

```

function Example() {
  const [count, setCount] = useState(0)

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() =>
setCount(count + 1)}>
        Click here to increase the
counter.
      </button>
    </div>
  )
}

```

```

constructor(props) {
  super(props);
  this.textInput = React.createRef();
}

componentDidMount() {
  this.textInput.current.focus();
}

render() {
  return (
    <input ref={this.textInput} />
  );
}
}

```

Higher Order Component

```

import React from 'react'
import Loading from '../components/Loading'

const withLoading = WrappedComponent => {
  return (props = {}) => {
    if (props.isLoading) {
      return <Loading />
    }
    return <WrappedComponent {...props} />
  }
}

export default withLoading

// -----

const MyComponent = ({}) => <div /> // ...
const WithLoadingComponent =
withLoading(MyComponent)

```

Render Props

```

import React from 'react'

const MOBILE_VIEW_WIDTH_THRESHOLD =
600

class MediaQuery {
  state = {
    shouldShowMobileView: false
  }

  componentDidMount() {
    addEventListener('resize',
this.updateResizeStatus)
  }

  componentWillUnmount() {
    removeEventListener('resize',
this.updateResizeStatus)
  }

  updateResizeStatus() {
    if (screen.width <=
MOBILE_VIEW_WIDTH_THRESHOLD) {
      this.setState({
        shouldShowMobileView: true
      })
    }
  }

  render() {
    return this.props.children(
      this.state.shouldShowMobileView
    )
  }
}

```