

Test-Driven Development with Predictions

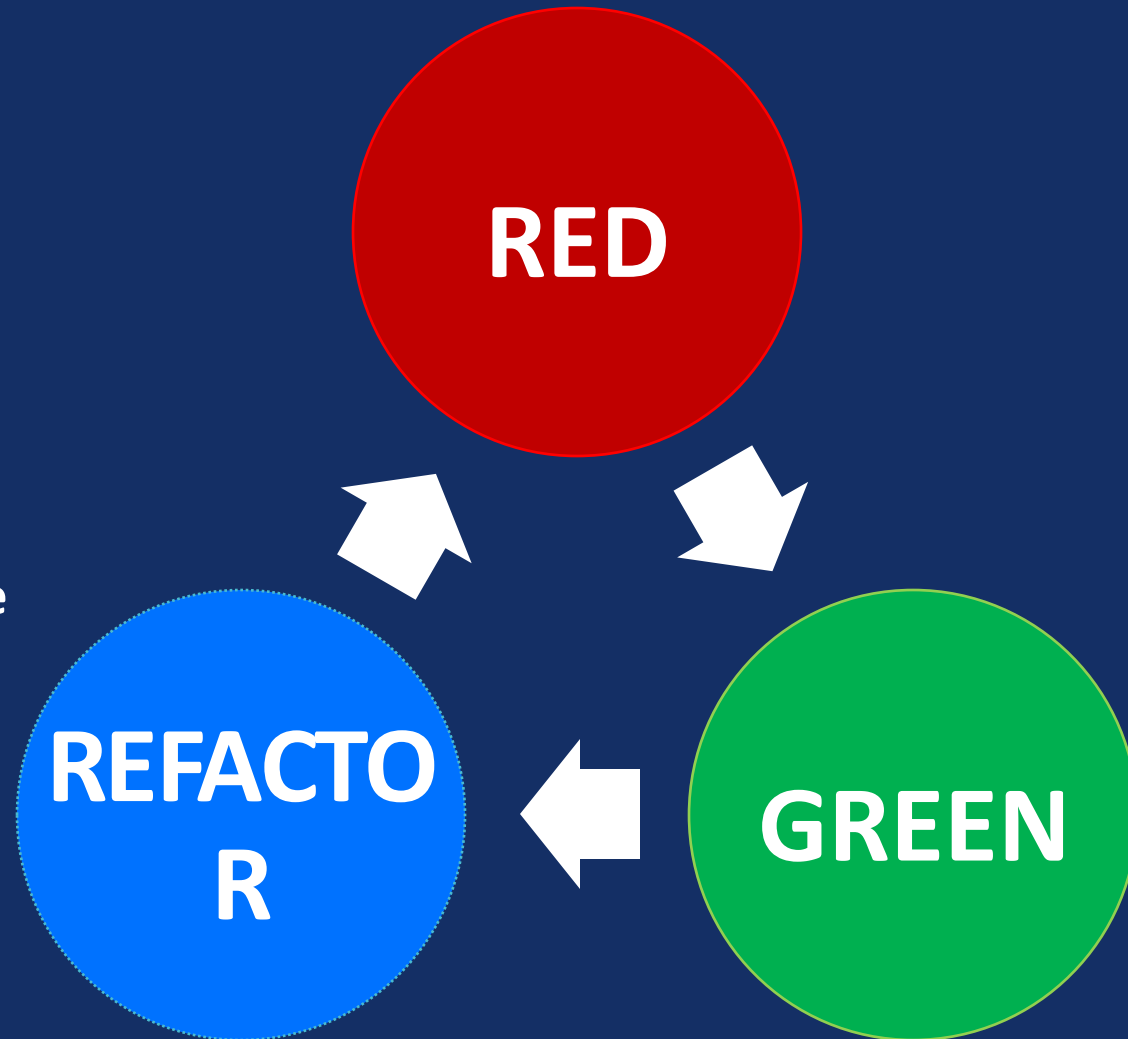
An Introduction to the
PREDICTIVE TDD PROCESS

Ted M. Young
<https://ted.dev>

<https://tddga.me>
@jitterted

TDD Cycle

1. Think and write next specification (as a test)
2. It should Fail (Red)
3. Write just enough code to pass (Green)
4. Refactor (tests still pass) until happy
5. Repeat from 1

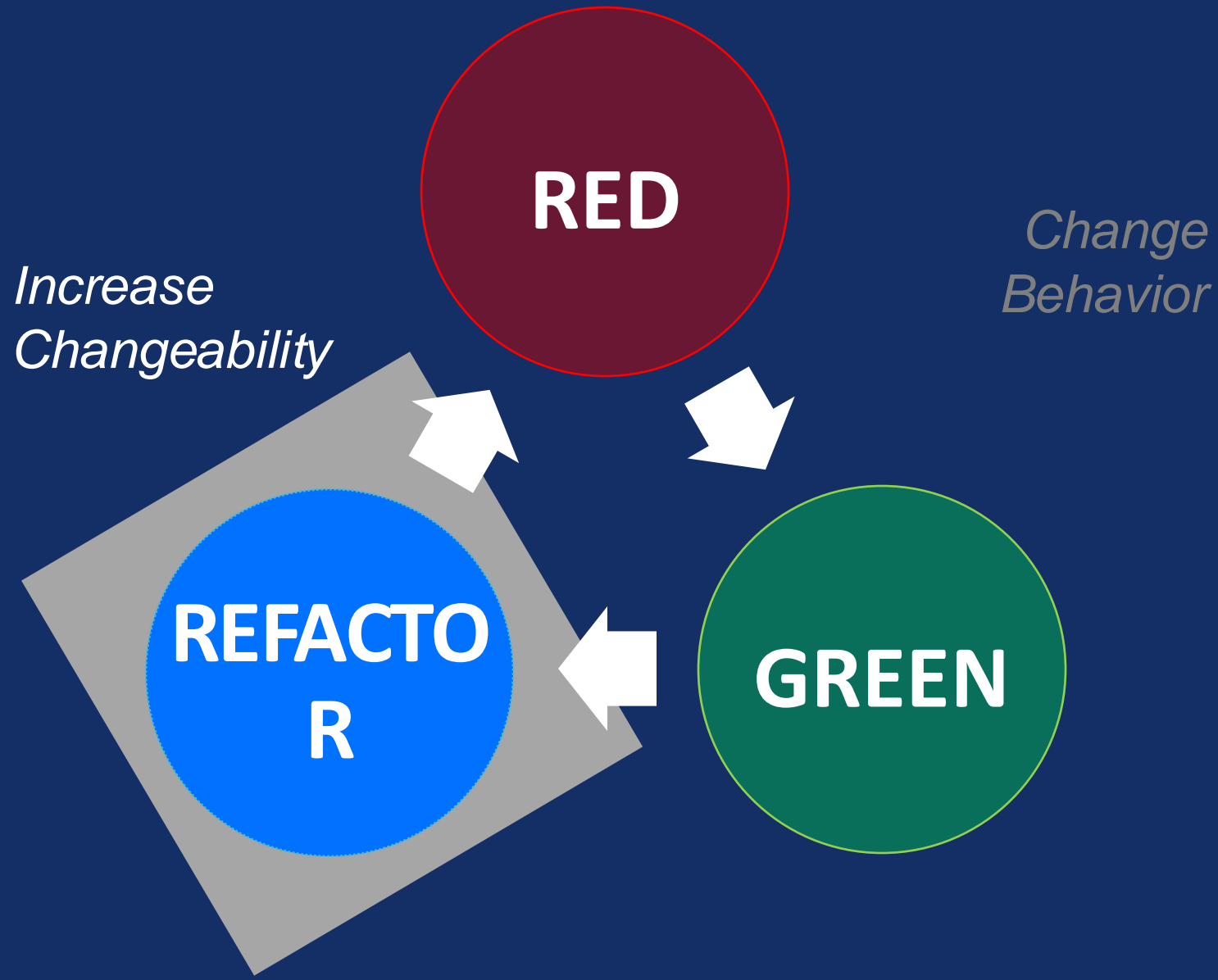


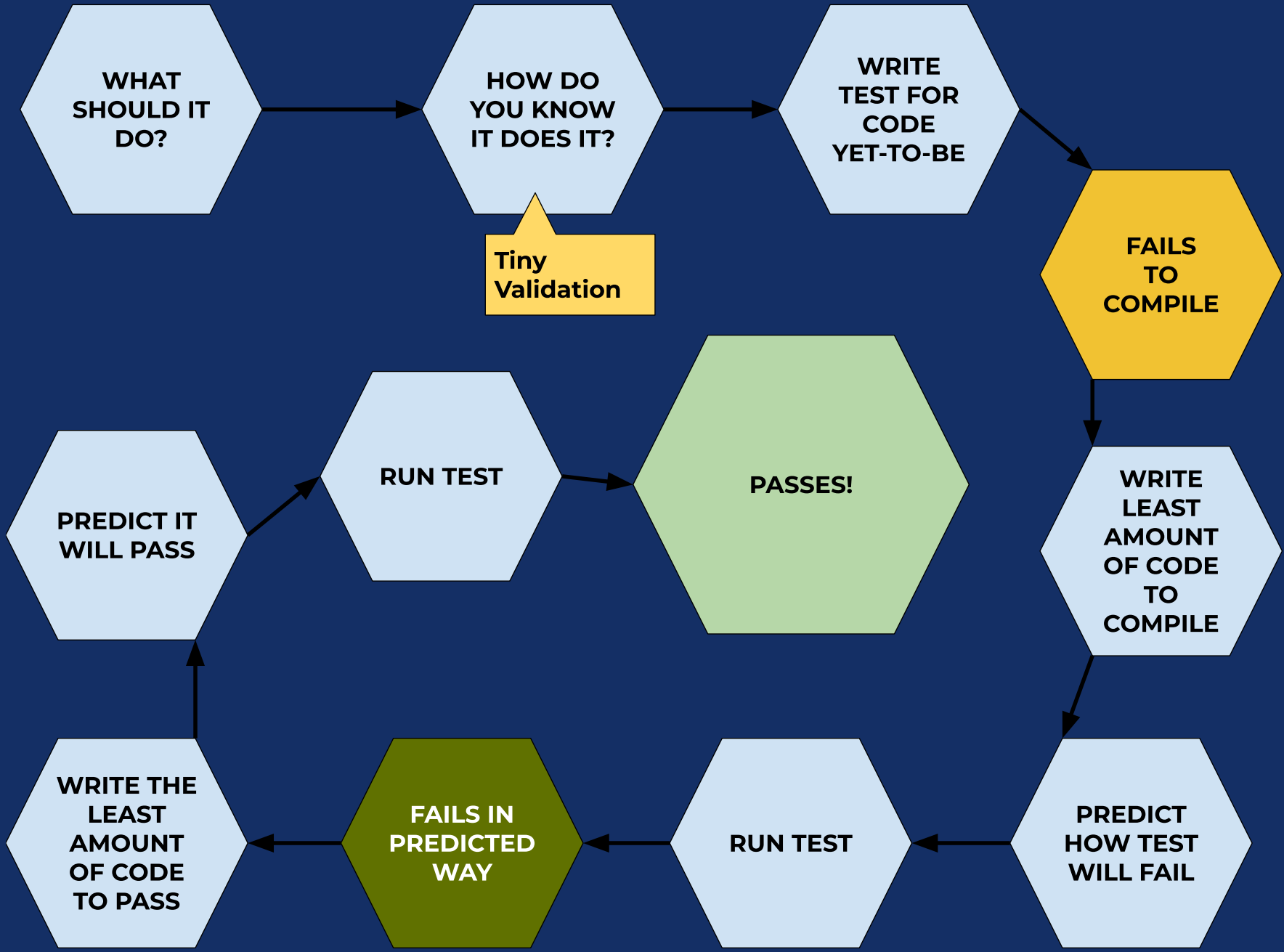
*Increase
Changeability*



*Change
Behavior*







WHAT SHOULD IT DO?

HOW DO YOU KNOW IT DOES IT?

WRITE TEST FOR CODE YET-TO-BE

Tiny Validation

FAILS TO COMPILE

PREDICT IT WILL PASS

RUN TEST

PASSES!

WRITE LEAST AMOUNT OF CODE TO COMPILE

WRITE THE LEAST AMOUNT OF CODE TO PASS

FAILS IN PREDICTED WAY

RUN TEST

PREDICT HOW TEST WILL FAIL

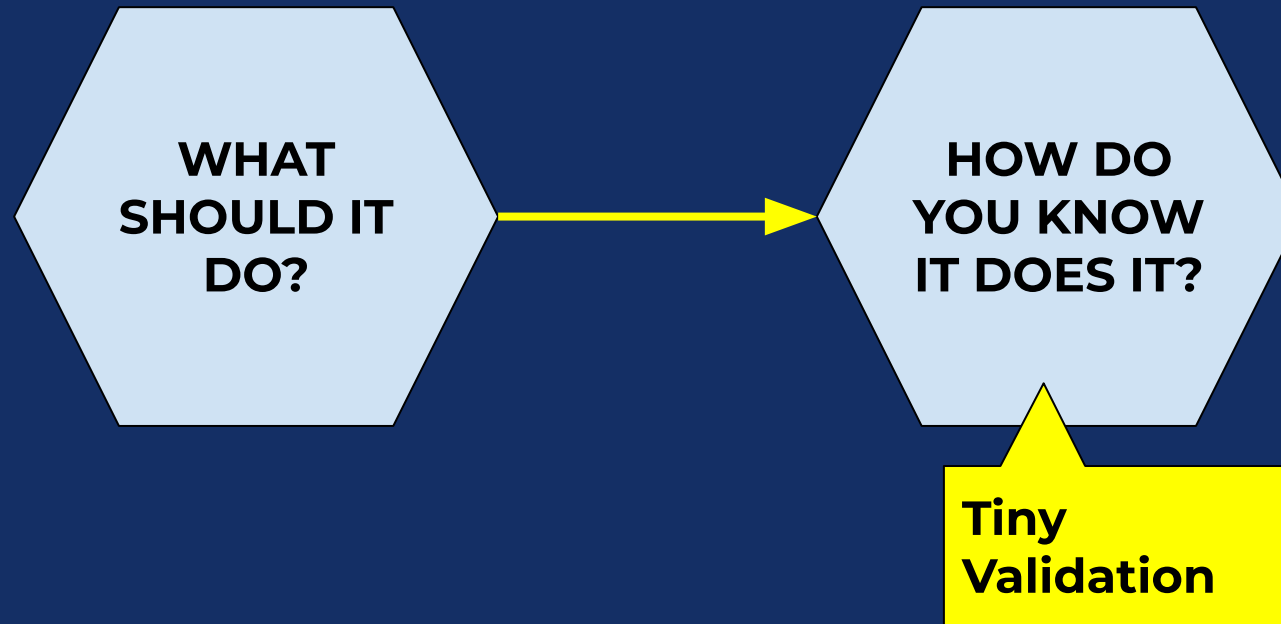
Buy at

<https://tddga.me>

Stay tuned for
\$10 off coupon



1: The Hard Part (Think!)



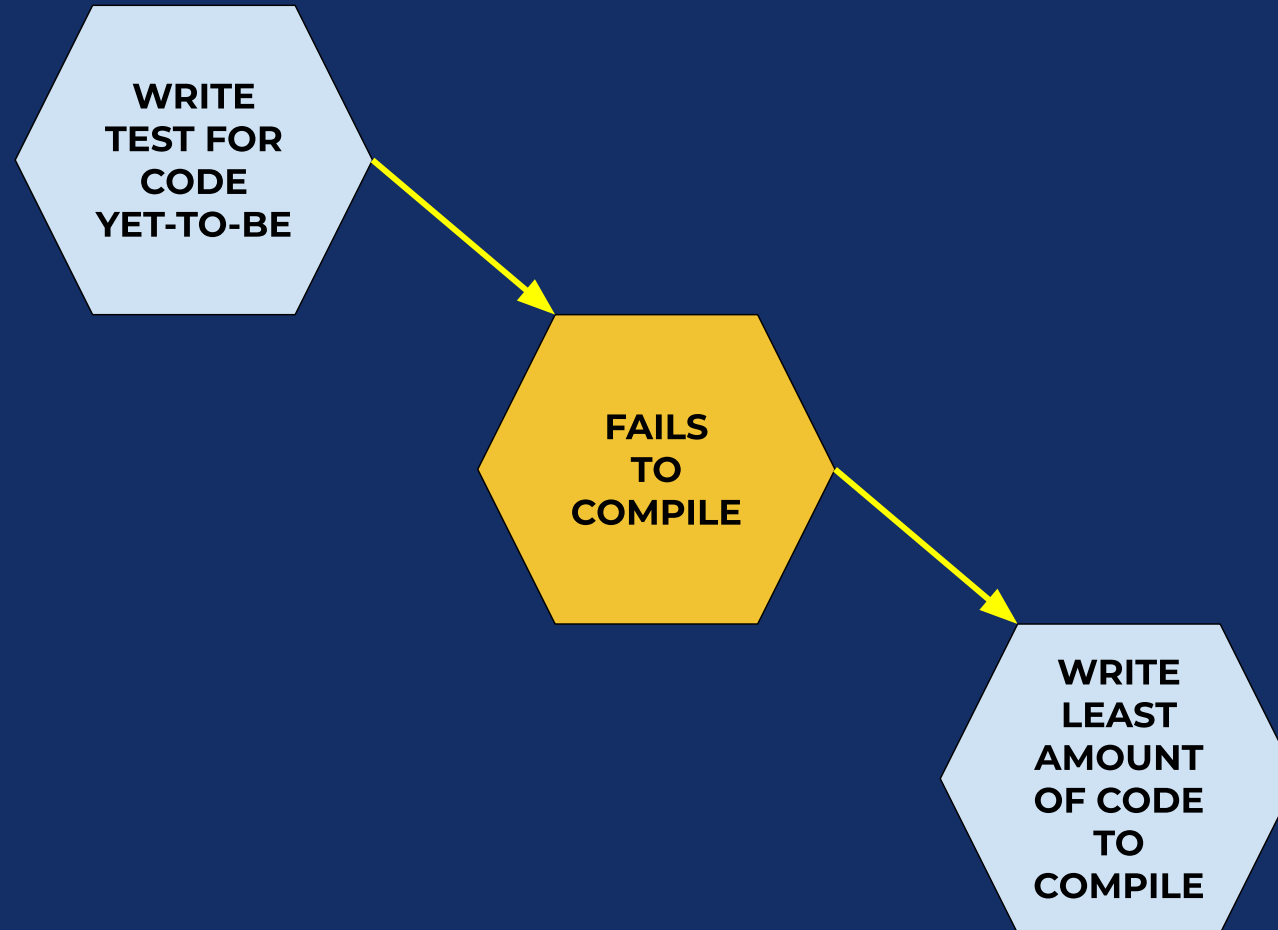
No Behavior Changes to Production Code without a Failing Test

TDD Principle #1

Test Runs & Fails Before Passing

TDD Principle #2

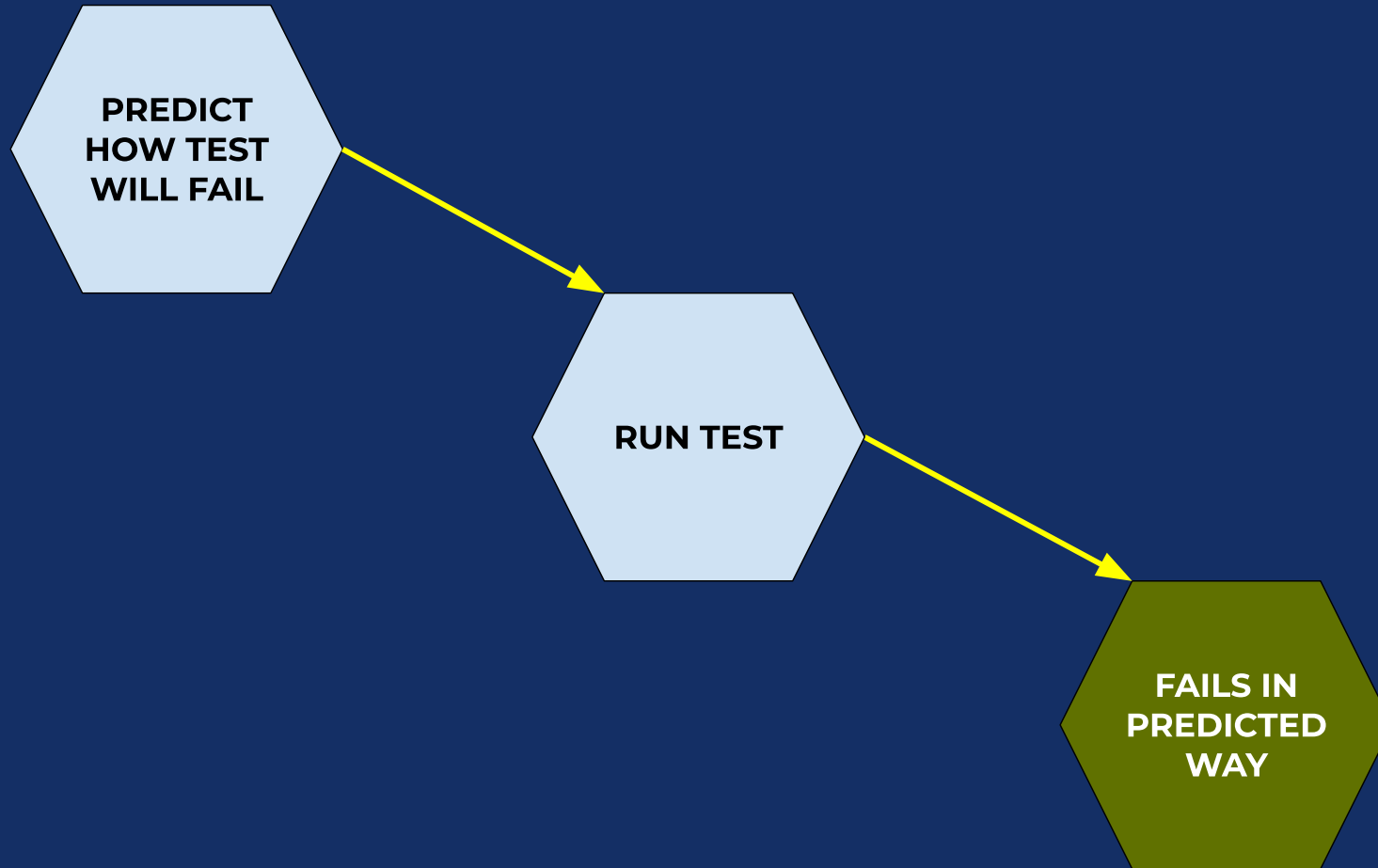
2: Doesn't Compile (optional)



Test Must Fail For Right Reason

TDD Principle #3

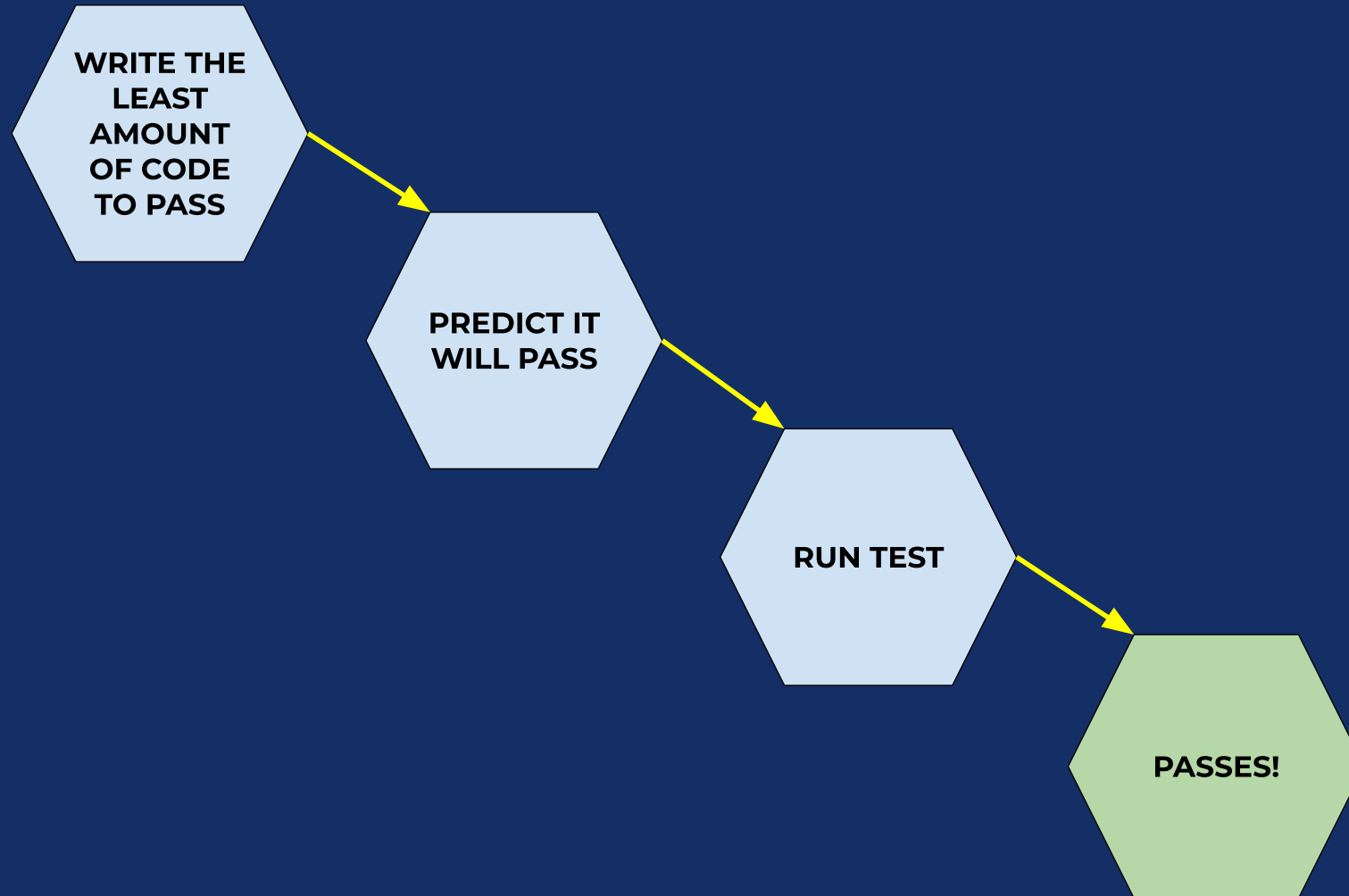
3: Test Fails Predictably



Minimal (Least Effort) Code to Passing Test

TDD Principle #4

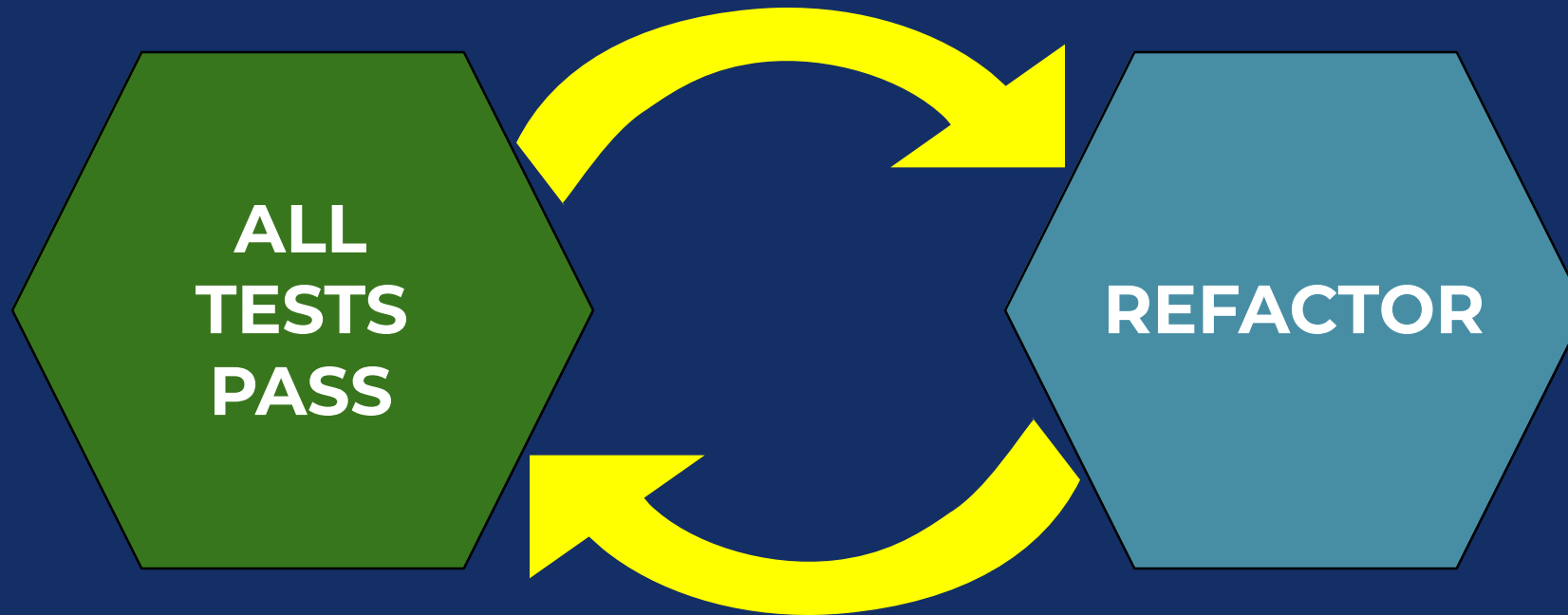
4: Write Production Code



Refactoring: Stay in the Green

TDD Principle #5

5: Refactoring (Don't Forget)



Questions

What Questions Do You Have Right Now?

NEXT: THE GAME

history of the game

JITTERED's TDD GAME



Goal of Game

First to 5 "commits" wins

or

Most commits wins (timeboxed)

The Action Cards

The Board

Multiple Paths: Choose Wisely

Player 1

Playing it Safe

Player 2

Taking a Risk

Player 1

Learned a Lesson

Player 2

Pays the Price

Technical Risk Cards

aka TECH DEBT

Questions

Last Chance for Questions...

Buy at

<https://tddga.me>

\$10 off coupon:

LNSAVE10*

* Discount good through February 29, 2024

That's All Folks!

Thank You.

Ted M. Young

@jitterted