Cheatography

REACTJS Cheat Sheet

by restran via cheatography.com/64008/cs/16540/

Components

```
import React from 'react'
import ReactDOM from 'react -dom'
class Hello extends React.C om ponent {
render () {
return <div classN ame ='m ess age -bo x'> <Us erA vatar src={a vatar} />
Hello {this.p ro ps.n ame}
</d iv>
}
}
const el = docume nt.body
ReactD OM.r en der (<Hello name=' John' />, el)
Use the React.js jsfiddle to start hacking. (or the unofficial jshin)
```

Properties

```
<Video fullsc ree n={ true} />
render () {
  this.p rop s.f ull screen
  ...
}
Use this.props to access properties passed to the component.
See: Properties
```

Nesting

```
class Info extends React.Component {
    render () {
      const { avatar, username } = this.props
      return <di v>
        <Us erA vatar src={a vatar} />
        <Us erP rofile userna me= {us ername}
      />
      </d iv>
    }
}
```

As of React v16.2.0, fragments can be used to return multiple children without adding extra wrapping nodes to the DOM.

States

```
constructor(props) {
    sup er( props)
    thi s.state = {}
}
this.s etS tate({ username:
'rstacruz' })
render () {
    thi s.s tat e.u sername
    ...
}
```

Use states (this.state) to manage dynamic data.

See: States

Children

```
<AlertBox>
    <h1 >You have pending
notifi cat ion s</ h1>
</A ler tBo x>
class AlertBox extends React.C -
om ponent {
    render () {
        return <div classN ame -
='a ler t-b ox'>
            {th is.p ro ps.c -
hi ldren}
        </d iv>
    }
}
```

Children are passed as the children property.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
http://crosswordcheats.com