Java: Refactoring to Design Patterns

WHY AND WHEN REFACTOR TO PATTERNS

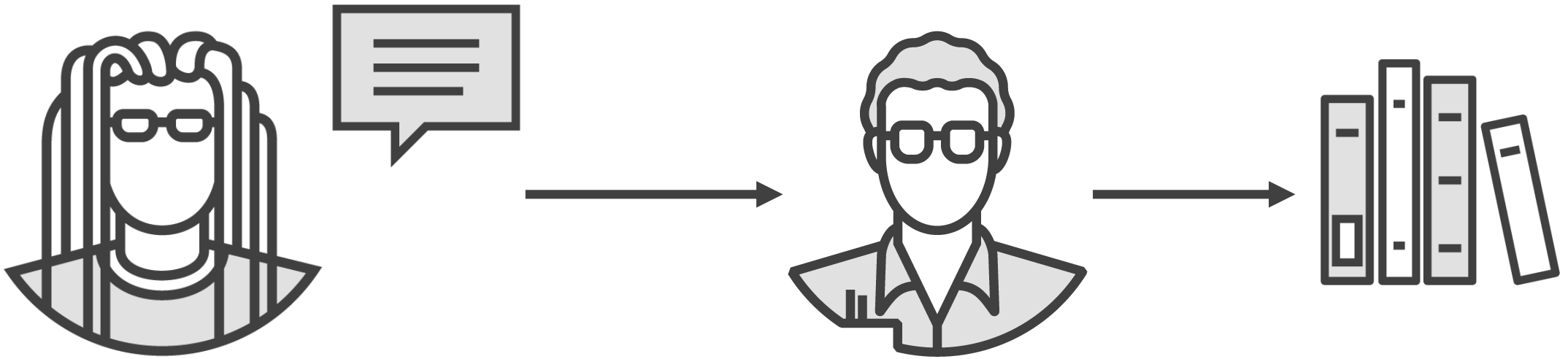
Andrejs Doronins
TEST AUTOMATION ENGINEER

Refactoring Techniques

SOLID Principles

Clean Code Principles

OO Design Patterns!



Challenges When Learning Design Patterns

Where do I start?

- Dozens of patterns

I still don't get it...

- Toy examples: dogs, cats, birds

Erich Gamma

Richard Helm

GoF Design Patterns

Ralph Johnson

John Vlissides

And many more
DTO
DAO
Null Object
Static Factory Method
...

1

How often do we create objects?

2

```
new SomeObject();
```

3

Creational

Behavioral

Structural

1 Creational

Implementing business logic?

2

if(){ } else { }; switch();

3

Behavioral

Structural

Singleton

Builder

Factory

Abstract Factory

Prototype

State

Command

Interpreter

Iterator

Strategy

Memento

Bridge

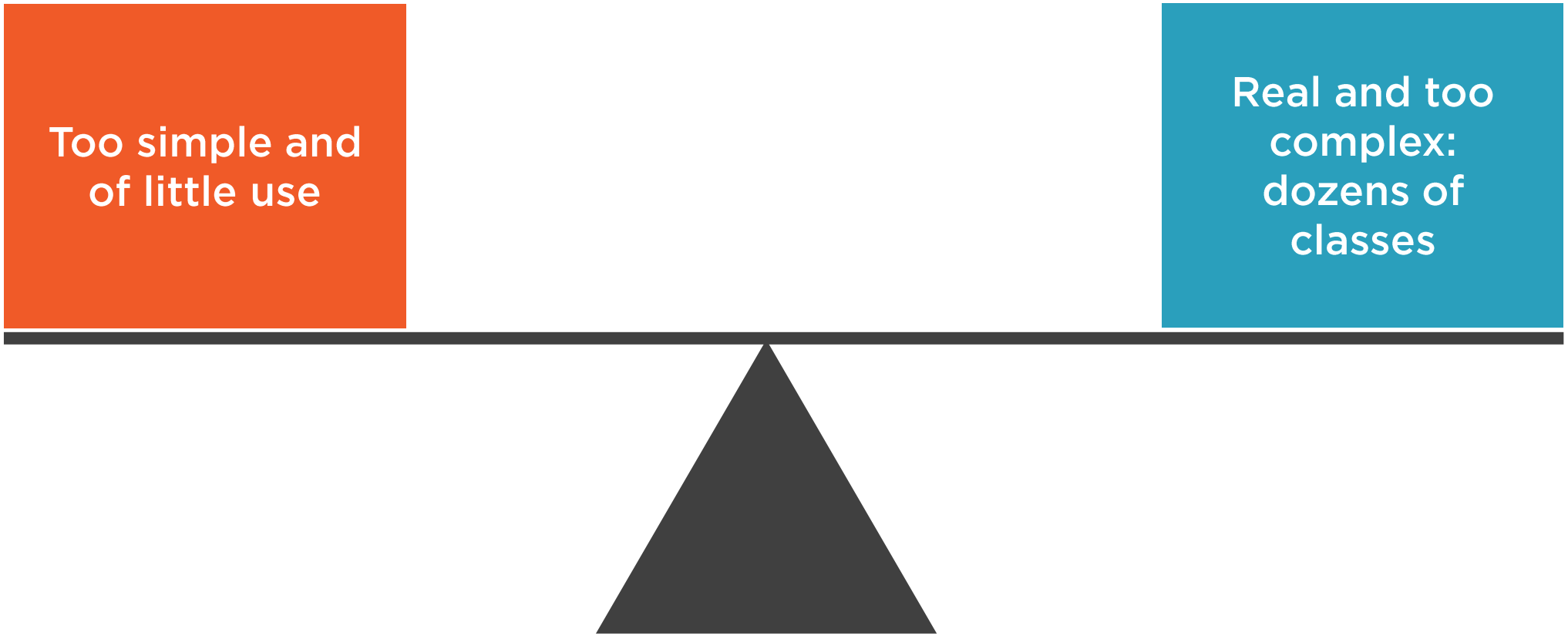
Adapter

Decorator

Composite

Proxy

Facade


Too simple and of little use
Real and too complex: dozens of classes

Text processor with dictionary functionality

Prerequisites

Java

- Encapsulation, Inheritance and Polymorphism

Any IDE

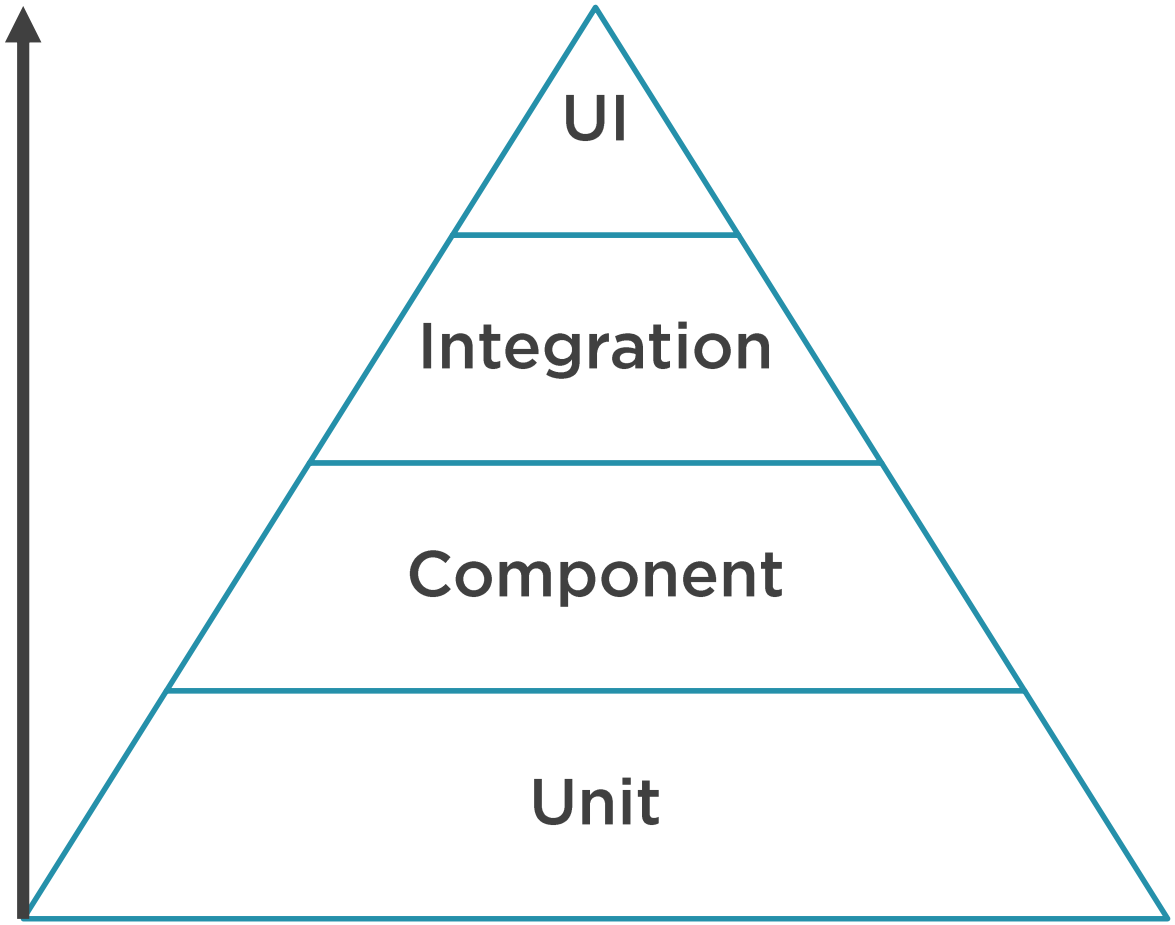
- e.g. IntelliJ

Basics of Java 8 streams and lambdas

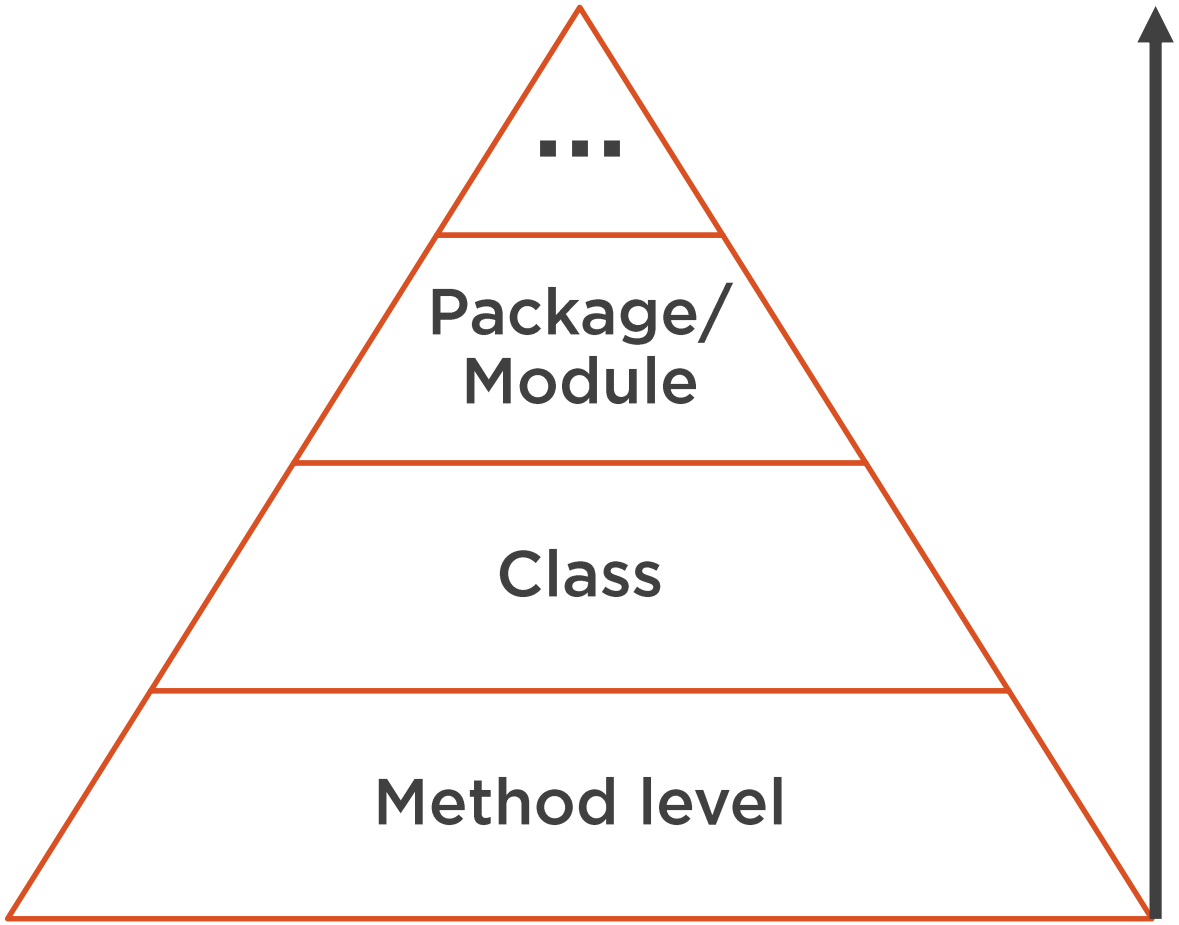
Basic clean code principles

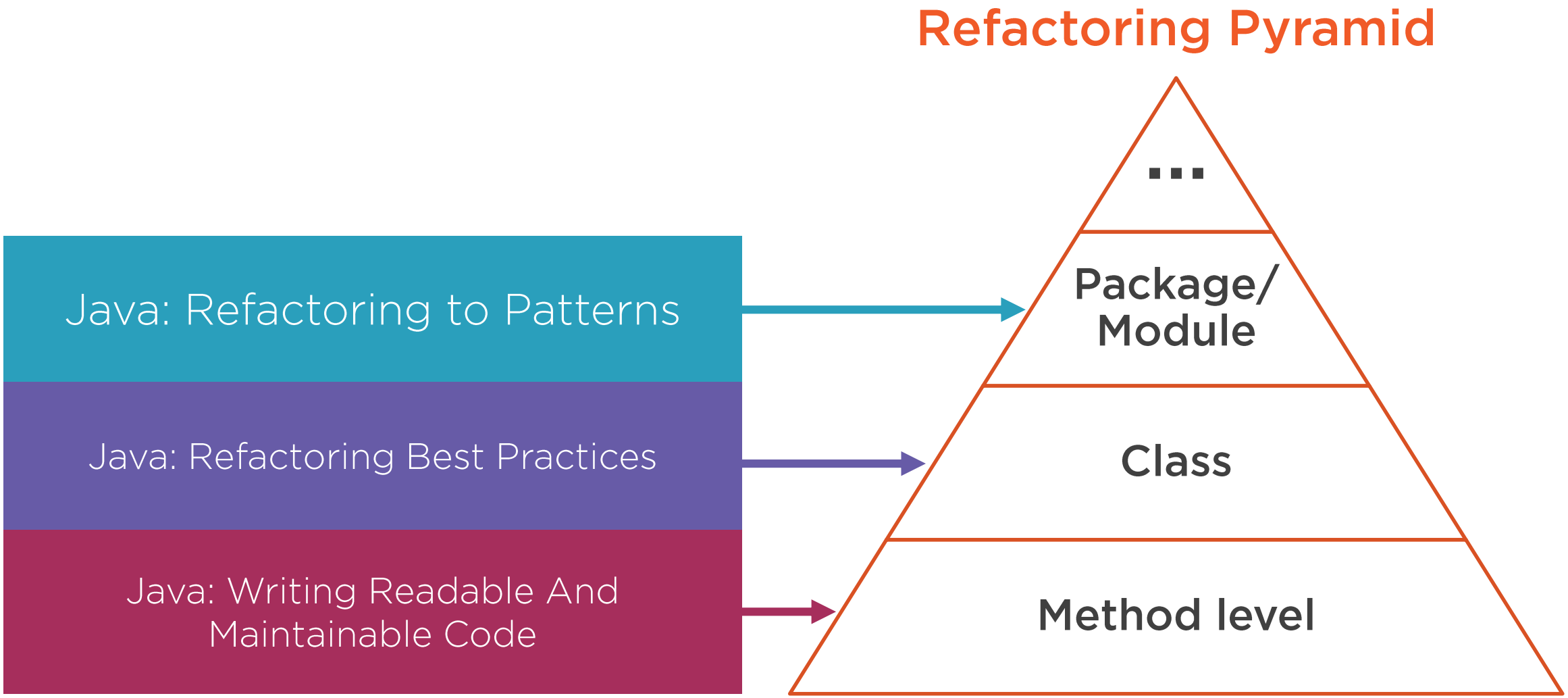
Basic knowledge of refactoring

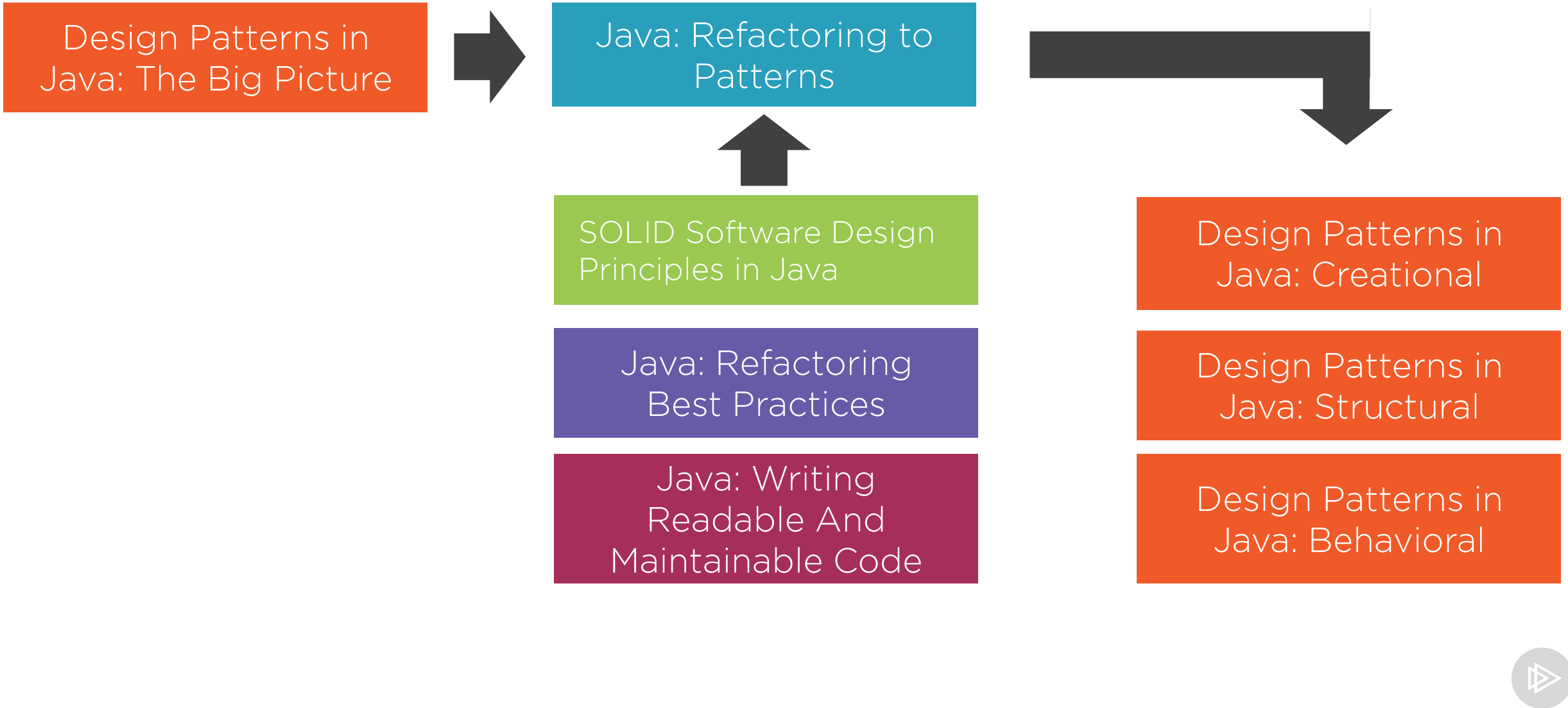
Testing Pyramid



Refactoring Pyramid




Refactoring Pyramid
Java: Refactoring to Patterns
Java: Refactoring Best Practices
Java: Writing Readable And Maintainable Code
...
Package/ Module
Class
Method level


Design Patterns in Java: The Big Picture
Java: Refactoring to Patterns
SOLID Software Design Principles in Java
Java: Refactoring Best Practices
Java: Writing Readable And Maintainable Code
Design Patterns in Java: Creational
Design Patterns in Java: Structural
Design Patterns in Java: Behavioral

What This Course Doesn't Do

Give an overview of design patterns

Demo every single design pattern

What This Course Does

Shows how to refactor poor code towards good code with design patterns

Uses the most frequently used design patterns

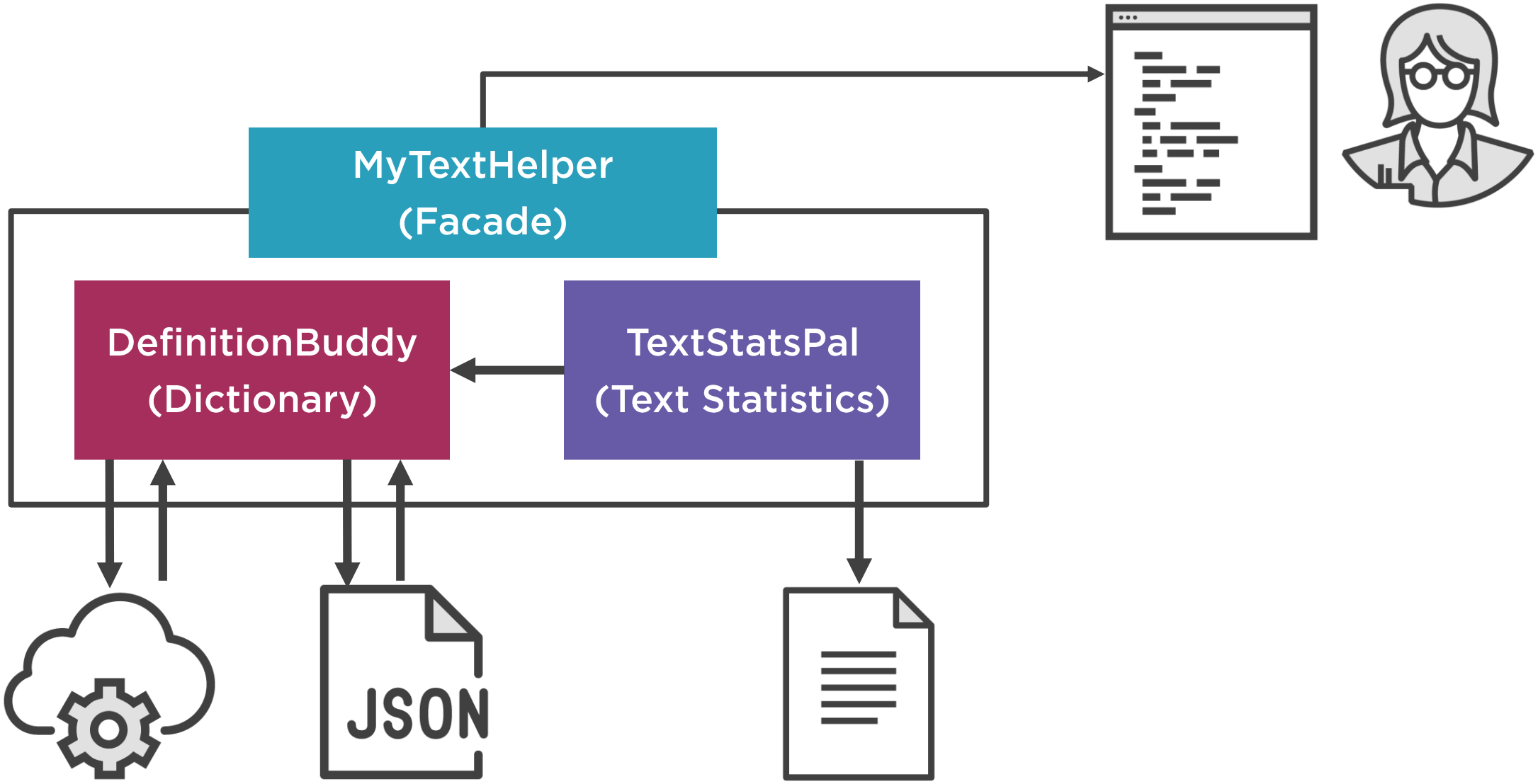
Gives hands-on demo on a real small app

Summary

Better object creation with factories

Eliminating conditional complexity

Improving interfaces with wrappers


MyTextHelper
(Facade)
DefinitionBuddy
(Dictionary)
TextStatsPal
(Text Statistics)
JSON