EXERCISE 0

Checkout and do composer install

https://github.com/lmuzinic/websummercamp-pragmatic-tdd

```
git clone git@github.com:lmuzinic/websummercamp-pragmatic-tdd.git
cd websummercamp-pragmatic-tdd
composer install
vendor/bin/phpunit
```

```
OK (1 test, 1 assertion)
```

There is <10 MB of packages to download, so you can use your mobile data if WIFI does not work.

If you already did this step before arriving do git pull latest changes :)

PRAGMATIC TDD

HELLO

Luka Muzinic
@lmuzinic

WORKSHOP RULES

ASK QUESTIONS

IF YOU STILL DO NOT UNDERSTAND, ASK QUESTIONS AGAIN

DISCUSS RIGHT NOW, DO NOT WAIT FOR THE “RIGHT MOMENT”

WHY WE
NEED TESTING?

EXERCISE 0

WHY AM I HERE?

ARE WE SOFTWARE TESTERS?

AND YET
WE KEEP ON SAYING
WE WRITE TESTS…

WHERE CAN I GET MORE OF THOSE TESTS?

EXCUSES, EXCUSES

TESTS SLOW US DOWN

WE WILL NEVER GET TO 100% CODE COVERAGE

WE DO NOT HAVE TIME TO LEARN TESTING, WE’RE TO BUSY SHIPPING CODE

EXCUSES, EXCUSES

TESTS SLOW US DOWN?

STOP TESTFILEING

DO YOU OFTEN DO THIS?

```
~ php test.php
```

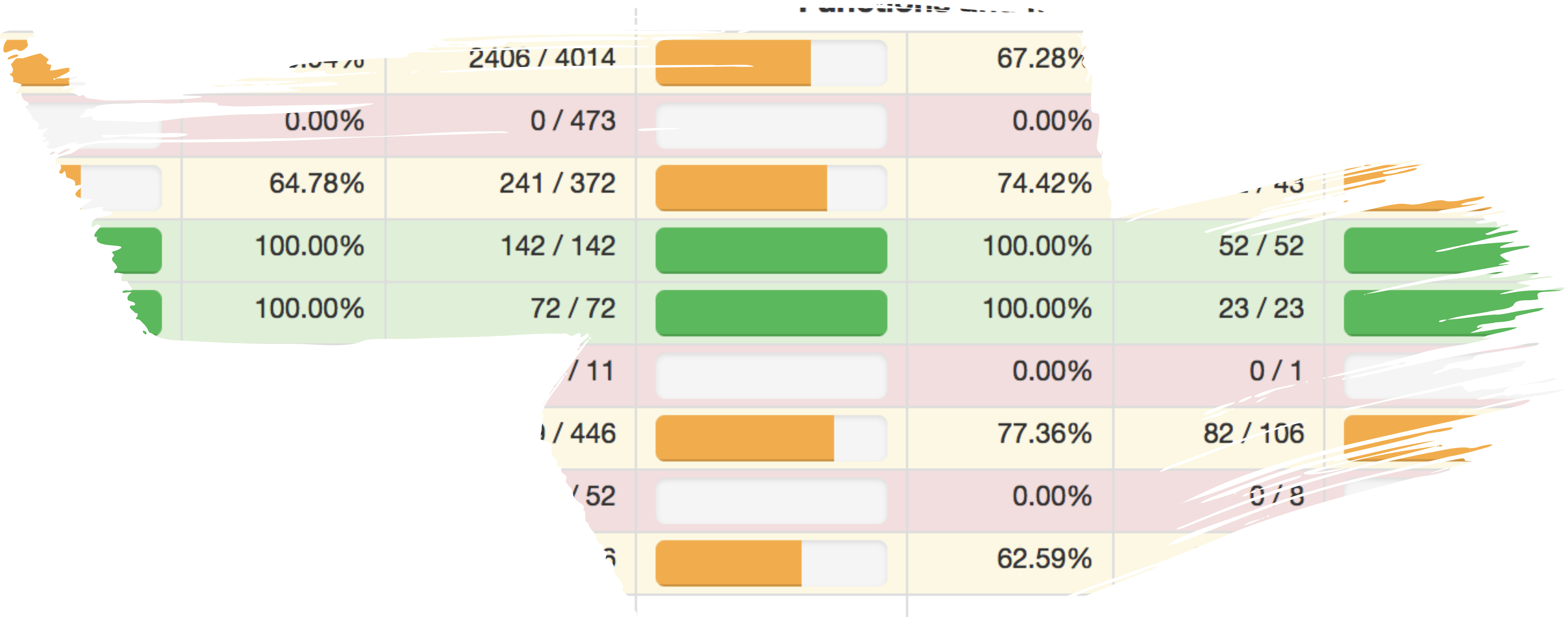
OR THIS

```
http://localhost/test_problem.php
```

EXCUSES, EXCUSES

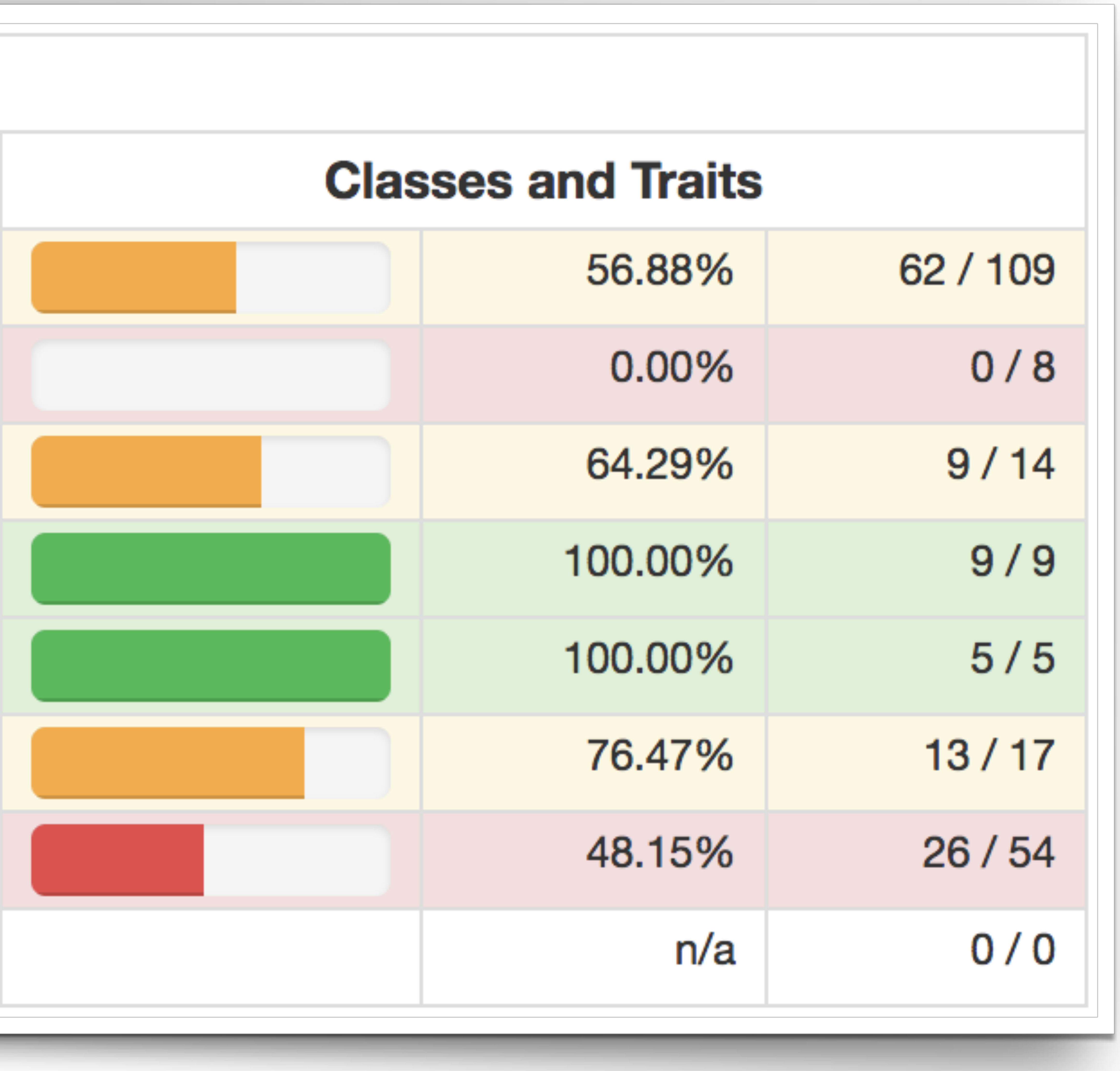
WE WILL NEVER GET TO 100% CODE COVERAGE

CODE COVERAGE



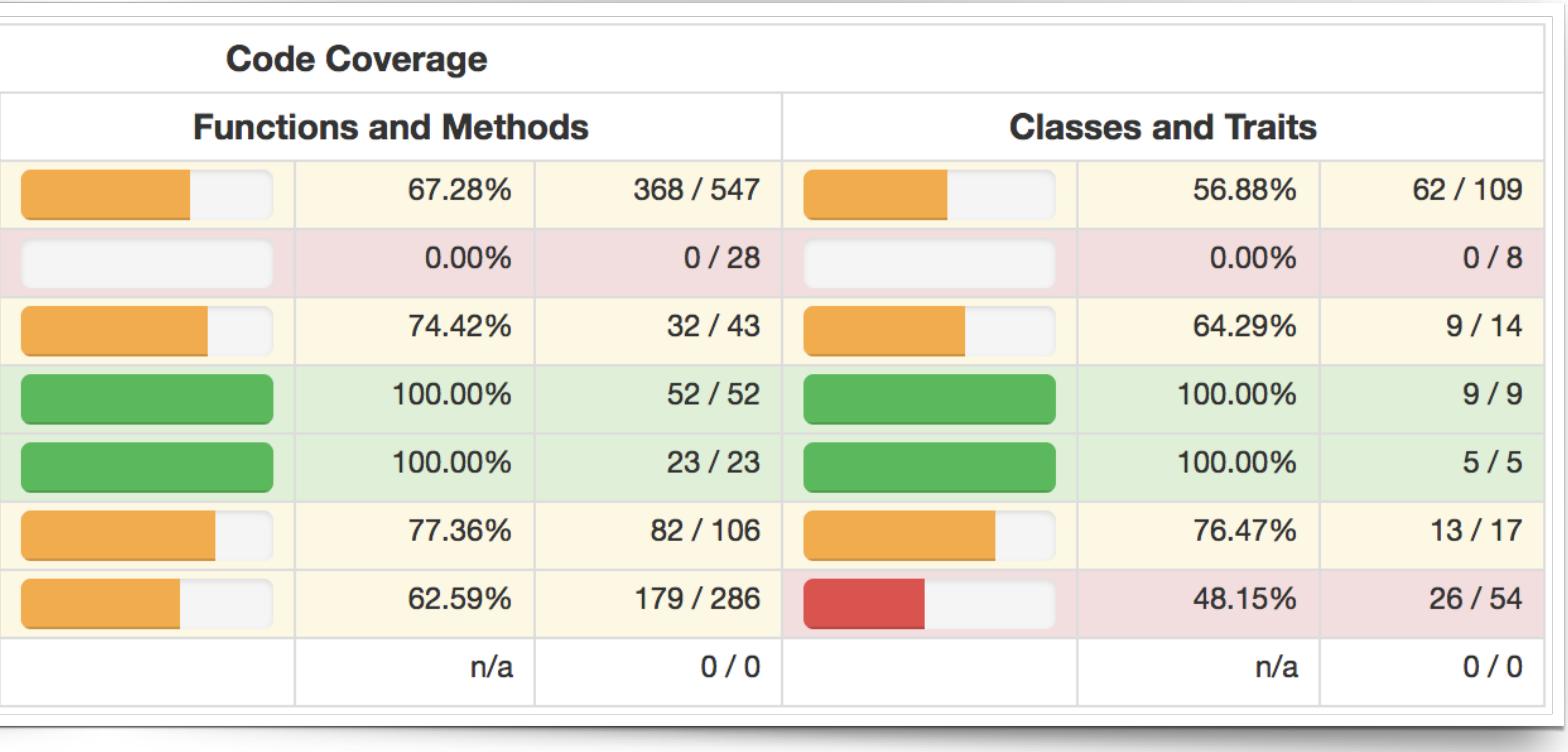
	2406 / 4014	67.28%	
0.00%	0 / 473	0.00%	
64.78%	241 / 372	74.42%	/ 43
100.00%	142 / 142	100.00%	52 / 52
100.00%	72 / 72	100.00%	23 / 23
	/ 11	0.00%	0 / 1
	/ 446	77.36%	82 / 106
	/ 52	0.00%	0 / 8
	6	62.59%	

CODE COVERAGE



Classes and Traits		
	56.88%	62 / 109
	0.00%	0 / 8
	64.29%	9 / 14
	100.00%	9 / 9
	100.00%	5 / 5
	76.47%	13 / 17
	48.15%	26 / 54
	n/a	0 / 0

CODE COVERAGE

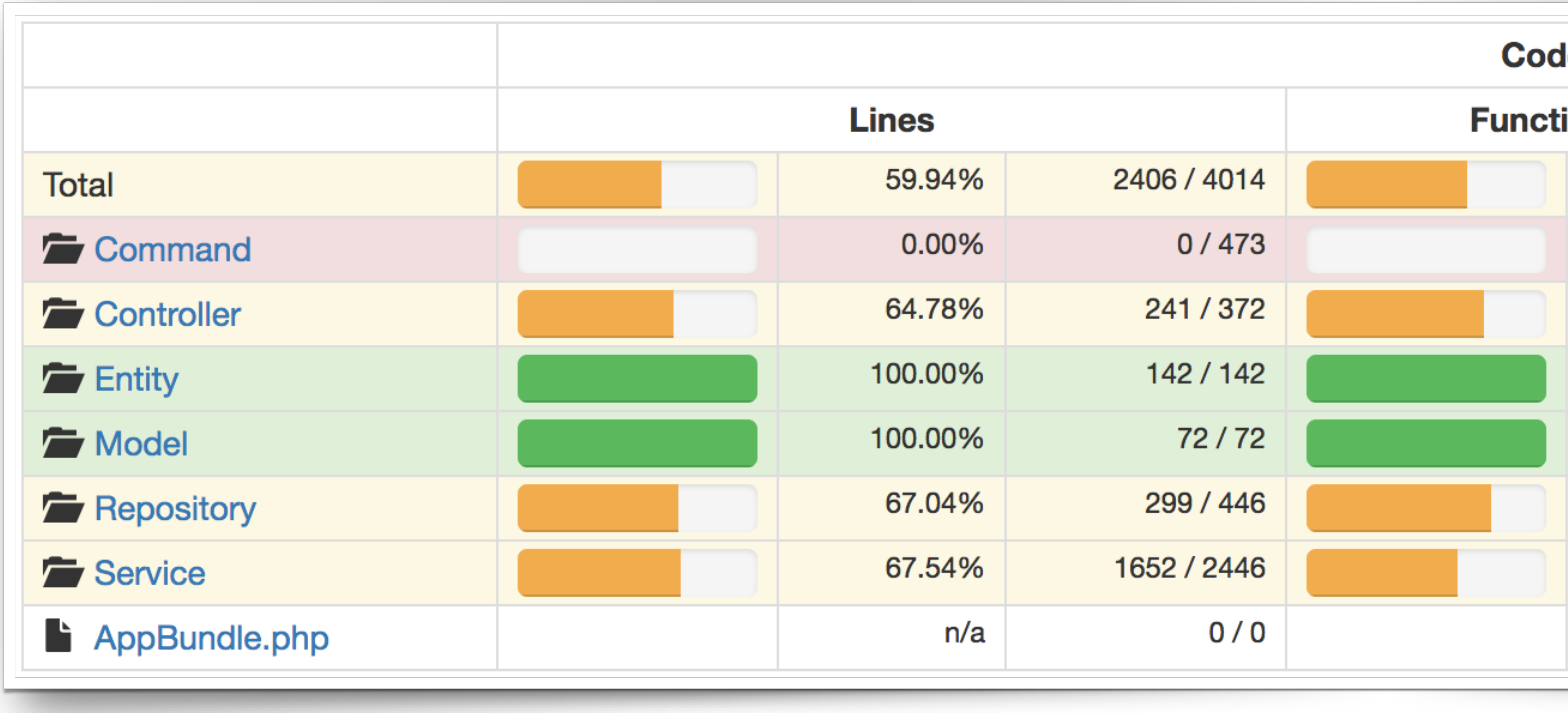


Code Coverage					
Functions and Methods			Classes and Traits		
	67.28%	368 / 547		56.88%	62 / 109
	0.00%	0 / 28		0.00%	0 / 8
	74.42%	32 / 43		64.29%	9 / 14
	100.00%	52 / 52		100.00%	9 / 9
	100.00%	23 / 23		100.00%	5 / 5
	77.36%	82 / 106		76.47%	13 / 17
	62.59%	179 / 286		48.15%	26 / 54
	n/a	0 / 0		n/a	0 / 0

CODE COVERAGE

Code Coverage					
Lines			Functions and Methods		
	59.94%	2406 / 4014		67.28%	368 / 547
	0.00%	0 / 473		0.00%	0 / 28
	64.78%	241 / 372		74.42%	32 / 43
	100.00%	142 / 142		100.00%	52 / 52
	100.00%	72 / 72		100.00%	23 / 23
	67.04%	299 / 446		77.36%	82 / 106
	67.54%	1652 / 2446		62.59%	179 / 286
	n/a	0 / 0		n/a	0 / 0

20 > 80



				Cod
	Lines			Functi
Total		59.94%	2406 / 4014	
Command		0.00%	0 / 473	
Controller		64.78%	241 / 372	
Entity		100.00%	142 / 142	
Model		100.00%	72 / 72	
Repository		67.04%	299 / 446	
Service		67.54%	1652 / 2446	
AppBundle.php		n/a	0 / 0	

EXCUSES, EXCUSES

WE DO NOT HAVE TIME TO LEARN TESTING, WE’RE TO BUSY SHIPPING CODE

//@TODO: STANDSTILL

DON’T BE SCARED OF PHPUNIT*

IT IS JUST A CODE RUNNER

UNIT, INTEGRATION OR ACCEPTANCE TESTS

SMOKE TESTS

WEBSITE SCRAPER

…

TESTS ANSWER QUESTIONS

IS MY CODE WORKING CORRECTLY?

WILL MY CODE WORK CORRECTLY?

HOW IS MY CODE SUPPOSE TO WORK?

IS MY CODE DESIGNED WELL?

CAN I REFACTOR MY CODE?

TESTS ANSWER QUESTIONS

IS MY CODE WORKING CORRECTLY?

WILL MY CODE WORK CORRECTLY?

HOW IS MY CODE SUPPOSE TO WORK?

IS MY CODE DESIGNED WELL?

CAN I REFACTOR MY CODE?

TESTS ANSWER QUESTIONS

IS MY CODE WORKING CORRECTLY?

WILL MY CODE WORK CORRECTLY?

HOW IS MY CODE SUPPOSE TO WORK?

IS MY CODE DESIGNED WELL?

CAN I REFACTOR MY CODE?

TESTS ANSWER QUESTIONS

IS MY CODE WORKING CORRECTLY?

WILL MY CODE WORK CORRECTLY?

HOW IS MY CODE SUPPOSE TO WORK?

IS MY CODE DESIGNED WELL?

CAN I REFACTOR MY CODE?

TESTS ANSWER QUESTIONS

IS MY CODE WORKING CORRECTLY?

WILL MY CODE WORK CORRECTLY?

HOW IS MY CODE SUPPOSE TO WORK?

IS MY CODE DESIGNED WELL?

CAN I REFACTOR MY CODE?

TESTS ANSWER QUESTIONS

IS MY CODE WORKING CORRECTLY?

WILL MY CODE WORK CORRECTLY?

HOW IS MY CODE SUPPOSE TO WORK?

IS MY CODE DESIGNED WELL?

CAN I REFACTOR MY CODE?

HOW TO WRITE TESTS?

GIVEN WHEN THEN

SETUP EXERCISE VERIFY TEARDOWN

ANATOMY OF PHPUNIT TEST CASE

```
class TeamTest extends PHPUnit\Framework\TestCase;
{
    private $team;

    public function setUp()
    {
        $this->team = Team::create(‘Hellas Verona');
    }

    public function testGetName()
    {
        $name = $this->team->getName();

        $this->assertEquals('Hellas Verona', $name);
    }
}
```

WORKSHOP RULES

ASK QUESTIONS

IF YOU STILL DO NOT UNDERSTAND, ASK QUESTIONS AGAIN

DISCUSS RIGHT NOW, DO NOT WAIT FOR THE “RIGHT MOMENT”

DISCUSSION

IMPLEMENT A FOOTBALL LEAGUE MANAGEMENT APP

- We want to display standings table on our website

WHAT IS YOUR BIGGEST CONCERN?

EXERCISE 0

Checkout and do composer install

https://github.com/lmuzinic/websummercamp-pragmatic-tdd

```
git clone git@github.com:lmuzinic/websummercamp-pragmatic-tdd.git
cd websummercamp-pragmatic-tdd
composer install
vendor/bin/phpunit
```

```
OK (1 test, 1 assertion)
```

Do git pull for latest changes :)

EXERCISE 1

WRITE A FUNCTIONAL TEST FOR OUR APP

- main goal of the test is to show how our code is supposed to work
- let us play with with the design, try out how it feels
- decide how the output is going to look

EXERCISE 2

RUN TEST

- understand why it is failing, look at error messages

DECIDE WHAT IS CRITICAL TO WORK

- estimate if you have time to write tests now

WRITE A TEST FOR THAT

- create a separate test file as our focus is shifting

GOTO RUN TEST :)

EXERCISE 3

TIE IT ALL TOGETHER

- make all tests green

QUESTIONS? /r/AMA?

Luka Muzinic

@lmuzinic

luka.muzinic.net/talks

HOMEWORK

READING LIST

Reading list

https://www.devmynd.com/blog/five-factor-testing/
https://martinfowler.com/articles/practical-test-pyramid.html
https://dev.to/theobendixson/the-problem-that-unit-tests-solve-b2l
https://blog.liplex.de/testing-private-and-protected-methods-with-phpunit/

Libraries

https://github.com/sebastianbergmann/phpunit
https://github.com/phpspec/phpspec
https://github.com/Codeception/Codeception
https://github.com/phpstan/phpstan
https://github.com/infection/infection

HOMEWORK

WRITE ANOTHER TEST

- given a scenario where two teams have played a tie
- implement the feature

REFACTOR

- given a scenario where RuleBook also defines how much points should each team get after match win, write additional test case where each win gets you 2 points
- implement the feature

BONUS POINTS

- have an idea how to make this whole thing better?
- write tests and demonstrate :)

KTHXBAI

Photos by Les Anderson, Joshua Earle, Ian Espinosa and Tom Roberts on Unsplash