

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

React Native0.82DevelopmentContributingCommunityShowcaseBlog

RefreshControl

ScrollView

SectionList

StatusBar

Switch

Text

TextInput

TouchableHighlight

TouchableOpacity

TouchableWithoutFeedback

View

VirtualizedList

Android Components

DrawerLayoutAndroid

TouchableNativeFeedback

iOS Components

Join us for React Conf on Oct 7-8. [Learn more.](#)

TextInput

A foundational component for inputting text into the app via a keyboard. Props provide configurability for several features, such as auto-correction, auto-capitalization, placeholder text, and different keyboard types, such as a numeric keypad.

The most basic use case is to plop down a `TextInput` and subscribe to the `onChangeText` events to read the user input. There are also other events, such as `onSubmitEditing` and `onFocus` that can be subscribed to. A minimal example:

TextInput Example

```
import React from 'react';
import {StyleSheet, TextInput} from 'react-native';
import {SafeAreaView, SafeAreaViewProvider} from 'react-native-safe-area-context';
```

multiline

numberOfLines

onBlur

onChange

onChangeText

onContentSizeChange

onEndEditing

onPressIn

onPressOut

onFocus

onKeyPress

onLayout

onScroll

onSelectionChange

e

onSubmitEditing

placeholder

placeholderTextColor

color

TextInput

onChangeText

Text Input Events

`onChangeText`, `onFocus`, `onBlur`, `onSubmitEditing`

```
import React, {useState} from 'react';
import { StyleSheet, TextInput, View, Button, Alert } from 'react-native';

export default function App() {
  const [name, setName] = useState('');

  const handleSubmit = () => {
    if (!name) Alert.alert('Please type your name');
    else Alert.alert(`Welcome, ${name}!`);
  };

  return (
    <View style={{ padding: 40 }}>
      <TextInput
        placeholder="Type your name"
        style={{ borderWidth: 1, padding: 10, marginBottom: 10 }}
        onChangeText={setName}
        onFocus={() => console.log('Focused')}
        onBlur={() => console.log('Lost focus')}
      />
      <Button title="Submit" onPress={handleSubmit} />
    </View>
  );
}
```

Another way to update Hook:

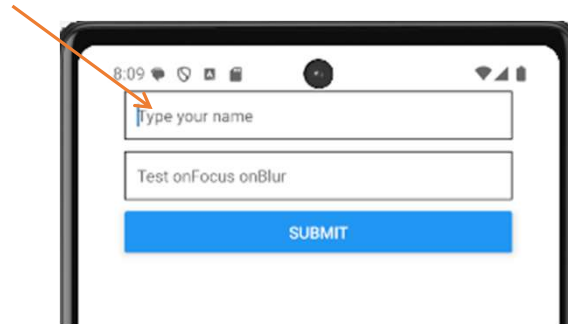
```
function functionSetName(thename){
  setName(thename);
  console.log("hello", thename)
}
```

. . .

```
onChangeText={(n)=>functionSetName(n)}
```

• **onFocus**: This callback function is triggered when the `TextInput` component receives focus. This typically happens when a user taps on the `TextInput`, making it active and ready to receive input, or when focus is programmatically shifted to it. You can use `onFocus` to perform actions like changing the styling of the input, displaying a specific UI element, or logging an event.

```
<TextInput
  placeholder="Type your name"
  style={{ borderWidth: 1, padding: 10, marginBottom: 10 }}
  onChangeText={setName}
  // onChangeText={(thisName)=>functionSetName(thisName)}
  onFocus={() => console.log('Focused')}
  onBlur={() => console.log('Lost focus')}
/>
<TextInput
  placeholder="Test onFocus onBlur"
  style={{ borderWidth: 1, padding: 10, marginBottom: 10 }}
  // onChangeText={setName}
  // onChangeText={(thisName)=>functionSetName(thisName)}
  onFocus={() => console.log('Focused2')}
  onBlur={() => console.log('Lost focus2')}
/>
```



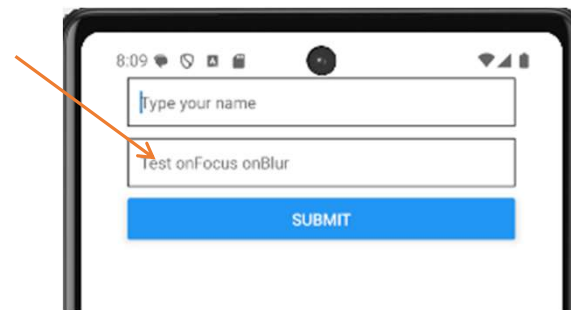
Click or Tap TextInput **Type ...**



Android Bundled 51ms index.js (1 module)
LOG Focused

• **onBlur**: This callback function is triggered when the `TextInput` component loses focus. This occurs when a user taps **outside** of the `TextInput`, or when focus is programmatically shifted to another element. `onBlur` is useful for tasks such as validating input, saving data, or reverting styling changes made during focus.

```
<TextInput
  placeholder="Type your name"
  style={{ borderWidth: 1, padding: 10, marginBottom: 10 }}
  onChangeText={setName}
  // onChangeText={(thisName)=>functionSetName(thisName)}
  onFocus={() => console.log('Focused')}
  onBlur={() => console.log('Lost focus')}
/>
<TextInput
  placeholder="Test onFocus onBlur"
  style={{ borderWidth: 1, padding: 10, marginBottom: 10 }}
  // onChangeText={setName}
  // onChangeText={(thisName)=>functionSetName(thisName)}
  onFocus={() => console.log('Focused2')}
  onBlur={() => console.log('Lost focus2')}
/>
```



Click or Tap TextInput **Test...**



Android Bundled 51ms index.js (1 module)

```
LOG Focused
LOG Lost focus
LOG Focused2
```