



Simply Good Tests

A Practical Approach

Thomas Much

 @thmuch

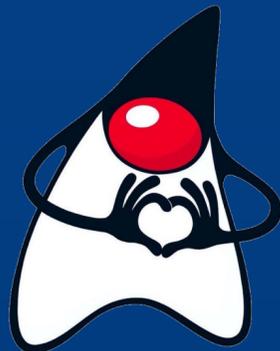
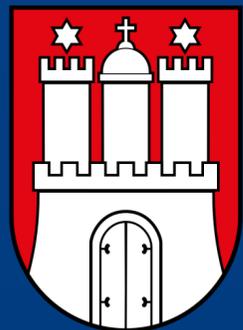
5 October 2021

Software Developer

Technical Agile Coach



 @thmuch



Brief introduction with a couple of slides

Live coding

Short roundup

Q & A

Why this session?

career changers

newcomers

returnees

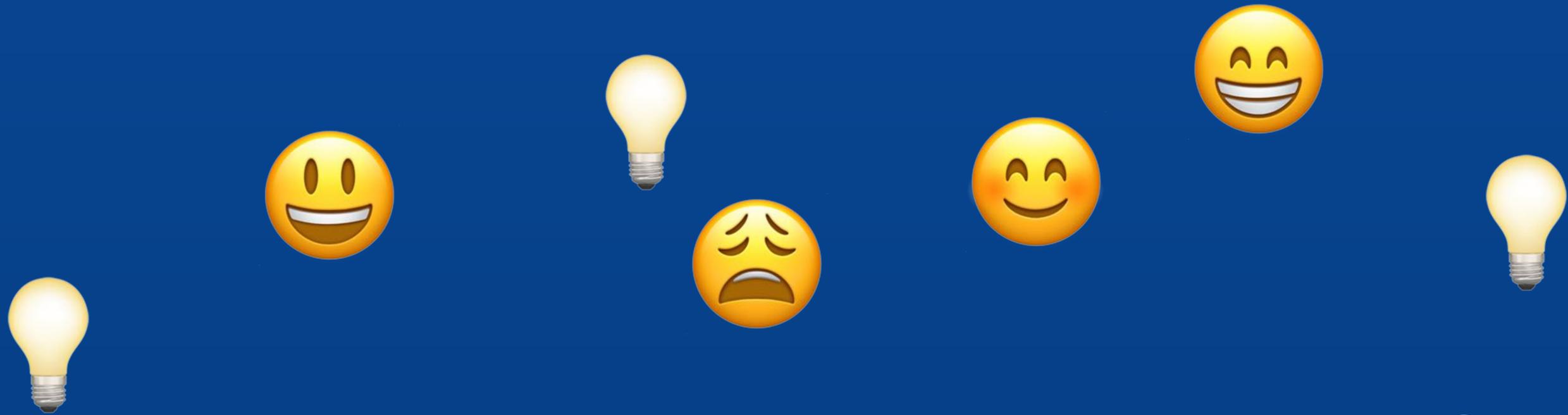
copy & paste

cargo
cult

Beginners

Tradition

Basics. Good practices.



What are "good tests"?



Reasons against good tests

not
maintainable
anyway

always
broken
anyway



not
readable
anyway

Let's just get
high code coverage



Not now,
we don't have time



So much quality
costs too much

Retesting is
stupid & expensive

How to test?

TDD?

Test Driven **Design** & Development



Test First?

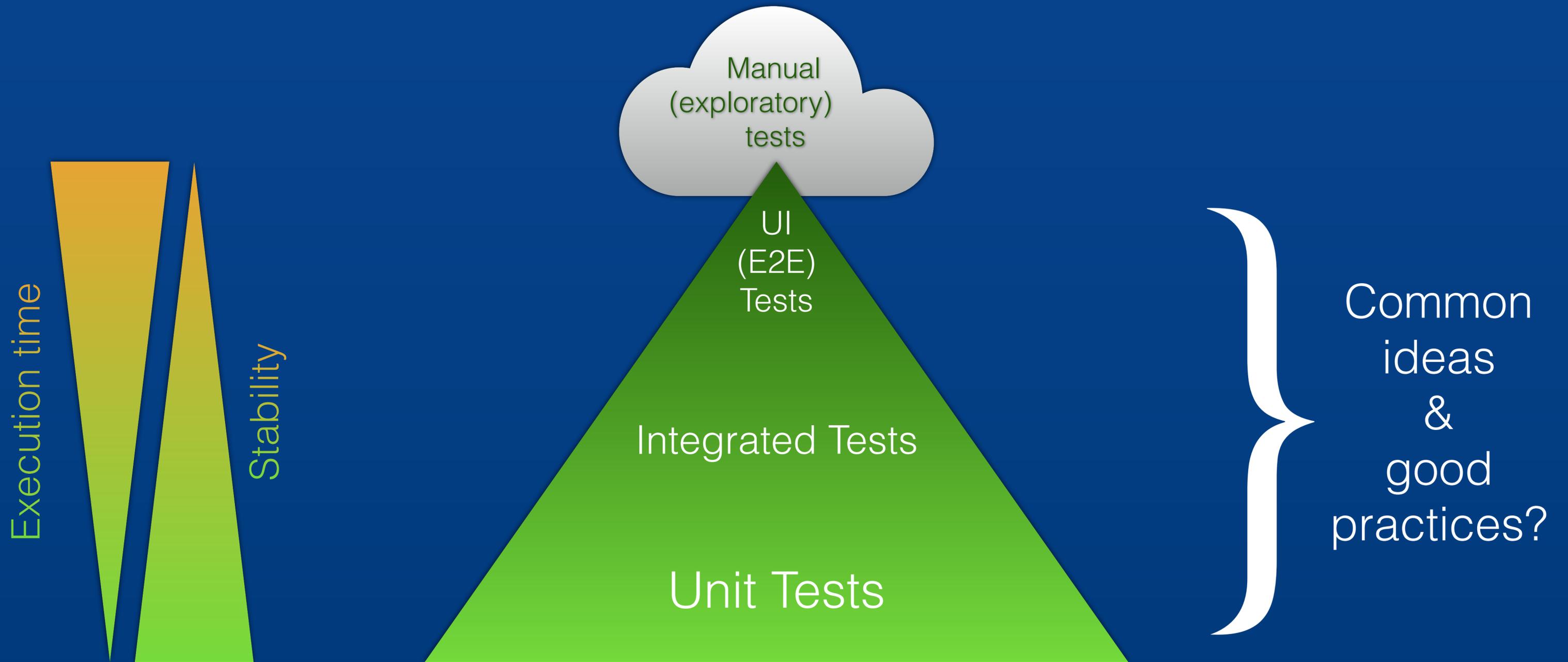
Definitely not "all tests up front"!

"Accompanying tests"

Step by step

Small experiments!

Test Pyramid



Requirements:

With the `SushiService`,
a `customer` can order a `menu`.

The service returns an `order number`.
An order number is an `8-digit number`
that starts with the digit "1".

Ordering à la carte is out of scope for this task.

Live Demo

[https://github.com/**thmuch**/**simply-good-tests**](https://github.com/thmuch/simply-good-tests)

Recap

Understandable, Maintainable Tests

Every test method tests **one thing**

Given/When/Then or Arrange/Act/Assert helps

Prefer **explicit setup**

Avoid surprises from implicit "magic"

Naming etc.: Don't think about it too long!

Better **improve little by little**

„Make it work, make it beautiful, make it fast“

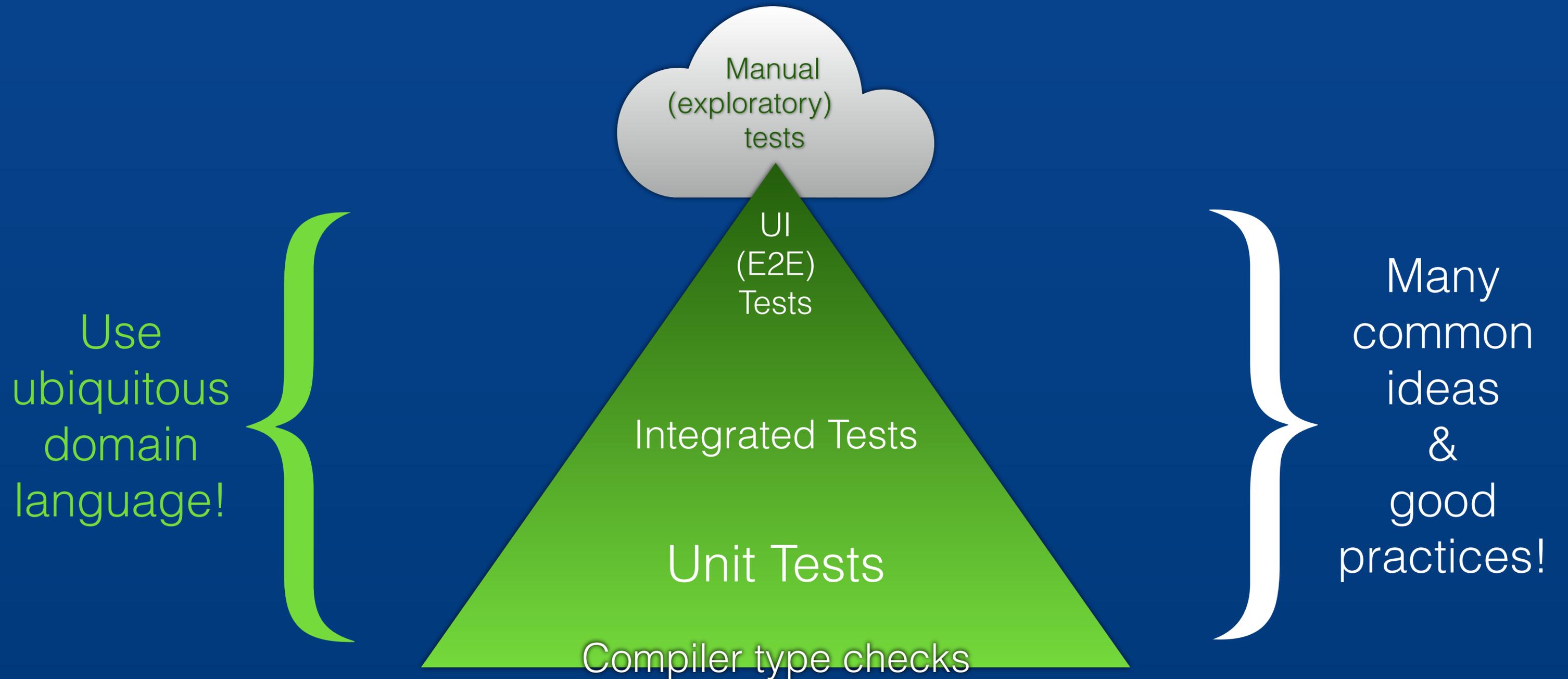


Programming Wisdom
@CodeWisdom

"Make it work, then make it beautiful, then if you really, really have to, make it fast. 90 percent of the time, if you make it beautiful, it will already be fast. So really, just make it beautiful!" – Joe Armstrong

6:15 nachm. · 7. Apr. 2021 · Buffer

Test Pyramid (Reprise)



Not "stupid"
Surely **clean!**

Optimized for
reading

Without
"magic"

Without surprises

Write. Boring. Code.

Even in tests.

Good Tests

Test Pyramid

Stability

Naming

Given-When-Then

Readability

Maintainability



TDD

Test First

Unit Tests

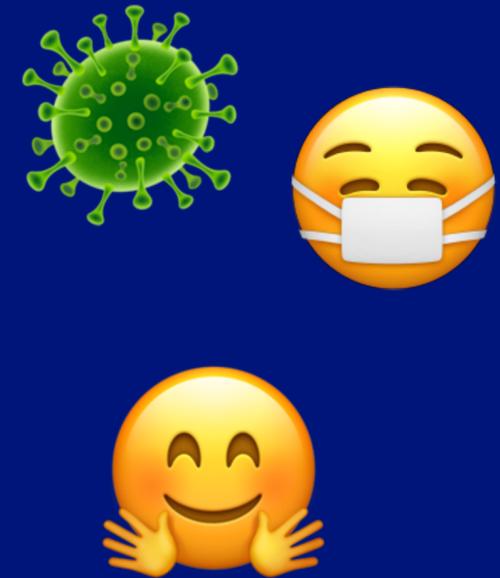
JUnit

Test data

Code Coverage

Questions?

The logo for Jax London features the word "Jax" in a white, stylized font with three yellow dots connected by lines to its left. To the right of "Jax" is the word "LONDON" in a bold, yellow, italicized sans-serif font.



Thank you!

Thomas Much
 @thmuch

Further Reading

- <http://agileinaflash.blogspot.com/2009/02/first.html>
- <https://docs.microsoft.com/en-us/dotnet/core/testing/unit-testing-best-practices>
- Kent Beck, "Test-Driven Development by Example", 2002
- Steve Freeman, Nat Pryce, "Growing Object-Oriented Software, Guided by Tests", 2009
- Michael C. Feathers, "Working Effectively with Legacy Code", 2013
- Emily Bache, "Mocks, Fakes and Stubs", 2013