原文地址：<https://www.cnblogs.com/vvjiang/p/8599980.html>

参考地址：https://www.cnblogs.com/susu8/p/9512393.html

Enzyme是用于React的测试，需搭配其他单元测试使用

**安装**

安装enzyme

npm install --save-dev enzyme

安装对应配适器

npm install --save-dev enzyme-adapter-react-16

配适器对应react的列表

enzyme-adapter-react-16 ^16.0.0

enzyme-adapter-react-15 ^15.5.0

enzyme-adapter-react-15.4 15.0.0-0 - 15.4.x

enzyme-adapter-react-14 ^0.14.0

enzyme-adapter-react-13 ^0.13.0

**基本示例（搭配Jest）**

import React from 'react'

import Enzyme from 'enzyme'

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

// 配置配适器

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

const {shallow}=Enzyme

test('test1', function () {

let button = shallow(<button>test</button>);

    expect(button.text()).toEqual("test");

})

**Shallow方法**

Shallow将一个组件渲染成虚拟DOM对象，但是只渲染第一层，不渲染所有子组件

**mount方法**

mount方法用于将React组件加载为真实DOM节点。然而真实DOM需要一个浏览器环境，为了解决这个问题，我们可以用到jsdom

安装jsdom：

npm install --save-dev jsdom

在test目录下新建一个文件setup.js：

import jsdom from 'jsdom';

const { JSDOM } = jsdom;

// 使用jsdom环境

if (typeof document === 'undefined') {

    const dom=new JSDOM('<!doctype html><html><head></head><body></body></html>');

    global.window =dom.window;

    global.document = global.window.document;

    global.navigator = global.window.navigator;

}

将setup.js配置到jest（运行每个测试文件前先运行setup.js）

setupFiles:["<rootDir>/test/setup.js"],

mount的使用方法和Shallow一样

**wrapper对象**

shallow和mount返回的是wrapper对象，其具有的方法如下：

（详细Api地址http://airbnb.io/enzyme/docs/api/ShallowWrapper/at.html）

.get(index)：返回指定位置的子组件的DOM节点

.at(index)：返回指定位置的子组件

.first()：返回第一个子组件

.last()：返回最后一个子组件

.type()：返回当前组件的类型

.text()：返回当前组件的文本内容

.html()：返回当前组件的HTML代码形式

.props()：返回根组件的所有属性

.prop(key)：返回根组件的指定属性

.state([key])：返回根组件的状态

.setState(nextState)：设置根组件的状态

.setProps(nextProps)：设置根组件的属性

.instance()：返回我们React类的实例，包含我们自定义定义的React类的方法