Imp courses next

solid principle

Micro Services Arcitecture

Kafka

Docker

Testing

With React Testing Library and Jest

What is Testing

!. Software testing is the process of finding errors inn a software product before it is launched.

Validate functioning and features of software

Types of testing

Manual

Automation

Testing By Developers

Unit Testing – Testing individual units or components

Integrating Testing- Testing between two units or compare

E-2E Testing – Test Start to end complete project

React Testing Tools

Jest Framework

React Testing Library

Where we will focus?

Mostly we focus on unit testing and integrating Testing

But we will try to cover some E-2-E testing topics

React developer needs to do component testing mostly that come under Unit testing

Run React Test

Install App

Check Testing Library

Run Test case

Check Output

Iq

In package.json

 "@testing-library/jest-dom": "^5.17.0",

    "@testing-library/react": "^13.4.0",

    "@testing-library/user-event": "^13.5.0",

These are here

Then go to App.test.js

To run test case….

npm run test

Write basic Test Case

Make Js file a simple function

Make Test file

What is Test function

What is Expect function

Write Test case

Run Test case

Sum.test.js

npm run test

Test Suites: 2 passed, 2 total (Means - file )

Tests: 2 passed, 2 total (means - how many cases )

Snapshots: 0 total

Time: 2.68 s

Ran all test suites.

In create-react-app automatic installed jest

Understand React Test Structure

What needs to import

How to render the component

How to Tests working with RTL

How test finding UI elements

Write First React Test

Make new test function

Write test case for check text on screen

Write test for case-insensitive text

Test title for image

Write multiple expect in the same test function

Test Input Box

Make input box in App Comp.

Write test case function

The test input box is present or not

Test input box

* Name
* Placeholder
* Id
* Value
* Type

In this error for Value Attribute Add -readOnly

Warning: You provided a `value` prop to a form field without an `onChange` handler. This will render a read-only field. If the field should be mutable use `defaultValue`. Otherwise, set either `onChange` or `readOnly`.

at input

at div

at App

Test Case Run Options

How to run specific test files.

What is watch Mode?

How to run the failed test case

How to run call test cases

How to quit watch mode

How to filter test files for run

How to filter test case

PS C:\Users\user\Desktop\testing> npm run test sum run for specific test file

Sum – name of file name

Watch Usage

› Press a to run all tests.

› Press f to run only failed tests.

› Press o to only run tests related to changed files.

› Press q to quit watch mode.

› Press p to filter by a filename regex pattern.

› Press t to filter by a test name regex pattern.

› Press Enter to trigger a test run.

Test grouping with describe

What is describe?

How to make test case group

Run test case with describe

Skip in describe

Only Describe

Nested describe

Describe function for identify----

(Use `node --trace-deprecation ...` to show where the warning was created)

PASS src/App.test.js

√ 123 Test First React App (81 ms)

√ Test Input box (39 ms)

UI test case group

√ Test case1 (7 ms)

√ Test case2 (5 ms)

√ Test case3 (5 ms)

UI2 test case group

√ Test case1 (8 ms)

√ Test case2 (4 ms)

√ Test case3 (4 ms)

Test Suites: 1 passed, 1 total

Tests: 8 passed, 8 total

Snapshots: 0 total

Time: 1.894 s, estimated 2 s

Ran all test suites matching /App/i.

Active Filters: filename /App/

› Press c to clear filters.

Watch Usage

› Press a to run all tests.

› Press f to run only failed tests.

› Press o to only run tests related to changed files.

› Press q to quit watch mode.

› Press p to filter by a filename regex pattern.

› Press t to filter by a test name regex pattern.

› Press Enter to trigger a test run.

If we want to run only one group

describe.only

Test OnChange Event with input Box

Make input box in the component

Define state and use with on change event

Import component in test file

Write code for test case

Run test case

Test Click Event with Button

Make Button and state in the component

Update state with button click event

Import component in test file

Write code for test click event

Run test case

File and folder naming convention

What file name we can use for test case file ?

Folder name for testing files

Run test case with naming convention

File\_name.test.js

File\_name.spec.js

File\_name.spec.jsx

\_\_test\_\_ (if we use this as folder)

Hooks

Before and after hooks

Use of before and after hooks  
before All and before Each

After All and after Each

Ex:-

Clean database- db

Snapshot Testing

What is snapshot testing?

Example?

When this useful

How to update snapshots

U- is for update changes

Important points for testing

What we should test?

* Testing compo. Rendering
* Ui elements that we write
* Functions which we write
* API Testing
* Event testing
* Props and states
* UI condition testing | UI state testing

What things we should not test?

Extended Ui library code

No needs to test default function of js and react

Sometime we should mock function rather than testing it in details.

Important points?

Do not write snapshots in starting of the project

Run test case after completing your functionality

Make a standard for code coverage

Class component method testing

Why Iam talking about class component?

Make class component

Install React test Renderer

Test class component method

> npm install react-test-renderer --save-dev

Functional component method

Discuss possible case for method testing

Define the button click event and method

Test method with event

Test method without event

RTL Query

What is RTL Query?

Why need RTL Query?

Steps in testing UI.

How RTL Query finds elements?

Type of RTL Queries.

Steps in testing UI

Render Component

Find Element and Action

Assertions

How RTL find elements

By Element Type

By Element name

By element id

By test id

Etc

Type of RTL Queries

Find Single Element

getBy

QueryBy

findBy

Find Multiple elements

getAllBy

queryAllBy

findAllBy

getByRoleQuery

GetByRoleQuery

What is the Role in getByRole?

What is semantic elements

* Button, heading tags and table are semantic element
* Div and span are not semantic elements

Test textbox with getByRole

* Text box present or not
* Text box value
* Text box disabled or not

Test button with getByRole

RTL Query: getByLabelText

* Use of getByLabelText
* Test input field with getByLabelText
* Test checkbox with getByLabelText

Multiple elements with Role Custom Role

* Multiple elements with the same role issue
* Multiple buttons with role
* Multiple input box with role
* Custom role (we use for non semantic tag)

Get All By Role

Handle Multiple Elements

Multiple buttons with same role

Multiple select options box with same role

When we call all the buttons it give error

received value must be an HTML Element or an SVGElement.

Received has type: array

Received has value: [<button>click me1</button>, <button>click me2</button>]

Way to resolve

1st method

**expect**(btns[0]).**toBeInTheDocument**();

2nd  method

for (let i=0; i<btns.length-1; i++){

**expect**(btns[i]).**toBeInTheDocument**();

}