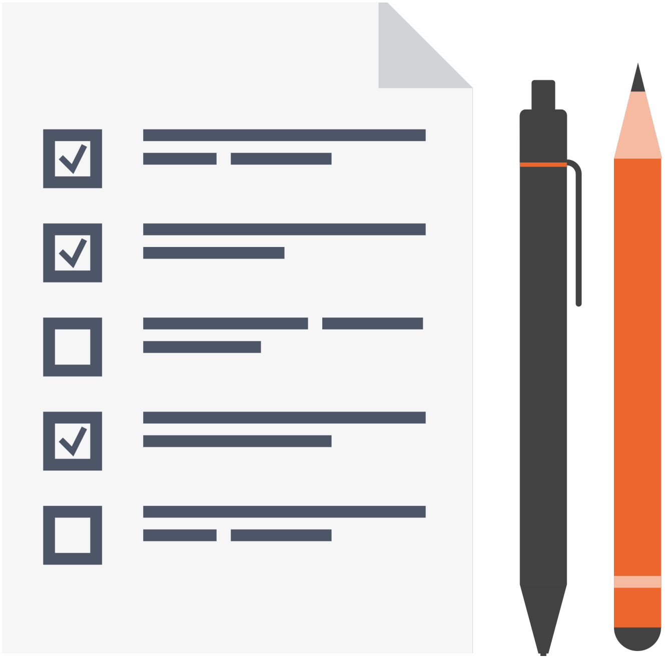
TDD 101

Rem Zolotykh

http://remzolotykh.net | @remzolotykh

Overview



Red - Green - Refactor cycle

Test structure

RSpec 101

Bowling game kata

Types of tests

Red - Green - Refactor

TDD Workflow

“Standard” Way of Doing Things

Understand | Code | Test

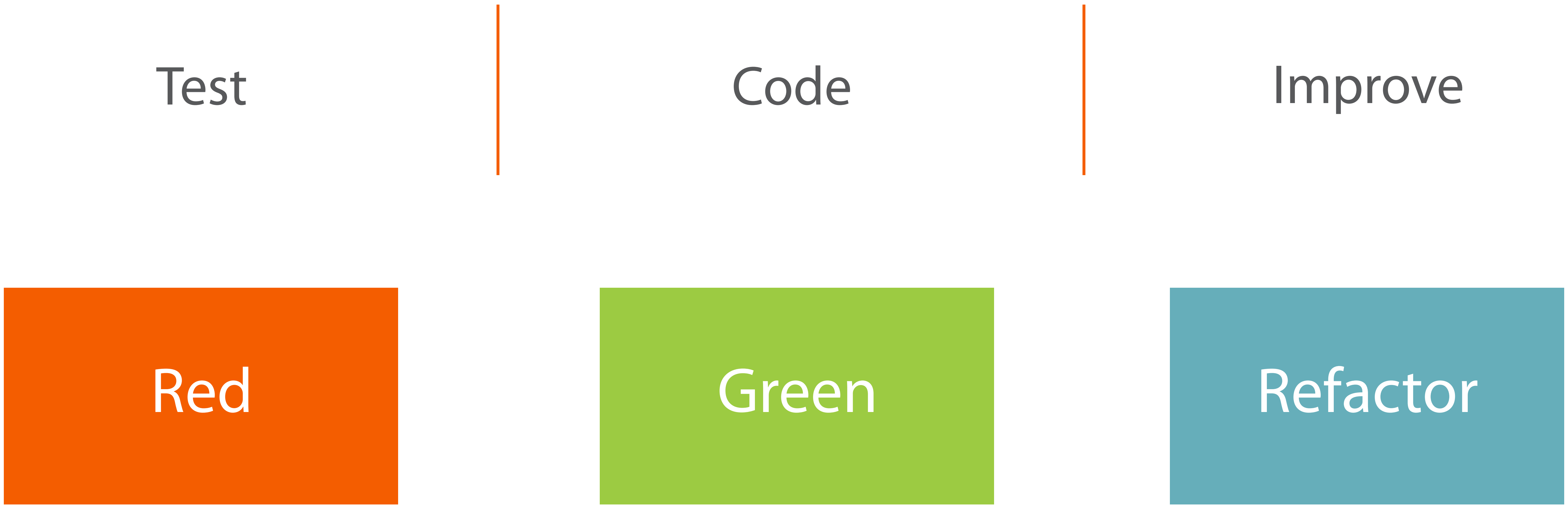
Sum two digits

```ruby
def sum(a, b)
  a + b
end
```

```
puts sum(2,3)

$ ruby sum.rb #=> 5
```

Red - Green - Refactor



Test	Code	Improve
Red	Green	Refactor

Red - Green - Refactor

Red

- Analyze problem
- Focus on public API
- Failing test

```
calc = Calculator.new
calc.sum(1,4)
```

Green

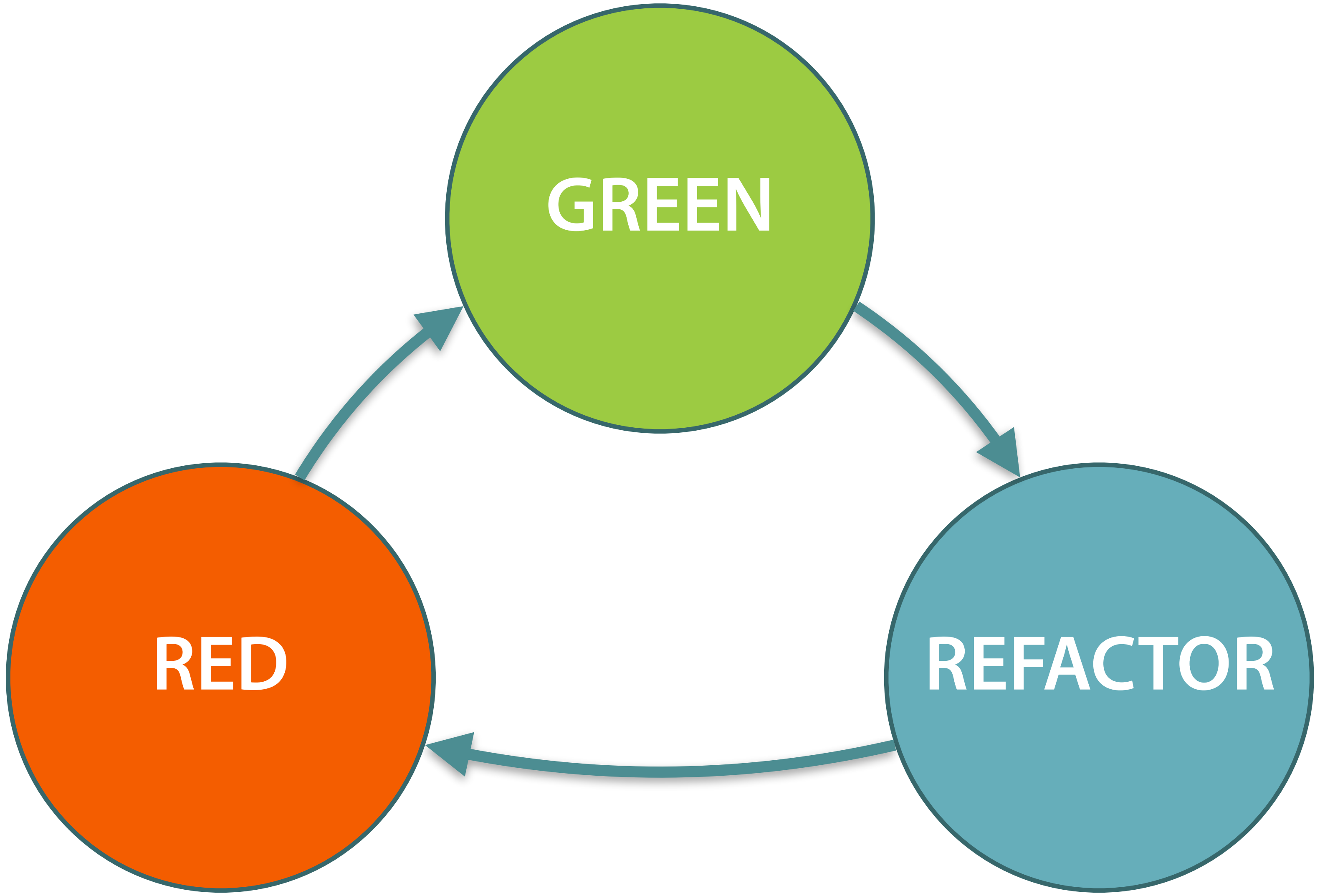
- Write just enough code to make test pass

```
calc = Calculator.new(1,4)
calc.sum
```

Refactor

- Improve code design
- **Tip**: Always refactor in green

Red - Green - Refactor



Red - Green - Refactor

Demo

Overview

- String calculator
  - ‘1,2,3,4,5’ => 15
- Use MiniTest

Test Structure

AAA

AAA

Arrange

```
calc = Calculator.new(‘1,2’)
```

Act

```
result = calc.sum
```

Assert

```
result.must_equal(3)
```

```
calc.sum.must_equal(3)     #=> Minitest
expect(calc.sum).to eq(3)  #=> RSpec
```

RSpec 101

Demo

Bowling Game Kata

Rules and Requirements

What Is a Code Kata?

- It’s an exercise you work on multiple times
- Each time you focus on different aspects
  - Different solution
  - Getting better at tools
  - New programming language
  - Use another paradigm

Public API

```
game = BowlingGame.new

susie_game = [1,4,6,4,5,5,10,0,1,7,3,6,4,10,2,8,6]

peter_game = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]


game.rolls(susie_game).score   #=> 133

game.rolls(peter_game).score   #=> 0
```

Susie’s Game

1	2	3	4	5	6	7	8	9	10
1 4	4 5	6 ◢	5 ◢	■	0 1	7 ◢	6 ◢	■	2 ◢ 6
5	14	29	49	60	61	77	97	117	133

SPARE
(+ 1 next roll)

STRIKE
(+ 2 next rolls)

Possible Games

Gutter game
(0 points)

Normal game
(like Susie’s one)

Perfect game
(300 points)

Try It Out!

Solve by yourself before venturing forth

Stop watching this course

DO IT!

Bowling Game Kata

Demo

Bowling Game Kata Review

Lessons Learned

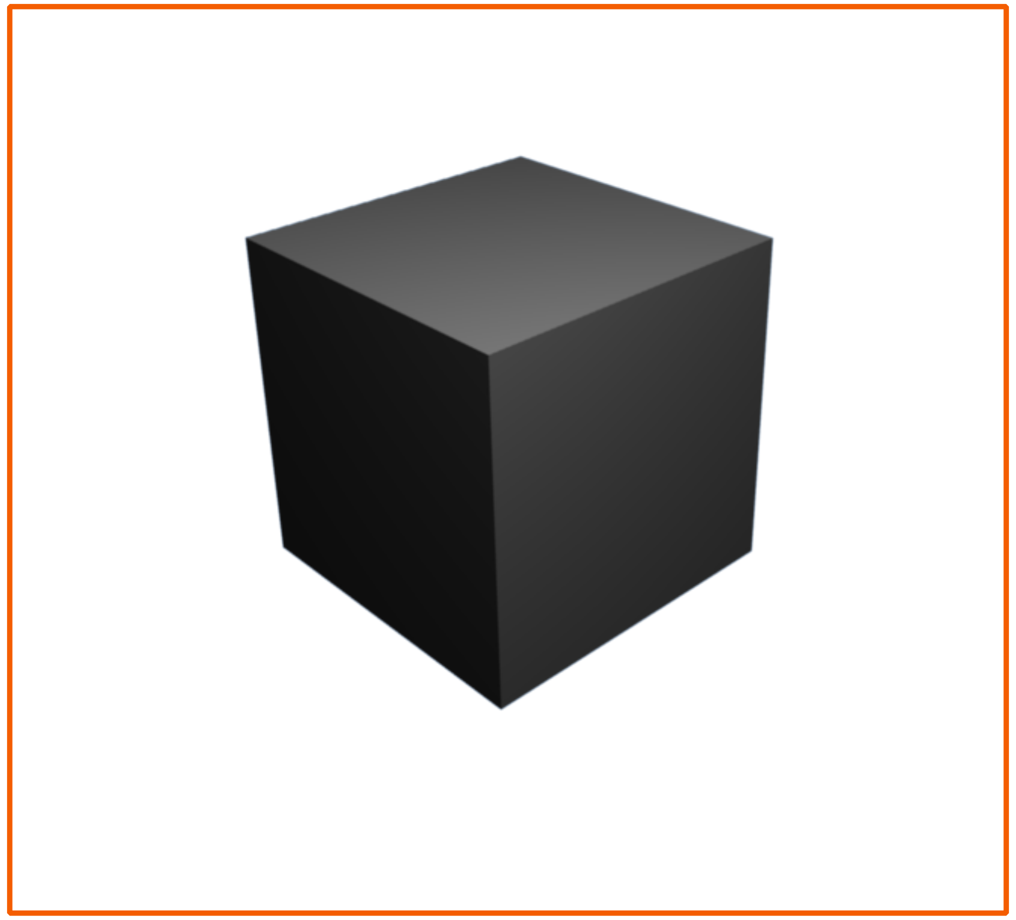
Tips

- Start from the easiest example
  - Clean slate state
- Add new examples and extend functionality
- Refactor
  - Start and finish “in green”
- Keep examples small with one assertion per example
- Keep tests readable and clear

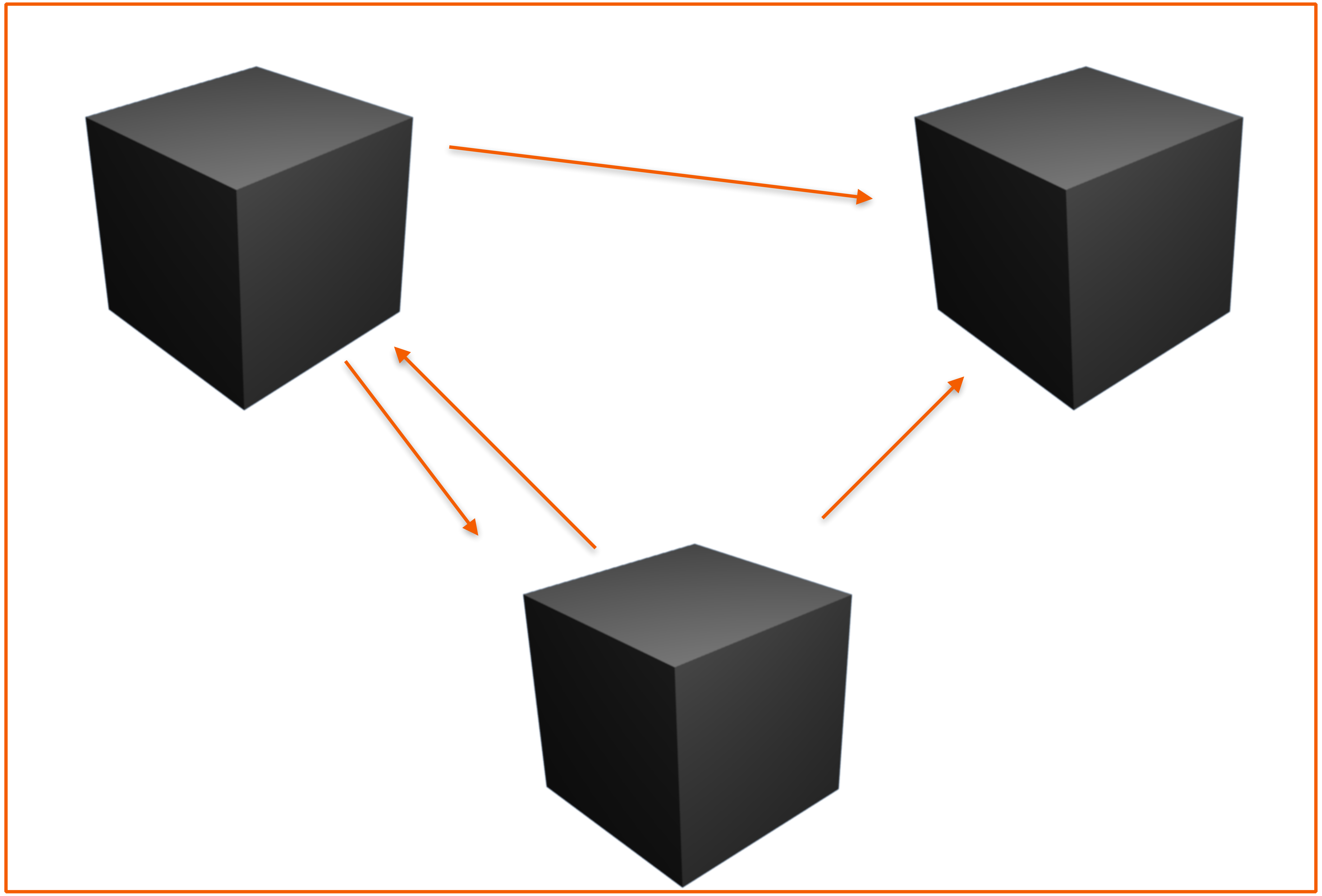
Different Types of Tests

Unit, Integration, and Acceptance Tests

Unit Tests

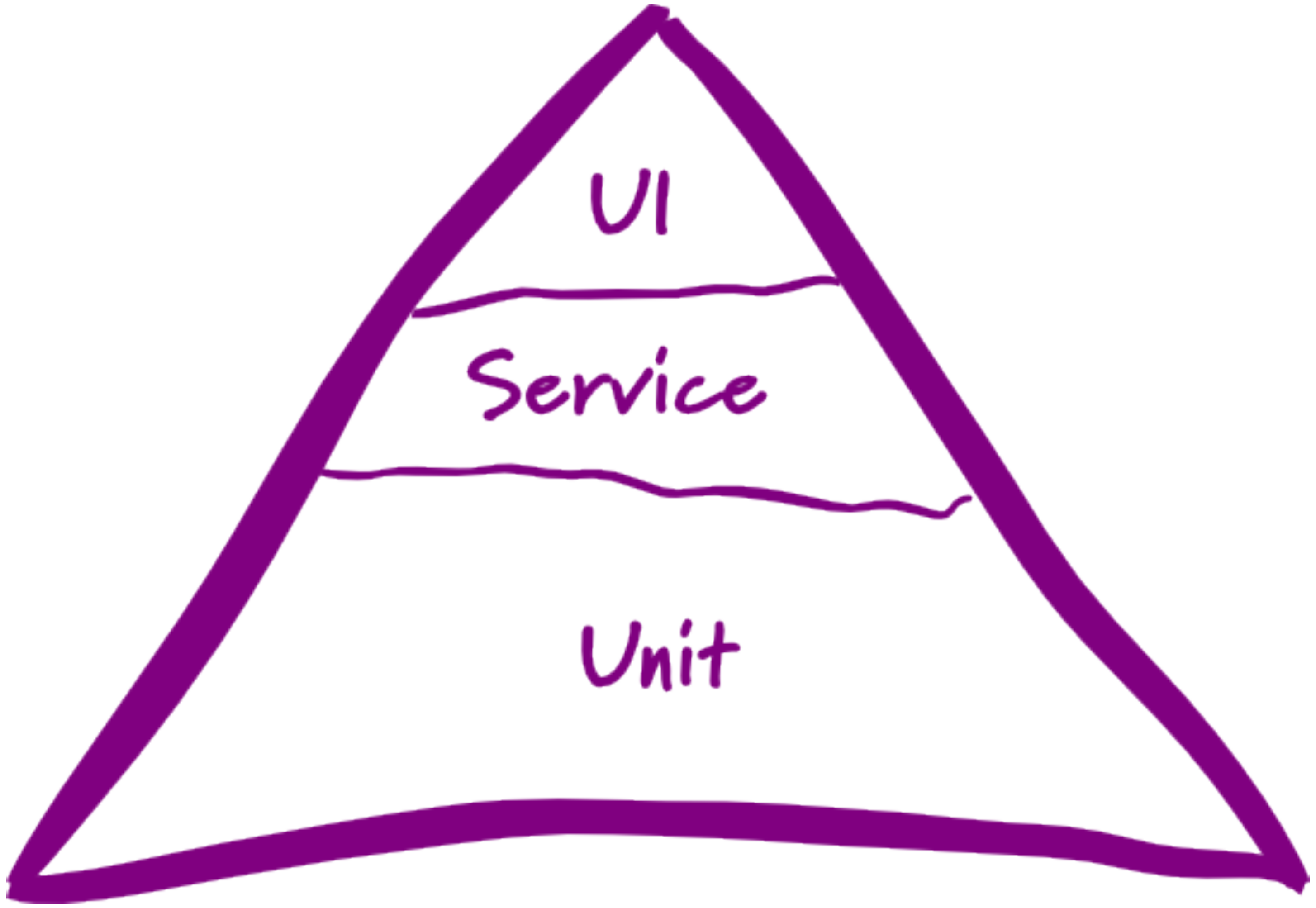


Integration Tests

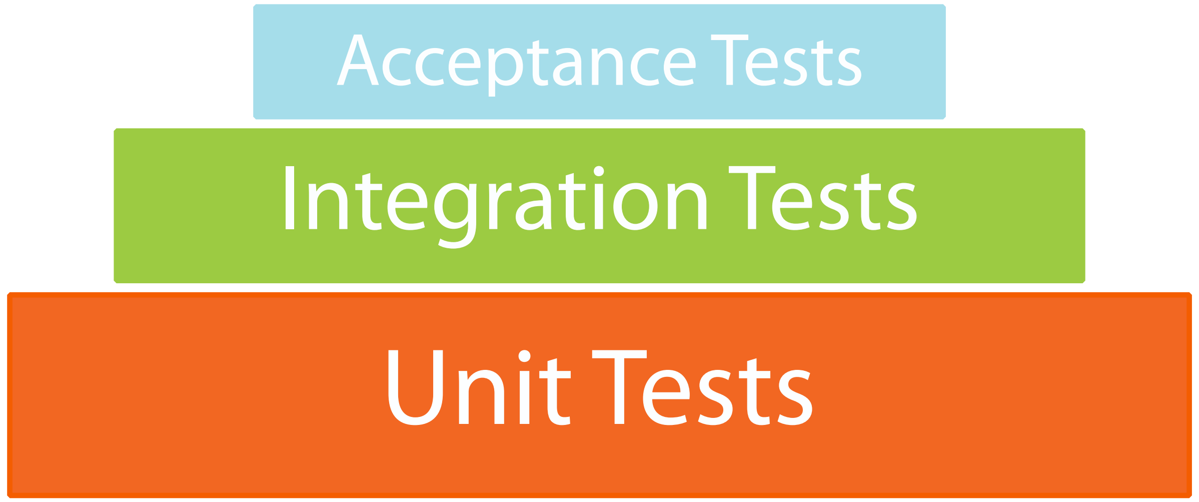


Acceptance Tests

Test Pyramid



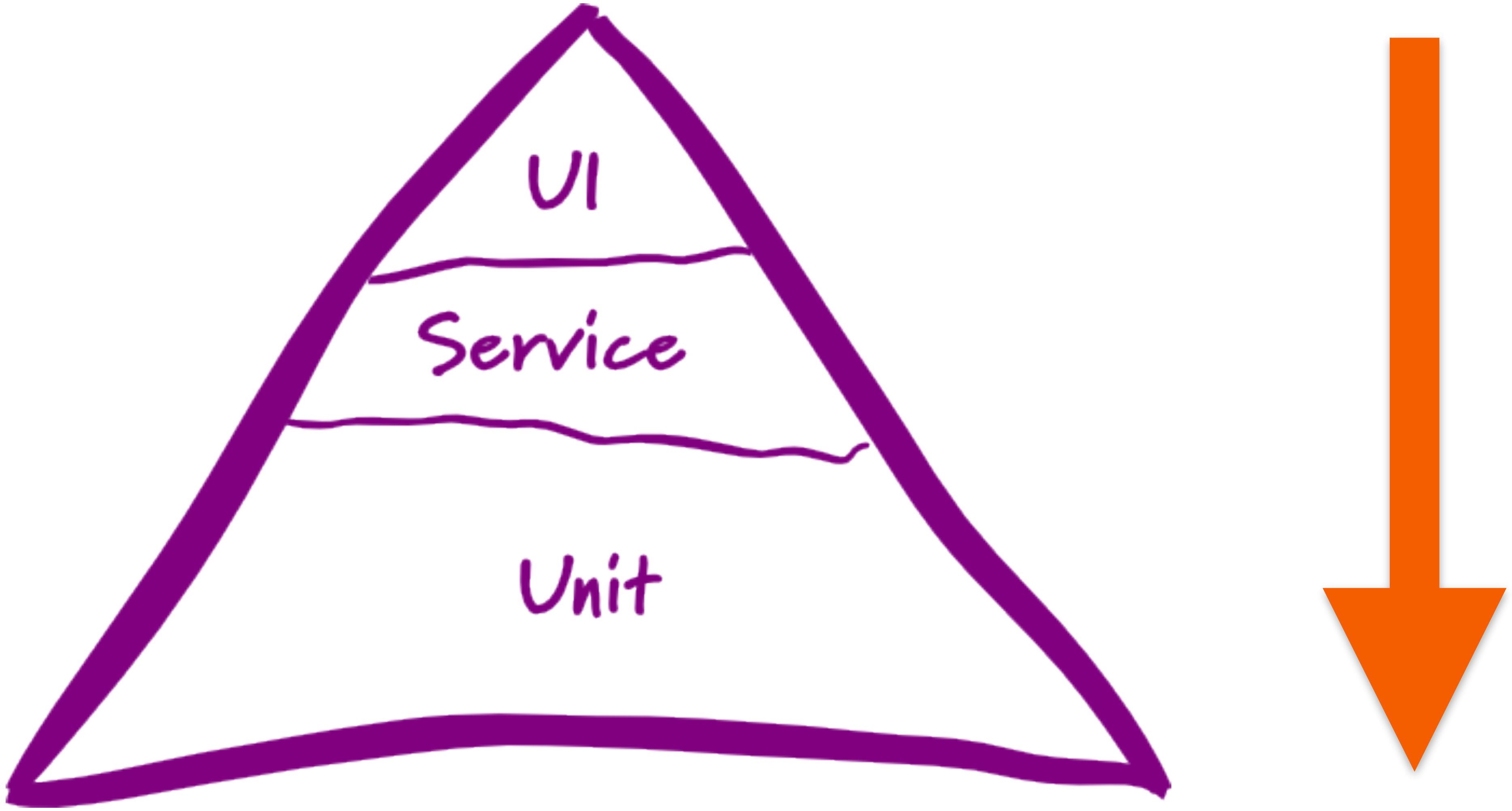
http://martinfowler.com/bliki/TestPyramid.html


Acceptance Tests

Integration Tests

Unit Tests

Outside-in Development



http://martinfowler.com/bliki/TestPyramid.html

Summary

TDD Red - Green - Refactor

RSpec 101

Bowling game code kata

Types of tests