



Native Testing Library

 github.com/testing-library/native-testing-library

Simple and complete cheat sheet

render a component

```
import { render } from 'native-testing-library';

// See render result for more details...
const result = render(<TextInput />);
```

search native nodes

```
const { getByText } = render(<Text>hello</Text>);

// Check out jest-native on npm for handy assertions...
const element = getByText('hello');
```

fire an event

```
import { fireEvent } from 'native-testing-library';

fireEvent.press(element);
// Or you can fire events manually with...
fireEvent(element, new NativeEvent('press', {}));
```

pretty print native nodes

```
const { debug } = render(<Text>hello</Text>);

// Pretty print the native nodes of your render
debug();
```

wait for something

```
import { wait } from 'native-testing-library';

// Retry search every 50ms for 4500ms
wait(() => getByText('async result'));
```

search variants

(return value)

getBy	Element or Error
getAllBy	Element[] or Error
queryBy	Element or null
queryAllBy	Element[] or []
findBy	Promise<Element> or Error
findAllBy	Promise<Element[]> or Error

search types

LabelText	accessibilityLabel="go back"
PlaceholderText	placeholder="username"
Text	<Text>About</Text>
HintText	accessibilityHint="movie poster"
Title	<Button title="hello" />
DisplayValue	Current value of Picker or Switch
Role	accessibilityRole="button"
TestId	testID="username-input"

text matches

```
const { getByText } = render(<Text>Hello World</Text>);

getByText('Goodbye World'); // ✗
getByText(/hello world/); // ✗
getByText('ello Worl', { exact: false }); // ✓
getByText('Hello World'); // ✓
```

wait for appearance

```
test('movie title appears', async () => {
  // element is initially not present...
  expect(() => getByText('aladdin')).toThrow();

  // wait for appearance
  await wait(() => {
    expect(getByText('aladdin')).toBeTruthy();
  });

  // wait for appearance and return the result
  await waitForElement(() => getByText('aladdin'));
});
```

assert for absence

```
expect(queryByText('submit')).toBeNull();
```

the render result

(description)

container	View containing your UI
baseElement	Root View of the render
debug(element)	Pretty print native elements
unmount()	Unmount your component
rerender(ui)	Render again at the container
...queries	Queries for the baseElement

scope your queries

```
import { within } from 'native-testing-library';

within(loginForm).getByText('submit');
```