TDD & Kata & You

Women Who Code (Feb 2017)

Cynthia Wu
(t) @wildstyle_cwu
(e) wu.x.cynthia@gmail.com

Agenda

- What is TDD? – Why should I care?
- A word of caution...
- What is Kata?
- Interactive Hands-on Example
- Where do I go from here?

TDD

Test Driven Development

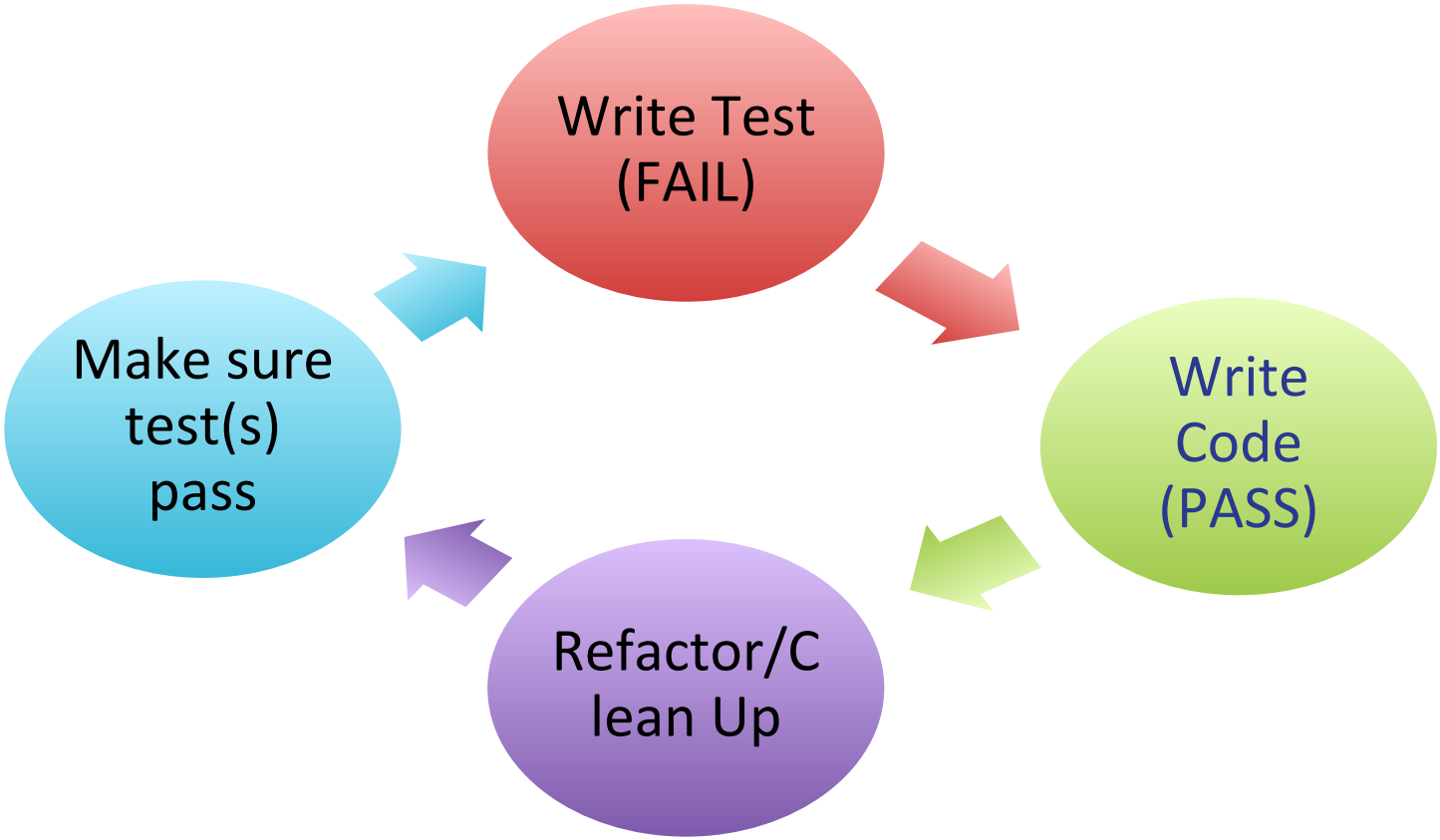
Goal:

Create simple unit-testable code
(*100% code coverage)

The Rules:

1. Create a test
2. Write the minimum amount of code to make the test pass
3. Refactor/Clean-up code
4. Run test(s) again to make sure all is green

TDD Lifecycle



Why Should I Care?

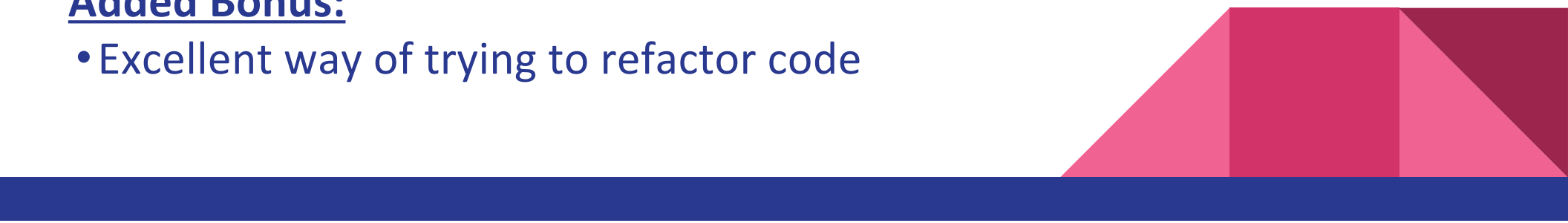
- Helps focus on what you “really” need out of your code
- Reduces debug time – your tests are already there
- Your tests will naturally outline the behavior of your code; unofficial documentation
- Overall code quality improves

➢ Easier to troubleshoot

➢ Helps ensure that new code doesn’t break existing features

Added Bonus:

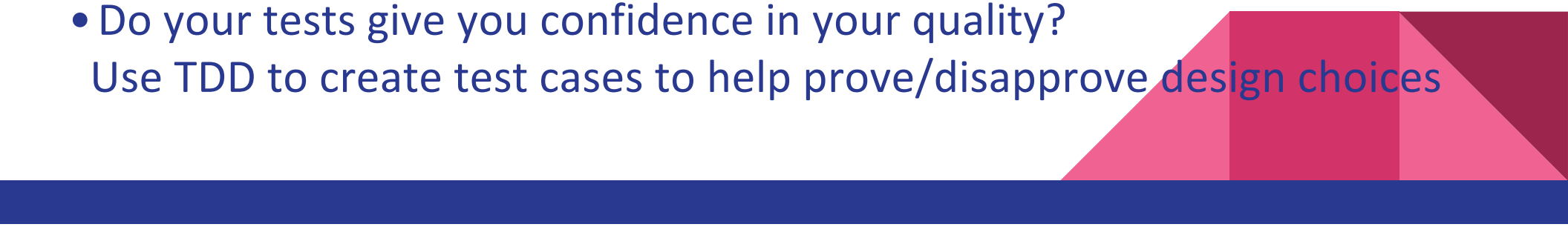
- Excellent way of trying to refactor code



A Word of Caution...

- Easy to fall into a rabbit hole of never-ending testing
  *(You're never done testing)
- Focus unit tests on **high** and **impactful** areas
  - Methods that have integration risks
  - Methods that are often used (central to processing)
  - POJOs or Objects with Getters/Setters may not add too much value
  - 100% code coverage != 100% quality

- Do your tests give you confidence in your quality?
  Use TDD to create test cases to help prove/disapprove design choices



What is Kata?

- Kata – Japanese for a series of movements & flow
- Katas are small exercises designed to help train your brain to think in TDD

Kata Activity:

Let's work together on the Calculator Kata Example

- http://osherove.com/tdd-kata-1/ << Today's exercise
- Git site: https://github.com/cynthisizer/tdd-exercises

Where Do I Go From Here? (30 mins/week)

- Another Kata: http://www.jamesshore.com/Blog/Lets-Play/
- Try other languages/exercises
- Use more advanced libraries (Exceptions, Design Patterns, Mocks)
- Start integrating it into your workflow
- Want to learn something new - try using TDD
- Integrate BDD with TDD

Additional Resources

More information about TDD:

- http://coderetreat.org/facilitating/activities/tdd-as-if-you-meant-it

TDD with BDD

- https://presentationtier.wordpress.com/2012/11/16/doing-tdd-bdd-style/

Some other helpful readings:

- Pragmatic Programmer (Andy Hunt and Dave Thomas)
- Clean Code (Robert C. Martin)
- Design Patterns [for your desired language]
    - Code reusability
    - Learn concepts by proof (TDD/BDD)