pessoas = [{
10     nome: "Jose",
11     idade: 45
12 },
13
14     {
15     nome: "Jose3",
16     idade: 45
17     }
18 ]
19 pessoas.map(obj => console.log(obj.nome))
20
```

# Desestruturação

```
1  var pessoa = {
2      nome: "Jose",
3      idade: 45
4  }
5  console.log(pessoa.nome)
6
7  //DESESTRUTURAÇÃO JS
8  //[{variaveis existentes}] = [Objeto em formato JSON];
9
10 const {nome, idade} = pessoa
11 console.log(nome)
```

# ScrollView

```
1  import React, { Component, useState } from 'react';
2  import { View, Text, TextInput, ScrollView } from 'react-native';
3
4  function App() {
5
6    return (
7      <View style={{ flex: 1 }}>
8        <ScrollView showsVerticalScrollIndicator={false}>
9          <View style={{ height: 300, backgroundColor: 'yellow' }}></View>
10         <View style={{ height: 300, backgroundColor: 'red' }}></View>
11         <View style={{ height: 300, backgroundColor: 'blue' }}></View>
12         <View style={{ height: 300, backgroundColor: 'yellow' }}></View>
13         <View style={{ height: 300, backgroundColor: 'black' }}></View>
14       </ScrollView>
15     </View>
16   );
17 }
18 export default App;
```

<https://github.com/halleygondim/react-native/tree/main/ScrollView>

# Lista

```
1  import React from 'react'
2  import { View, Text, StatusBar, ScrollView } from 'react-native'
3  export default function App() {
4    const contatos = [
5      { id: 1, apelido: 'Calvin Candie', twitter: '@calvin', avatar: '' },
6      { id: 2, apelido: 'Dr. King Schutlz', twitter: '@schutlz', avatar: '' }
7    ]
8    return (
9      <View>
10        <StatusBar hidden={true}></StatusBar>
11        <Text>Twitters Populares</Text>
12        {contatos.map(contato => {
13          return (
14            /*fácil interpolar texto com variável `Olá meu nome é ${variavel}`*/
15            <View key={`_${contato.id}`}>
16              <Text >{contato.apelido} - {contato.twitter}</Text>
17            </View>
18          )
19        })}
20      </View>
21    );
22  }
```

<https://github.com/halleygondim/react-native/tree/main/Lista%20com%20Map>

[https://github.com/halleygondim/react-native/tree/main/Lista%20com%20Map%20\(Com%20Estilo\)](https://github.com/halleygondim/react-native/tree/main/Lista%20com%20Map%20(Com%20Estilo))

# FlatList

```
1  import React, { Component, useState } from 'react';
2  import { View, Text, StyleSheet, FlatList } from 'react-native';
3
4  export default function App(){
5    const [nomes, setNomes] = useState([
6      { id: '10', nome: 'Maria', telefone: '(62)994444-444', email: 'maria123@gmail.com' },
7      { id: '20', nome: 'José', telefone: '(62)997777-444', email: 'josese@gmail.com' } ]);
8
9    return (
10     <View style={styles.container}>
11       <FlatList
12         data={nomes} keyExtractor={(item) => item.id}
13         renderItem={({ item }) => <Contato data={item} />}
14       />
15     </View>
16   );
17 }
```

```
18  const styles = StyleSheet.create({
19    container: {
20      flex: 1,
21    },
22    contato: {
23      backgroundColor: '#751fb1',
24      height: 100,
25      marginBottom: 5
26    },
27    pessoa: {
28      color: '#e6e6fa',
29      fontSize: 15,
30    }
31  });
32
```

<https://github.com/halleygondim/react-native/tree/main/FlatList>

# FlatList

```
33 function Contato(props) {  
34   return (  
35     <View style={styles.contato}>  
36       <Text style={styles.pessoa}>Nome: {props.data.nome} </Text>  
37       <Text style={styles.pessoa}>Telefone: {props.data.telefone} </Text>  
38       <Text style={styles.pessoa}>Email: {props.data.email} </Text>  
39     </View>  
40   );  
41 }
```

<https://github.com/halleygondim/react-native/tree/main/FlatList>