API Design Consideration In Java

Dr. Ridi Ferdiana

Microsoft MVP – Developer Tech
L: https://mic.ms/ridi
T: @ridife
B: https://ridilabs.net

Microsoft Learn – Free!

Build your skills fast with free, interactive tutorials at Microsoft Learn, a new training experience for technical users.

- Step-by-step training to fit your schedule
- Interactive coding environments for hands-on experience
- Earn achievements and recognition for your Azure skills

Start today at aka.ms/LearnMS

Microsoft

Hi There!

I’m Ridi

I used to become Java Programmer in my first job,

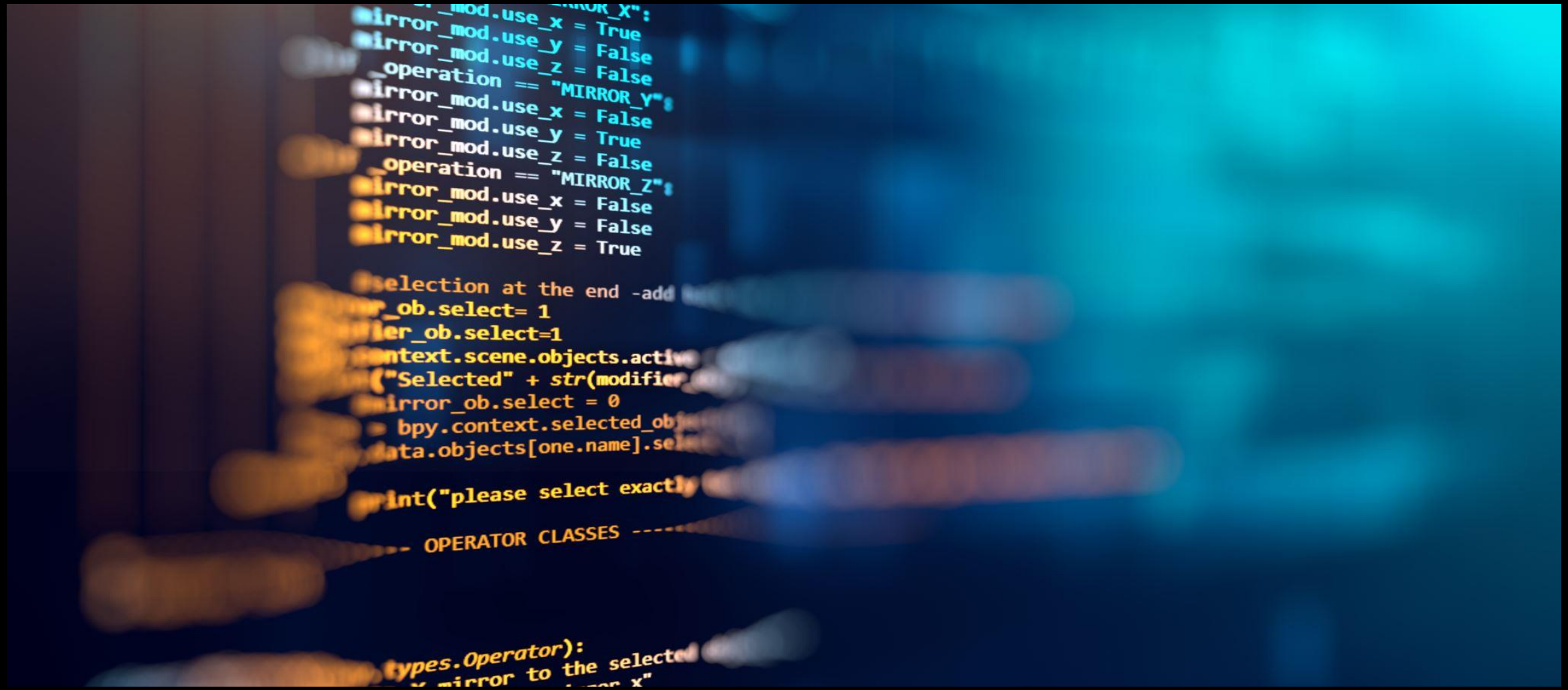
Now, I am Microsoft MVP for more than 14 years. Microsoft Certified Trainers for DevOps, Office 365, and Azure, and AWS Accredited Educator for Solution Architect

Reach me at https://ridilabs.net

The Codes is right
Until...

slow defects
anomaly untraceable

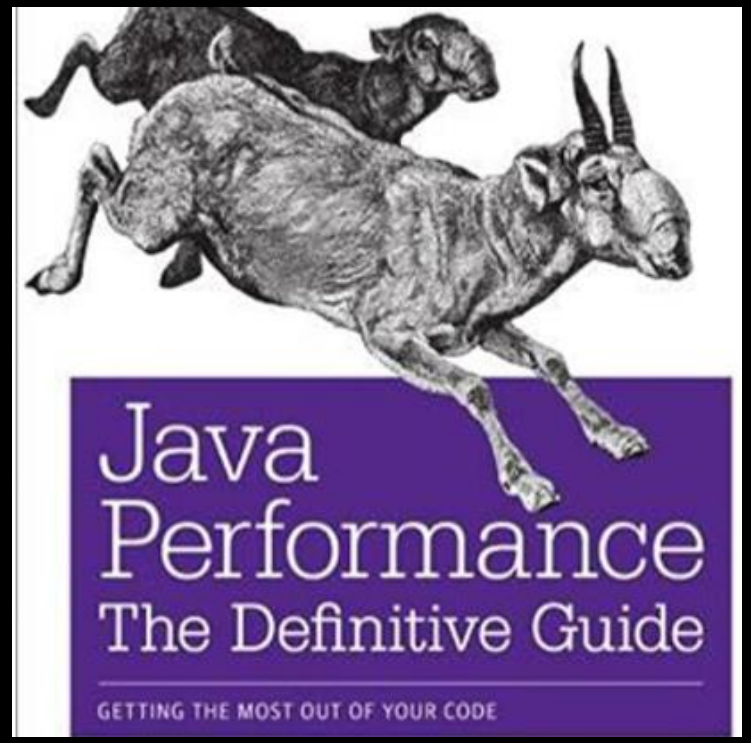
‘Good’ Code Design is like Insurance



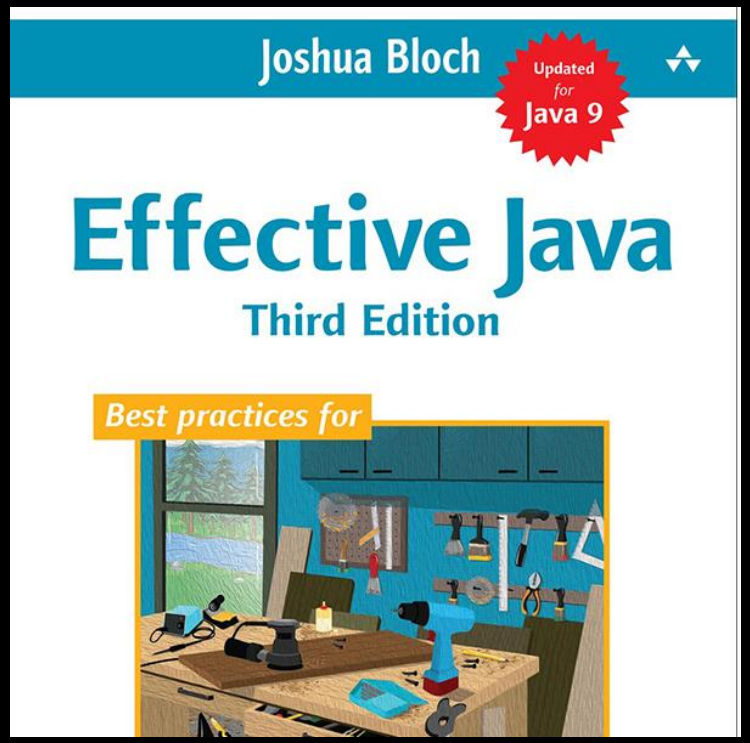
API (Application Programming Interface) provides a ‘socket’ to communicate between component or system

Quick References

Traceable?



Slow?



Defects? Anomaly?

What is API Design?

Abstracts implementation Eliminating the replication

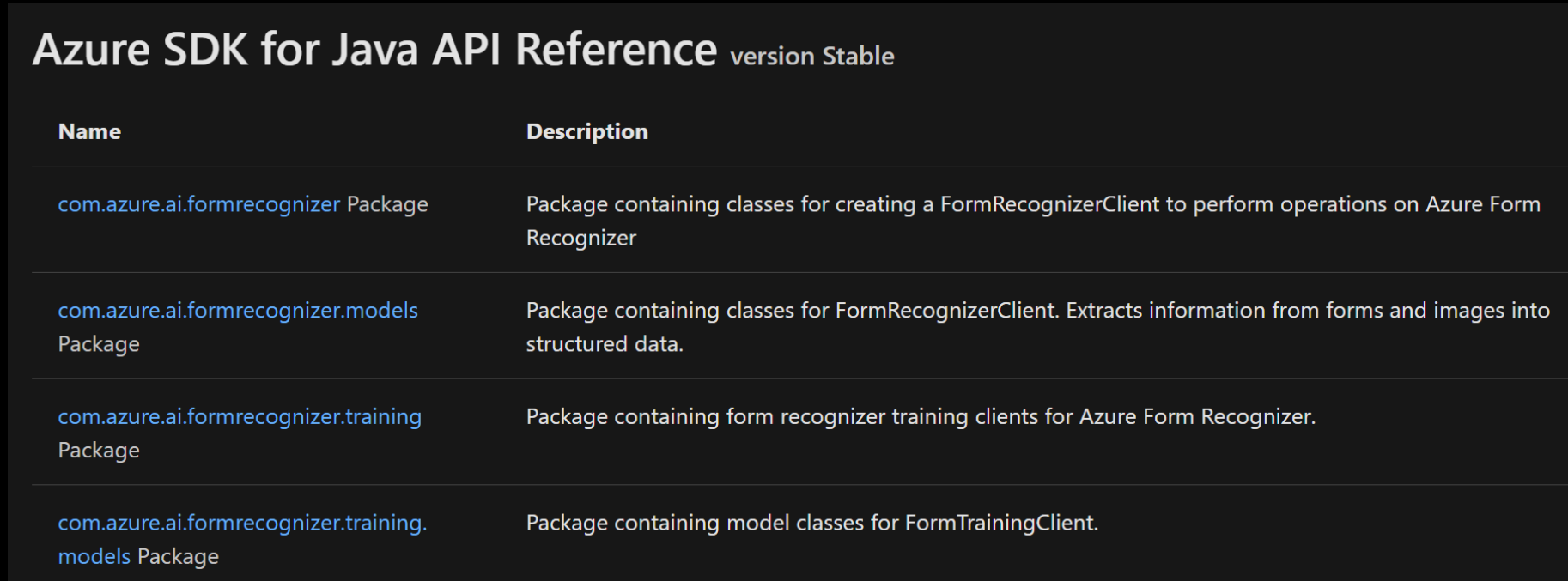
Providing Interaction

API Characteristic

Understandable	Documented	Consistent
Fit for Purpose	Restrained	Evolvable

Understandable

- Clear entry points
- Discover benefits and use of API
- Minimum External Documentation



Azure SDK for Java API Reference version Stable

Name	Description
com.azure.ai.formrecognizer Package	Package containing classes for creating a FormRecognizerClient to perform operations on Azure Form Recognizer
com.azure.ai.formrecognizer.models Package	Package containing classes for FormRecognizerClient. Extracts information from forms and images into structured data.
com.azure.ai.formrecognizer.training Package	Package containing form recognizer training clients for Azure Form Recognizer.
com.azure.ai.formrecognizer.training.models Package	Package containing model classes for FormTrainingClient.

Consistency (1/2)

- Consistency in Class Name, Method, Arguments, Constants, and Process
  - set and get
  - .of(), .valueOf()
  - Follow java coding standard and conventions
- A minimal Set of Return Types
- Avoid Surprise (Inconsistency method in similar classes)
- Returning non-null values instead (Defensive Programming)
  - String – ""
  - List - .emptyList()
  - Stream - .empty()
  - Array – empty array

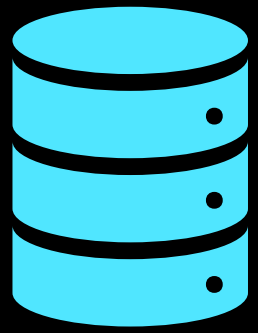
Consistency (1/2)

- Argument should be consistent
  - Order consistent
  - If argument size become unwieldly consider introducing a object parameter
- Refactoring will help consistency

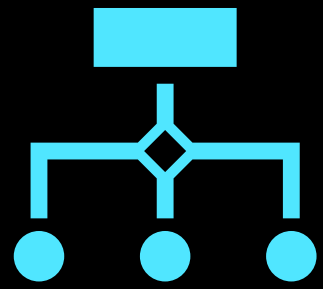
Refactoring In Visual Studio Codes

Demonstration

Fit for Purpose



Do only one things and do it right (Single API call for One Scenario)



Understand the user and the customization that needed (Multiple API calls for One Scenario)

Restrained

- Justify the Public Method when creating new API
- Introduce protected API
- Convenience API is important
  - E.g List.toString(), list.add(Object)
- Default position is make classes and public method final
- Start with private modifiers

Evolvable

- Writing sample code with your API and discuss with real users
- Sample codes should be FULL Sample
- Review the sample codes
  - Unclear
  - Duplicate or redundant
  - Abstraction is too high-level / low level
- Quality JavaDoc
  - Add annotation (@since, @link, @see)
  - Include small code snippets as direct example

API Design in Azure

Case Study

Join Visual Studio Youtube Channel

https://mic.ms/vsid

Azure Fundamentals Training

Join the Webinar and Get free $100 Exam Voucher to take Azure Fundamental Exam

aka.ms/AzureFundJan-Mar

Microsoft

Join our community!

Receive regular updates from our community experts and exclusive invites to our community events and workshops.

aka.ms/devID

Microsoft Source Newsletter