React Testing Library

github.com/kentcdodds/react-testing-library

Simple and complete cheat sheet v1

render a component

```
import {render} from ‘react-testing-library’

// See render result for more details…
const result = render(<MyComponent />)
```

search the DOM

```
const {getByText} = render(<span>hello</span>)

// Check out jest-dom on npm for handy assertions…
const element = getByText(‘hello’)
```

fire an event

```
import {fireEvent} from ‘react-testing-library’

fireEvent.click(element)
// Or you can fire events manually with…
fireEvent(element, new MouseEvent(‘click’)
```

pretty print the DOM

```
const {debug} = render(<div>hello</div>)

// Pretty print the DOM tree of your render
debug()
```

wait for something

```
import {wait} from ‘react-testing-library’

// Retry search every 50ms for 4500ms
wait(() => getByText(‘async result’))
```

search variants

	(return value)
getBy	Element *or* Error
getAllBy	Element[] *or* Error
queryBy	Element *or* null
queryAllBy	Element[] *or* []
findBy	Promise<Element> *or* Error
findAllBy	Promise<Element[]> *or* Error

search types

	(DOM attribute)
LabelText	<label for=“element” />
PlaceholderText	<input placeholder=“username” />
Text	<a href="/about">About</a>
AltText	<img alt=“movie poster” />
Title	<span title="Delete" /> *or* <title />
DisplayValue	Current value of input element
Role	<div role="dialog">...</div>
TestId	<input data-testid="username-input" />

text matches

```
const {getByText} = render(<div>Hello World</div>)

getByText('Goodbye World’)                        // ❌
getByText(/hello world/)                          // ❌
getByText(‘ello Worl’, {exact: false})            // ✅
getByText(‘Hello World’)                          // ✅
```

wait for appearance

```
test('movie title appears', async () => {
  // Element is initially not present...
  expect(() => getByText(‘aladdin’)).toThrow()

  // Wait for appearance
  await wait(() => {
    expect(getByText(‘aladdin’)).toBeTruthy()
  })

  // Wait for appearance and return the result
  await waitForElement(() => getByText(‘aladdin’))
})
```

assert for absence

```
expect(queryByText('submit')).toBeNull()
```

the render result

	(description)
container	The target of ReactDOM.render()
baseElement	App wrapper (usually <body> tag)
debug(element)	Pretty print the DOM
unmount()	Unmount your component
rerender(ui)	Render again at the container
…queries	Queries for the baseElement

cleanup the DOM

```
import ‘react-testing-library/cleanup-after-each’

// Or, for more control…
import {cleanup} from ‘react-testing-library’

afterEach(cleanup)
```