



# Testing



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)

But...why?



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



We want to be confident in  
our code, at all times.



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



When is this helpful?



#hackundsoehne

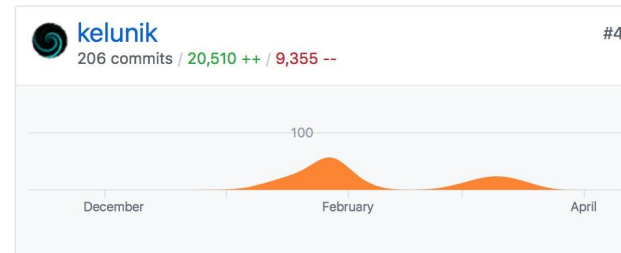
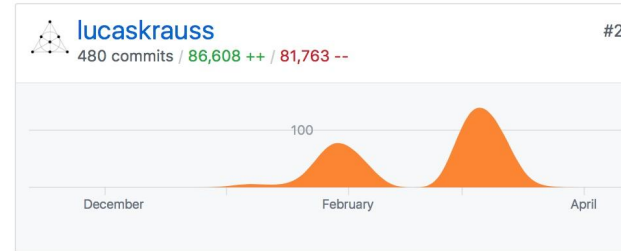
[www.hackundsoehne.de](http://www.hackundsoehne.de)



Nov 15, 2015 – Apr 14, 2016

Contributions: **Commits** ▾

Contributions to master, excluding merge commits



#hackundsoehne

www.hackundsoehne.de



# Fix Continous Integration #328

[Edit](#)[Open](#) phimonk wants to merge 9 commits into `master` from `fix-ci-mysql`

💬 Conversation 1

🔗 Commits 9

📄 Files changed 5

+10 -5



phimonk commented an hour ago

Owner



This fixes errors with travis, gradle and tests that came from updated/changed/unavailable dependencies.

Reviewers



Suggestions

kelunik



lucaskrauss



LeanderK



Assignees



No one—assign yourself

Labels



None yet

Projects



None yet

phimonk and others added some commits 2 days ago

ci-test

✗ bbbf2af

Fix outdated mysql install script for travis

✗ 1370fbb

Add allow unauthenticated flag to apt install

✗ 0195059

Try another install script that might work

✗ 89838bb

Use deprecated travis image

✗ ab6d600

gradle: fix codegen version ...

✗ 37023f7

ImageSimilarityTest: Ignore test case that failes because of outdated... ...

✗ c23eff0

Merge pull request #325 from coolcrowd/gradle ...

✗ fe4c32a

Merge pull request #327 from coolcrowd/fix-test-duplicate-detection ...

✓ cea823d



# Changed mnist download source to web.archive.org, downloads also handle http 302 redirects #81

**Open** avctrh wants to merge 9 commits into tensorflow:master from avctrh:webarchive

💬 Conversation 18

🔗 Commits 9

📄 Files changed 1

+19 -10 ■ ■ ■



avctrh commented on Mar 8

Contributor



MNIST downloads from Yann Lacun's website have been unreliable, leading to build (& CI) failures since this is part of Setup.hs in tensorflow-mnist-input-data.

I've made changes to Setup.hs to point to a webarchive.org cache of YL's site. This required code changes handle http 302 redirects, but still uses the HTTP library and doesn't add new dependencies.

Tested locally and resolves build failures due to MNIST downloads for me.

Reviewers

blackgneздо



Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

📄 avctrh added some commits on Mar 6

🔗 added osx dependency installation script

✓ 7f1f9f2

🔗 added script to download mnist from webarchive since Yann LeCun's web... ...

✓ ce8603f

🔗 requested changes from review

✓ 554579b

🔗 whitespace

✓ 4584a92



```
public class BadTests {  
    @Test  
    public void testCalculateSalary() {  
        MockEmployee sample = new MockEmployee();  
        sample.setLevel("Developer");  
        DB db = new DBMock(sample);  
        SalaryCalculator salaryCalculator = new SalaryQueries(db).getCalculator();  
        assertEquals(55000, salaryCalculator.averageSalary());  
    }  
}
```





```
@Test
public void testImageSimilarity() {
    Future<Image> imageAsync = loadImage("/test/image.jpg");
    Image image = block(imageAsync);
    Future<Integer> hashAsync = image.getHash();
    int hash = block(hashAsync);
    ...
}
```



# What are my options?



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# 1. Proving Correctness



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



```
/*@ public behavior
@  ensures (myInt == \old(myInt) + 1);
@
@*/
public void addOneToMyInt() {
    myInt = myInt + 1;
}
```



## 2. Testing



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



## **“Traditional” Testing**

- The programmer must provide the sample
- The output is tested for one or more properties

## **Property based testing**

- The programmer must specify properties
- The testing-library will now generate samples and test whether the property holds



# Testing Framework: **JU**nit



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# Why use it?

- Simple
- Battle-tested
- Well-supported



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)





```
47  
48     @Test  
49     public void testExpectEmail() throws Exception {  
50         Context context = prepareContext(data.getExperimentRecord().getIdExperiment(), data.getPla  
51         Queries queries = prepareQuery(data, Optional.empty(), taskChooserAlgorithm: null);  
52         View view = queries.getNext(context);  
53         assertTrue(view.getType().equals(View.Type.EMAIL));  
54     }  
55
```



Run QueriesTest

All 13 tests passed – 5s 23ms

Test Name	Duration
testStopgapAnswer	3s 819ms
testCreative	72ms
testInvalidQuality	93ms
testProvideWorkerID	212ms
testPreview	48ms
testWrongCalibration	100ms
testExpectEmailFail	86ms
testExpectGetWorkerI	127ms
testStopgapAnswer	68ms

```
23:14:50.914 [main] DEBUG org.jooq.tools.StopWatch - Finishing : Total: 0.1302ms, +0.1302ms
23:14:50.916 [main] DEBUG edu.kit.ipd.crowdcontrol.workerservice.query.Queries - handling no worker-id
23:14:50.918 [main] DEBUG org.jooq.tools.LoggerListener - Executing query : select `crowdcontrol`.`Platform`.`id_platform`,
23:14:50.918 [main] DEBUG org.jooq.tools.LoggerListener - -> with bind values : select `crowdcontrol`.`Platform`.`id_platform`,
23:14:50.919 [main] DEBUG org.jooq.tools.StopWatch - Query executed : Total: 0.587ms
23:14:50.921 [main] DEBUG org.jooq.tools.LoggerListener - Fetched result
23:14:50.922 [main] DEBUG org.jooq.tools.LoggerListener - : +-----+-----+
23:14:50.922 [main] DEBUG org.jooq.tools.LoggerListener - : |id_platform|name|
23:14:50.922 [main] DEBUG org.jooq.tools.LoggerListener - : +-----+-----+
23:14:50.922 [main] DEBUG org.jooq.tools.LoggerListener - : |rlikf397379nq22hqa4a9brbdv|v0t5tlcesv3giri3v5h|
23:14:50.922 [main] DEBUG org.jooq.tools.LoggerListener - : +-----+-----+
23:14:50.922 [main] DEBUG org.jooq.tools.StopWatch - Finishing : Total: 3.673ms, +3.086ms
```



# Demo time!



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# JUnit Cheatsheet

- **testCompile('junit:junit:4.12')** - your gradle dependency
- **@Test** - to mark your tests with
- **@Test(expected = RuntimeException.class)** - if you expect them to throw Exceptions
- **import static org.junit.Assert.\*** - to use the assertions
  - **assertTrue(x)** - whether x is true
  - **assertEquals(x, y)** - whether x equals y
  - ... (check Assert class)



# How many tests are enough?



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# Mocking using **mockito**



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)

# What is mocking?



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# Example time!



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)





# Mockito Cheatsheet

- **testCompile(org.mockito:mockito-core:2.+')** - your gradle dependency
- **import static org.mockito.Mockito.\*** - to use the assertions
  - **mock(x)** - to mock x
  - **when(x).thenReturn(y)** - to fake functionality without input  
Example: **when(mockedList.get(0)).thenReturn("first");**
  - **when(x.y(Mockito.anyInt())).thenReturn()** - to fake functionality dynamically
  - ... (check Mockito class)



# Mock all the things?



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# Test Driven Development (TDD)



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



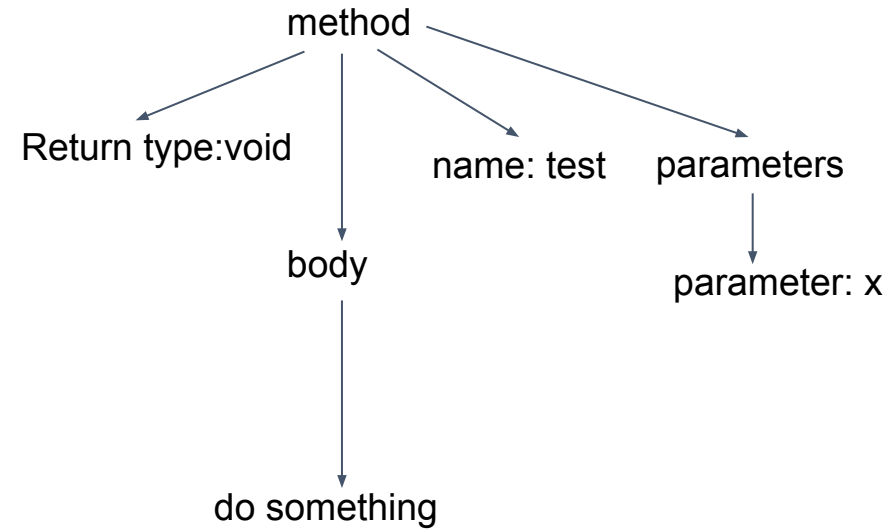
# Test Driven Development

- Write your test(s) before you write your code!
- Can be a whole paradigm
- You don't have to buy all in
- Especially useful when working with new, complicated libraries



# AST

```
public void test(String x) {  
    doSomething();  
}
```



# Demo Time!



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# JUnit Cheatsheet

- **testCompile('junit:junit:4.12')** - your gradle dependency
- **@Test** - to mark your tests with
- **@Test(expected = RuntimeException.class)** - if you expect them to throw Exceptions
- **import static org.junit.Assert.\*** - to use the assertions
  - **assertTrue(x)** - whether x is true
  - **assertEquals(x, y)** - whether x equals y
  - ... (check Assert class)



# Mocha/Chai



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)





# Mocha/Chai

**Mocha** => JUnit for Javascript, but without the asserts, provides other primitives (BDD)

**Chai** => the missing asserts



# Behaviour Driven Development (BDD)



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# Demo Time!



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# Mocha Cheatsheet

- **npm install mocha** - your npm dependency
- **require('assert')** - to use asserts
- **describe('name', function)** - describe your use-case
- **it('description', function)** - your test



# Mocha/Chai Cheatsheet

- **npm install mocha** - your npm dependency
- **require('assert')** - to use asserts
- **describe('name', function)** - describe your use-case
- **it('description', function)** - your test



# Coverage



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# Why do we care about coverage

- Convenient Metric whether your tests are thoroughly
- Provides a nice badge, looks good on paper



# Problems with Coverage

- Does not measure quality of your tests, only quantity
  - Many bugs result from the absence of code, not incorrect code
- Everybody focuses on getting 100% coverage
  - A lot of work
  - Tests tailored to the implementation
  - Diminishing returns





# Some general advice about testing software



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)



# About testing

- Ask yourself: How much testing is necessary, how much testing can i afford?
- Listen to your tests, if they are a pain to write there is probably something wrong.
- Don't tailor your tests to your implementation
  - Implementations change
  - Do you really eliminate bugs?
- Outdated tests are a problem, never ever comment out tests!
- Write them early, a bit of discipline helps you in the long run
- Don't accept non-reproducible tests



# Questions?



#hackundsoehne

[www.hackundsoehne.de](http://www.hackundsoehne.de)

