Enzyme cheatsheet

Proudly sponsored by
Airbrake.io Full-stack error tracking & analytics
for JS developers. Try it Free!
ethical ad by CodeFund

Enzyme lets you write unit tests for React components. This guide covers Enzyme 3.x.

Mounting

```
import shallow, mount from 'enzyme'

wrap = shallow(MyComponent />)

wrap = mount(MyComponent />)
```

Shallow wrapping doesn't descend down to sub-components. A full mount also mounts sub-components.

See: Shallow rendering, Full rendering

Debugging

```
console.log(wrap.debug())
```

Shows HTML for debugging purposes.

See: debug()

Examples

Basic example

```
import { shallow } from 'enzyme'
import MyComponent from '../MyComponent'

it('works', () => {
  const wrap = shallow(
    <MyComponent name="Groot" />
  )

  expect(wrap.text()).toEqual('I am Groot')
})
```

Props and state

Setting

```
wrap.setProps({ name: 'Moe' })
wrap.setState({ show: true })
```

Asserting

```
expect(wrap.props('name')).toEqual('Moe')
expect(wrap.state('show')).toEqual(true)

expect(wrap.name() wrap.prop('name')).toEqual(true)
expect(wrap.name() wrap.state())).toEqual(true)
```

Matching elements

```
expect(
  wrap.containsMatchingElement(
    <span>I am groot</span>
  )
).toBeTruthy()
```

`containsMatchingElement()` is probably the most useful assertion in Jest.

Snapshots

```
expect(wrap).toMatchSnapshot()
```

Be sure you've set up enzyme-to-json for snapshots (see Installing below).

Traversions

```
expect(
  wrap.find('button').text()
).toEqual('Submit')
```

Use `.find()` to traverse down to nodes. It will return wrapper objects, too.

Simulating events

```
wrap.find('input').simulate("click")
```

With event object props

```
wrap.find('input').simulate('change', {
  target: { value: 'Hello' }
})
```

Installing

Initial setup

```
npm install --save-dev enzyme \
  enzyme-adapter-react-16 \
  react-test-renderer
```

test/setup.js

```
import Enzyme from 'enzyme'
import Adapter from 'enzyme-adapter-react-16'

Enzyme.configure({ adapter: new Adapter() })
```

package.json

```
"jest": {
  "setupFiles": [
    "test/setup.js"
]
}
```

This configures Enzyme for React v16, and Jest to automatically configure Enzyme for you. There are other adapters in Enzyme's installation instructions.

See: Installation

Jest snapshots

```
npm install --save-dev enzyme-to-json
```

package.json

```
"jest": {
  "snapshotSerializers": [
    "enzyme-to-json/serializer"
]
}
```

Test

```
it('works', () => {
  wrap = mount(MyComponent />)
  expect(wrap).toMatchSnapshot()
})
```

Optional, but recommended: This allows you to use Enzyme wrappers with Jest snapshots.

See: enzyme-to-json

ReactWrapper

Traversing

```
wrap.find('button')  // => ReactWrapper
wrap.find(Foo)       // => ReactWrapper
wrap.not('span')     // => ReactWrapper (inverse of f
wrap.children()      // => ReactWrapper
wrap.parent()        // => ReactWrapper
wrap.closest('div')  // => ReactWrapper
wrap.childAt(0)      // => ReactWrapper
wrap.at(0)           // => ReactWrapper
wrap.first()         // => ReactWrapper
wrap.last()          // => ReactWrapper

wrap.get(0)          // => ReactElement
wrap.getNode()       // => ReactElement
wrap.getNodes()      // => Array<ReactElement>
wrap.getDOMNode()    // => DOMComponent
```

See: Full rendering API

Actions

```
wrap.simulate('click')
```

React components

```
wrap.setState({ xxx: 3})
wrap.setProps({ xxx: 3})
wrap.setContext({ xxx: 1})

wrap.state()         // get full state
wrap.props()         // get full props
wrap.context()       // get full context

wrap.state('key')    // => any
wrap.prop('key')     // => any
wrap.context('key')  // => any

wrap.instance()      // => ReactComponent
```

Mount

```
wrap.mount()
wrap.unmount()
wrap.update()        // calls forceUpdate()
```

Tests

```
wrap.debug()                 // => string
wrap.html()                  // => string
wrap.text()                  // => string
wrap.type()                  // => string | function
wrap.name()                  // => string
wrap.is('.classname')        // => boolean
wrap.hasClass('class')       // => boolean
wrap.exists()                // => boolean
wrap.contains(<div />)       // => boolean
wrap.contains([ <div />, <h1 />  // => boolean
wrap.some('.child')          // => boolean

wrap.someWhere(n => n.hasClass('foo'))

wrap.containsMatchingElement(<div />)      // => bo
wrap.containsAllMatchingElements([ <div />, <h1 />  // => bo
wrap.containsAnyMatchingElements([ <div />, <h1 />  // => bo
```

References

- Enzyme website (airbnb.io)
- Enzyme v2 cheatsheet (devhints.io) (old version)

3 Comments for this cheatsheet. Write yo

Search 380+ cheatsheets

Over 380 curated cheatsheets, by developers for developers.
Devhints home

Other React cheatsheets

React.js cheatsheet · Redux cheatsheet · Enzyme v2 cheatsheet · Awesome Redux cheatsheet · Flux architecture cheatsheet · React-router cheatsheet

Top cheatsheets

Elixir cheatsheet · ES2015+ cheatsheet · React.js cheatsheet · Vim/Emacs cheatsheet · Vim cheatsheet · Vim scripting cheatsheet