

Mastering React Testing with Jest

INTRODUCTION TO TESTING REACT APPLICATIONS WITH JEST



Daniel Stern

Github – [danielstern](#) | [@danieljackstern](#)



Before Beginning



Know how to build apps with React

Understand stores, dispatchers, helpers and components

Comfortable working in a Node.js environment with **Gulp** and **Express**



Course Suggestion:

Build a Full Stack Application with React and Express

Daniel Stern, Pluralsight

“In Jest, there is truth.”

William Shakespeare, King Lear





What is Jest?

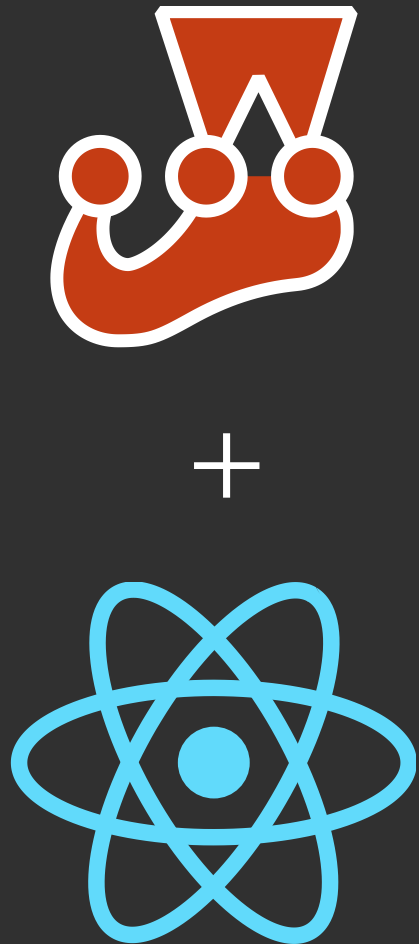
Jasmine-based testing framework

Test React applications and much more

Runs in Node (via CLI, Gulp, etc.)

Automated mocking of dependencies





What is Jest?

React's virtual DOM
facilitates easy testing

Jest is a great framework for testing
component-based applications

React and Jest **work together** to simplify
testing and reduce errors



Technologies Covered in This Course



Focus on **Jest** and **React**

Some **ES6** and **JSX**

Istanbul (built-in to Jest) for coverage to enhance tests

Node as the foundation



What You Will Need

Windows, Mac or Linux workstation with **Node 4.0+** and installed

Able to install **NPM** packages globally

Access to course files not required but highly recommended

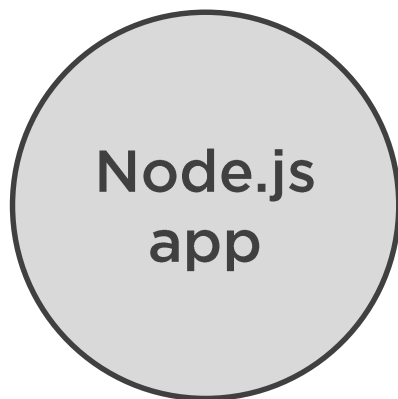
Git (recommended)

Text editor (**Sublime**, **IntelliJ**, **Brackets** or any high-quality IDE)



```
// __tests__/sum-test.js
jest.dontMock(' ../sum');

describe('sum', function() {
  it('adds 1 + 2 to equal 3', function() {
    var sum = require(' ../sum');
    expect(sum(1, 2)).toBe(3);
  });
});
```



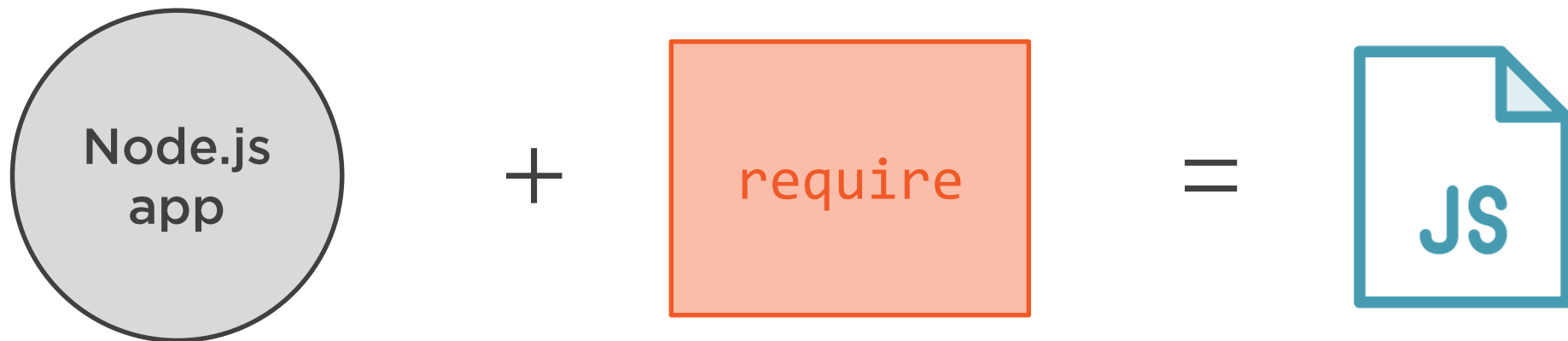
+



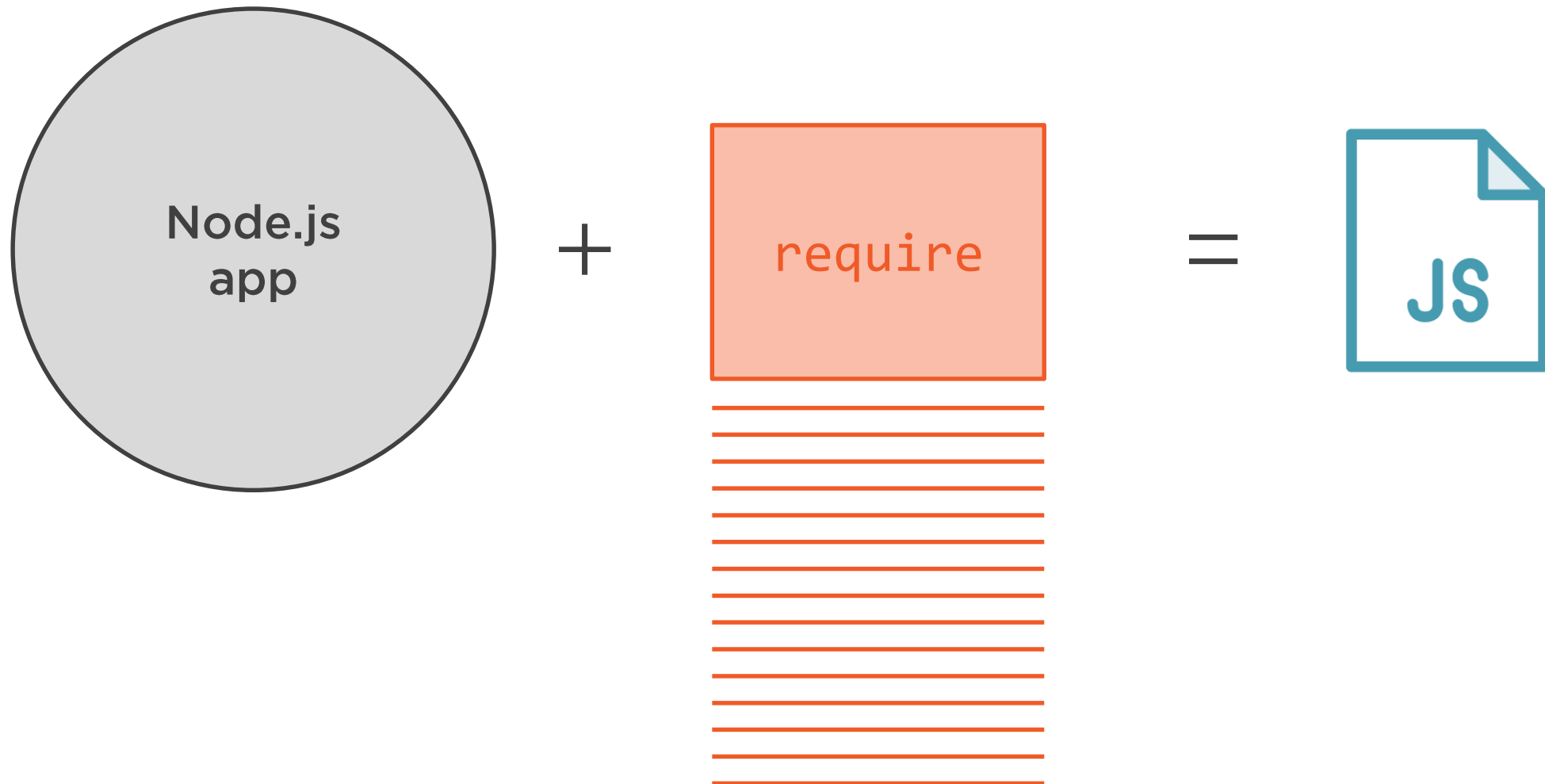
=



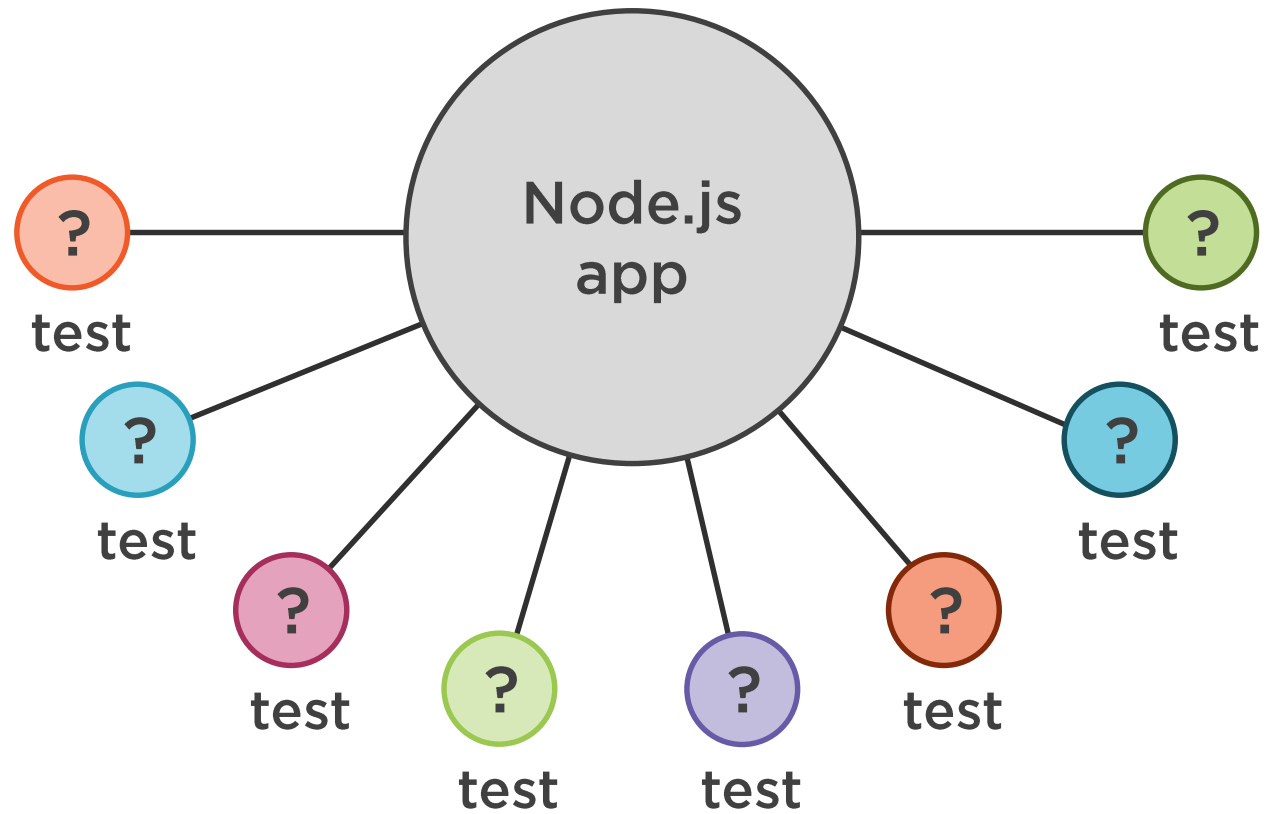
Understanding Jest — Automatic Mocking



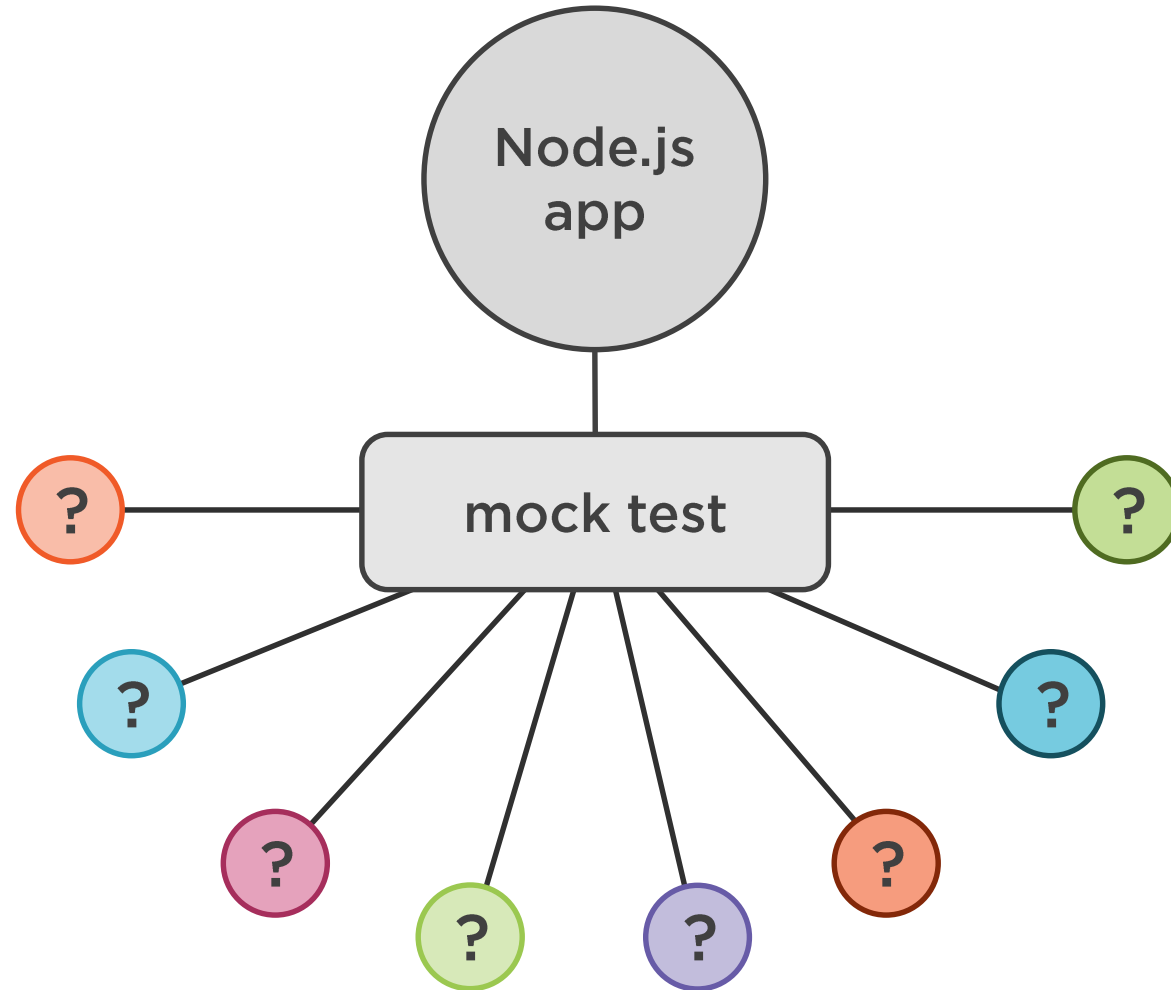
Understanding Jest — Automatic Mocking



Understanding Jest — Automatic Mocking



Understanding Jest — Automatic Mocking



What is Mocking?



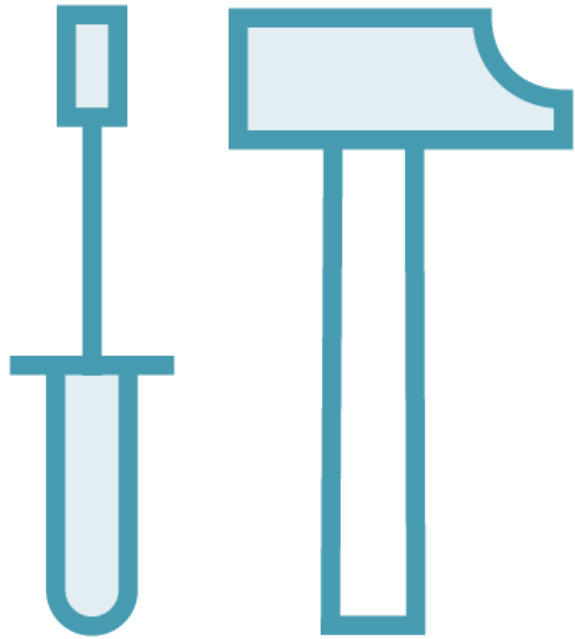
Mocking

The process of replacing a module with a generic and mostly empty object, called a “mock”

- The functionality of a mock is often defined inside the test

- Mocks have more features than stubs

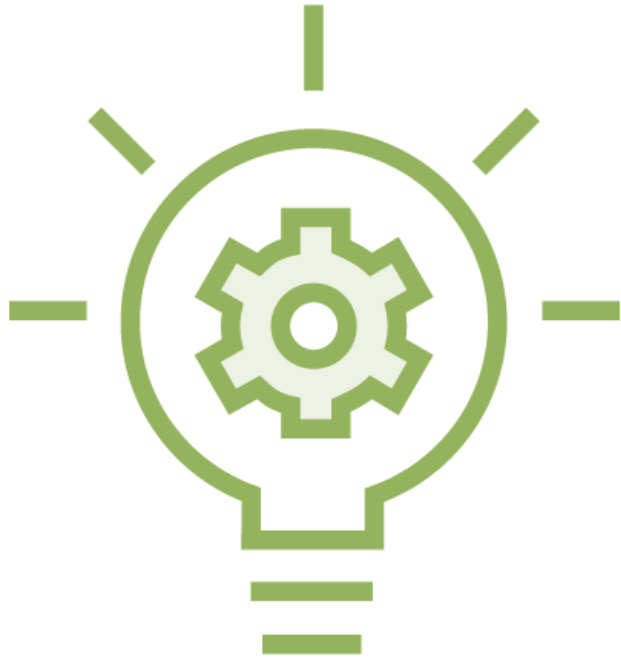




Most testing frameworks require that you manually specify mocks

Some do not support mocking at all





Contrarily, Jest automatically mocks all the modules in an application by default



Automatic Mocking

Typical

Test Framework

User defines what to mock

Jest

Automatic Mocking
Test Framework

User defines what not to mock



In short, all modules are
mocked by default with
automatic mocking



Advantages of Mocking

Isolates your code from its dependencies

Write tests with no knowledge of dependencies

No need to change tests after change to dependencies

Reduces false positives

Tests are less fragile

Easier for other developers to understand

