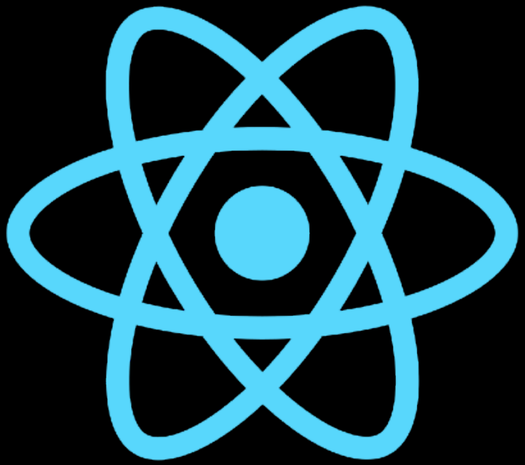
Snapshot

A Painless JavaScript Testing

Timo 王權

Icon made by Madebyoliver from www.flaticon.com

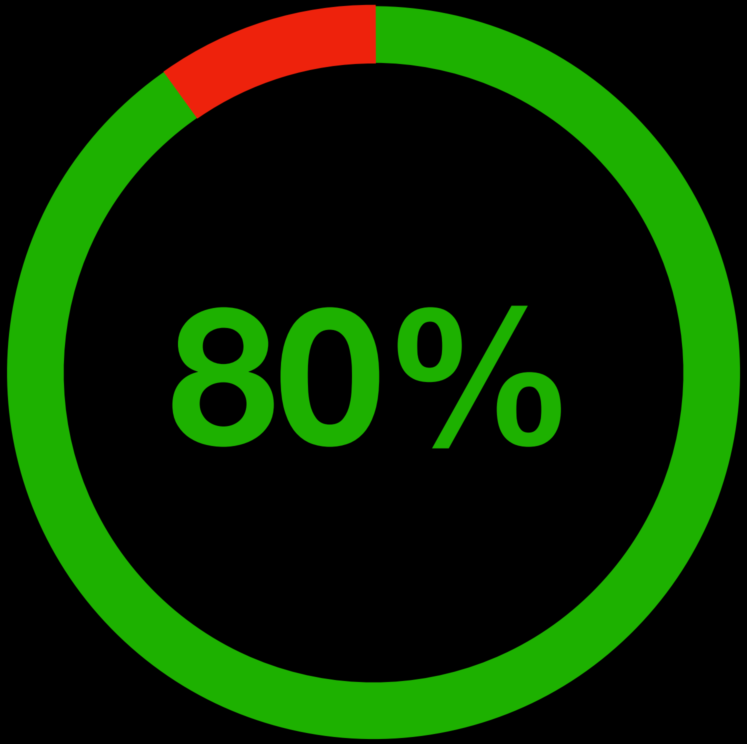
Prerequisites



```
// file: sum.js

export default (a, b) => a + b;
```

```
// file: sum.test.js

import sum from './sum';

it('should sum 2 numbers', () => {
    const result = sum(1, 2);

    expect(result).to.equal(3);
});
```

Why test?

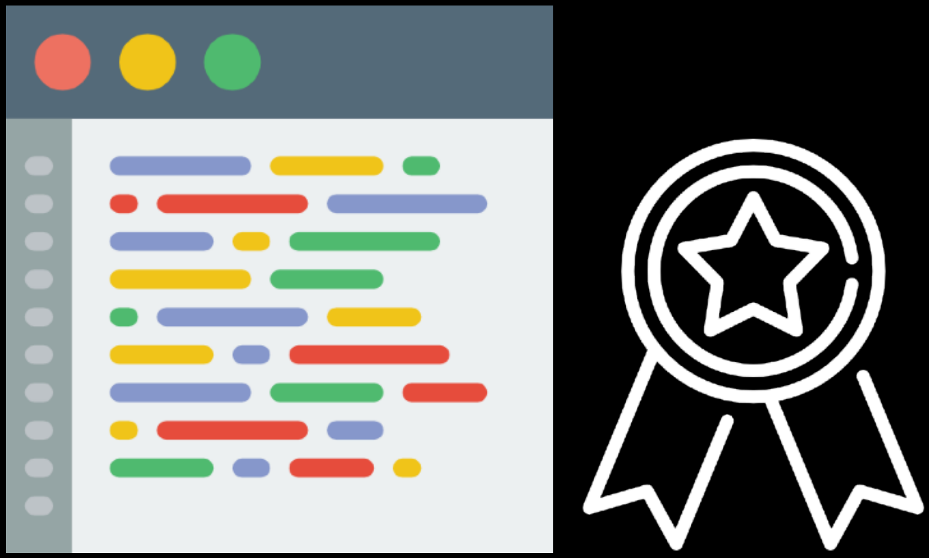


Code Coverage

Test, plz

Can
you add some tests
please?

y20



Code Quality

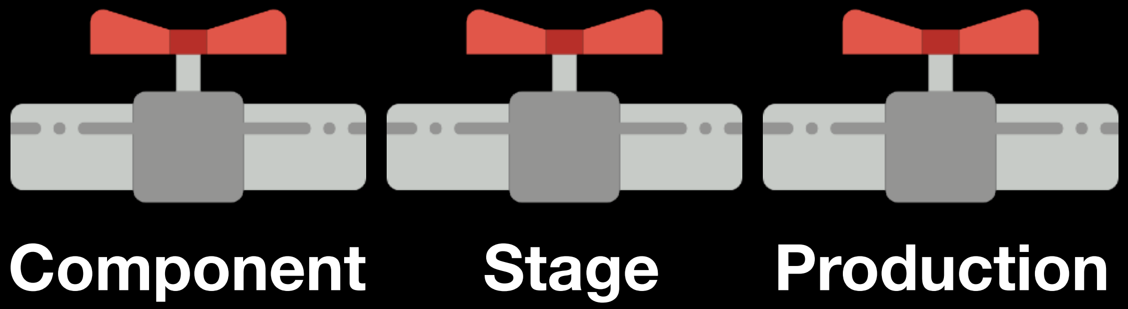
Icon made by Madebyoliver from www.flaticon.com

Icon made by Freepik from www.flaticon.com

Paranoia

Icon made by Freepik from www.flaticon.com

Protection

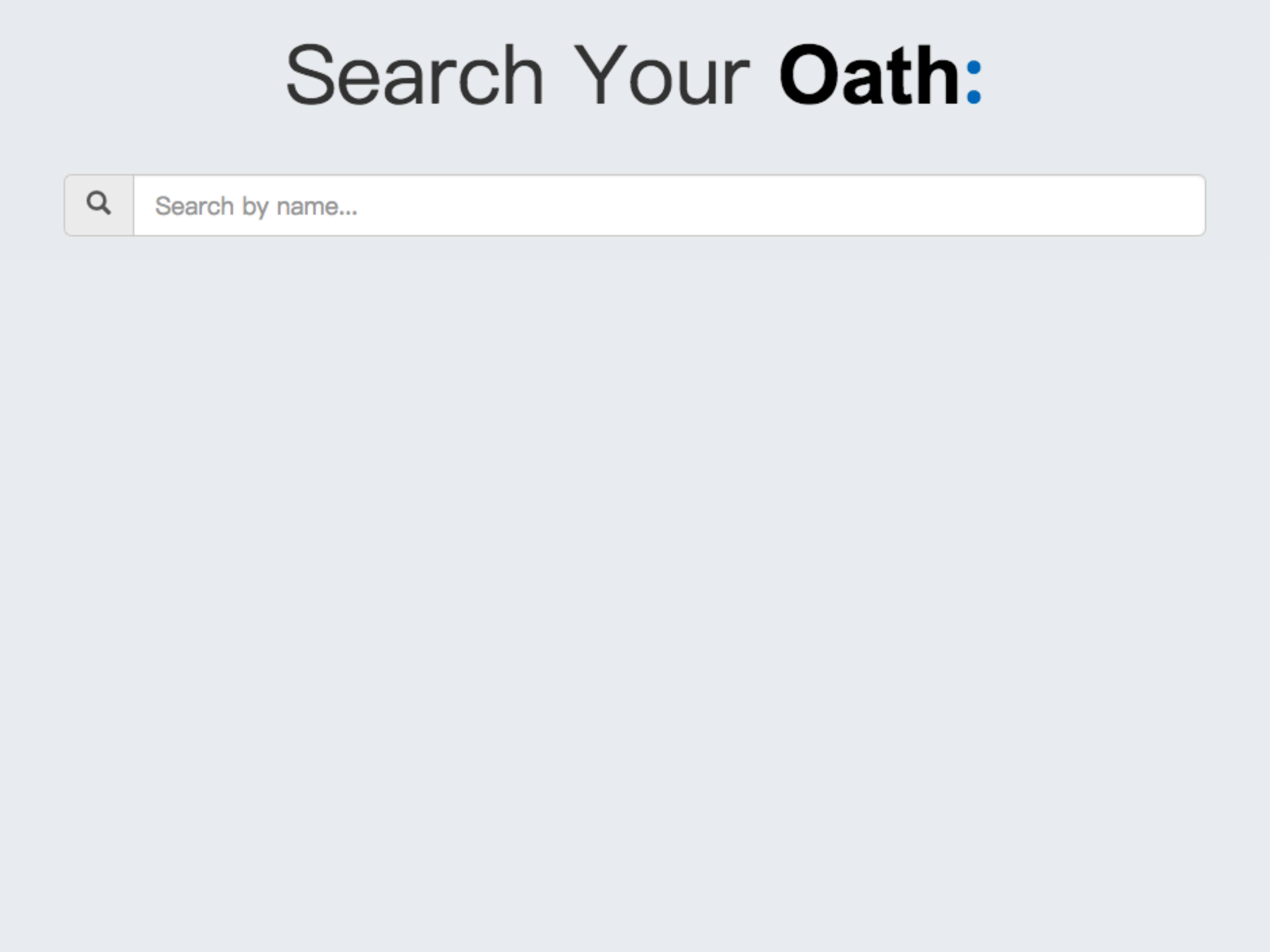


Continuous Delivery Pipeline

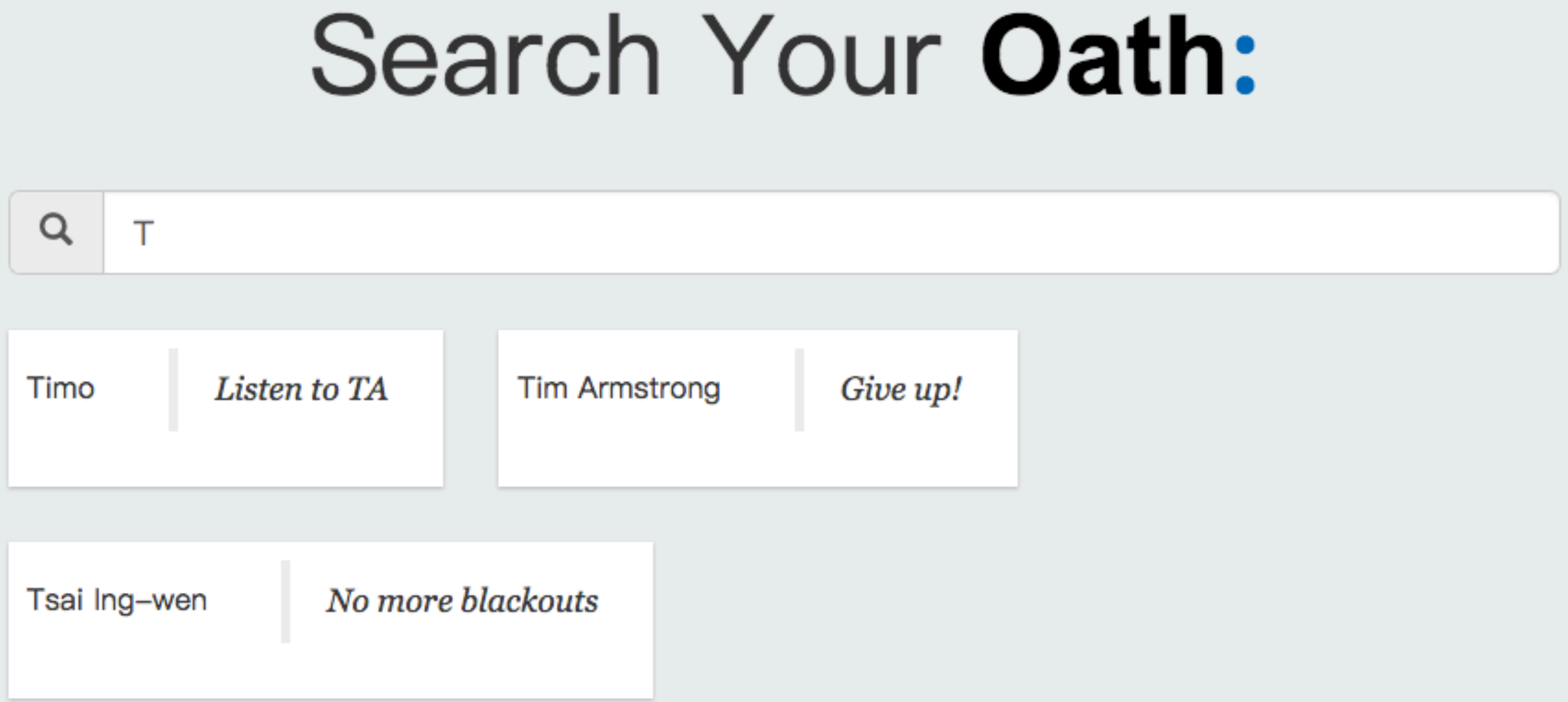
Icon made by ThoseIcons from www.flaticon.com

Icon made by Madebyoliver from www.flaticon.com

How we write tests today?



Search Your **Oath**:



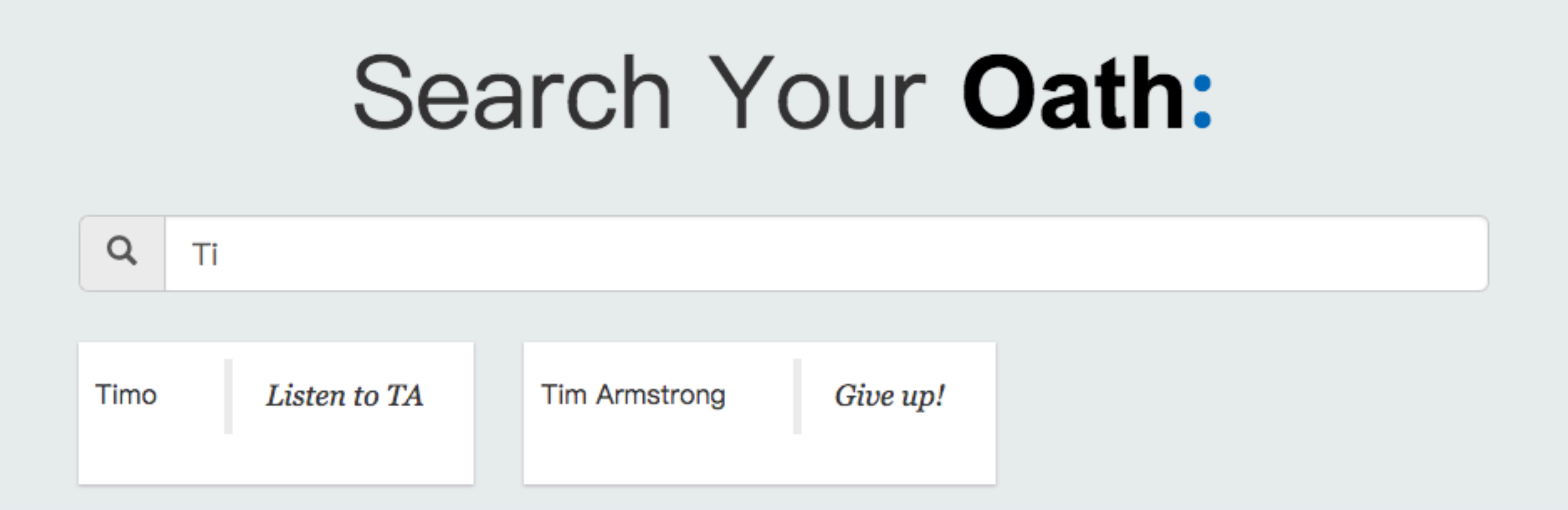
Search Your **Oath**:

T

Timo | *Listen to TA*

Tim Armstrong | *Give up!*

Tsai Ing–wen | *No more blackouts*



Search Your **Oath**:

Ti

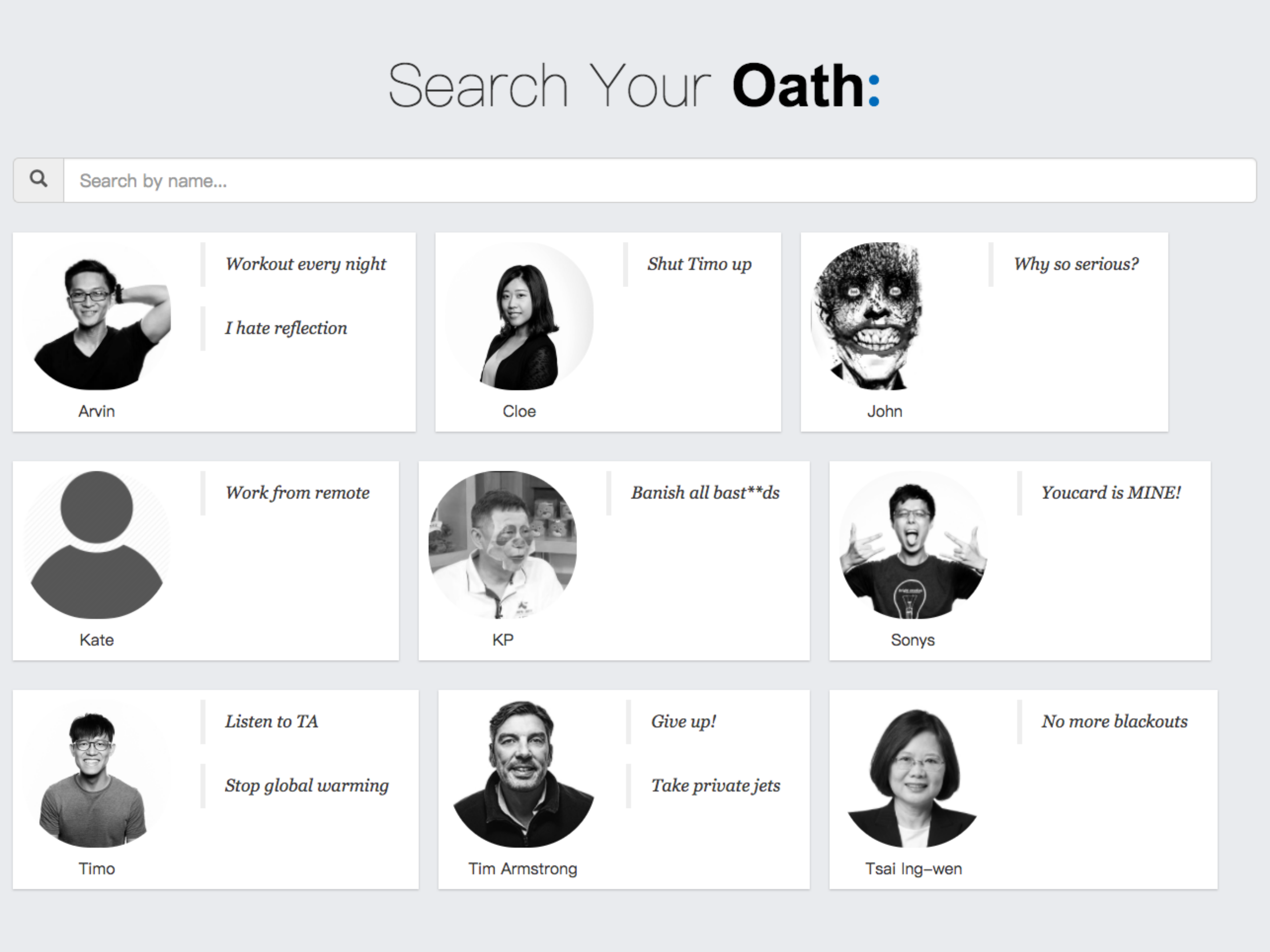
Timo | *Listen to TA*

Tim Armstrong | *Give up!*

```
/* data format:
    {
        person1: oath1,
        person2: oath2,
        ...
    }
*/
function searchOathData(query) {
    const data = someApiCall(query);
    ...
    return data;
}
```

```
const oathDb = {
    Arvin: 'Workout every night',
    Cloe: 'Shut Timo up',
    John: 'Why so serious?',
    Kate: 'Work from remote',
    Sonys: 'Youcard is MINE!',
    Timo: 'Listen to TA',
    'Tim Armstrong': 'Give up!',
    'Tsai Ing-wen': 'No more blackouts'
};



it('should return names starting with A’, () => {

    const oath = searchOathData(‘A');

    expect(oath).to.deep.equal({
        Arvin: 'Workout every night'
    });
});
```

```
const oathDb = {
    Arvin: 'Workout every night',
    Cloe: 'Shut Timo up',
    John: 'Why so serious?',
    Kate: 'Work from remote',
    Sonys: 'Youcard is MINE!',
    Timo: 'Listen to TA',
    'Tim Armstrong': 'Give up!',
    'Tsai Ing-wen': 'No more blackouts'
};


it('should return name starting with T', () => {

    const oath = searchOathData(’T');

    expect(oath).to.deep.equal({
        Timo: 'Listen to TA',
        'Tim Armstrong': 'Give up!',
        'Tsai Ing-wen': 'No more blackouts'
    });
});
```



Search Your **Oath:**

Arvin

Workout every night

I hate reflection

Cloe

Shut Timo up

John

Why so serious?

Kate

Work from remote

KP

*Banish all bast**ds*

Sonys

Youcard is MINE!

Timo

Listen to TA

Stop global warming

Tim Armstrong

Give up!

Take private jets

Tsai Ing-wen

No more blackouts

```
const oathDb = {
    Arvin: {
        image: 'arvin.png',
        oath: ['Workout every night', 'I hate reflection']
    },
    Cloe: {
        image: 'cloe.jpg',
        oath: ['Shut Timo up']
    },
    John: {
        image: 'john.png',
        oath: ['Why so serious?']
    },
    Kate: {
        oath: ['Work from remote']
    },
    ...
    Sonys: {
        image: 'sonys.jpg',
        oath: ['Youcard is MINE!']
    },
    Timo: {
        image: 'timo.jpg',
        oath: ['Listen to TA', ‘Stop global warming’]
    },
    'Tim Armstrong': {
        image: 'tim-armstrong.jpg',
        oath: ['Give up!', 'Take private jets']
    },
    'Tsai Ing-wen': {
        image: 'tsai-ing-wen.jpg',
        oath: ['No more blackouts']
    }
};
```

```
it('should return name starting with T', () => {

    const oath = searchOathData(’T');

    expect(oath).to.deep.equal({
        Timo: 'Listen to TA',
        'Tim Armstrong': 'Give up!',
        'Tsai Ing-wen': 'No more blackouts'
    });
});
```

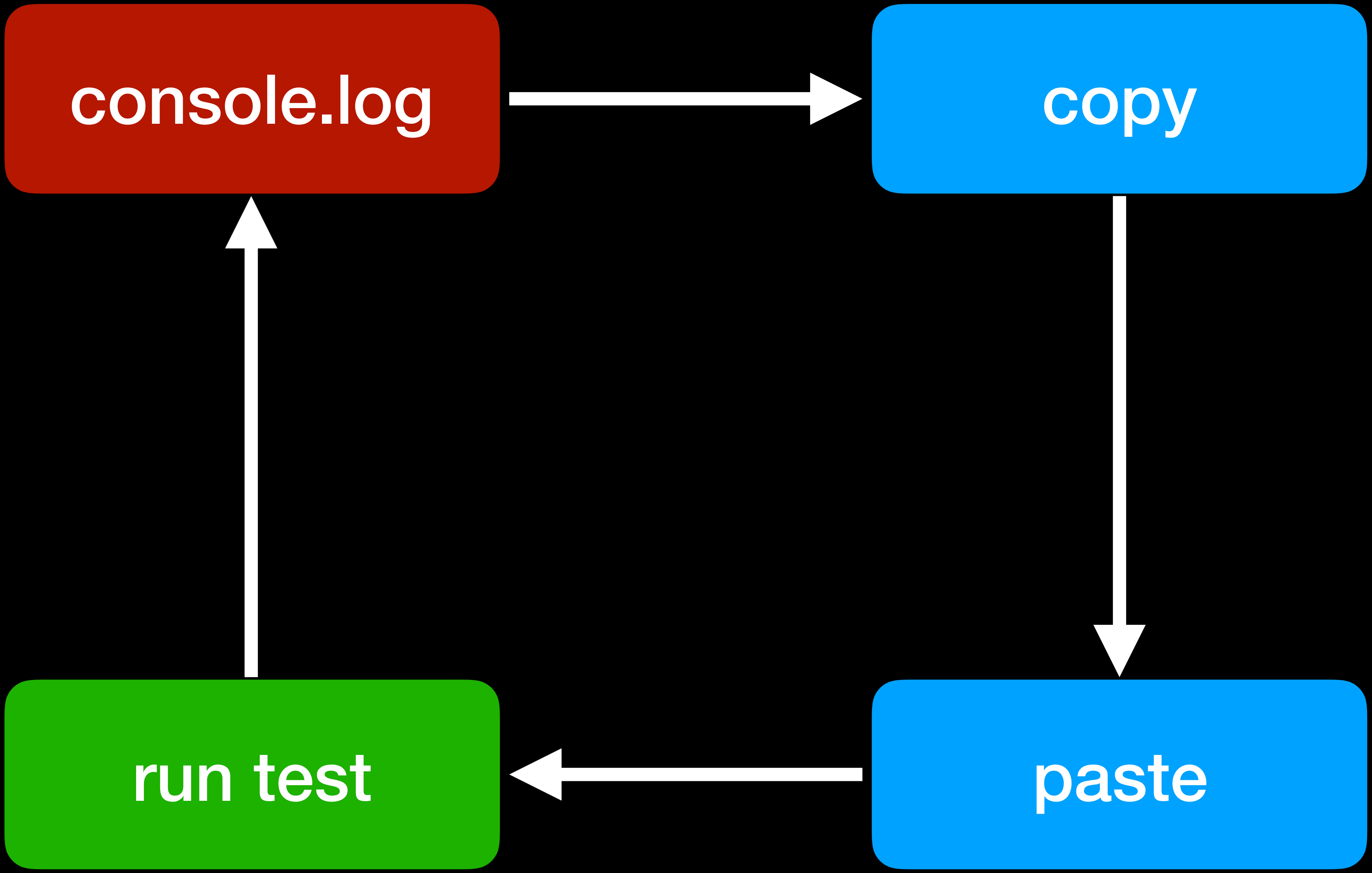
```
it('should return name starting with T', () => {

    const oath = searchOathData(’T');

    console.log(oath);
});
```

```
it('should return name starting with T', () => {

    const oath = searchOathData(’T');

    expect(oath).to.deep.equal({
        Timo: {
            image: 'timo.jpg',
            oath: ['Listen to TA', 'Must Be Crazy']
        },
        'Tim Armstrong': {
            image: 'tim-armstrong.jpg',
            oath: ['Give up!']
        },
        'Tsai Ing-wen': {
            image: ‘tsai-ing-wen.jpg',
            oath: ['No more blackouts']
        }
    });
});
```

```
it('should return name starting with T', () => {
    ...
});
```

```
it('should return name starting with Ti’, () => {
    ...
});
```

```
it('should return name starting with Tim’, () => {
    ...
});
```

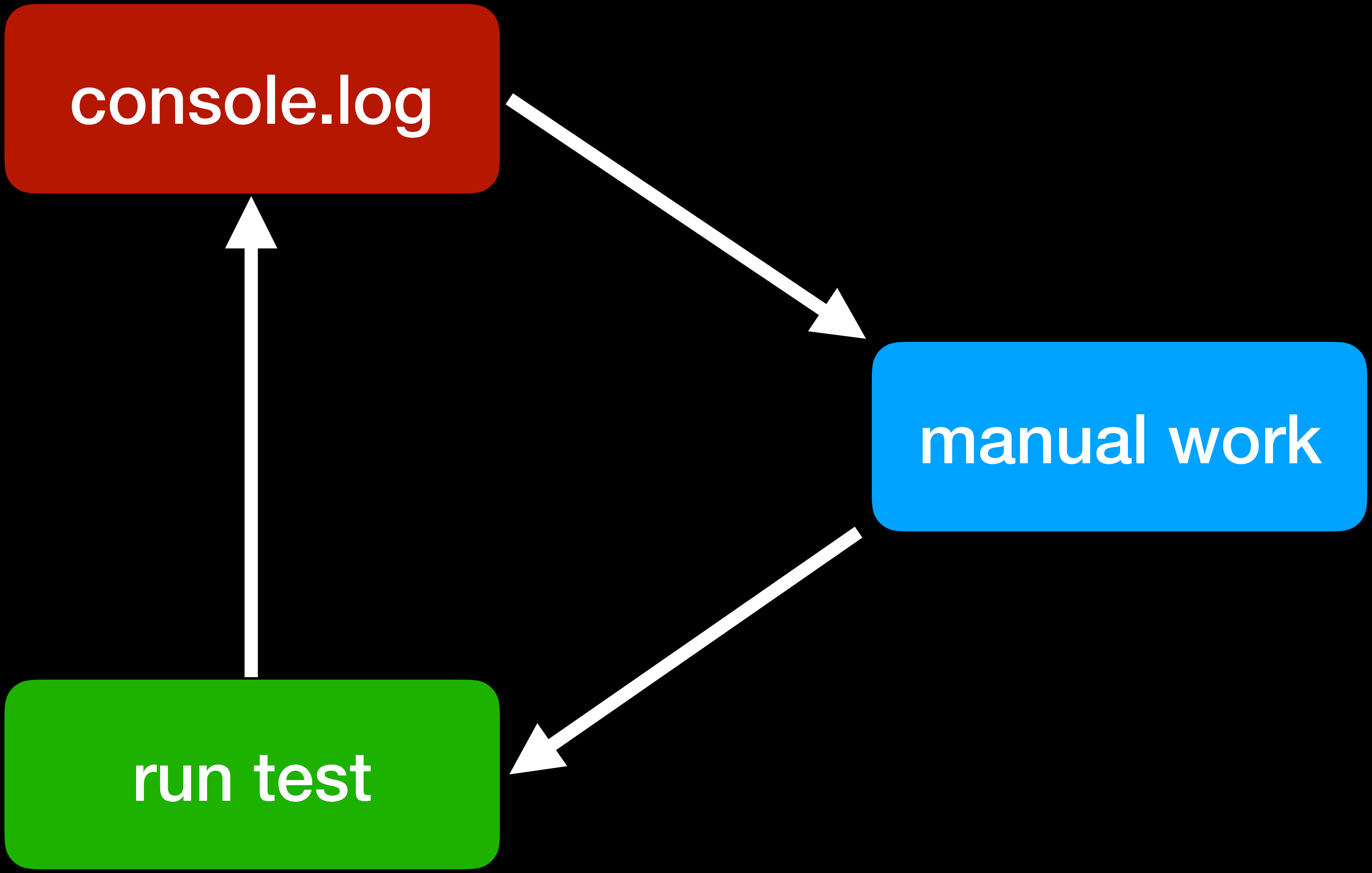


console.log

copy

paste

run test

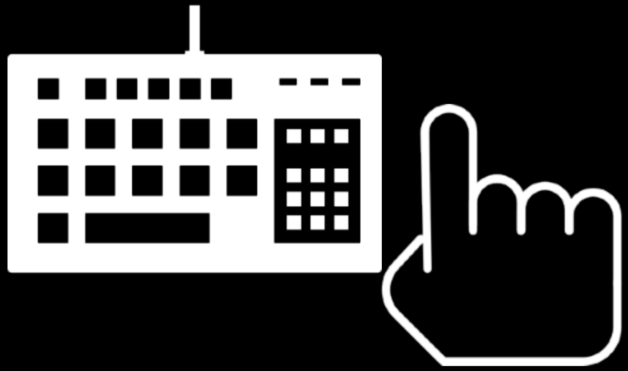


console.log

manual work

run test

Tedious



Manual

Extra code to maintain

Icon made by Freepik from www.flaticon.com

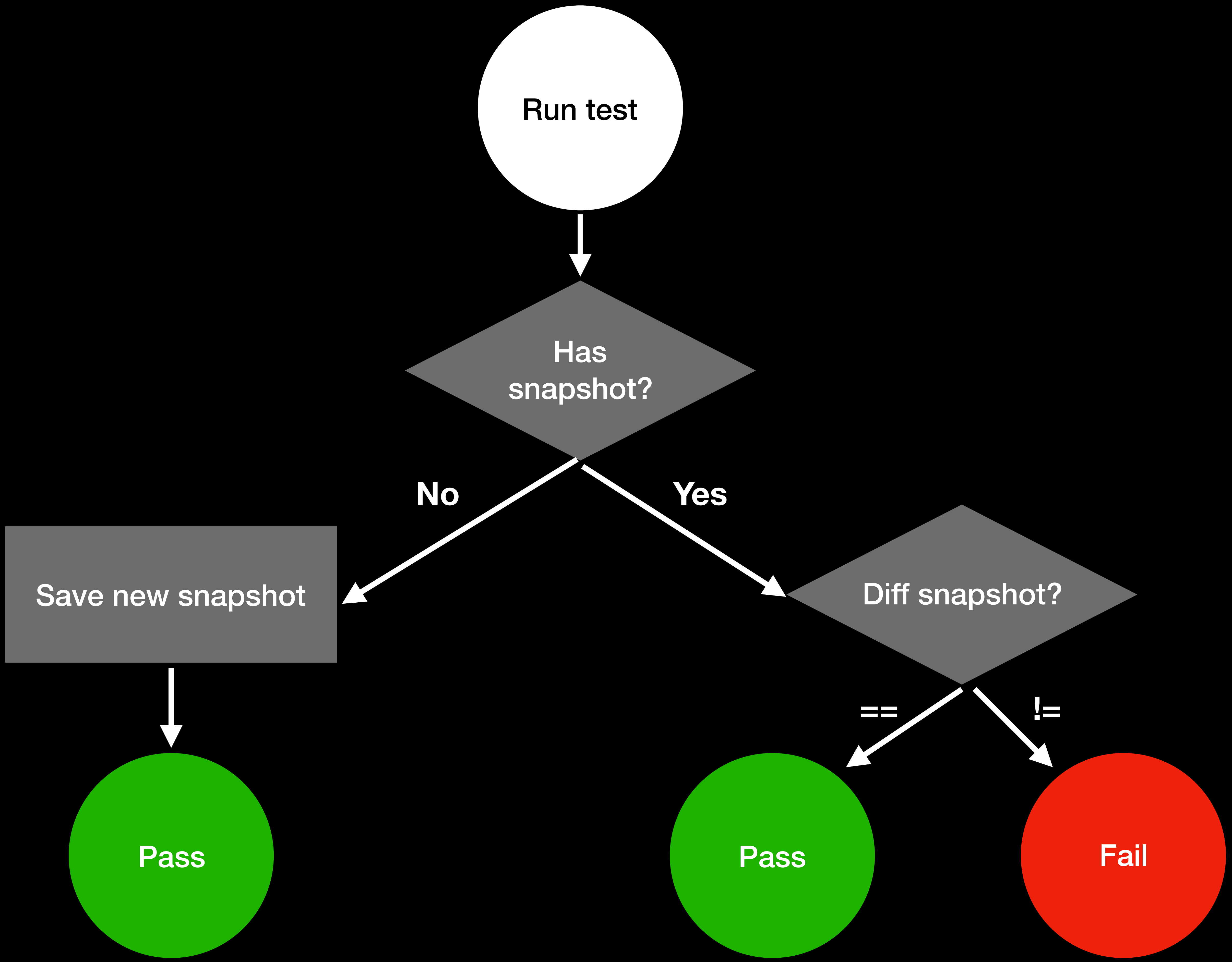
Icon made by Freepik from www.flaticon.com

Icon made by Madebyoliver from www.flaticon.com

Icon made by Iconnice from www.flaticon.com

Icon made by Freepik from www.flaticon.com

Snapshot


Run test
Has snapshot?
No
Yes
Save new snapshot
Diff snapshot?
==
!=
Pass
Pass
Fail

Snapshot Tools

```
npm i --save-dev jest enzyme enzyme-to-json
npm i --save-dev jest enzyme enzyme-to-json
```

Run

```
it('should ...', () => {
    expect(...).toBe(...);
});
```

Snap

```
it('should ...', () => {
    expect(...).toMatchSnapshot();
});
```

Icon made by Madebyoliver from www.flaticon.com

```
npm i --save-dev jest enzyme enzyme-to-json
```

Run

```
$ jest

$ jest src/sum.test.js
```

```
npm i --save-dev jest enzyme enzyme-to-json
```

```
// file: sum.test.js
it('should sum 2 numbers snapshot', () => {
    expect(sum(1, 2)).toMatchSnapshot();
});
```

PASS

‘./__snapshots__/sum.test.js’

```
exports[`should sum 2 numbers snapshot 1`] = `3`;
```

Icon made by Freepik from www.flaticon.com

```
export default function sum(a, b) {
    return a + b;
}
```

```
export default function sum(a, b) {
    return a + b + 1;
}
```

```
npm i --save-dev jest enzyme enzyme-to-json
```

```
// file: sum.test.js
it('should sum 2 numbers snapshot', () => {
    expect(sum(1, 2)).toMatchSnapshot();
});
```

📷 FAIL

```
 FAIL  src/sum.test.js
  ● should sum 2 numbers snapshot

    expect(value).toMatchSnapshot()

    Received value does not match stored snapshot 1.

    - 3
    + 4
 FAIL  src/sum.test.js
  ● should sum 2 numbers snapshot

    expect(value).toMatchSnapshot()

    Received value does not match stored snapshot 1.

    - 3
    + 4
```

```
$ jest -u src/sum.test.js
$ jest -u src/sum.test.js
```

```
FAIL
```

```
$ jest -u src/sum.test.js

PASS
```

```
$ jest -u src/sum.test.js
```

PASS

📄 ‘./__snapshots__/sum.test.js’

```
exports[`should sum 2 numbers snapshot 1`] = `4`;
```

```
npm i --save-dev jest enzyme enzyme-to-json
```

Run

```
it('should ...', () => {
    expect(...).toBe(...);
});
```

Snap

```
it('should ...', () => {
    expect(...).toMatchSnapshot();
});
```

Icon made by Madebyoliver from www.flaticon.com

```
npm i --save-dev jest enzyme enzyme-to-json
```

```
import React from 'react';
import { render } from ‘enzyme';


it('should render react div DOM', () => {
    const div = render(
        <div className="div-class">
            <span className=“span-class">
                I love Oath
            </span>
        </div>
    );

    console.log(div);
});
```

```
npm i --save-dev jest enzyme enzyme-to-json
```

```
console.log(div);

{
    '0': {
        type: 'root',
        name: 'root',
        attribs: {},
        children: [
            [Object]
        ],
        next: null,
        prev: null,
        parent: null
    },
    options: {
        withDomLvl1: true,
        normalizeWhitespace: false,
        xmlMode: false,
        decodeEntities: true
    },
    length: 1,
    _root: [Circular]
}
```

```
npm i --save-dev jest enzyme enzyme-to-json
```

```
import React from 'react';
import { render } from 'enzyme';
import enzymeSerializer from 'enzyme-to-json/serializer';

expect.addSnapshotSerializer(enzymeSerializer);

it('should render react div DOM', () => {
    const div = render(
        <div className="div-class">
            <span className=“span-class">
                I love Oath
            </span>
        </div>
    );

    expect(div).toMatchSnapshot();
});
```

```
<div className="div-class">
    <span className=“span-class">
        I love Oath
    </span>
</div>
```

‘./__snapshots__/Test.test.js’

```
exports[`should render react div DOM 1`] = `
<div
  class="div-class"
>
  <span
    class="span-class"
  >
    I love Oath
  </span>
</div>
`;
```

Live Demo

x 3

Tedious

Manual

Extra code to maintain

Icon made by Freepik from www.flaticon.com

Icon made by Freepik from www.flaticon.com

Icon made by Madebyoliver from www.flaticon.com

Icon made by Iconnice from www.flaticon.com

Icon made by Freepik from www.flaticon.com

Dynamic/Interactive tests

Icon made by Anton from www.flaticon.com

Icon made by Iconnice from www.flaticon.com

```
const wrapper = shallow(<Foo />);

wrapper.setState({ name: 'bar' });
wrapper.setProps({ name: 'bar' });
wrapper.setContext({ name: 'bar' });
```

```
it('should ...', () => {
    expect(...).to.be.a(...);
    expect(...).to.equal(...);
    expect(...).to.be.some.test.that.i.hate.to.write(...);
});
```

```
it('should ...', () => {
    expect(...).toMatchSnapshot();
});
```

☺

FAQ / Q&A

1

```
exports[`should return name starting with Tim Armstrong 1`] = `
Array [
    Object {
        "info": Object {
            "image": "tim-armstrong.jpg",
            "oath": Array [
                "Give up!",
                "Take private jets",
            ],
        },
        "name": "Tim Armstrong",
    },
]
`;
```

Prettify snapshot files ?

Icon made by Freepik from www.flaticon.com

1

```
exports[`should return name starting with Tim Armstrong 1`] = `
Array [
   {
       "name": "Tim Armstrong",
       "image": "tim-armstrong.jpg",
       "oath": [“Give up!”, "Take private jets"],
   },
]
`;
```

Prettify snapshot files ?

✓

Icon made by Freepik from www.flaticon.com

1

```
import yourSerializer from ‘./yourSerializer’;

expect.addSnapshotSerializer(yourSerializer);
```

Customize your serializer

2

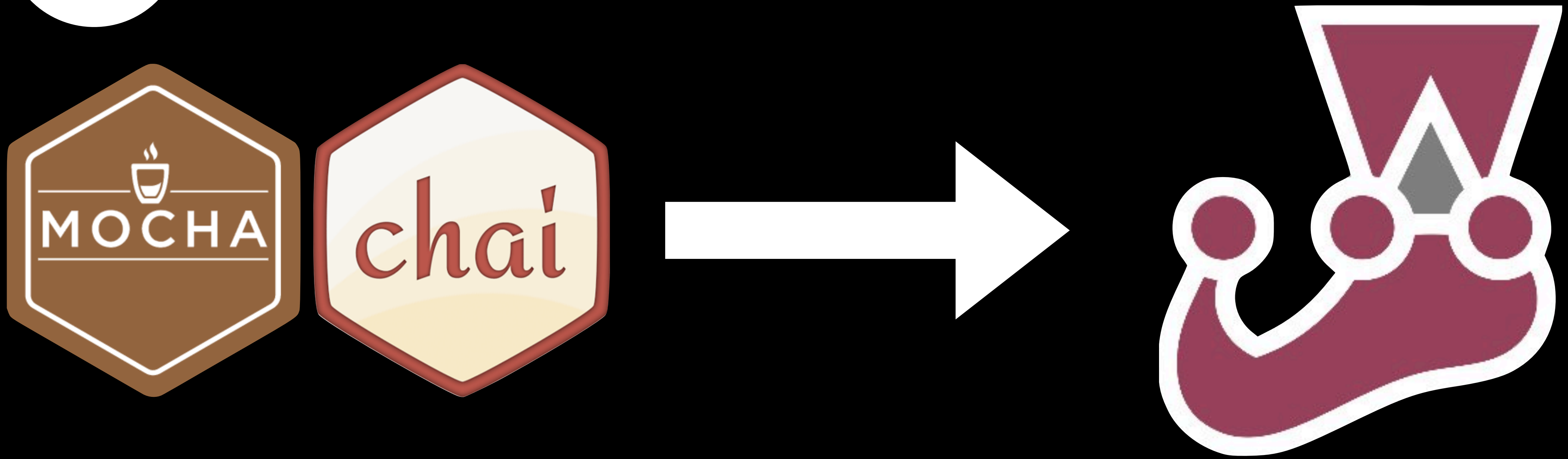
`‘./__snapshots__/*.snap’`

gitignore this folder ?

✗

Icon made by Hadrien from www.flaticon.com

3



Painless Migration?

Icon made by Chris Veigt from www.flaticon.com

```
$ npm i create-react-app -g

$ create-react-app snapshot && \

> cd snapshot/ && \

> npm run eject && \

> npm i enzyme enzyme-to-json --save-dev
```