

# 자바스크립트 프로젝트의 단위 테스트, 테스트 커버리 지 그리고 CI

넷스루 개발2팀 이병주

# #1. 단위 테스트

# 단위 테스트 프레임워크

- JsUnit
- QUnit
- Jasmine
- Mocha
- Chai
- BusterJS
- JsTestDriver

# JsUnit

- <http://www.jsunit.net/>
- JUnit을 자바스크립트로 포팅한 단위 테스트 프레임워크
- 현재는 유지보수를 중단한 상태

# QUnit

- <http://docs.jquery.com/QUnit>
- jQuery에서 만든 자바스크립트 단위 테스트 프레임워크
- jQuery를 테스트하기 위해서 사용한다
- 참고할 문서들이 많이 존재한다
- jsTestDriver와 연동하기에 용이하다

# Jasmine

- <http://pivotal.github.com/jasmine/>
- JsUnit을 관리하던 Pivotal Labs. 에서 만들었다
- BDD(Behavior Driven Development) 프레임워크
- 보고서 유형을 손쉽게 추가할 수 있다
- 사용자들이 많아 다른 라이브러리들과 호환성이 좋다

# Mocha

- <http://visionmedia.github.com/mocha/>
- Node 기반의 테스트 프레임워크 (브라우저도 지원한다)
- BDD, TDD, QUnit 등 다양한 인터페이스 제공
- 다양한 형태의 보고서 제공
- node-jscoverage와의 연동 지원

# Chai

- <http://chaijs.com/>
- Node 기반의 테스트 프레임워크 (브라우저도 지원한다)
- BDD, TDD 등 다양한 인터페이스 제공
- 공식 홈페이지에서 다양한 플러그인 제공



# BusterJS

- <http://busterjs.org/>
- 브라우저 자동화를 통한 테스트 (JsTestDriver와 비슷하다)
- Headless Browser 테스트를 지원할 계획
- 요즘 자바스크립트의 추세를 많이 반영하고 있는 것 같다

# JsTestDriver

- <http://code.google.com/p/js-test-driver/>
- 브라우저 자동화를 통한 테스트
- 이클립스 플러그인 제공
- Adapter를 사용하여 다른 테스트 프레임워크를 사용할 수 있다

# QUnit – Sample Code

```
module("module1", {
  setup: function() {
    ok(true, "once extra assert per test");
    this.testData = "foobar";
  },
  teardown: function() {
    ok(true, "and one extra assert after each test");
  }
});

test("test with setup and teardown", function() {
  expect(3);
  same(this.testData, "foobar");
});

module("module2");
test("test with neither setup nor teardown", function() {
  expect(0);
});
```

# Jasmine – Sample Code

```
describe("A spec (with setup and tear-down)", function() {  
  var foo;  
  
  beforeEach(function() {  
    foo = 0;  
    foo += 1;  
  });  
  
  afterEach(function() {  
    foo = 0;  
  });  
  
  it("is just a function, so it can contain any code", function() {  
    expect(foo).toEqual(1);  
  });  
  
  it("can have more than one expectation", function() {  
    expect(foo).toEqual(1);  
    expect(true).toEqual(true);  
  });  
});
```

# SinonJS

- <http://sinonjs.org/>
- 자바스크립트 테스트 스파이, 스텝, 목 라이브러리
- 다른 라이브러리와 전혀 의존관계가 없으며, 아무 단위 테스트 프레임워크에서나 잘 동작한다
- Spies, Stubs, Mocks, Fake timers, Fake XHR, Fake server, Sandboxing, Assertions, Matchers

# SinonJS – Sample Code

```
"test should call subscriber": function () {  
  var spy = sinon.spy();  
  PubSub.subscribe("message", sp);  
  PubSub.publishSync("message", undefined);  
  
  assertTrue(spy.called);  
}  
"test should call subscriber": function () {  
  var myAPI = { method: function () {} };  
  var mock = sinon.mock(myAPI);  
  mock.expects("method").once();  
  
  PubSub.subscribe("message", myAPI.methd);  
  PubSub.publishSync("message", undefined);  
  
  mock.verify();  
}
```

# SinonJS – Sample Code

```
{
  setUp: function () {
    this.server = sinon.fakeServer.create();
  },
  tearDown: function () {
    this.server.restore();
  },
  "test should fetch comments from server": function () {
    this.server.respondWith("GET", "/some/article/comments.json",
      [200, { "Content-Type": "application/json" },
        '[{ "id": 12, "comment": "Hey there" }]']);
    var callback = sinon.spy();
    myLib.getCommentsFor("/some/article", callback);
    this.server.respond();

    assert(callback.calledWith([{ id: 12, comment: "Hey therer" }]));
  }
}
```



현재 사용하는 테스트 프레임워크

Jasmine  
+  
SinonJS



# SimpleApp 단위 테스트

Jasmine 1.2.0 revision 1337005947

finished in 0.025s

.....

Passing 9 specs

model.Person

객체가 정상적으로 생성된다.

getName

attributes의 name의 값을 반환한다.

getPart

attributes의 part의 값을 반환한다.

collection.PersonList

객체가 정상적으로 생성된다.

view.DataInputView

객체가 정상적으로 생성된다.

view.DataRecordView

객체가 정상적으로 생성된다.

view.DataListView

객체가 정상적으로 생성된다.

addRecord

tbody에 tr이 추가된다.

<http://pillarlee16.github.com/SimpleApp/test/>

## #2. 테스트 커버리지

# JSCoverage

- <http://siliconforks.com/jscoverage/>
- version 0.5.1 (2010.09.23)
- 브라우저에서 라인 커버리지 측정 가능
- node-jscoverage

# JSCoverage

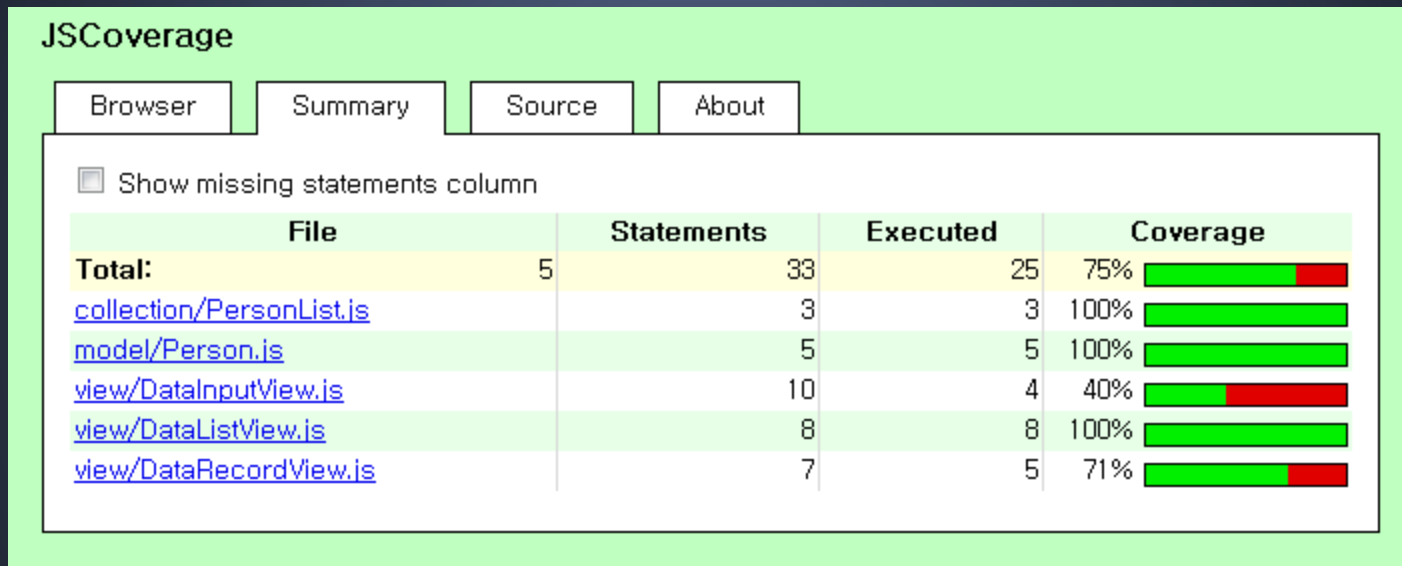
```
jscoverage SOURCE-DIRECTORY DESTINATION-DIRECTORY
```

SOURCE-DIRECTORY/  
app/  
index.html  
script.js



DESTINATION-DIRECTORY/  
app/  
index.html  
script.js [instrumented]  
jscoverage.html

# SimpleApp 테스트 커버리지



<http://pillarlee16.github.com/SimpleApp/coverage/src-copy/jscoverage.html>

#3. CI

# Hudson & Ant

## Hudson

seetoc | 로그인

[Hudson](#)
자동 재실행 켜기

새 작업
 개발자
 빌드 기록
 Project Relationship
 Check File Fingerprint
 New View
 My Views

**빌드 대기 목록**  
 빌드 대기 항목이 없습니다.

**빌드 실행 상태**  
**Status 0/2**  
 Idle

All	SeeToc 1.5	SmartOffer	+					
S	W	작업 ↓	최근 성공	최근 실패	최근 소요 시간	Console		
		<a href="#">ABCAudit</a>	1 mo 25 days (#78)	4 mo 7 days (#33)	37 sec			
		<a href="#">ABCLogging</a>	11 days (#69)	4 mo 6 days (#20)	16 sec			
		<a href="#">ABCReport</a>	5 mo 1 day (#47)	5 mo 8 days (#25)	17 sec			
		<a href="#">ABCSyncSite</a>	5 mo 3 days (#11)	5 mo 3 days (#1)	8.3 sec			
		<a href="#">ClientScript</a>	11 days (#25)	—	16 sec			
		<a href="#">CloseBeta admin admin</a>	1 yr 2 mo (#3)	—	15 sec			
		<a href="#">CloseBeta chat config</a>	1 yr 2 mo (#2)	—	9.5 sec			
		<a href="#">CloseBeta web ROOT</a>	1 yr 1 mo (#2)	—	20 sec			
		<a href="#">LoggingScript(deprecated)</a>	10 hr (#23)	1 day 4 hr (#22)	5.5 sec			
		<a href="#">Maven Cobertura Test</a>	7 mo 28 days (#1)	—	54 sec			
		<a href="#">MonitorWebapp</a>	10 days (#26)	1 mo 26 days (#1)	1 min 1 sec			
		<a href="#">OpenBeta admin admin</a>	6 mo 23 days (#26)	—	36 sec			

# PhantomJS

- <http://phantomjs.org/>
- Headless Webkit with JavaScript API

```
// example
console.log('Loading a web page');
var page = new WebPage();
var url = 'http://www.phantomjs.org/';
page.open(url, function (status) {
    // Page is loaded;
    phantom.exit();
});
```



# Jasmine Reporter Plugins

- HTML Reporter
- Console Reporter
- JUnit Reporter
- JSCoverage Reporter (\*)
- PhantomJS Reporter (\*)

# SimpleApp CI

## Hudson

search

sectoc | 로그아웃

Hudson » SimpleApp

Back to Dashboard

Status

Changes

Workspace

Build Now

Delete Project

Configure

Redmine

Coverage Report

Project SimpleApp

Coverage Report

Workspace

Recent Changes

Latest Test Result (no failures)

Latest Console output

Permalinks

- Last build (#13), 10 hr ago
- Last stable build (#13), 10 hr ago
- Last successful build (#13), 10 hr ago
- Last failed build (#11), 10 hr ago
- Last unsuccessful build (#11), 10 hr ago

Build History (trend)

#13 2012. 7. 9 오전 10:26:51

#12 2012. 7. 9 오전 9:54:10

#11 2012. 7. 9 오전 9:48:33

#10 2012. 7. 9 오전 9:46:38

#9 2012. 7. 8 오후 6:44:42

#8 2012. 7. 8 오후 6:44:40

#7 2012. 7. 8 오후 6:43:42

#6 2012. 7. 8 오후 6:41:23

#5 2012. 7. 8 오후 6:21:35

#4 2012. 7. 8 오후 6:18:57

#3 2012. 7. 8 오후 5:59:08

#2 2012. 7. 8 오후 5:54:23

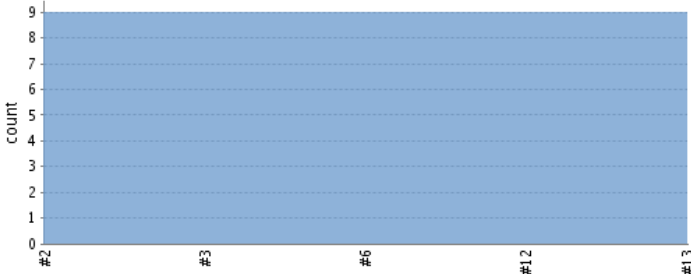
for all

for failures

소개 내용 입력

Disable Project

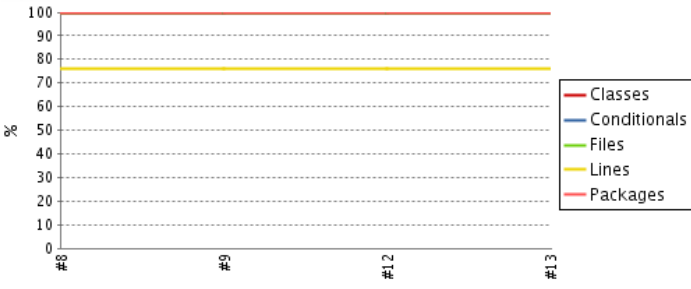
Test Result Trend



(just show failures) enlarge

Code Coverage

Classes 100% Conditionals 100% Files 100% Lines 76% Packages



Classes  
Conditionals  
Files  
Lines  
Packages