

SPRING BOOT W/ KOTLIN

Made with ♥ by



INTRODUCTION



4-5 years of experience as a **java developer**
in the **automotive industry**

project leader of the **sch-pincér** and the **cmsch**

kir-dev member since 2019

SZABÓ GERGELY

<https://github.com/gerviba>

ABOUT THE COURSE

- This is the 3rd iteration
- 4 sessions (2 beginner level and 2 advanced)
 - **Introduction and API basics** [~1.5h]
 - Databases, Authentication [~1.5h]
 - Dependency Injection, IoC and more Spring Data [~ 3h]
 - Websocket and others [~ 3h]

STRUCTURE

- Live coding on incomplete demo project(s)
- All the sources are available to download
- Optional extra tasks to practice the topics

WHY KOTLIN?

- Java but better
- JVM interoperation
- We won't get into the details

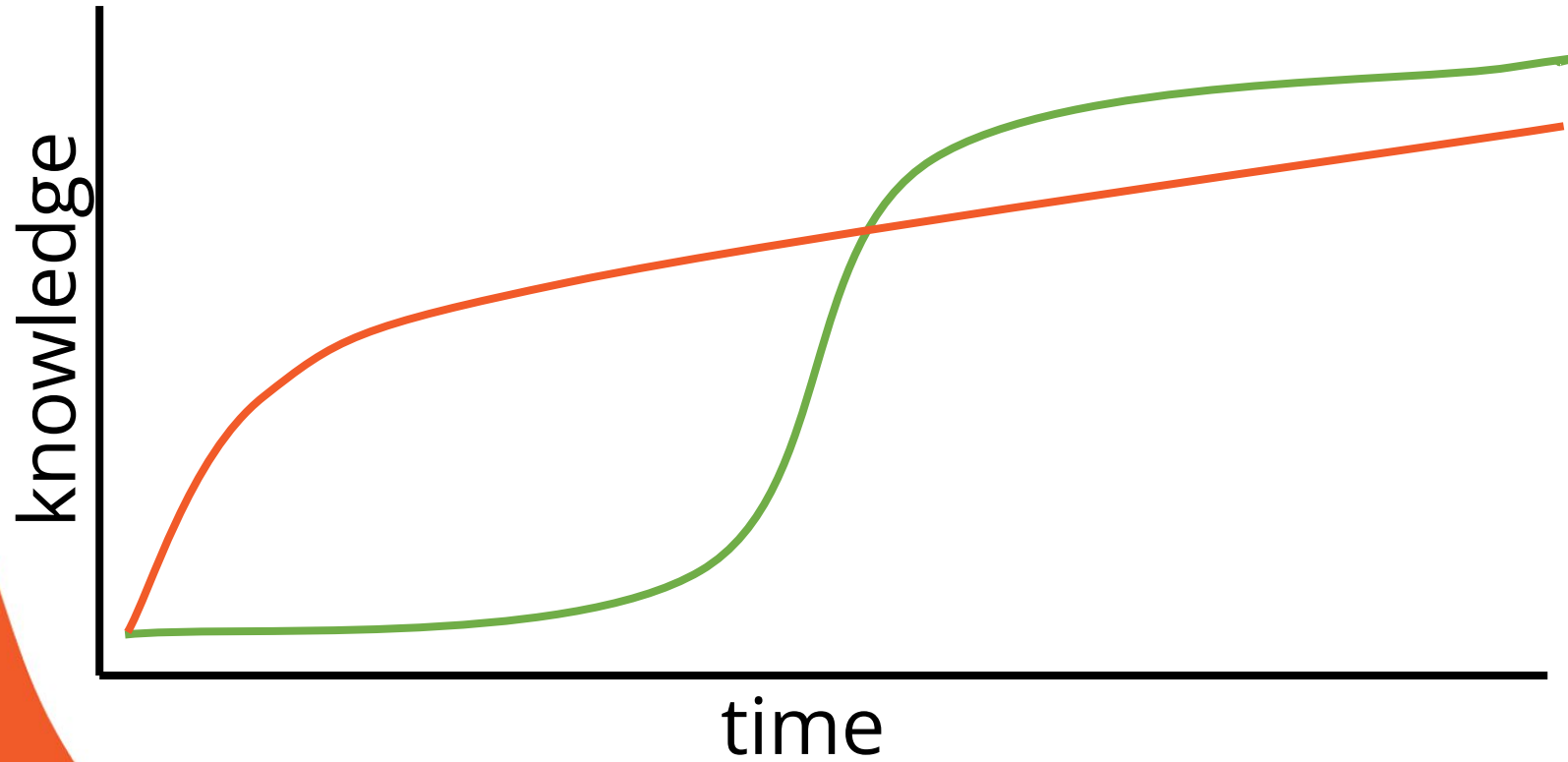
WHY SPRING?

- Active community
- Opensource
- Built on JVM ecosystem
- QA
- Performance

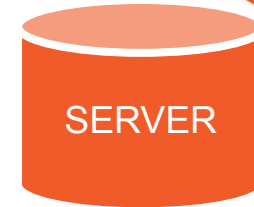
LEARNING CURVE

- Steeper learning curve
 - More effort to create basic sites
 - Pays off for more complex applications
- The steeper learning curve is a common argument against the framework

LEARNING CURVE



HTTP REQ. AND RESP.



HTTP REQ. AND RESP.



CLIENT

GET http://site.com/path?arg=val



SERVER

HTTP REQ. AND RESP.

CLIENT

GET http://site.com/path?arg=val

SERVER



HTTP REQ. AND RESP.

CLIENT

GET http://site.com/path?arg=val

200 OK 🙌 (+ payload)

SERVER



HTTP REQ. AND RESP.

CLIENT

GET http://site.com/path?arg=val

200 OK 🙌 (+ payload)

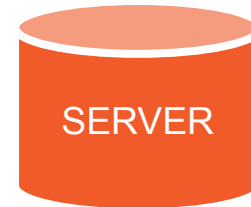
SERVER



CLIENT

POST http://site.com/path (+ payload)

SERVER



HTTP REQ. AND RESP.

CLIENT

GET http://site.com/path?arg=val

200 OK 🙌 (+ payload)

SERVER



CLIENT

POST http://site.com/path (+ payload)

200 OK 🙌 (+ payload)

SERVER



HTTP REQ. AND RESP.

CLIENT

GET http://site.com/path?arg=val

200 OK 🙌 (+ payload)

SERVER



POST / PUT / DELETE / ...

CLIENT

POST http://site.com/path (+ payload)

200 OK 🙌 (+ payload)

SERVER



SPRING INIT

<https://start.spring.io>

LIVE CODING

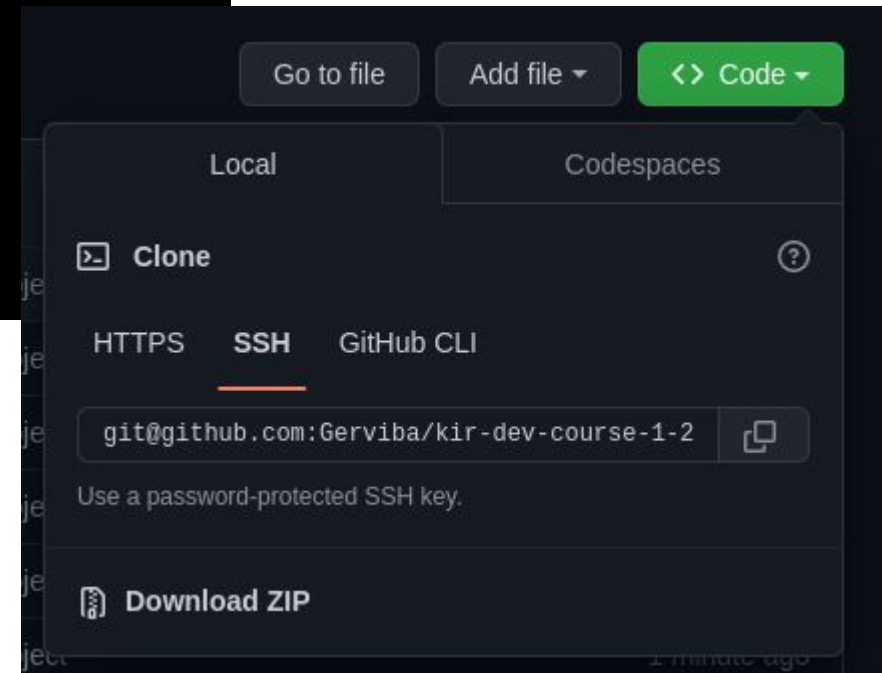
LIVE CODING

<https://github.com/gerviba/kir-dev-course-1-2023>

<https://github.com/gerviba/kir-dev-course-1-2023>

TASK 0

- clone the project
- open in IDEA
- start the project
- open in a browser



TASK 1

- create an API endpoint that
 - shows the name of the app
 - shows the current date in a readable format
 - counts the requests
- create a HTML page that show all of these

TASK 2

- create a message board page
 - users can submit messages and their names
 - all the previous messages displayed (with creation time)
- create API endpoint to show all the messages

<https://github.com/gerviba/kir-dev-course-1-2023>

EXTRA TASK

see the README.md

**THANKS FOR YOUR
PARTICIPATION!**

SEE YOU NEXT WEDNESDAY