**Lesson 10 Demo 02**

**react-component-login-testing**

**Objective:** To demonstrate the login component testing which contains emailed and password if successfully login get success message else failure message.

**Tools required:** Node JS and React JS

**Prerequisites:** HTML, CSS, JavaScript ES5/ES6, Basic React Concept

**Note** : All react js project already created with version 18.x with Sample App.js file

**Steps to be followed:**

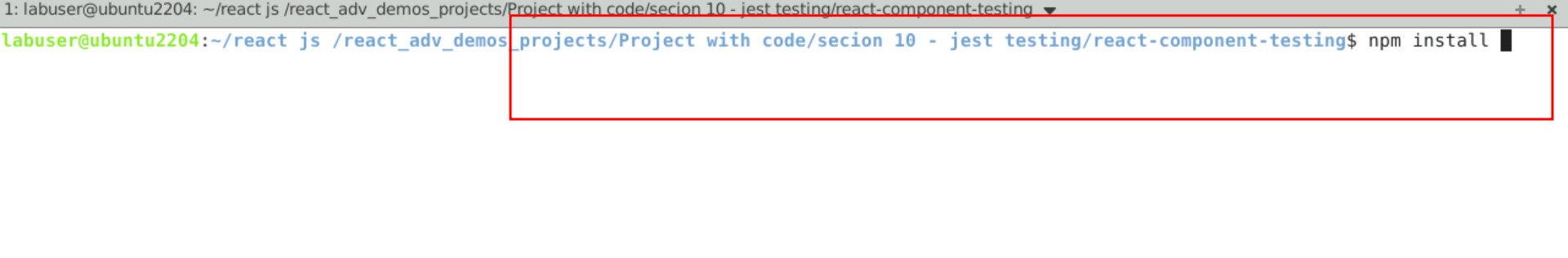
1. Set up for react jest js project
2. verify testing dependencies in package.json file
3. Create the login component
4. Create the login testing file responsible to the testing using jest
5. Now we test the application.

**Step 1: Set up for next js project**

1. Open the terminal window inside the **react-component-testing** folder



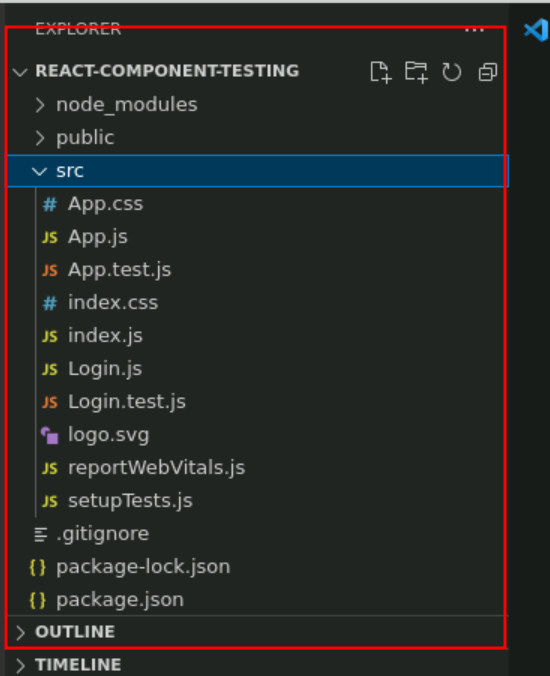
1.2 run npm install to download the dependencies.



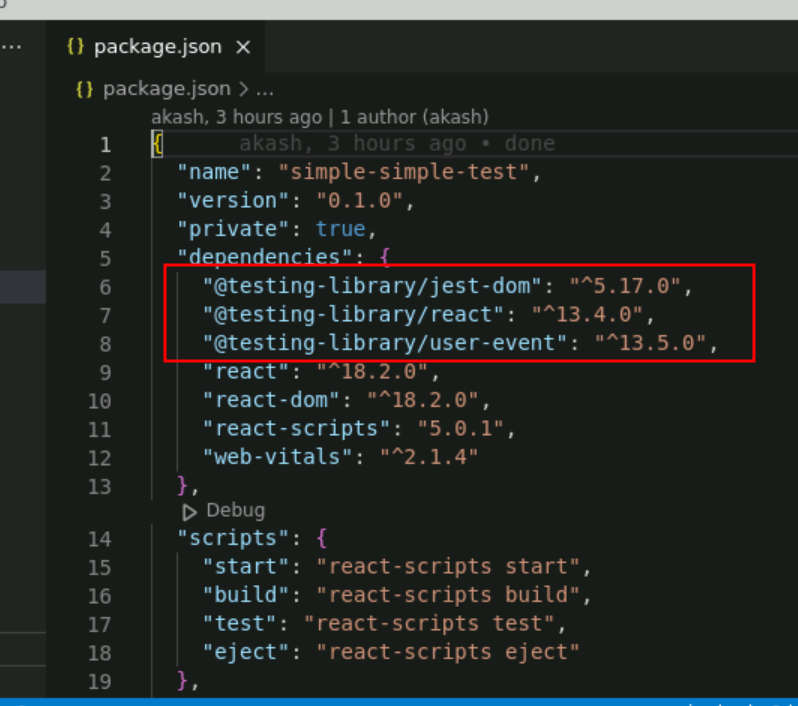
* 1. Open the project in VS code

Note: short cut to open write **code .**

* 1. view the project structure



**Step 2 : verify testing dependencies in package.json file**

****

**Step 3: Create the login component**

3.1 create Login.js file which is responsible to do the hardcoding login

Login.js

import { useState } from "react";

function Login() {

let[email,setEmailId]=useState("");

let[password,setPassword]=useState("");

let [msg,setMessage]=useState("");

let checkUser= (event)=> {

if(email=="admin@gmail.com" && password=="admin@123"){

setMessage("successfully login");

}else {

setMessage("failure try once again");

}

}

return(

<div>

<h3>Login Page</h3>

<form onSubmit={checkUser}>

<label>EmailId</label>

<input type="email" name="emailid" onChange={(event)=>setEmailId(event.target.value)} placeholder="enter emailid"/><br/>

<label>Password</label>

<input type="password" name="password" onChange={(event)=>setPassword(event.target.value)} placeholder="enter password"/><br/>

<input type="submit" value="SignIn" data-testid="submit-button"/>

<input type="reset" value="reset" data-testid="reset-button"/>

</form>

<br/>

<span data-testid="result">{msg}</span>

</div>

)

}

export default Login;

**Step 4 : Create the login testing file responsible to the testing using jest**

This file testing with valid emailed and password as well as invalid emailed and password.

**Login.test.js**

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

import Login from './Login';

describe("Login Testing ",()=> {

test('React Login test with success', async() => {

render(<Login />);

// succcessfully login testing code

const email = screen.getByPlaceholderText('enter emailid');

fireEvent.change(email,{target:{value:"admin@gmail.com"}});

const password = screen.getByPlaceholderText('enter password');

fireEvent.change(password,{target:{value:"admin@123"}});

const submitButton = screen.getByTestId('submit-button');

fireEvent.submit(submitButton);

const result1 = screen.getByTestId("result");

console.log(result1.textContent);

expect(result1.textContent).toEqual("successfully login");

});

test('React Login test with failure', async() => {

render(<Login />);

// failure login testing code

const email = screen.getByPlaceholderText('enter emailid');

fireEvent.change(email,{target:{value:"admin@gmail.com"}});

const password = screen.getByPlaceholderText('enter password');

fireEvent.change(password,{target:{value:"admin@1234"}});

const submitButton = screen.getByTestId('submit-button');

fireEvent.submit(submitButton);

const result1 = screen.getByTestId("result");

console.log(result1.textContent);

expect(result1.textContent).toEqual("failure try once again");

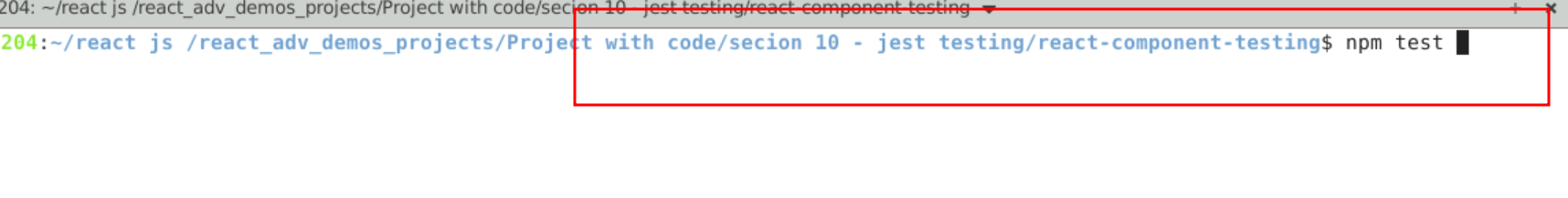
});

})

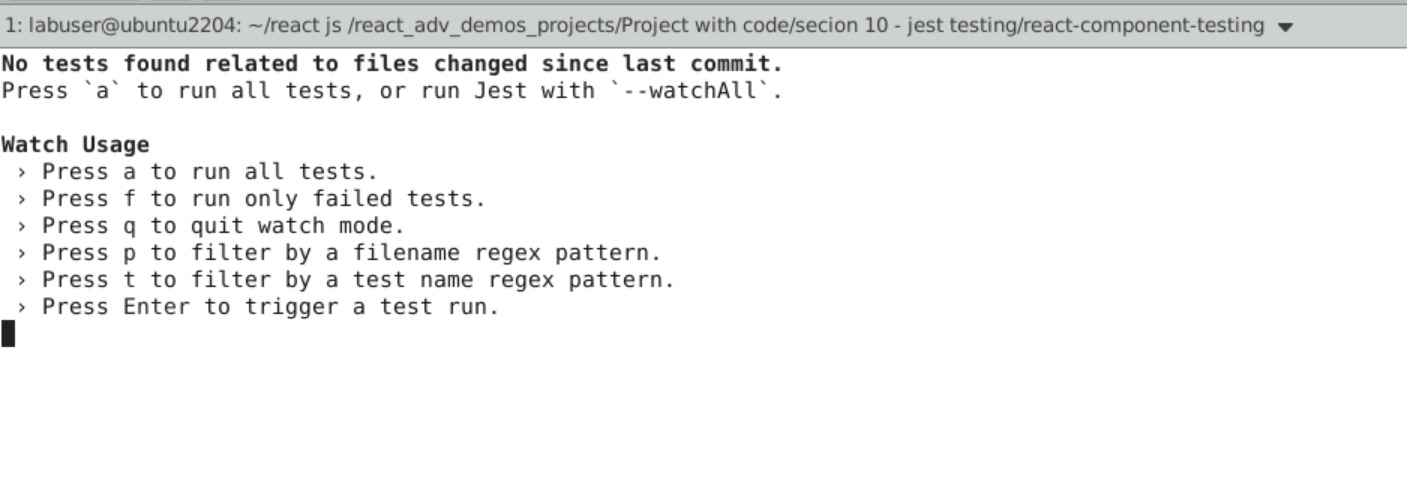
Step 5 : Now we test the application.

5.1 open the terminal inside the project and run the application we need to use the command as

**npm test**

****

5.2 You can view the output on console.



Press a button to run all test

