Alert

Launches an alert dialog with the specified title and message.

Optionally provide a list of buttons. Tapping any button will fire the respective on Press callback and dismiss the alert. By default, the only button will be an 'OK' button.

This is an API that works both on Android and iOS and can show static alerts. Alert that prompts the user to enter some information is available on iOS only.

Example

```
Alert Example
                                                                 ∧ Expo
import React from 'react';
import {View, StyleSheet, Button, Alert} from 'react-native';
const App = () => {
  const createTwoButtonAlert = () =>
    Alert.alert('Alert Title', 'My Alert Msg', [
      {
        text: 'Cancel',
        onPress: () => console.log('Cancel Pressed'),
        style: 'cancel',
      },
      {text: 'OK', onPress: () => console.log('OK Pressed')},
    ]);
  const createThreeButtonAlert = () =>
    Alert.alert('Alert Title', 'My Alert Msg', [
        text: 'Ask me later',
        onPress: () => console.log('Ask me later pressed'),
      },
      {
        text: 'Cancel',
        onPress: () => console.log('Cancel Pressed'),
        style: 'cancel',
      },
      {text: 'OK', onPress: () => console.log('OK Pressed')},
                                   Preview
                                                   My Device
                                                              iOS
                                                                   Android
```

iOS

On iOS you can specify any number of buttons. Each button can optionally specify a style or be emphasized, available options are represented by the <u>AlertButtonStyle</u> enum and the <u>isPreferred</u> field on AlertButton.

Android

On Android at most three buttons can be specified. Android has a concept of a neutral, negative and a positive button:

- If you specify one button, it will be the 'positive' one (such as 'OK')
- Two buttons mean 'negative', 'positive' (such as 'Cancel', 'OK')
- Three buttons mean 'neutral', 'negative', 'positive' (such as 'Later', 'Cancel', 'OK')

Alerts on Android can be dismissed by tapping outside of the alert box. It is disabled by default and can be enabled by providing an optional <u>Options</u> parameter with the cancelable property set to true i.e.

{cancelable: true}.

The cancel event can be handled by providing an onDismiss callback property inside the options parameter.

Example Android

```
Alert Android Dissmissable Example
                                                                 ∧ Expo
import React from 'react';
import {View, StyleSheet, Button, Alert} from 'react-native';
const showAlert = () =>
  Alert.alert(
    'Alert Title',
    'My Alert Msg',
        text: 'Cancel',
       onPress: () => Alert.alert('Cancel Pressed'),
       style: 'cancel',
     },
    ],
      cancelable: true,
     onDismiss: () =>
        Alert.alert(
          'This alert was dismissed by tapping outside of the alert
dialog.',
        ),
    },
  );
const App = () => (
  <View style={styles.container}>
                                        Preview
                                                        My Device
                                                                   Android
```

Reference

Methods

alert()

```
static alert (
  title: string,
  message?: string,
  buttons?: AlertButton[],
  options?: AlertOptions,
);
```

Parameters:

NAME	TYPE	DESCRIPTION
title Required	string	The dialog's title. Passing $_{\text{null}}$ or empty string will hide the title.
message	string	An optional message that appears below the dialog's title.
buttons	AlertButton[]	An optional array containing buttons configuration.
options	Options	An optional Alert configuration.

prompt()

```
static prompt: (
  title: string,
  message?: string,
  callbackOrButtons?: ((text: string) => void) | AlertButton[],
  type?: AlertType,
  defaultValue?: string,
  keyboardType?: string,
);
```

Create and display a prompt to enter some text in form of Alert.

Parameters:

NAME	TYPE	DESCRIPTION
title Required	string	The dialog's title.
message	string	An optional message that appears above the text input.
callbackOrButtons	function	If passed a function, it will be called with the prompt's value (text: string) => void, when the user taps 'OK'.
	AlertButton[]	If passed an array, buttons will be configured based on the array content.

NAME	TYPE	DESCRIPTION
type	AlertType	This configures the text input.
defaultValue	string	The default text in text input.
keyboardType	string	The keyboard type of first text field (if exists). One of TextInput keyboardTypes.
options	Options	An optional Alert configuration.

Type Definitions

AlertButtonStyle ◀ ios

An iOS Alert button style.

TYPE	
enum	

Constants:

VALUE	DESCRIPTION
'default'	Default button style.
'cancel'	Cancel button style.
'destructive'	Destructive button style.

AlertType ◀ ios

An iOS Alert type.

ТҮРЕ		
enum		

Constants:

9/6/23, 12:47 AM

VALUE	DESCRIPTION
'default'	Default alert with no inputs
'plain-text'	Plain text input alert
'secure-text'	Secure text input alert
'login-password'	Login and password alert

AlertButton

An object describing the configuration of a button in the alert.

TYPE				
array of o	bjects			

Objects properties:

NAME	TYPE	DESCRIPTION
text	string	Button label.
onPress	function	Callback function when button is pressed.
style (ios	AlertButtonStyle	Button style, on Android this property will be ignored.
isPreferred ₫ ios	boolean	Whether button should be emphasized, on Android this property will be ignored.

AlertOptions

TYPE	
object	

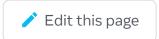
Properties:

NAME	TYPE	DESCRIPTION
cancelable Android	boolean	Defines if alert can be dismissed by tapping outside of the alert box.
userInterfaceStyle d ios	string	The interface style used for the alert, can be set to light or dark, otherwise the default system style will be used.
onDismiss (Android	function	Callback function fired when alert has been dismissed.

Is this page useful?







Last updated on **Jun 21, 2023**