

TESTING

React Applications





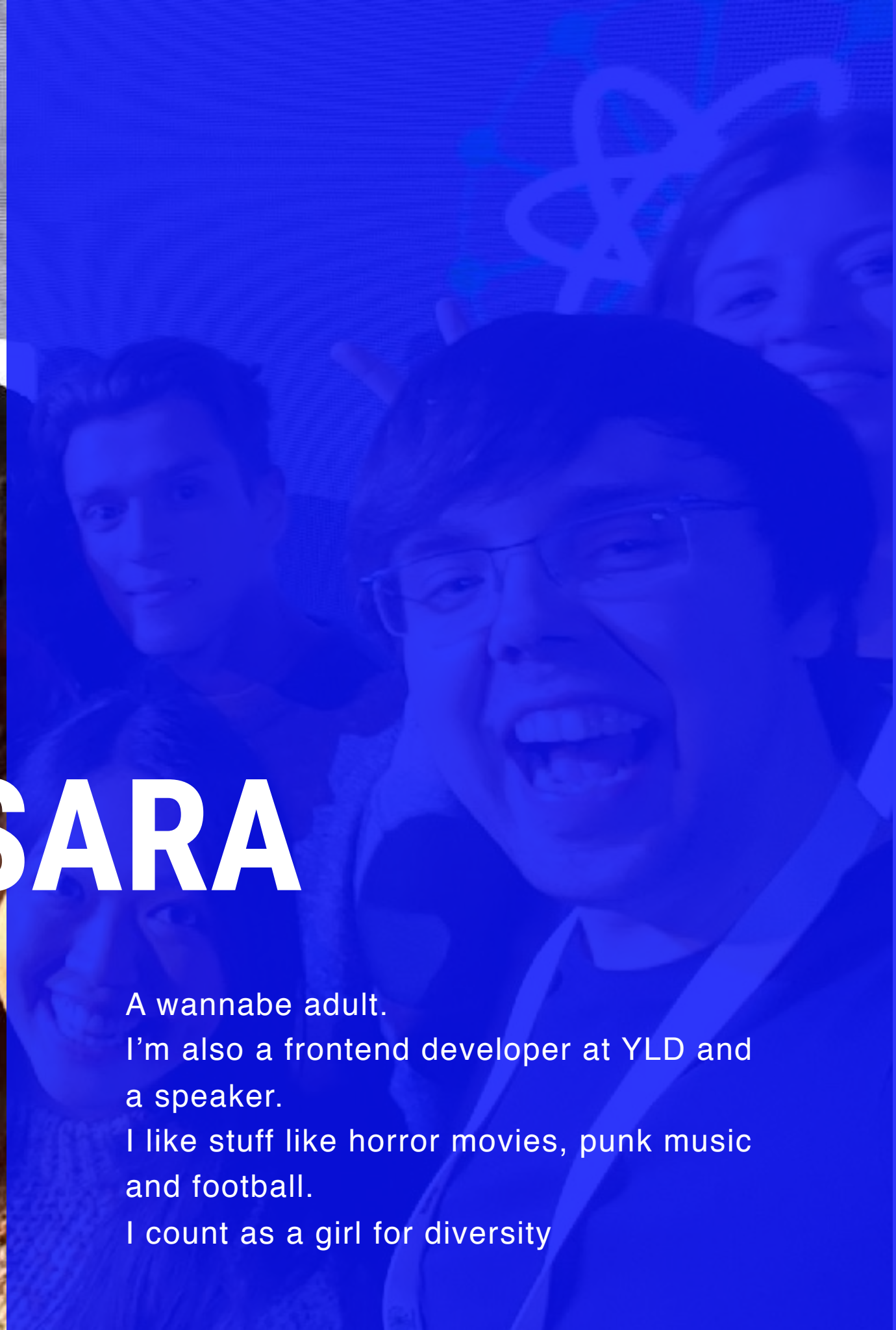
I'M SARA

A wannabe adult.

I'm also a frontend developer at YLD and a speaker.

I like stuff like horror movies, punk music and football.

I count as a girl for diversity



STUFF Y'ALL NEED

- **NODE**
- **GIT**
- **JS AND REACT KNOWLEDGE**
- **CODESANDBOX ACCOUNT**
- **PATIENCE FOR TESTING AND MY HORRIBLE JOKES**

CODESANDBOX = 

MATERIAL

**GITHUB.COM/SARAVIEIRA/TESTING-REACT-
WORKSHOP**

The background of the slide is a landscape photograph. The left half of the image is covered by a solid blue vertical bar. The right half shows a scenic view of a valley with mountains in the distance under a clear sky. The text 'WHY SHOULD I TEST?' is overlaid in large white letters, spanning across the blue bar and the landscape.

**WHY
SHOULD I
TEST?**

**BECAUSE NO ONE LIKES TO
BE WRONG**



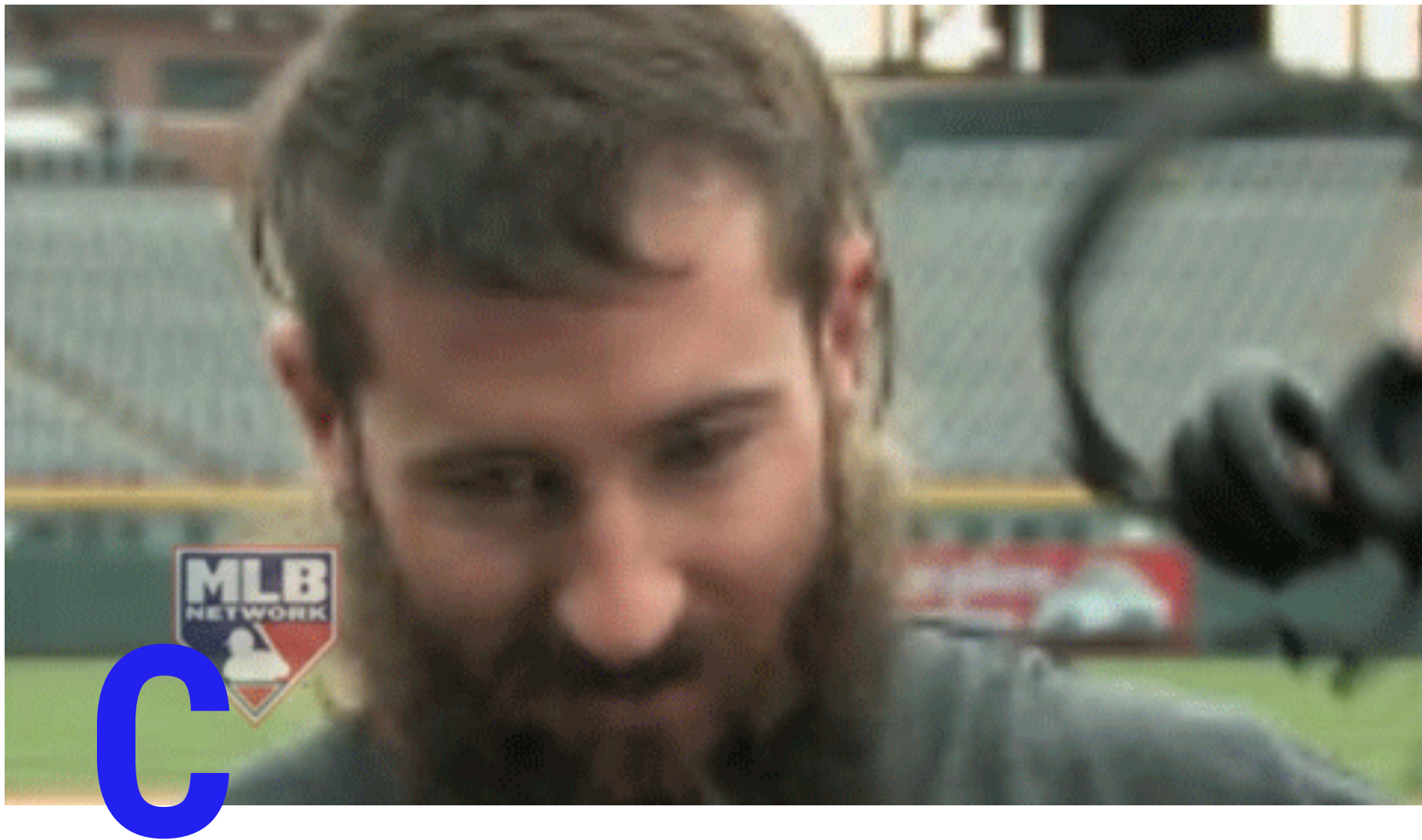
A

TESTS ARE DOCUMENTATION



B

TESTS ALLOW REFACTOR



The background of the slide is a landscape photograph of a mountain range under a clear sky. The image is split vertically by a solid blue bar on the left side. The text 'UNIT TEST ?' is positioned in the upper half, and 'WTF ?' is in the lower half, both in white, bold, sans-serif font.

UNIT TEST ?
WTF ?

“

**A UNIT TEST IS AN AUTOMATED PIECE OF
CODE THAT TESTS A UNIT OF WORK IN
ISOLATION.**

**A UNIT OF WORK IN OUR CASE IS MOSTLY A
FUNCTION.**

```
const sum = (a, b)  $\Rightarrow$  a + b
```

```
assert(sum(2, 2), 4);
```


CREATE A SIMPLE TEST RUNNER


01

[HTTPS://JSBIN.COM/
KIJISAHASA](https://jsbin.com/KIJISAHASA)

CONSOLE.ASSERT()

**WRITES AN ERROR MESSAGE TO THE
CONSOLE IF THE ASSERTION IS FALSE. IF
THE ASSERTION IS TRUE, NOTHING
HAPPENS.**

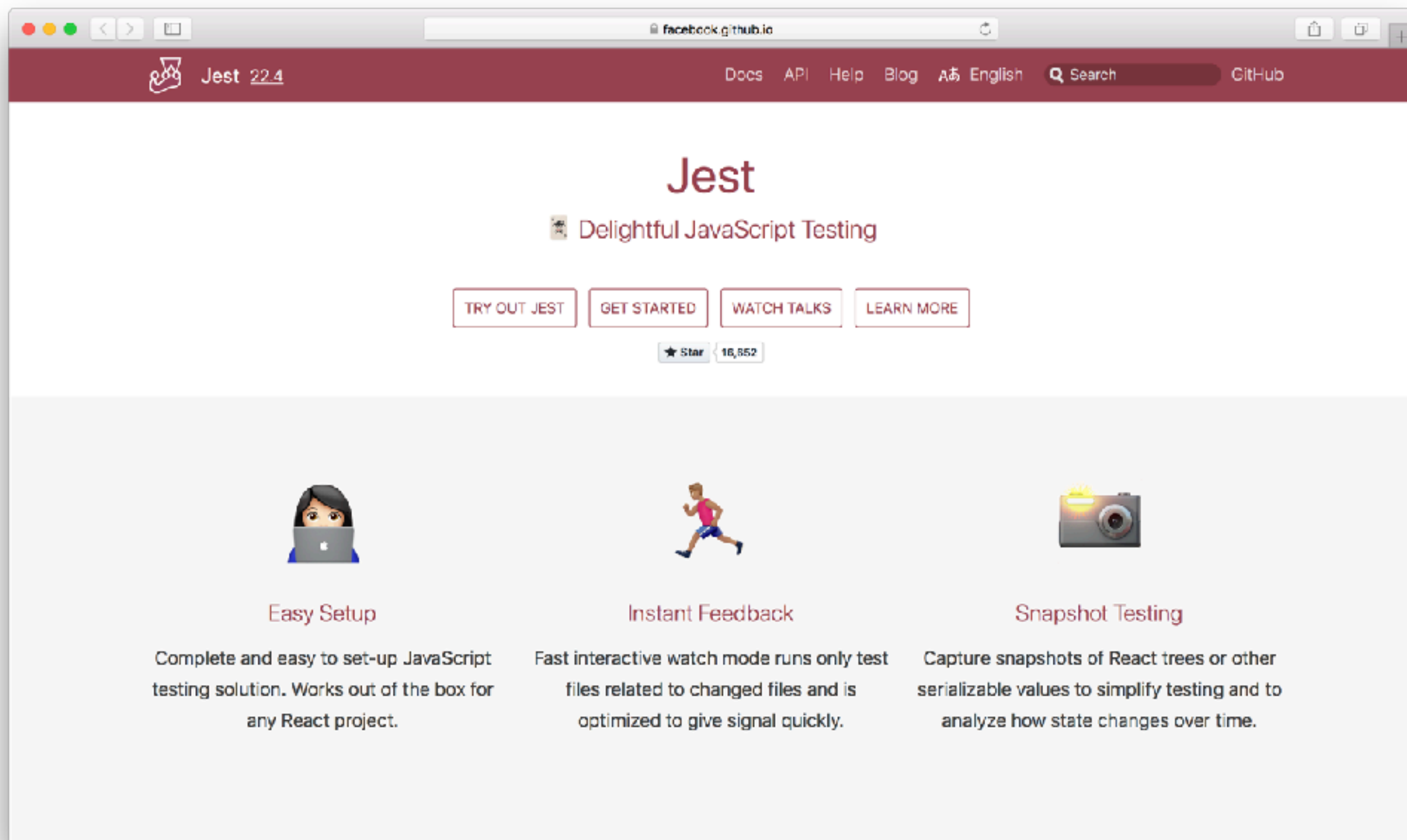
[HTTPS://DEVELOPER.MOZILLA.ORG/EN-US/DOCS/WEB/API/CONSOLE/ASSERT](https://developer.mozilla.org/en-US/docs/Web/API/Console/assert)



**[HTTPS://JSBIN.COM/
KIJISAHASA](https://jsbin.com/kijisahasa)**

INSTALL JEST

02



Jest 22.4

Docs

API

Help

Blog

⌂

English

🔍 Search

GitHub

Jest



Delightful JavaScript Testing

TRY OUT JEST

GET STARTED

WATCH TALKS

LEARN MORE

★ Star 16,852



Easy Setup

Complete and easy to set-up JavaScript testing solution. Works out of the box for any React project.



Instant Feedback

Fast interactive watch mode runs only test files related to changed files and is optimized to give signal quickly.



Snapshot Testing

Capture snapshots of React trees or other serializable values to simplify testing and to analyze how state changes over time.

```
yarn add jest -dev
```


LET'S TEST JS

03

[CODESANDBOX.IO/S/
QLVM92Z92J](https://codesandbox.io/s/QLVM92Z92J)

[HTTPS://CODESANDBOX.IO/S/QLVM92Z92J](https://codesandbox.io/s/QLVM92Z92J)

**LET'S TEST A SIMPLE SUM FUNCTION AND
SEE HOW JUST BY CREATING SOME TESTS
WE CAN CATCH A LOT OF BUGS**

EXERCISE

Please fork before making
changes

<https://codesandbox.io/s/z3w1k7nx94>

REACT AND UNIT TESTING

04

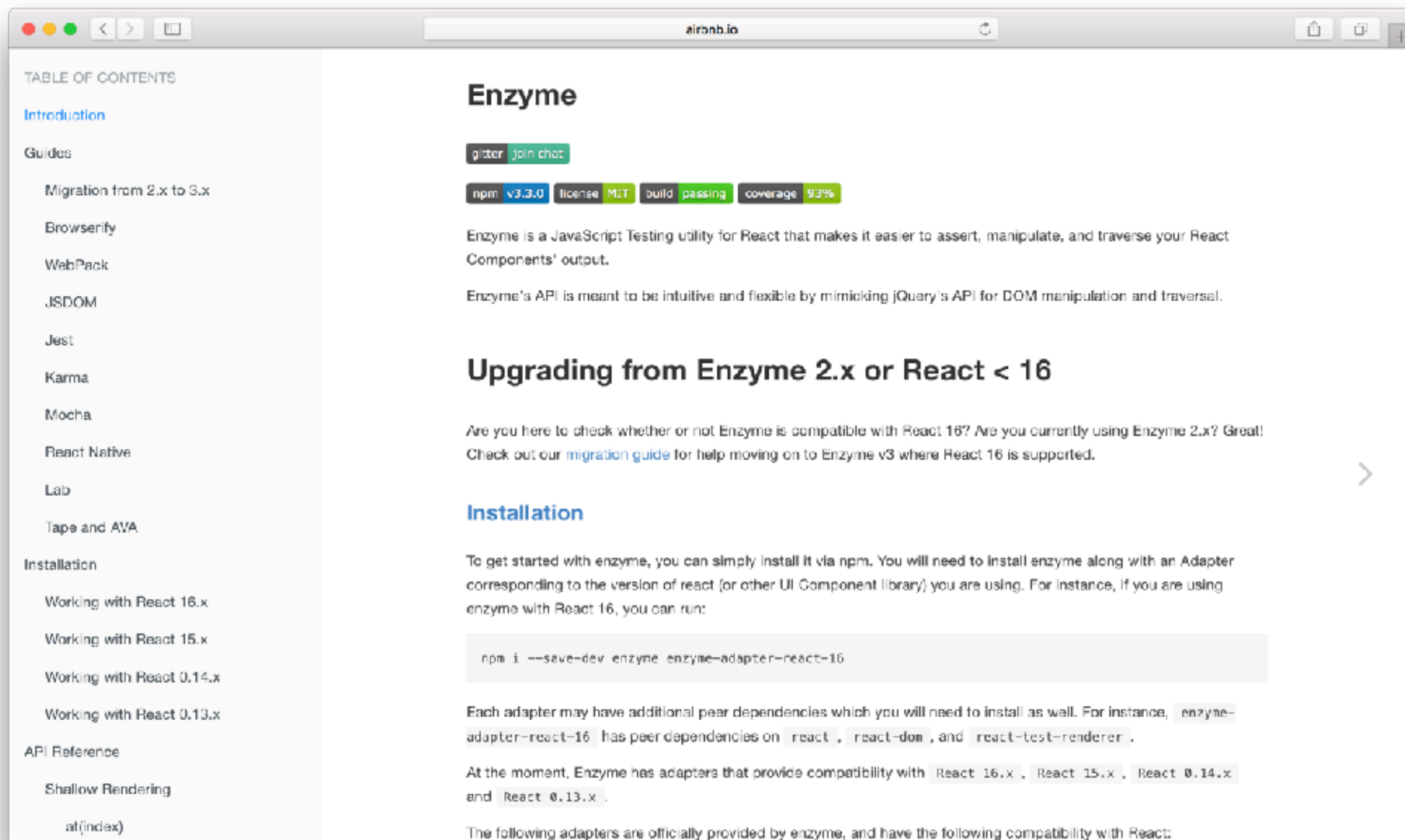
**IN ANY JAVASCRIPT MODULE THE IDEA OF
A UNIT TEST IS TO TEST A FUNCTION**

**IN REACT OUR FUNCTIONS ARE OUR
COMPONENTS SO THE MECHANICS ARE A
BIT DIFFERENT**

**WE NEED TO TEST FOR WHAT THE USER
SEES ON THE PAGE**

INSTALL ENZYME

05



**“ENZYME IS A JAVASCRIPT TESTING
UTILITY FOR REACT THAT MAKES IT EASIER
TO ASSERT, MANIPULATE, AND TRAVERSE
YOUR REACT COMPONENTS' OUTPUT.**

**ENZYME'S API IS MEANT TO BE INTUITIVE
AND FLEXIBLE BY MIMICKING JQUERY'S API
FOR DOM MANIPULATION AND
TRAVERSAL.**

```
yarn add enzyme enzyme-adapter-  
react-16 --dev
```

CONFIGURE ENZYME

06


```
"jest": {  
  "setupTestFrameworkScriptFile": "<rootDir>/  
src/setupTests.js"  
}
```

```
import Enzyme from "enzyme";  
import Adapter from "enzyme-adapter-  
react-16";  
  
// React 16 Enzyme adapter  
Enzyme.configure({ adapter: new Adapter() });
```

The background of the image is a landscape photograph. The left half of the image is covered by a solid blue vertical bar. The right half shows a landscape with mountains under a clear sky. The mountains in the foreground are dark and rocky, while the mountains in the background are lighter and have some snow or light-colored rock. The sky is a pale blue with a slight gradient. The text "LET'S TEST REACT" is overlaid on the image, with "LET'S" and "TEST" on the blue bar and "REACT" on the landscape. The text is in a bold, white, sans-serif font.

**LET'S
TEST
REACT**

LET'S TEST THE DOM

07

[CODESANDBOX.IO/S/
V3VNN457OL](https://codesandbox.io/s/v3vnn457ol)

EXERCISE

Please fork before making
changes

<https://codesandbox.io/s/7wk640x9zq>

**MOUNT ?
SHALLOW ?
RENDER ?
WTF ?**

08

SHALLOW

**SHALLOW RENDERS THE CURRENT NODE
AND RETURNS A SHALLOW WRAPPER
AROUND IT.**

**DOES NOT RENDER CHILDREN OF THE
COMPONENT. USEFUL FOR A SINGLE
COMPONENT**

MOUNT

A METHOD THAT RE-MOUNTS THE COMPONENT. THIS CAN BE USED TO SIMULATE A COMPONENT GOING THROUGH AN UNMOUNT/MOUNT LIFECYCLE.

USEFUL FOR PROPS AS WE NEED TO RE RENDER THE COMPONENT AND ALSO RENDERS THE WHOLE TREE

RENDER

FUNCTION IS USED TO RENDER REACT COMPONENTS TO STATIC HTML AND ANALYZE THE RESULTING HTML STRUCTURE.

VERY USEFUL FOR SNAPSHOT TESTING

LET'S TEST STATE CHANGES

09

[CODESANDBOX.IO/S/
34LM7YQ87M](https://codesandbox.io/s/34LM7YQ87M)

EXERCISE

Please fork before making
changes

<https://codesandbox.io/s/k9y4jkxo95>

SNAPSHOT TESTING

10

PLEASE CLONE

github.com/SaraVieira/testing-react-workshop

RUN

```
cd snapshots/start  
npm install  
npm run test --watch
```




**ADD
COVERAGE**

11

```
"jest": {  
  "snapshotSerializers": [  
    "enzyme-to-json/serializer"  
  ],  
  "coverageThreshold": {  
    "global": {  
      "branches": 80,  
      "functions": 80,  
      "lines": 80,  
      "statements": -10  
    }  
  }  
}
```

CLI FLAGS

11

— COVERAGE

Allows Jest to get coverage from
the your files

— CI

When this option is provided, Jest will assume it is running in a CI environment.

— CLEARCACHE

Clears all Jest Cache. Will delete
cacheDirectory if the option is
passed



QUESTIONS ?



THANK YOU