JEST Labs

# Task 1

Locate the *App.test.js* file in your project, this was the test that we just executed. Upon inspection this test is very simple, it is a Smoke Test and merely checks that *App.js* renders something without failing.

# Task 2

We will do one more thing with this file, currently it is in the *src* folder amidst our other React Components, which is going to quickly fill up. We want to create a new folder here called \_\_*tests*\_\_ this is conventional and is where the tool we are using to run our tests will first look for our tests. It also helps keep or Project structure neat and organised.

# Task 3

Using Enzyme write a simple test that shallow renders your *App* component

# Task 4

Using Enzyme create a content test that shallow renders one of your components and then checks for a specific value in that component.

# Task 5

Create a snapshot test for a component of your choosing. Run this this test at least twice to ensure it is accurate.

# Task 6

Modify the component above, run the Snapshot test again to ensure that it fails. Now remove the original snapshot so that the test now passes.

# Task 7

Create a component that takes in *props*, and that use these *props* when it renders. shallow render this component and then ensure that the *props* are used correctly by creating a snapshot.

# Task 8

Create a component that has a button, this button should change some *state* of the component, this *state* should in turn be used by the component to render some part of it. Create a snapshot test that shallow renders the component, takes a snapshot of the state, simulates clicking the button, and finally takes another snapshot to ensure the functionality.

# Task 9

Change the functionality of the button in the Task above to ensure it works correctly.

# Task 10

Create a component that has a form element, this form, when submitted, should change the state of the component. This state should be used to render something of the page, this component can simply look like the user can type something into a box, click a button, and get that text to appear on the page.

Using *Mount*, test the functionality of this component.