# **Touchable Highlight**

If you're looking for a more extensive and future-proof way to handle touch-based input, check out the Pressable API.

A wrapper for making views respond properly to touches. On press down, the opacity of the wrapped view is decreased, which allows the underlay color to show through, darkening or tinting the view.

The underlay comes from wrapping the child in a new View, which can affect layout, and sometimes cause unwanted visual artifacts if not used correctly, for example if the backgroundColor of the wrapped view isn't explicitly set to an opaque color.

TouchableHighlight must have one child (not zero or more than one). If you wish to have several child components, wrap them in a View.

### **Example**

```
TouchableHighlight Example
                                                                                                ∧ Expo
import React, {useState} from 'react';
import {StyleSheet, Text, TouchableHighlight, View} from
'react-native';
const TouchableHighlightExample = () => {
  const [count, setCount] = useState(0);
  const onPress = () => setCount(count + 1);
  return (
    <View style={styles.container}>
      <TouchableHighlight onPress={onPress}>
        <View style={styles.button}>
          <Text>Touch Here</Text>
        </View>
      </TouchableHighlight>
      <View style={styles.countContainer}>
        <Text style={styles.countText}>{count || null}
</Text>
      </View>
    </View>
  );
};
const styles = StyleSheet.create({
  container: {
    flex: 1,
                                                             Preview
                                                                             My Device
                                                                                       iOS Android
                                                                                                     Web
```

# Reference

### **Props**

#### TouchableWithoutFeedback Props

Inherits TouchableWithoutFeedback Props.

#### activeOpacity

Determines what the opacity of the wrapped view should be when touch is active. The value should be between 0 and 1. Defaults to 0.85. Requires underlayColor to be set.

ТҮРЕ	
number	

#### onHideUnderlay

Called immediately after the underlay is hidden.

TYPE function

#### onShowUnderlay

Called immediately after the underlay is shown.

TYPE function

#### style

TYPE
View.style

#### underlayColor

The color of the underlay that will show through when the touch is active.

ТҮРЕ	
color	

## hasTVPreferredFocus ◀ iOS

(Apple TV only) TV preferred focus (see documentation for the View component).

TYPE bool

## 

TV next focus down (see documentation for the View component).

number

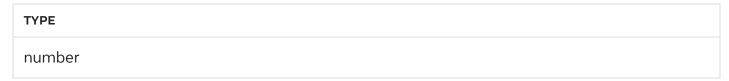
# nextFocusForward Android

TV next focus forward (see documentation for the View component).

TYPE number

### nextFocusLeft | Android

TV next focus left (see documentation for the View component).



# nextFocusRight | Android

TV next focus right (see documentation for the View component).

TYPE number

# 

TV next focus up (see documentation for the View component).

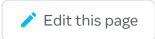
TYPE number

#### testOnly\_pressed

Handy for snapshot tests.

TYPE bool

# Is this page useful?



Last updated on **Jun 21, 2023**