

Who's More Functional Groovy, Kotlin, Scala or Java?

Andrey Breslav



Why you should care

- Everybody says "FP is Good!"
 - ➔ Every other person can't explain what it is



About Me

- Project lead of Kotlin
 - ➔ at JetBrains since 2010
- EG member of JSR-335
 - ➔ Project **Lambda**
 - ➔ at Java Community Process



Outline

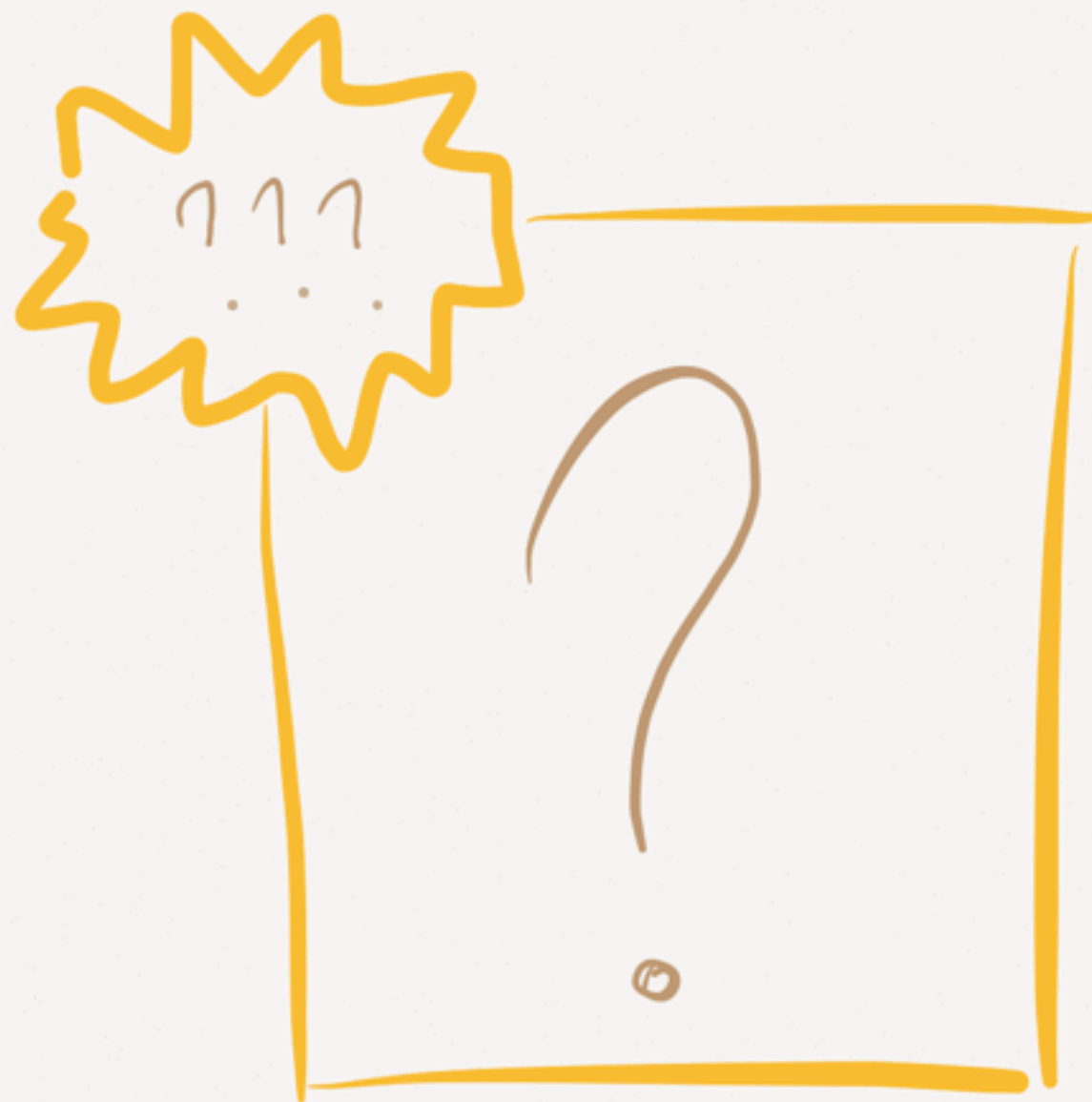
- What is FP
- What's so good about FP
- **You've** been using FP for years
- How a language helps you



As Opposed To...

What is (not) FP









Imperative

A hand-drawn speech bubble with a black outline, containing the text "Code is data".

Code is data

Functional \neq Declarative

A hand-drawn speech bubble with a black outline, containing the text "Order doesn't matter".

Order doesn't
matter





Mutable
State

java.lang.Integer

was

Immutable

before Haskell'98 was released!



java.util.Date

is

a Mistake,

not a paradigm limitation!



And Other GoF Patterns

The FP you've been using for years



OOP and Patterns

- 1995
 - ➔ Java 1.0
 - ➔ The Patterns Book (Gang of Four)



Strategy Pattern → FP

- Coding demo



Intermediate Summary

- Passing code around is natural
- Java 6 can do FP, but it's **inconvenient**
- Type system support matters



Useful Abstractions

What a good language is good for



#1 Source of boilerplate

- Your compiler is stupid
- (Demo)



#2 Source of boilerplate

- You can't put it into a library
- (Demo)



Summary

- FP is not something unfamiliar
 - ➔ Strategy pattern has been known for ages
- Language support matters
- Ivory towers are easy, but impractical
 - ➔ Pick a language that fits your needs



Kotlin Resources

- Docs: <http://kotlin.jetbrains.org>
- Demo: <http://kotlin-demo.jetbrains.com>
- Code: <http://github.com/jetbrains/kotlin>
- Twitter:
 - ➔ @project_kotlin
 - ➔ @abreslav

