From Zero to Hero
Towards Front-End Craftsmanship
What shall I do next?
You either die a JavaScript hero, or you live long enough to read those JavaScript fatigue articles.
Sample Project
github.com/ariya/from-zero-to-hero

Test Runner

TodoMVC

Test Library

Assertion Library

mocha

JSCS

Style Checker
> npm start
> from-zero-to-hero@0.0.1 start /home/ariya/from-hero-to-zero
> http-server -p 9000

Starting up http-server, serving ./
Available on:
  http://127.0.0.1:9000
  http://192.168.1.69:9000
Hit CTRL-C to stop the server
Prevent regressions with Git hooks
git commit

Run **pre-commit** hook

Exit code = 0?

- **Abort** the commit process
- **Continue**
git push

Run **pre-push** hook

Exit code = 0 ?

- Abort the push process
- Continue

N Y
Create a pre-commit hook script

```bash
cat << EOF >> .git/hooks/pre-commit
npm run jscs
EOF
```

Ensure that it is executable

```bash
chmod +x .git/hooks/pre-commit
```
> git commit -a -m "Fix bug #666"

> from-zero-to-hero@0.0.1 jscs /home/ariya/from-hero-to-zero
> jscs -p idiomatic js/*/*.js

Invalid quote mark found at js/todoModel.js:

```
1 | var app = app || {};  
2 | (function() {       
3 |     'use strict';   
3 |     var Utils = app. 
4 |     var Utils = app. 
5 |     // Generic "model" object. You can use whatever
```

1 code style error found.
Manage code complexity
function searchConsultant() {
    var candidate = [];
    for (var i = 0; i < depts.length; ++i) {
        var dept = depts[i];
        if (dept !== 'Service') continue;
        for (var j = 0; j < dept.members[j].length; ++i) {
            var e = dept.members[j];
            if (e.state === 'CA') {
                if (e.keywords.indexOf('JavaScript') >= 0) {
                    candidate.push(e);
                }
            }
        }
    }
}
> cr -f plain search.js

search.js

**Function:** searchConsultant

- **Line No.:** 1
- **Physical LOC:** 17
- **Logical LOC:** 13
- **Parameter count:** 0
- **Cyclomatic complexity:** 6
- **Cyclomatic complexity density:** 46.15384615384615%
- **Halstead difficulty:** 15.235294117647058
- **Halstead volume:** 371.5647232790157
- **Halstead effort:** 5660.897842897945
Monitor and track code coverage
Istanbul

Instrumenter + Coverage analysis
module.exports = function(config) {
  config.set({
    basePath: '..', 
    autoWatch: true, 
    frameworks: ['mocha', 'chai'], 
    files: ['js/*.js', 'test/*.js'], 
    plugins: ['karma-mocha', 'karma-chai', 'karma-coverage', 'karma-phantomjs-launcher'], 
    browsers: ['PhantomJS'], 
    singleRun: true, 
    preprocessors: { 'js/*.js': ['coverage'] }, 
    reporters: ['progress', 'coverage'], 
    coverageReporter: {
      dir: 'coverage/',
      reporters: [{ type: 'html', subdir: 'html' }] 
    }
  });
};
Coverage Report

Statistics

Uncovered Statements
Code Coverage Dashboard

83.87% ✓

<table>
<thead>
<tr>
<th>Files</th>
<th>Statements</th>
<th>Hit</th>
<th>Missing</th>
<th>Partial</th>
<th>Branches</th>
<th>Methods</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>todoModel.js</td>
<td>35</td>
<td>27</td>
<td>8</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>77.14%</td>
</tr>
<tr>
<td>utils.js</td>
<td>27</td>
<td>25</td>
<td>2</td>
<td>0</td>
<td>7</td>
<td>0</td>
<td>92.59%</td>
</tr>
<tr>
<td>Project Totals</td>
<td>62</td>
<td>52</td>
<td>10</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>83.87%</td>
</tr>
</tbody>
</table>
Coverage Tracking in Pull Request

Current coverage is 80.00%

Merging #3 into master will decrease coverage by -20.00% as of d85bc86

<table>
<thead>
<tr>
<th></th>
<th>master</th>
<th>#3</th>
<th>diff</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Stmts</td>
<td>4</td>
<td>5</td>
<td>+1</td>
</tr>
<tr>
<td>Branches</td>
<td>1</td>
<td>2</td>
<td>+1</td>
</tr>
<tr>
<td>Methods</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Hit | 4 | 4      |
Partial | 0 | 1 | +1 |
Missed | 0 | 0      |

Review entire Coverage Diff as of d85bc86

Powered by Codecov. Updated on successful CI builds.

“If you think JSLint hurts your feelings, wait until you use Istanbul.”

@davglass
Run the tests with evergreen browsers
#1 Continuous Delivery service for Windows

Your new build server in a cloud. Start in minutes. Enjoy faster results.

SIGN UP FOR FREE

Free for open-source projects. Plans start from $29/month.

install:
  - choco install firefox
  - choco upgrade firefox
  - choco install googlechrome
  - choco upgrade googlechrome
  - ps: Install-Product node $env:nodejs_version
  - node --version
  - npm --version
  - npm install

test_script:
  - npm test
Mozilla Firefox 44.0.2

This package was approved as a trusted package on 2/11/2016.

Bringing together all kinds of awesomeness to make browsing better for you.

This package installs Firefox in the first language which matches this list:

1. Install arguments override parameter if present, e.g. choco install Firefox -packageParameters "l=en-GB". To get a list of all available locales have a look at this file: https://releases.mozilla.org/pub/firefox/releases/latest/README.txt.
2. If Firefox is already installed: the same language as the already installed Firefox.
3. The Windows system language where the Firefox package gets installed.
4. If Firefox does not support the system language, it will fall back to “en-US”


To install Mozilla Firefox, run the following command from the command line or from PowerShell:

C:\> choco install firefox

https://chocolatey.org/packages/Firefox
Running Install scripts

choco install firefox
Installing the following packages:
firefox
By installing you accept licenses for the packages.
Firefox v41.0.2 already installed.
   Use --force to reinstall, specify a version to install, or try upgrade.
Chocolatey installed 0/1 package(s). 0 package(s) failed.

choco upgrade firefox
Upgrading the following packages:
firefox
By upgrading you accept licenses for the packages.
You have Firefox v41.0.2 installed. Version 44.0.2 is available based on your source(s).

Firefox v44.0.2
   Downloading Firefox 32 bit
   Installing Firefox...
Firefox has been installed.
The upgrade of firefox was successful.
5 Ensure a consistent development setup with Docker
New (happy) kid on the block
Docker Toolbox on OS X/Windows

http://ariya.ofilabs.com/2016/02/easy-docker-on-os-x.html
In-container Build

http://ariya.ofilabs.com/2014/12/docker-and-phoenix-how-to-make-your-continuous-integration-more-awesome.html
docker run -it -v $PWD:/src node:5 bash
$ docker run -it -v $PWD:/src node:5 bash

root@f28390ec6f58:/src# cat /etc/debian_version
8.3

root@f28390ec6f58:/src# cd /src

root@f28390ec6f58:/src# npm install

```
npm info it worked if it ends with ok
npm info using npm@3.6.0
npm info using node@v5.7.1
npm info attempt registry request try #1 at 11:09:19 PM```

npm run coverage

npm info it worked if it ends with ok
npm info using npm@3.6.0
npm info using node@v5.7.1
npm info lifecycle from-zero-to-hero@0.0.1~precoverage: from-zero-to-hero@0.0.1
npm info lifecycle from-zero-to-hero@0.0.1~coverage: from-zero-to-hero@0.0.1

> from-zero-to-hero@0.0.1 coverage /src
> cd test && karma start coverage.conf

06 03 2016 23:00:46.690:INFO [karma]: Karma v0.13.21 server started at http://localhost:9876/
06 03 2016 23:00:46.707:INFO [launcher]: Starting browser PhantomJS
06 03 2016 23:00:48.280:INFO [PhantomJS 2.1.1 (Linux 0.0.0)]: Connected on socket /#kp_WwtJE66DQ3pwZAAAA with id 56126974
PhantomJS 2.1.1 (Linux 0.0.0): Executed 6 of 6 SUCCESS (0.027 secs / 0.003 secs)

npm info lifecycle from-zero-to-hero@0.0.1~postcoverage: from-zero-to-hero@0.0.1
npm info ok
Launch the tests with many web browsers
wget -q https://saucelabs.com/downloads/sc-4.3.11-linux.tar.gz

tar -xzf sc-4.3.11-linux.tar.gz

export PATH=$PATH:`pwd`/sc-4.3.11-linux/bin

sc -u $SAUCE_USERNAME -k $SAUCE_ACCESS_KEY -f ~/sc_ready &

sleep 25

while [ ! -e ~/sc_ready ]; do sleep 5; done
5 Tests Found

<table>
<thead>
<tr>
<th>NAME</th>
<th>P</th>
<th>B</th>
<th>DURATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hero</td>
<td>🌌 9</td>
<td>🍊</td>
<td>8s</td>
</tr>
<tr>
<td>Hero</td>
<td>🌌 9</td>
<td>🍊</td>
<td>10s</td>
</tr>
<tr>
<td>Hero</td>
<td>🌌 13</td>
<td>🍊</td>
<td>10s</td>
</tr>
<tr>
<td>Hero</td>
<td>🐈 5</td>
<td>🍊</td>
<td>52s</td>
</tr>
<tr>
<td>Hero</td>
<td>🌌 8.4</td>
<td>🍊</td>
<td>1m 2s</td>
</tr>
</tbody>
</table>

Missing Tests?
Sauce Labs Dashboard

Internet Explorer 9 on Windows 7
Safari 9 on OS X 10.11
Edge 13 on Windows 10
Chrome on Android 5.0
Mobile Safari 8 on iOS 8.4
$ npm run saucelabs

> from-zero-to-hero@0.0.1 saucelabs /home/ubuntu/src/github.com/ariya/from-zero-to-hero
> cd test && karma start saucelabs.conf

04:11:58.484:INFO [karma]: Karma v0.13.21 server started at http://localhost:9876/
04:11:58.500:INFO [launcher]: Starting browser microsoftedge (Windows 10) on SauceLabs
04:11:58.524:INFO [launcher]: Starting browser safari 9.0 (OS X 10.11) on SauceLabs
04:11:58.537:INFO [launcher]: Starting browser android (Linux) on SauceLabs
04:11:58.550:INFO [launcher]: Starting browser iphone 8.4 (OS X 10.11) on SauceLabs
04:11:58.566:INFO [launcher]: Starting browser internet explorer 9.0 (Windows 7) on SauceLabs
Test your downstream projects
function test_downstream_project(project, repo) {
    console.log('Cloning', repo, '...');
    execute('git clone --depth 1 ' + repo + ' ');
    process.chdir(project);
    console.log();
    execute('npm install');
    console.log();
    console.log('Replacing awesome with a fresh one...');
    copy_file(source, './node_modules/awesome/awesome.js');
    console.log();
    try {
        execute('npm test');
    } catch (e) {
        console.log('Failing: ', e.toString);
        process.exit(1);
    }
    process.chdir('..');
}

Clone a downstream project

Install dependencies

Replace the library with a fresh one

Run the test suite
Run different tests in parallel
Parallel Execution

- Verify code coverage
- Monitor complexity
- Run cross-browser tests
- Check downstream project
This is the heroes you’re looking for
Thank You

@ariyahidayat

speakerdeck.com/ariya

shapesecurity.com

Booth #208

Some artworks are from http://openclipart.org.