



KOTLIN AND SPRING BOOT

A path toward a more declarative configuration

Me, myself and I

- **Kotlin fan**
- **Spring Boot lover**





European alternative to the “big” cloud-computing players

- **Privacy-minded**
- **Great support**



@nicolas_frankel

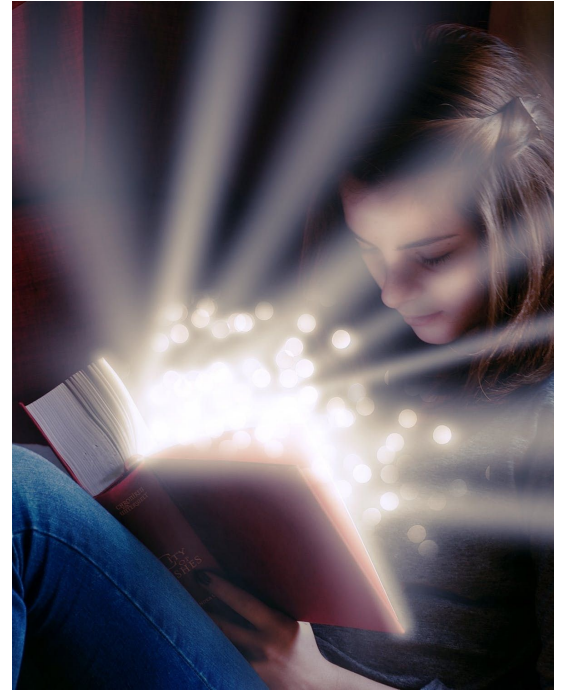
Spring Boot

- **Open Source**
- **Pivotal**
- **Spring Framework**
- **Convention over configuration**



The issue

- **Magic**
 - Via annotations
- **The solution?**
 - Functional configuration



A black laptop is shown from a front-facing perspective, slightly angled. The screen is white and displays the text "Time for DEMO" in a bold, red, sans-serif font. The text is tilted upwards from left to right. The laptop's keyboard and trackpad are visible below the screen.

Time for **DEMO**

Takeaways

- **Reactive all the way down**
- **Controller to routes**
 - Logic in handlers
- **Kotlin Beans DSL**
- **Kotlin Routes DSL**



Going further

- **Spring Fu**
 - Kofu
 - Jafu
- **Experimental!**



Direct from Spring One!

Kofu configuration for Spring Boot

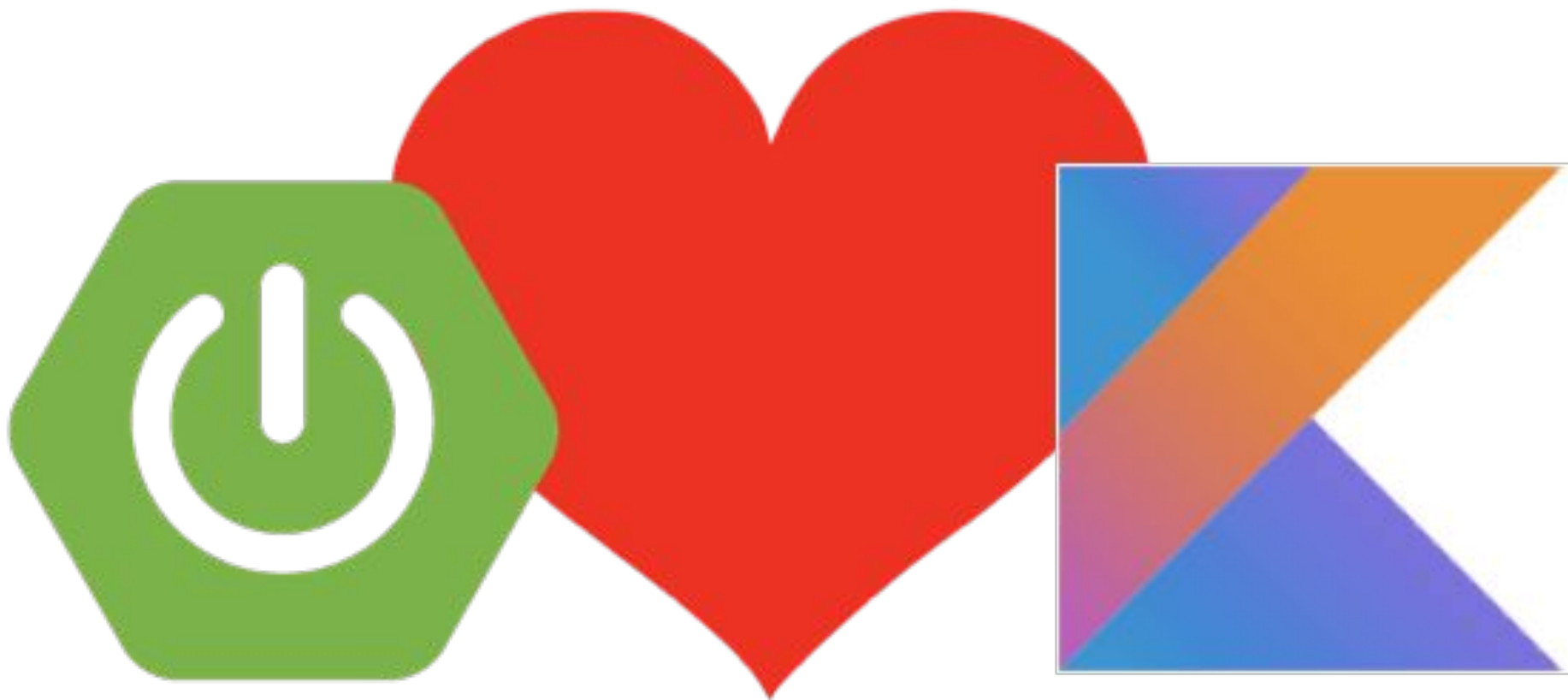
Kofu configuration is currently developed in Spring Fu, an incubator intended to ultimately contribute features related to Kotlin and functional configuration to Spring Framework, Boot and Data.

Direct from Spring One!

Kofu configuration for Spring Boot

- **Explicit configuration via a Kotlin DSL**
- **Based on Spring Boot infrastructure used in a functional way**
- **No feature enabled based on classpath detection**
- **Both declarative and programmatic**
- **Faster startup and lower memory consumption**
- **Minimal reflection & annotations usage**
- **Pure lambdas, no CGLIB proxy**





Thanks!

- <https://blog.frankel.ch/>
- @nicolas_frankel
- <https://git.io/fN3Qv>

