

**Version: 3.x**

# useAnimatedKeyboard

## ! INFO

This page was ported from an old version of the documentation.

As we're rewriting the documentation some of the pages might be a little outdated.

## ! CAUTION

Android implementation of `useAnimatedKeyboard` is an experimental feature.

With the `useAnimatedKeyboard` hook, you can create animations based on current keyboard position.

On Android, make sure to set `android:windowSoftInputMode` in your `AndroidManifest.xml` to `adjustResize`. Then, using the `useAnimatedKeyboard` hook disables the default Android behavior (resizing the view to accommodate keyboard) in the whole app. Using values from `useAnimatedKeyboard` hook you can handle the keyboard yourself. Unmounting all components that use `useAnimatedKeyboard` hook brings back the default Android behavior.

## Arguments

`options` [**`AnimatedKeyboardOptions`**]

Optional object containing additional configuration.

## Returns

Hook `useAnimatedKeyboard` returns an instance of [`AnimatedKeyboardInfo`](#);

## Types

AnimatedKeyboardInfo: [object]

Properties:

- `height`: [[SharedValue](#)] contains [number] contains current height of the keyboard
- `state`: [[SharedValue](#)] contains [enum] contains current state of the keyboard. Possible states: { CLOSED, OPEN, CLOSING, OPENING }

AnimatedKeyboardOptions: [object]

Properties:

- `isStatusBarTranslucentAndroid` [bool] - if you want to use translucent status bar on Android, set this option to `true`. Defaults to `false`. Ignored on iOS.

## Example

```
function AnimatedKeyboardExample() {
  const keyboard = useAnimatedKeyboard();
  const translateStyle = useAnimatedStyle(() => {
    return {
      transform: [{ translateY: -keyboard.height.value }],
    };
  });

  return (
    <ScrollView
      contentContainerStyle={{
        flex: 1,
        justifyContent: 'center',
        alignItems: 'center',
      }}>
      <Animated.View style={translateStyle}>
        <TextInput />
      </Animated.View>
    </ScrollView>
  );
}
```

 [Edit this page](#)