ReactJS Unit Testing Lab Report: CohortDetails Component

# Objectives

- Explain the need for Unit Testing in React  
- Working with Jest and Enzyme in React  
- List the types of Router Components

# In this hands-on lab, you will learn how to:

- Install and configure Enzyme  
- Create unit tests using describe(), test()  
- Mount component and test them using matchers  
- Capture snapshots of the React components

# Prerequisites

- Node.js  
- NPM  
- Visual Studio Code

# Step-by-Step Instructions

## Step 1: Setup the Project

1. Download and unzip the project.  
2. Open the folder in Visual Studio Code.  
3. In terminal, run:  
 npm install

## Step 2: Install Enzyme and Adapter

Run the following command:  
npm install --save-dev enzyme enzyme-adapter-react-16

## Step 3: Configure Enzyme

Modify setupTests.js:

import { configure } from 'enzyme';  
import Adapter from 'enzyme-adapter-react-16';  
  
configure({ adapter: new Adapter() });

## Step 4: Create Unit Test File

Create CohortDetails.test.js and use the following code:

import React from 'react';  
import { shallow, mount } from 'enzyme';  
import CohortDetails from './CohortDetails';  
import { CohortData } from './Cohort';  
  
describe('Cohort Details Component', () => {  
  
 test('should create the component', () => {  
 const wrapper = shallow(<CohortDetails />);  
 expect(wrapper.exists()).toBe(true);  
 });  
  
 test('should initialize the props', () => {  
 const cohort = CohortData[0];  
 const wrapper = mount(<CohortDetails cohort={cohort} />);  
 expect(wrapper.props().cohort).toEqual(cohort);  
 });  
  
 test('should display cohort code in h3', () => {  
 const cohort = CohortData[0];  
 const wrapper = mount(<CohortDetails cohort={cohort} />);  
 const heading = wrapper.find('h3');  
 expect(heading.text()).toContain(cohort.code);  
 });  
  
 test('should always render same html', () => {  
 const cohort = CohortData[0];  
 const wrapper = shallow(<CohortDetails cohort={cohort} />);  
 expect(wrapper).toMatchSnapshot();  
 });  
  
});

## Step 5: Run the Tests

Execute the tests using:  
npm test

