🏠   API reference   Gesture Handlers   Tap

Version: 2.6.0 – 2.12.0

TapGestureHandler

> 🔥 **DANGER**
>
> Consider using the new gestures API instead. The old API is not actively supported and is not receiving the new features. Check out RNGH 2.0 section in Introduction for more information.

A discrete gesture handler that recognizes one or many taps.

Tap gestures detect one or more fingers briefly touching the screen. The fingers involved in these gestures must not move significantly from their initial touch positions. The required number of taps and allowed distance from initial position may be configured. For example, you might configure tap gesture recognizers to detect single taps, double taps, or triple taps.

In order for a handler to activate, specified gesture requirements such as minPointers, numberOfTaps, maxDist, maxDurationMs, and maxDelayMs (explained below) must be met. Immediately after the handler activates, it will END.

Properties

See set of properties inherited from base handler class. Below is a list of properties specific to the `TapGestureHandler` component:

`minPointers`

Minimum number of pointers (fingers) required to be placed before the handler activates. Should be a positive integer. The default value is 1.

`maxDurationMs`

Maximum time, expressed in milliseconds, that defines how fast a finger must be released after a touch. The default value is 500.

`maxDelayMs`

Maximum time, expressed in milliseconds, that can pass before the next tap — if many taps are required. The default value is 500.

`numberOfTaps`

Number of tap gestures required to activate the handler. The default value is 1.

`maxDeltaX`

Maximum distance, expressed in points, that defines how far the finger is allowed to travel along the X axis during a tap gesture. If the finger travels further than the defined distance along the X axis and the handler hasn't yet activated, it will fail to recognize the gesture.

`maxDeltaY`

Maximum distance, expressed in points, that defines how far the finger is allowed to travel along the Y axis during a tap gesture. If the finger travels further than the defined distance along the Y axis and the handler hasn't yet activated, it will fail to recognize the gesture.

`maxDist`

Maximum distance, expressed in points, that defines how far the finger is allowed to travel during a tap gesture. If the finger travels further than the defined distance and the handler hasn't yet activated, it will fail to recognize the gesture.

Event data

See set of event attributes from base handler class. Below is a list of gesture event attributes specific to the `TapGestureHandler` component:

`x`

X coordinate, expressed in points, of the current position of the pointer (finger or a leading pointer when there are multiple fingers placed) relative to the view attached to the handler.

`y`

Y coordinate, expressed in points, of the current position of the pointer (finger or a leading pointer when there are multiple fingers placed) relative to the view attached to the handler.

`absoluteX`

X coordinate, expressed in points, of the current position of the pointer (finger or a leading pointer when there are multiple fingers placed) relative to the window. It is recommended to use `absoluteX` instead of `x` in cases when the view attached to the handler can be transformed as an effect of the gesture.

`absoluteY`

Y coordinate, expressed in points, of the current position of the pointer (finger or a leading pointer when there are multiple fingers placed) relative to the window. It is recommended to use `absoluteY` instead of `y` in cases when the view attached to the handler can be transformed as an effect of the gesture.

Example

See the multitap example from GestureHandler Example App.

```
export class PressBox extends Component {
  doubleTapRef = React.createRef();
  render() {
    return (
      <TapGestureHandler
        onHandlerStateChange={this._onSingleTap}
        waitFor={this.doubleTapRef}>
        <TapGestureHandler ref={this.doubleTapRef} numberOfTaps={2}>
          <View style={styles.box} />
        </TapGestureHandler>
      </TapGestureHandler>
    );
  }
}
```