

Date : 27 - 07 - 2020

Morning Session : 9 am – 11.00 PM

By ~ Sundeep Charan Ramkumar Today

Topics: React component testing

Types:

Unit:

Verify that individual, isolated parts work as expected.

Integration:

Verify that several units work together in harmony.

End to End:

A helper robot that behaves like a user to click around the app and verify that it functions correctly. Sometimes called "functional testing" or e2e.

<https://jestjs.io/docs/en/tutorial-react>

react-testing-library/cheatsheet

Cheatsheet

EDIT

Get the printable cheat sheet

A short guide to all the exported functions in **React Testing Library**

- **render** `const {/* */} = render(Component)` returns:
 - all the queries from **DOM Testing Library**, bound to the document so there is no need to pass a node as the first argument
 - **unmount** function to unmount the component
 - **container** reference to the DOM node where the component is mounted

```
import { render, fireEvent } from '@testing-library/react'

test('loads items eventually', async () => {
  const { getByText, findByText } = render(<Page />)

  // Click button
  fireEvent.click(getByText('Load'))

  // Wait for page to update with query text
  const items = await findByText(/Item #[0-9]: /)
  expect(items).toHaveLength(10)
})
```

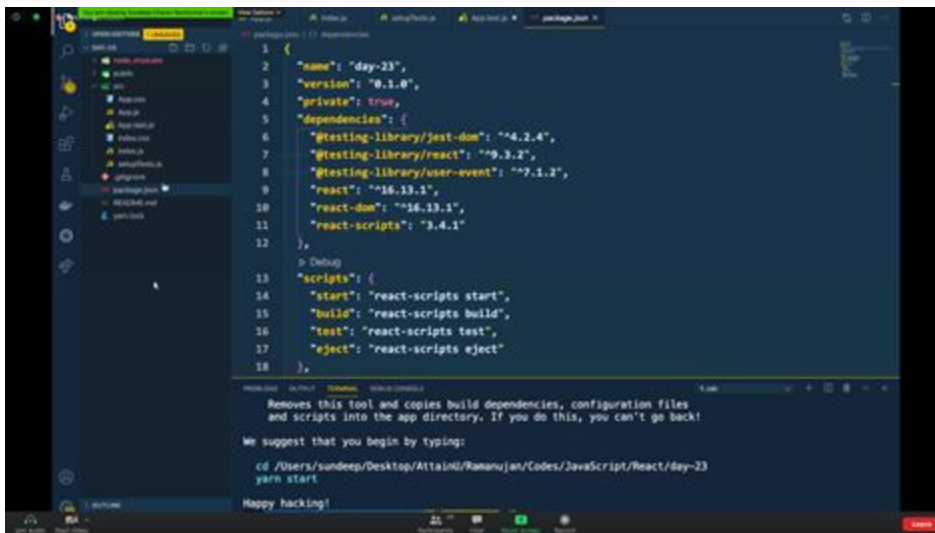
 Copy

<https://testing-library.com/docs/react-testing-library/cheatsheet>

The React Testing Library is a very lightweight solution for testing React components. It provides light utility functions on top of react-dom and react-dom/test-utils, in a way that encourages better testing practices. Its primary guiding principle is:

The more your tests resemble the way your software is used, the more confidence they can give you.

React has everything already setup for us.



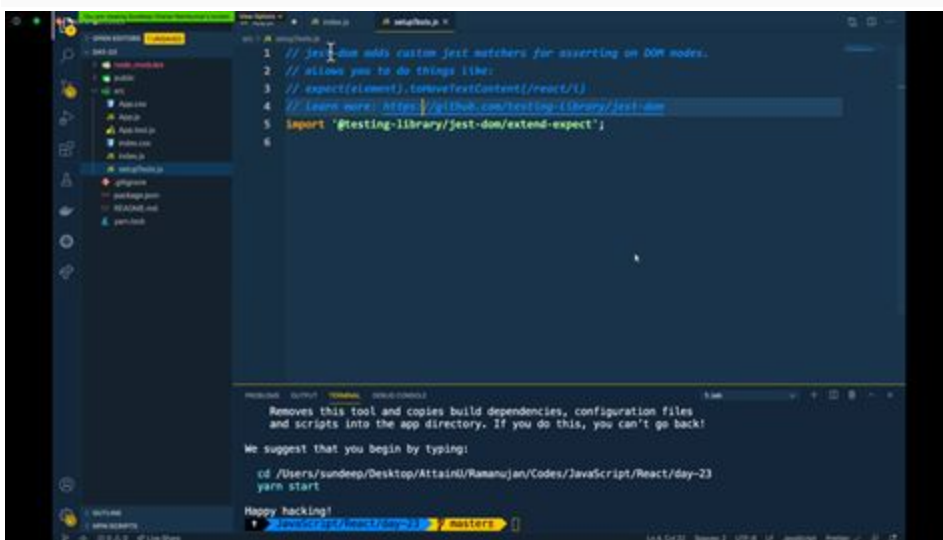
```
1 {
2   "name": "day-23",
3   "version": "0.1.0",
4   "private": true,
5   "dependencies": {
6     "@testing-library/jest-dom": "^4.2.4",
7     "@testing-library/react": "^9.3.2",
8     "@testing-library/user-event": "^7.1.2",
9     "react": "^16.13.1",
10    "react-dom": "^16.13.1",
11    "react-scripts": "3.4.1"
12  },
13   "scripts": {
14     "start": "react-scripts start",
15     "build": "react-scripts build",
16     "test": "react-scripts test",
17     "eject": "react-scripts eject"
18  },
19}
```

Removes this tool and copies build dependencies, configuration files and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

```
cd /Users/sundee/Desktop/AttainU/Ramanujan/Codes/JavaScript/React/day-23
yarn start
```

Happy hacking!



```
1 // jest-dom adds custom jest matchers for asserting on DOM nodes.
2 // allows you to do things like:
3 // expect(element).toHaveTextContent(/react/i)
4 // learn more: https://github.com/testing-library/jest-dom
5 import '@testing-library/jest-dom/extend-expect'
```

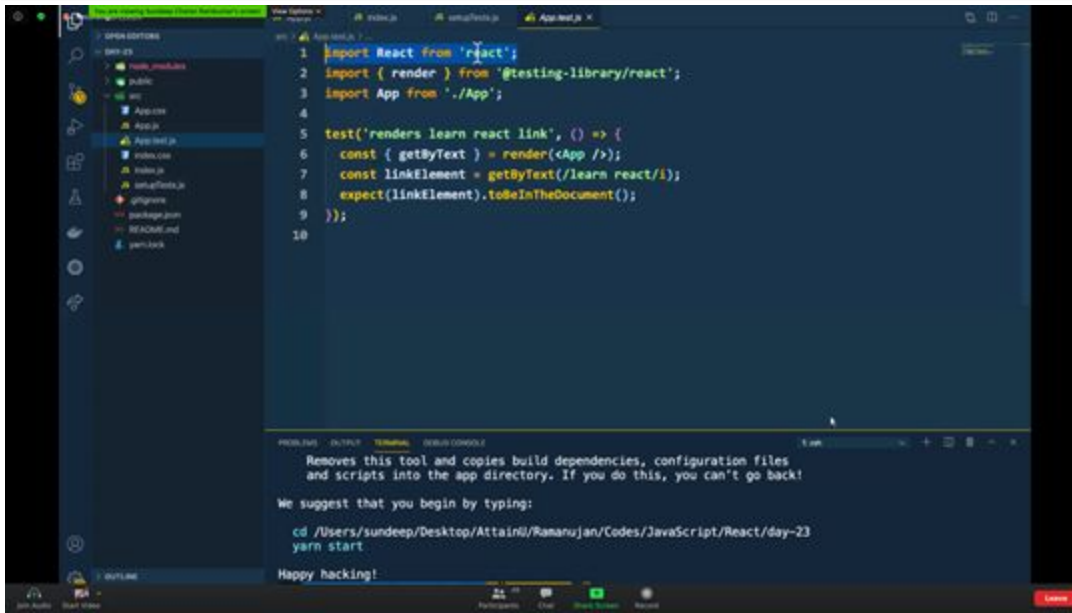
Removes this tool and copies build dependencies, configuration files and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

```
cd /Users/sundee/Desktop/AttainU/Ramanujan/Codes/JavaScript/React/day-23
yarn start
```

Happy hacking!

Boilerplate for Testing:



```
1 import React from 'react';
2 import { render } from '@testing-library/react';
3 import App from './App';
4
5 test('renders learn react link', () => {
6   const { getByText } = render(<App />);
7   const linkElement = getByText(/learn react/i);
8   expect(linkElement).toBeInTheDocument();
9 });
10
```

Removes this tool and copies build dependencies, configuration files and scripts into the app directory. If you do this, you can't go back!

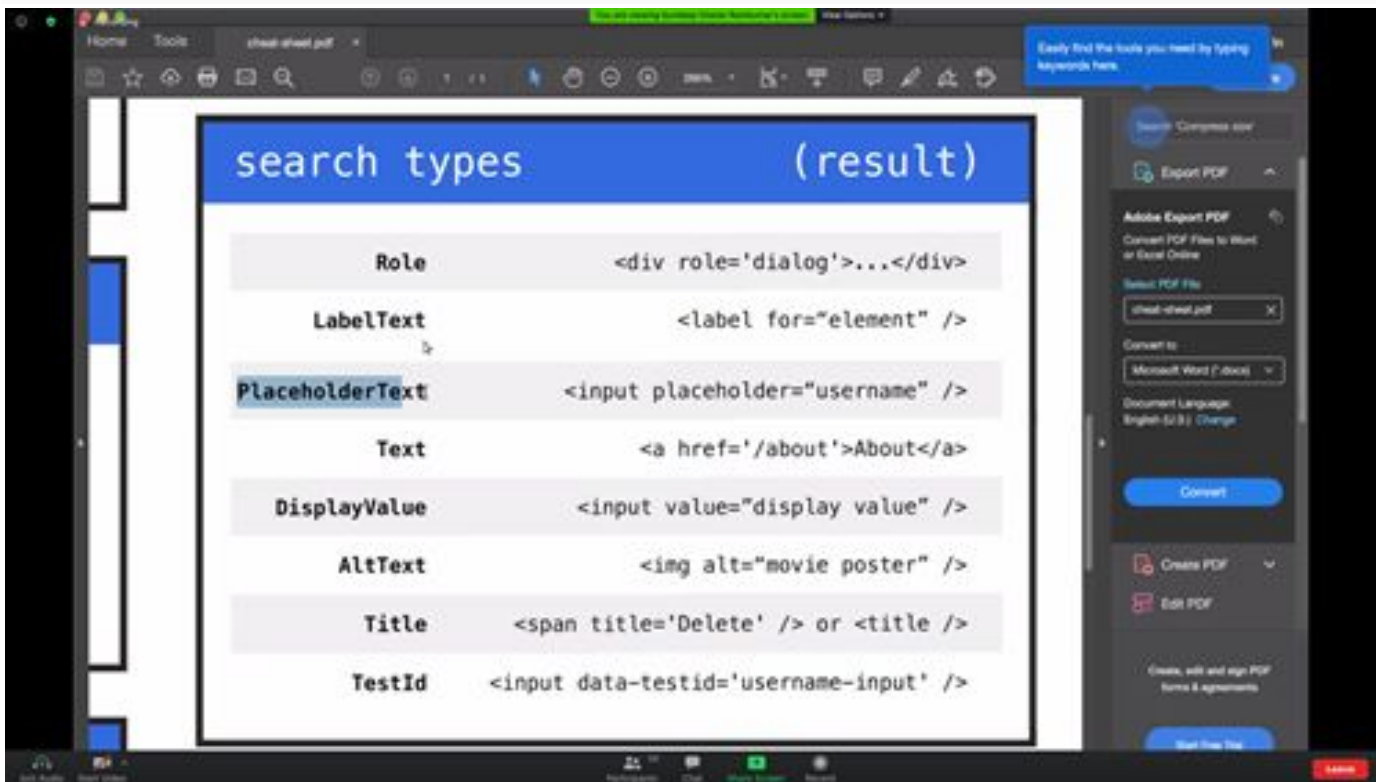
We suggest that you begin by typing:

```
cd /Users/sundeep/Desktop/AttainU/Ramanujan/Codes/JavaScript/React/day-23
yarn start
```

Happy hacking!

Node doesn't have DOM so yesterday we were not able to test DOM elements but React needs it, so we have that in React-Library

check for different DOM elements



Testing: getBy queryBy and findBy → selecting elements

getby : Gives Element or error

queryBy :Gives Element on null

findBy : Useful when you expect an element to pop up in due time

React testing library allows us to test Regular DOM nodes.
True or False?

Attempted - 30
(100%)

EASY ^

- | | |
|--|-----|
| <input checked="" type="checkbox"/> True | 80% |
| <input type="checkbox"/> False | 20% |

Which method allows us to check for conditionally rendered elements?

Attempted - 33
(110%)

EASY ^

- | | |
|--|--------|
| <input type="checkbox"/> getBy | 12.12% |
| <input type="checkbox"/> queryBy | 60.61% |
| <input checked="" type="checkbox"/> findBy | 30.3% |

The ability to mimic a real world functionality is called as?

Attempted - 34
(113.33%)

EASY v

Which method allows us to record events?

Attempted - 33 (110%)

EASY ^

- | | |
|---|--------|
| <input type="checkbox"/> listenEvent | 42.42% |
| <input checked="" type="checkbox"/> fireEvent | 45.45% |
| <input type="checkbox"/> callEvent | 12.12% |

The ability to mimic a real world functionality is called as?

Attempted - 34
(113.33%)

EASY



- | | | |
|-------------------------------------|---------|--------|
| <input type="checkbox"/> | cloning | 26.47% |
| <input checked="" type="checkbox"/> | mocking | 73.53% |

Resources:

<https://medium.com/@alialhaddad/a-beginners-guide-to-jest-testing-858d10198032>

<https://github.com/puppeteer/puppeteer>

<https://www.youtube.com/watch?v=D9DdY2WmM-s>