



HOW WE BUILD AND MIGRATED

our Spring Boot Applications to Kotlin

WARNING

This meeting will contain live coding,
so everything could (potentially) go wrong.

You will learn how to migrate a Spring Boot application
from Java to Kotlin

Spoiler: It's not only the IntelliJ IDEA migration wizard....



5 Tips for moving





I'm gonna sue
you for this!!!!

BIDNESS ETC



A security guard is shown from the back, wearing a black uniform with the word "SECURITY" in large white letters. He is wearing a black cap and a blue earpiece. He is holding a black walkie-talkie to his mouth with his right hand. The background is a blurred view of a building with white railings and a blue sky.

SECURITY





Ko Turk

Developer



<https://github.com/KoTurk>

I'm telling my experiences

Kotlin ambassador

Not a Kotlin Champion



<https://twitter.com/KoTurk77>

Agenda

- Different strategies
 - Class
 - Package
 - All
- It's not only the IntelliJ IDEA wizard













Key takeaways:

- simplicity **-> migrating is simple!**
- begin simple **-> use data classes, do not start with logic**
- do safe calls **-> with .let, .also and ?:**
- make your code readable **-> with idiomatic Kotlin!**

A photograph of two people sitting on a wooden floor, leaning against a white wall. They are wearing cardboard boxes on their heads, which have simple smiley faces drawn on them. The person on the left is wearing a red shirt and blue jeans, and the person on the right is wearing a green shirt and blue jeans. They are both barefoot. There are several stacks of cardboard boxes around them: a stack of three on the left, a stack of two on the right, and a single box in front of the person on the left. The text 'THANK YOU' is overlaid in large white letters across the middle of the image.

THANK YOU

Github: <https://github.com/KoTurk/>

Twitter: @KoTurk77

ABOUT COROUTINES

Coroutines aren't a snap-in replacement for threads. The fundamental difference is in the kind of API you're using. The rule of thumb is this:

- blocking API -> use a thread pool
- non-blocking (async) API -> use coroutines

So, if you can get a hold of an asynchronous mail-sending API, then by all means use coroutines with it. But if you're stuck with a blocking API, coroutines won't bring you much value. They can make it a bit more convenient to transfer a blocking operation out of the UI thread, but the mechanics will be the same with or without coroutines.