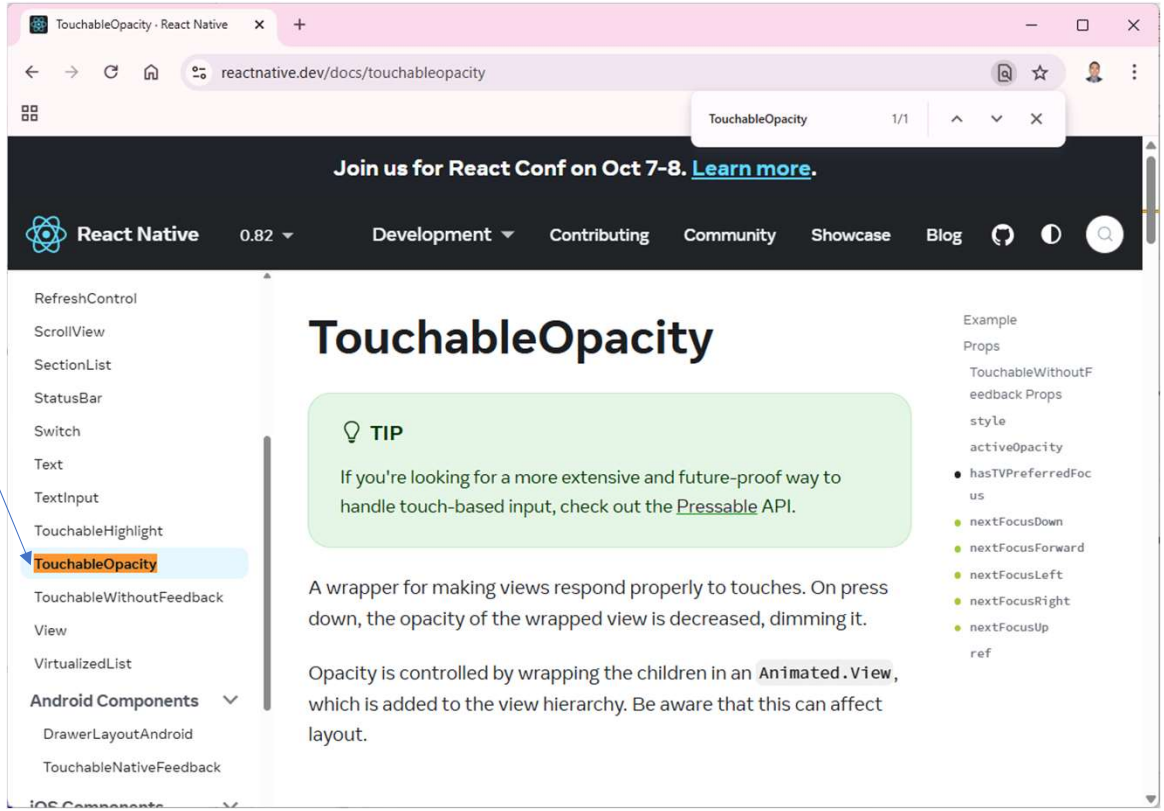
Reference:
https://reactnative.dev/docs/touchableopacity

**TouchableOpacity**



# Press Events

`onPress` in `<Button>`, `<TouchableOpacity>`, `<Pressable>`

It makes any component clickable/tappable with a visual feedback effect..

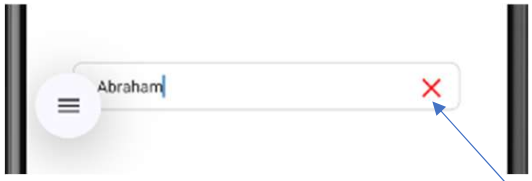
**1.Visual Feedback** - When pressed, it automatically reduces opacity (fades slightly), giving users visual confirmation that they tapped something
**2.Makes Non-Button Elements Clickable** - You can wrap any component (Text, View, Image, Icon) to make it tappable:

```jsx
import React, { useState } from 'react';
import { StyleSheet, View, TextInput, TouchableOpacity, Text } from 'react-native';

export default function App() {
  const [text, setText] = useState('');

  const handleClear = () => {
    setText('');
  };

  return (
    <View style={styles.container}>
      <View style={styles.inputContainer}>
        <TextInput
          placeholder="Type something..."
          style={styles.input}
          value={text}
          onChangeText={setText}
        />
        {text.length > 0 && (
          <TouchableOpacity onPress={handleClear} style={styles.clearButton}>
            <Text style={styles.clearButtonText}>X</Text>
          </TouchableOpacity>
        )}
      </View>
    </View>
  );
}
```

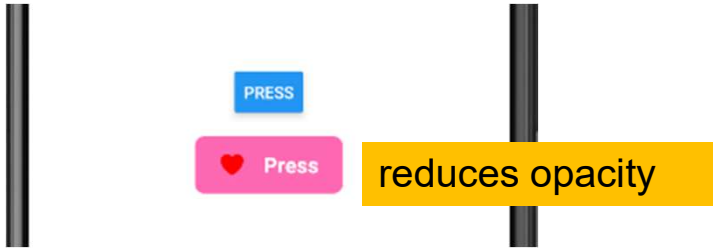



reduces opacity

```jsx
const styles = StyleSheet.create({
  container: {
    flex: 1,
    padding: 40,
    justifyContent: 'center',
  },
  inputContainer: {
    flexDirection: 'row',
    alignItems: 'center',
    borderWidth: 1,
    borderColor: '#ccc',
    borderRadius: 8,
    paddingHorizontal: 10,
  },
  input: {
    flex: 1,
    padding: 10,
  },
  clearButton: {
    padding: 5,
  },
  clearButtonText: {
    color: 'red',
    fontSize: 20,
    fontWeight: 'bold',
  },
});
```

# onPress in <Button>, <TouchableOpacity>, <Pressable>

**Compare <Button -vs- <TouchableOpacity**



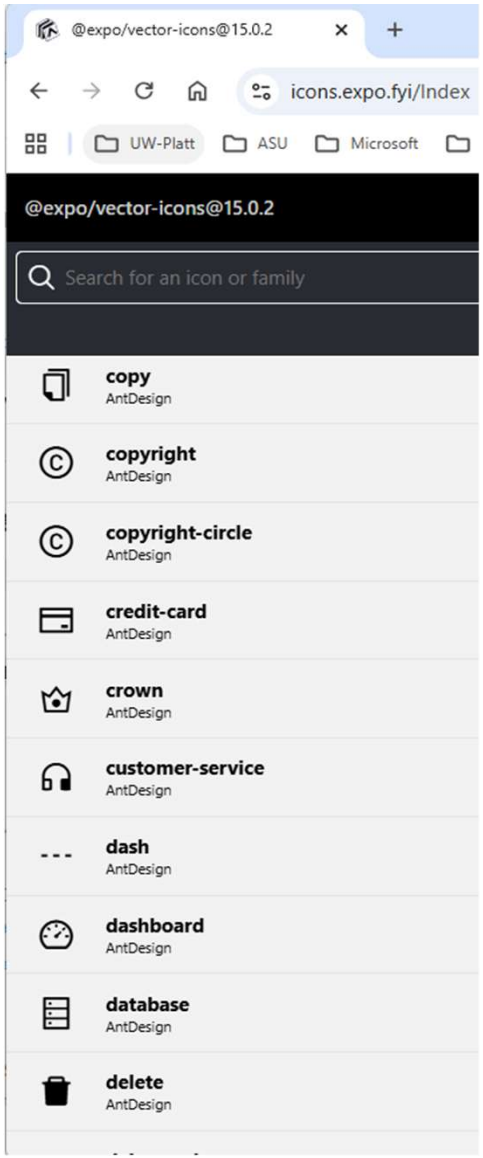

```jsx
import React from 'react';
import { StyleSheet, View, Button, TouchableOpacity, Text } from 'react-native';
import { Ionicons } from '@expo/vector-icons';
export default function App() {

  const handlePress = () => {
    console.log('Button pressed!');
    alert('You pressed the button!');
  };

  return (
    <View style={styles.container}>
      {/* Button - limited styling */}
      <Button title="Press" onPress={handlePress} />

      {/* TouchableOpacity - full control */}
      <TouchableOpacity onPress={handlePress} style={styles.customButton}>
        <Ionicons name="heart" size={24} color="red" style={styles.icon} />
        <Text style={styles.customText}>Press</Text>
      </TouchableOpacity>
    </View>
  );
}
```

```jsx
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    padding: 40,
    gap: 20,
  },
  customButton: {
    flexDirection: 'row',
    alignItems: 'center',
    backgroundColor: '#ff69b4',
    paddingVertical: 12,
    paddingHorizontal: 20,
    borderRadius: 8,
    gap: 10,
  },
  customText: {
    color: 'white',
    fontSize: 18,
    fontWeight: 'bold',
  },
  icon: {
    marginRight: 5,
  },
});
```

https://icons.expo.fyi/Index

```
import { Ionicons } from '@expo/vector-icons';

<Ionicons name="home" size={30} color="blue" />
<Ionicons name="search" size={30} color="green" />
<Ionicons name="heart" size={30} color="red" />
<Ionicons name="settings" size={30} color="gray" />
<Ionicons name="close-circle" size={30} color="red" />
<Ionicons name="checkmark-circle" size={30} color="green" />
```