

LISTEN

Shortcuts

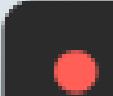
react-keys-shortcuts

Slide to know



Blessed raj

Installation



```
// installation
npm install react-keys-shortcuts
```



2
Blessed raj

Hooks

- **useListenEffect** - Globally listen to keyboard shortcuts and trigger actions.
- **useListenEvent** - Bind a shortcut formula to an existing click event.

```
// installation
import {useListenEvent, useListenEffect} from 'react-keys-shortcuts';
```



Usage

useListenEffect



```
import { useListenEffect } from 'react-keys-shortcuts';

const App = () => {
  useListenEffect('ctrl+shift+l', () => {
    alert('Shortcut triggered!');
  });

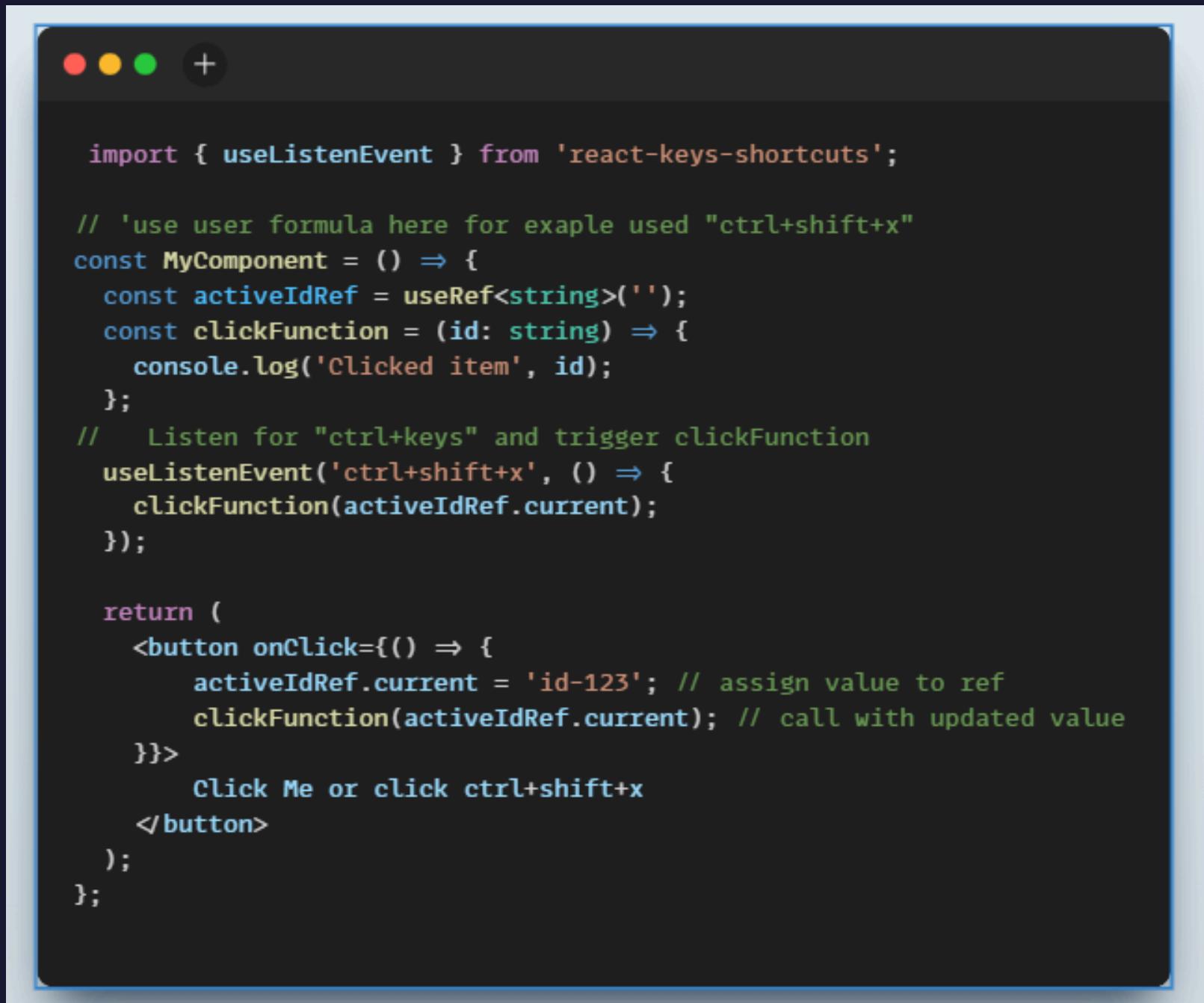
  return <div>Press ctrl+shift+l</div>;
};

export default App;
```



Usage

useListenEvent



A screenshot of a code editor window showing a React component. The component imports `useListenEvent` from `'react-keys-shortcuts'`. It defines a function `MyComponent` that uses a ref `activeIdRef` and a click function `clickFunction`. It then uses `useListenEvent` to listen for the `'ctrl+shift+x'` combination and triggers the `clickFunction` with the current value of `activeIdRef`. Finally, it returns a button that calls the click function when clicked or triggers the event when `ctrl+shift+x` is pressed.

```
import { useListenEvent } from 'react-keys-shortcuts';

// 'use user formula here for example used "ctrl+shift+x"
const MyComponent = () => {
  const activeIdRef = useRef<string>('');
  const clickFunction = (id: string) => {
    console.log('Clicked item', id);
  };
  // Listen for "ctrl+keys" and trigger clickFunction
  useListenEvent('ctrl+shift+x', () => {
    clickFunction(activeIdRef.current);
  });

  return (
    <button onClick={() => {
      activeIdRef.current = 'id-123'; // assign value to ref
      clickFunction(activeIdRef.current); // call with updated value
    }}>
      Click Me or click ctrl+shift+x
    </button>
  );
};

export default MyComponent;
```





Comment your
thoughts, for more
content lets connect



Blessed raj