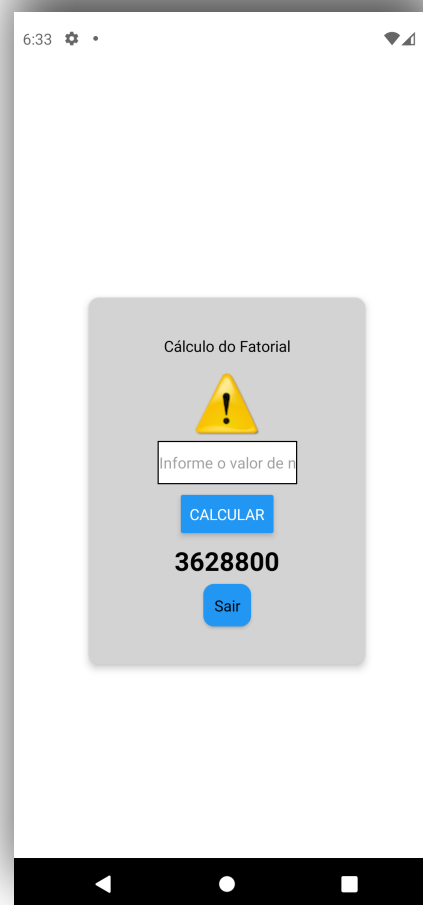
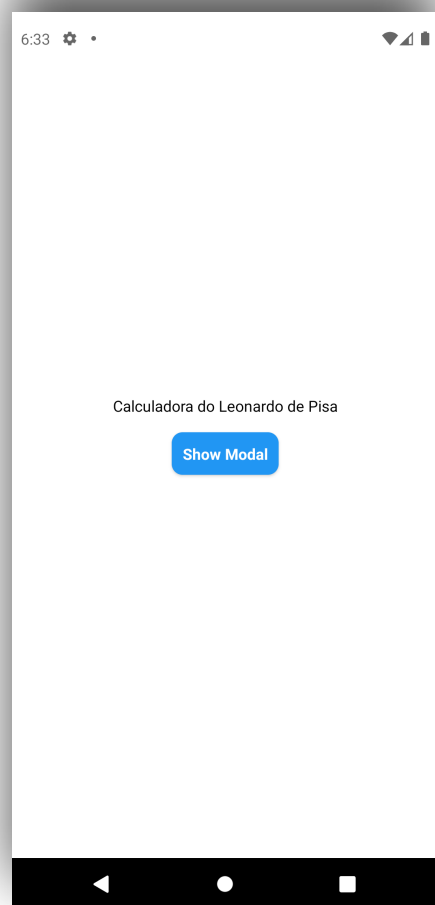
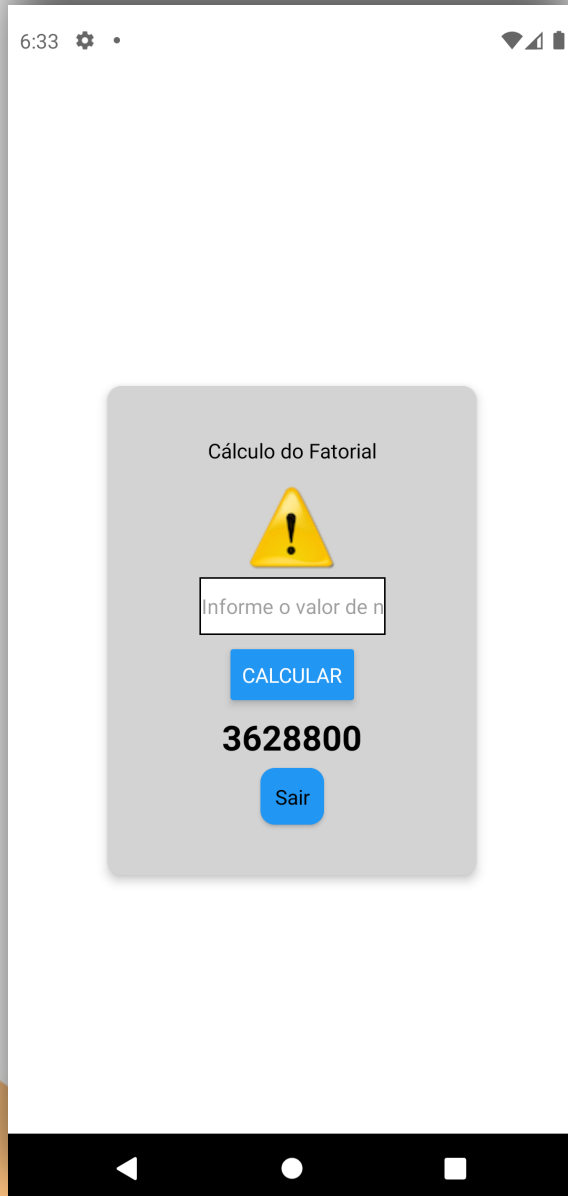


# Prog Disp Móveis



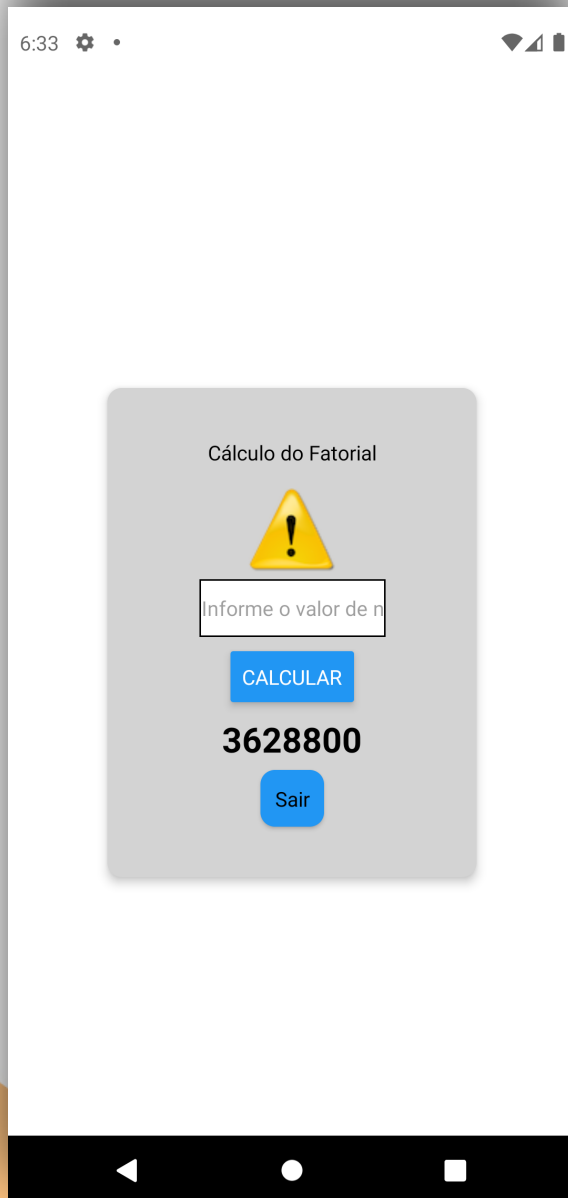
# Prog Disp Móveis



## Modal

```
<Modal animationType="slide" transparent={true}
  visible={this.state.modalVisible}
  onRequestClose={() => {
    Alert.alert("A janela será fechada.");
    setModalVisible(!modalVisible);}}>
  <View style={styles.centeredView}>
    <View style={styles.modalView}>
      <Text style={styles.modalText}>Cálculo do Fatorial</Text>
      <Image source={require('./assets/alert.png')}/>
      <TextInput
        style={styles.entrada}
        placeholder="Informe o valor de n"
        keyboardType="numeric"
        onChangeText={this.getNumber}/>
      <TouchableHighlight
        style={styles.botao}>
        <Button title="CALCULAR"
          onPress={() => this.fatorial(this.state.n)} />
        </TouchableHighlight>
      <Text style={styles.entrada, styles.titulo}>
        {this.state.F}
      </Text>
      <Pressable
        style={[styles.button, styles.buttonClose]}
        onPress={() => this.setModalVisible(!
          this.state.modalVisible)}>
        <Text>Sair</Text>
      </Pressable>
    </View>
  </View>
</Modal>
```

# Prog Disp Móveis



## Modal

```
<Modal animationType="slide" transparent={true}
  visible={this.state.modalVisible}
  onRequestClose={() => {
    Alert.alert("A janela será fechada.");
    setModalVisible(!modalVisible);}}>
```

```
<View style={styles.centeredView}>
```

### animationType

**slide** desliza pela parte inferior,  
**fade** desaparece à vista,  
**none** aparece sem animação.

### Transparent

O transparent determina se seu modal preencherá toda a visualização. Definir como true renderizará o modal sobre um fundo transparente.

```
onPress = {() => this.fatorial(this.state.n) }/;>
```

### onRequestClose

retorno de chamada é entregue quando o usuário toca no botão Voltar do hardware no Android ou no botão de menu na Apple TV.



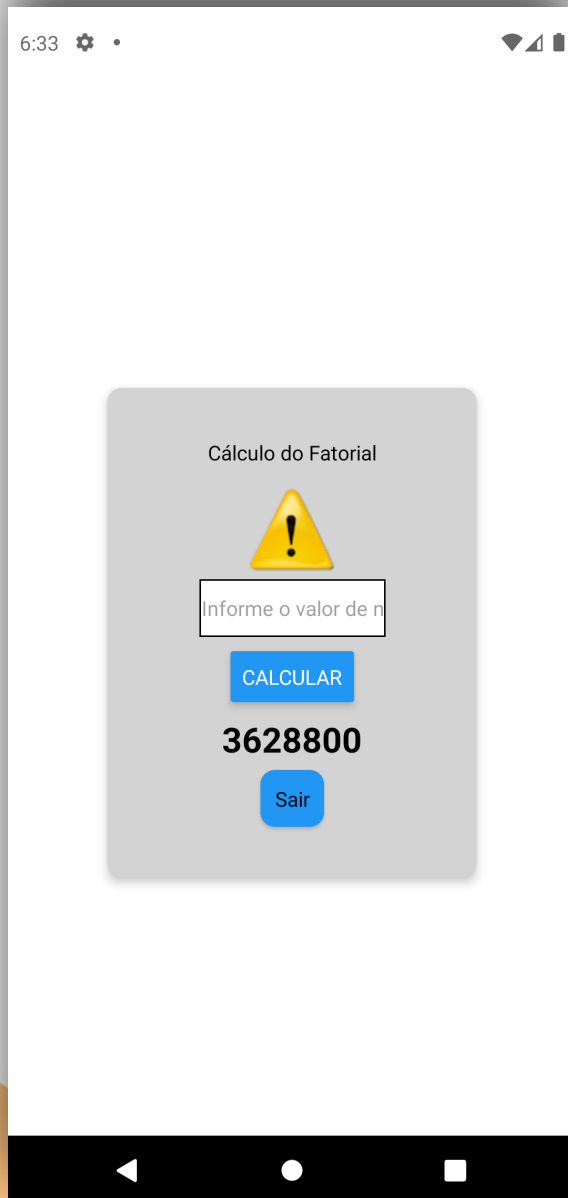
### Visible

Determina se o seu modal será visível.

```
</View>
```

```
</Modal>
```

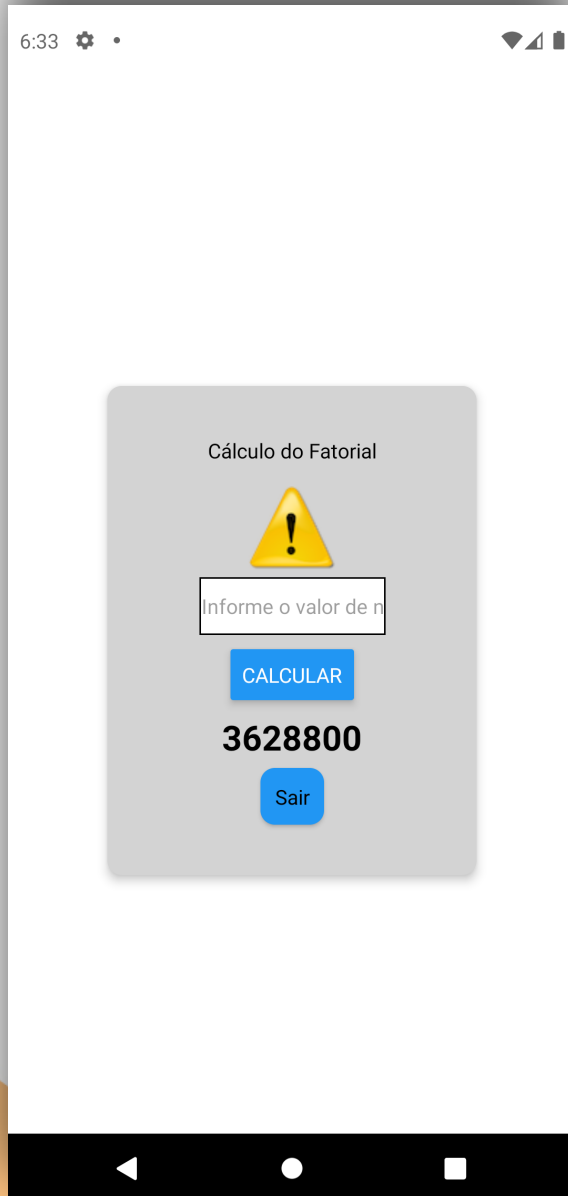
# Prog Disp Móveis



## Modal

```
<Modal animationType="slide" transparent={true}
  visible={this.state.modalVisible}
  onRequestClose={() => {
    Alert.alert("A janela será fechada.");
    setModalVisible(!modalVisible);}}>
<View style={styles.centeredView}>
  <View style={styles.modalView}>
    <Text style={styles.modalText}>Cálculo do Fatorial</Text>
    <Image source={require('./assets/alert.png')}/>
    <TextInput
      style={styles.entrada}
      placeholder="Informe o valor de n"
      keyboardType="numeric"
      onChangeText={this.getNumber}/>
    <TouchableHighlight
      style={styles.botao}>
      <Button title="CALCULAR"
        onPress={() => this.fatorial(this.state.n) }/>
    </TouchableHighlight>
    <Text style={styles.entrada, styles.titulo}>
      {this.state.F}
    </Text>
    <Pressable
      style={[styles.button, styles.buttonClose]}
      onPress={() => this.setModalVisible(!
        this.state.modalVisible)}>
      <Text>Sair</Text>
    </Pressable>
  </View>
</View>
</Modal>
```

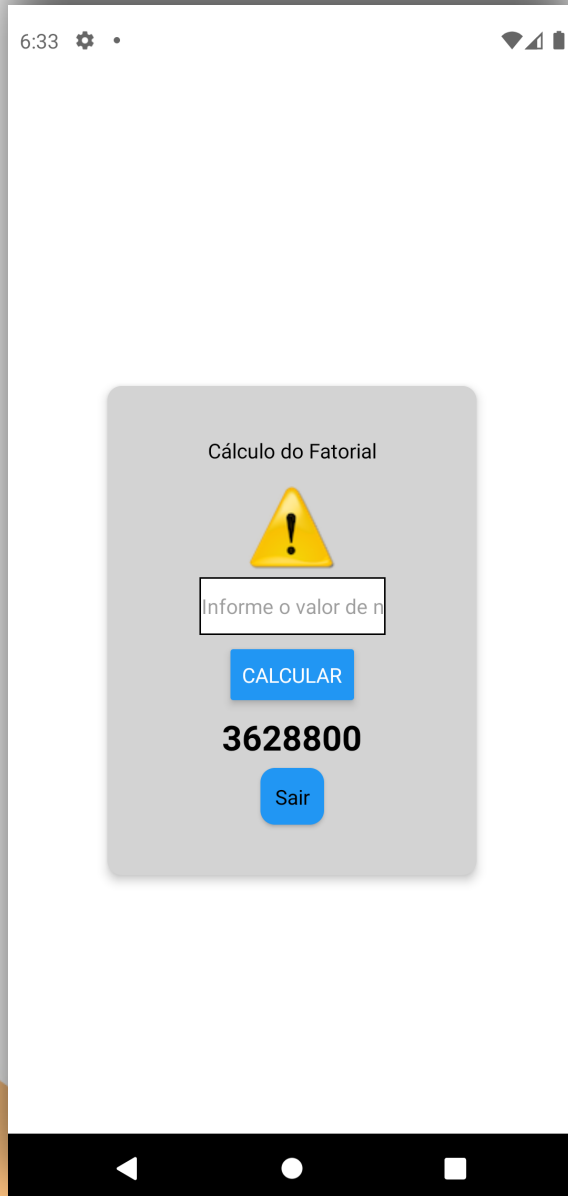
# Prog Disp Móveis



## Modal

```
<Modal animationType="slide" transparent={true}
  visible={this.state.modalVisible}
  onRequestClose={() => {
    Alert.alert("A janela será fechada.");
    setModalVisible(!modalVisible);}}>
  <View style={styles.centeredView}>
    <View style={styles.modalView}>
      <Text style={styles.modalText}>Cálculo do Fatorial</Text>
      <Image source={require('./assets/alert.png')} />
      <TextInput
        style={styles.entrada}
        placeholder="Informe o valor de n"
        keyboardType="numeric"
        onChangeText={this.getNumber} />
      <TouchableHighlight
        style={styles.botao}>
        <Button title="CALCULAR"
          onPress={() => this.fatorial(this.state.n)} />
        </TouchableHighlight>
      <Text style={styles.entrada, styles.titulo}>
        {this.state.F}
      </Text>
      <Pressable
        style={[styles.button, styles.buttonClose]}
        onPress={() => this.setModalVisible(!
          this.state.modalVisible)}>
        <Text>Sair</Text>
      </Pressable>
    </View>
  </View>
</Modal>
```

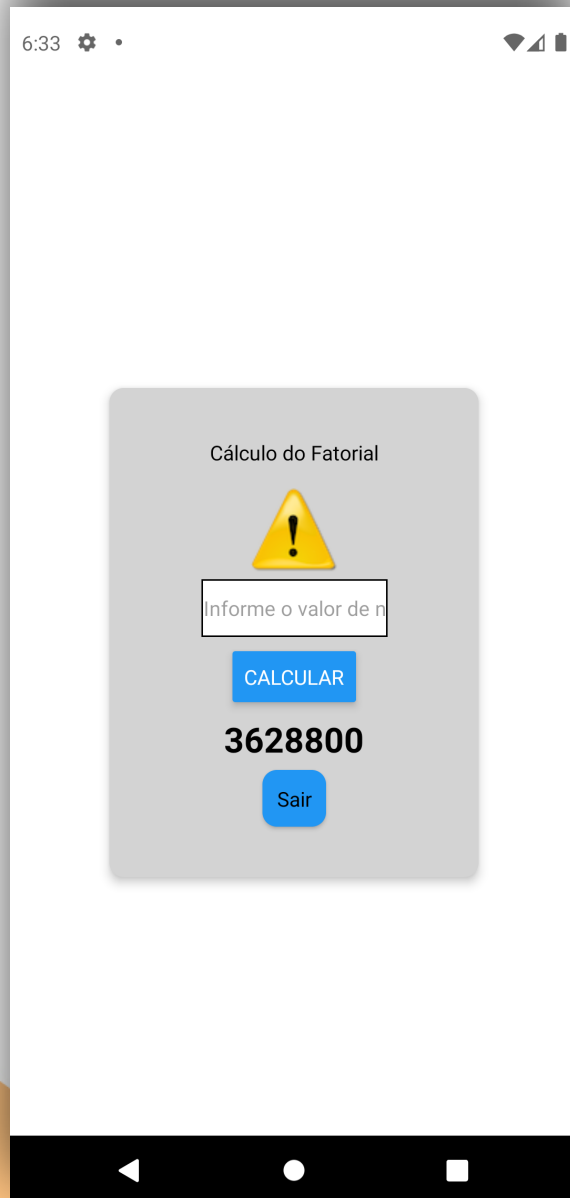
# Prog Disp Móveis



## Modal

```
<Modal animationType="slide" transparent={true}
  visible={this.state.modalVisible}
  onRequestClose={() => {
    Alert.alert("A janela será fechada.");
    setModalVisible(!modalVisible);}}>
  <View style={styles.centeredView}>
    <View style={styles.modalView}>
      <Text style={styles.modalText}>Cálculo do Fatorial</Text>
      <Image source={require('./assets/alert.png')} />
      <TextInput
        style={styles.entrada}
        placeholder="Informe o valor de n"
        keyboardType="numeric"
        onChangeText={this.getNumber} />
      <TouchableHighlight
        style={styles.botao}>
        <Button title="CALCULAR"
          onPress={() => this.fatorial(this.state.n)} />
        </TouchableHighlight>
      <Text style={styles.entrada, styles.titulo}>
        {this.state.F}
      </Text>
      <Pressable
        style={[styles.button, styles.buttonClose]}
        onPress={() => this.setModalVisible(!
          this.state.modalVisible)}>
        <Text>Sair</Text>
      </Pressable>
    </View>
  </View>
</Modal>
```

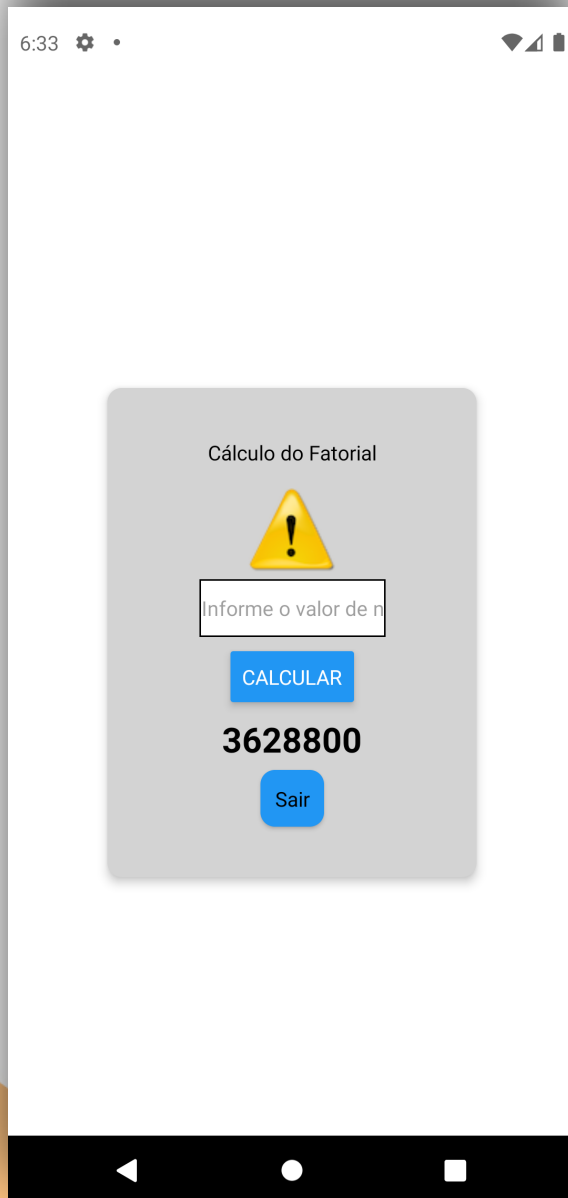
# Prog Disp Móveis



## Modal

```
<Modal animationType="slide" transparent={true}
  visible={this.state.modalVisible}
  onRequestClose={() => {
    Alert.alert("A janela será fechada.");
    setModalVisible(!modalVisible);}}>
  <View style={styles.centeredView}>
    <View style={styles.modalView}>
      <Text style={styles.modalText}>Cálculo do Fatorial</Text>
      <Image source={require('./assets/alert.png')} />
      <TextInput
        style={styles.entrada}
        placeholder="Informe o valor de n"
        keyboardType="numeric"
        onChangeText={this.getNumber} />
      <TouchableHighlight
        style={styles.botao}>
        <Button title="CALCULAR"
          onPress={() => this.fatorial(this.state.n)} />
        </TouchableHighlight>
      <Text style={styles.entrada, styles.titulo}>
        {this.state.F}
      </Text>
      <Pressable
        style={[styles.button, styles.buttonClose]}
        onPress={() => this.setModalVisible(!
          this.state.modalVisible)}>
        <Text>Sair</Text>
      </Pressable>
    </View>
  </View>
</Modal>
```

# Prog Disp Móveis



## Modal

### Pressable

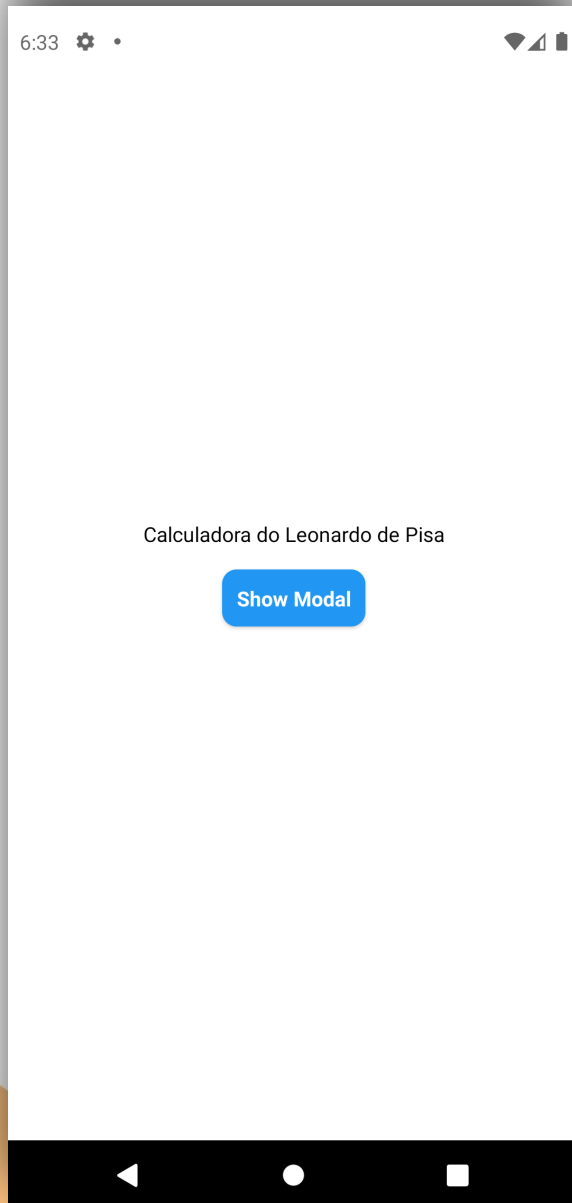
Determina se o seu modal será criada.");  
visível.



```
<Modal animationType="fade"
  visible={this.state.modalVisible}
  onRequestClose={() => this.setState({modalVisible: false})}
  style={styles.modal}
  backgroundColor="white">
  <View style={styles.modalContent}>
    <Text style={styles.titulo}>Cálculo do Fatorial</Text>
    <Image source={require('./assets/icon-warning.png')} style={styles.iconWarning}/>
    <TextInput style={styles.entrada}
      placeholder="Informe o valor de n"
      keyboardType="numeric"
      onChangeText={this.getNumber}/>
    <TouchableHighlight style={styles.botao}>
      <Button title="CALCULAR"
        onPress={() => this.fatorial(this.state.n)}/>
    </TouchableHighlight>
    <Text style={styles.entada, styles.titulo}>
      {this.state.F}
    </Text>
    <Pressable style={[styles.button, styles.buttonClose]}
      onPress={() => this.setState({modalVisible: false})}
      this.state.modalVisible)}>
      <Text>Sair</Text>
    </Pressable>
  </View>
</Modal>
```



# Prog Disp Móveis



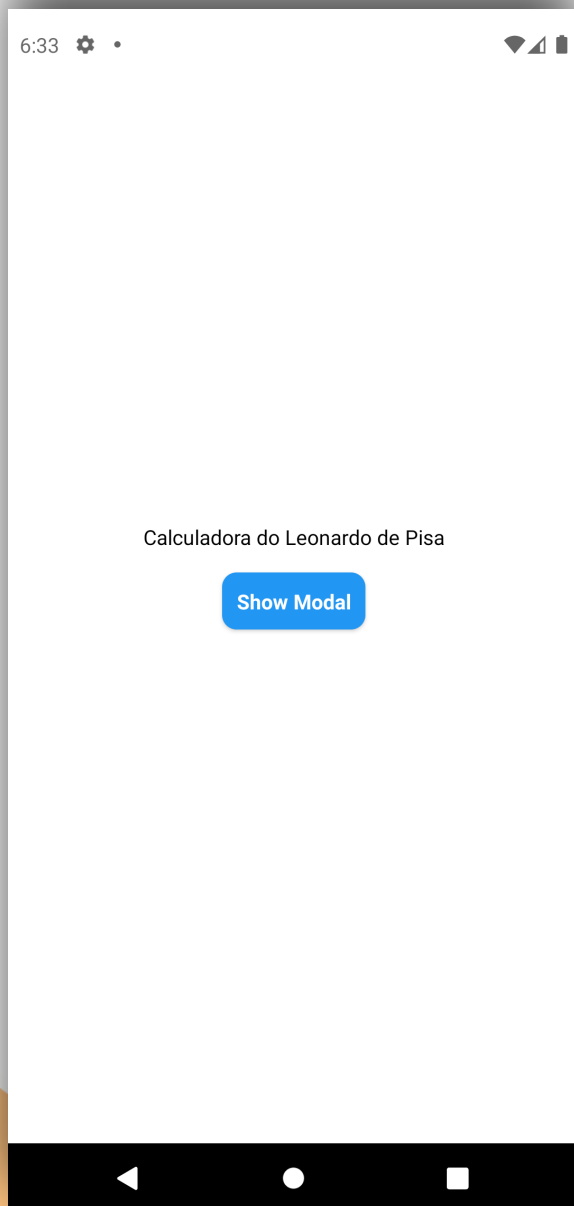
## Modal

```
export default class Fatorial extends Component {
  state = {
    n: 0,
    F: 0,
    modalVisible: false,
  }
  getNumber = (numero) => {
    this.setState({ n: numero })
  }
  setModalVisible(visible) {
    this.setState({ modalVisible: visible })
  }
  setF = (numero) => {
    this.setState({ F: numero })
  }

  fatorial=(numero)=>
  {
    var n = parseInt(numero);
    var a = 0, b = 1, t, fat=1;

    for(let i=1; i<=n; i++){
      fat*=i;
      console.log(fat);
    }
    this.setF(fat);
  }
}
```

# Prog Disp Móveis



## Modal

```
<Pressable
  style={[styles.button, styles.buttonOpen]}
  onPress={() => this.setModalVisible(true)}
>
<Text style={styles.textStyle}>Show Modal</Text>
</Pressable>
```