



Week 6: Additional Topics

Unit 4: Spring Boot – Motivation and Introduction

Spring Boot – Motivation and Introduction

The way towards Spring Boot

Spring Framework – like the individual ingredients...

- Full flexibility and transparency: libraries, configuration,...
- Some integration efforts



Spring Boot – like the ready-to-bake cake...

- Simplified dependency management with starter packages
- Spring auto-configuration
- Some uncertainties in what's included



Spring Boot – Motivation and Introduction

Spring Boot

Spring Boot *by Pivotal Software* ©

- Simplifies Spring Cloud application development
- Huge OS community: tutorials, code examples, troubleshoot support
- Integrates smoothly with **Spring Cloud** components

Spring Cloud for implementing common patterns in distributed systems:

- (Cloud Foundry) connectors (already used)
- Version-controlled configuration management, e.g. Spring Cloud Config
- Service discovery, e.g. Eureka, Zookeeper, Consul
- Instance-based routing and load balancing, e.g. Zuul + Ribbon
- Distributed tracing, e.g. Sleuth or Zipkin
- Distributed messaging, e.g. Spring Cloud Bus, Stream



Spring Boot – Motivation and Introduction

Getting started

The screenshot shows the Spring Initializr web application. At the top, it says "SPRING INITIALIZR bootstrap your application now". Below this, there are three dropdown menus: "Generate a" with "Maven Project" selected, "with" with "Java" selected, and "and Spring Boot" with "1.5.8" selected. The interface is divided into two main sections: "Project Metadata" and "Dependencies".

Project Metadata

Artifact coordinates

Group:

Artifact:

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies:

Selected Dependencies

- Security X
- Web X
- Cloud Connectors X
- Actuator X
- Hystrix X
- JPA X

Generate Project alt + ↵

Don't know what to look for? Want more options? [Switch to the full version.](#)

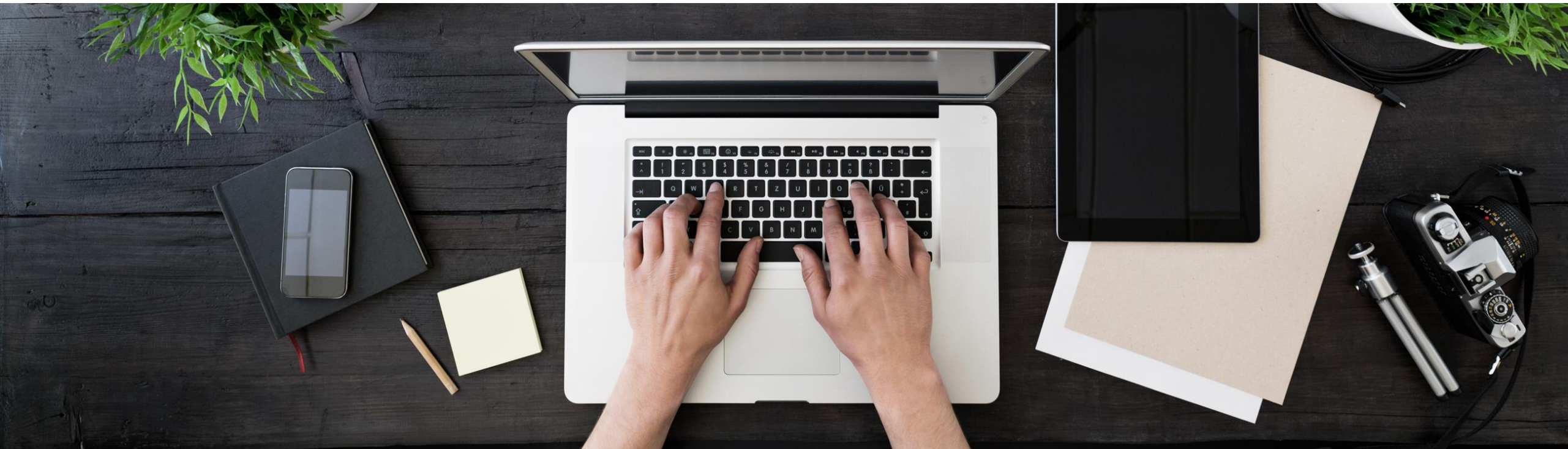
start.spring.io is powered by [Spring Initializr](#) and [Pivotal Web Services](#)

<https://start.spring.io/>

Spring Boot projects can also be created via Spring Tool Suite (Eclipse-based IDE) or IntelliJ.

Spring Boot – Motivation and Introduction

Demo: getting started



Spring Boot – Motivation and Introduction

The paradigm of Spring Boot: convention over configuration

Convention over configuration: without code generation and XML configuration

‘Starter’ Project Object Models (POMs)

Third-party libraries and *versions* are predefined to simplify Maven configuration

Example:

JPA Starter POM (spring-boot-starter-data-jpa) comes with Hibernate as default JPA implementation

Spring Auto-Configuration

Tries to configure Spring automatically to reduce boilerplate code and lower integration efforts

Example:

If there is H2 on the class path and no DataSource Bean is configured, then H2 as embedded database is auto-configured

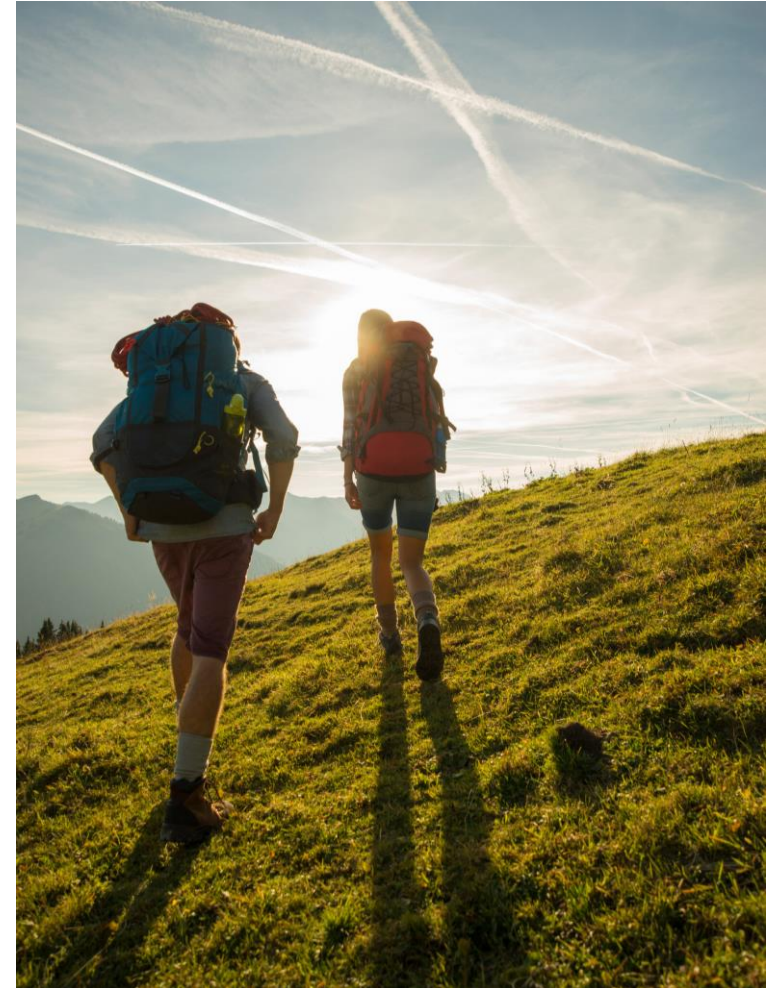
Spring Boot – Motivation and Introduction

Further reading



Additional Material

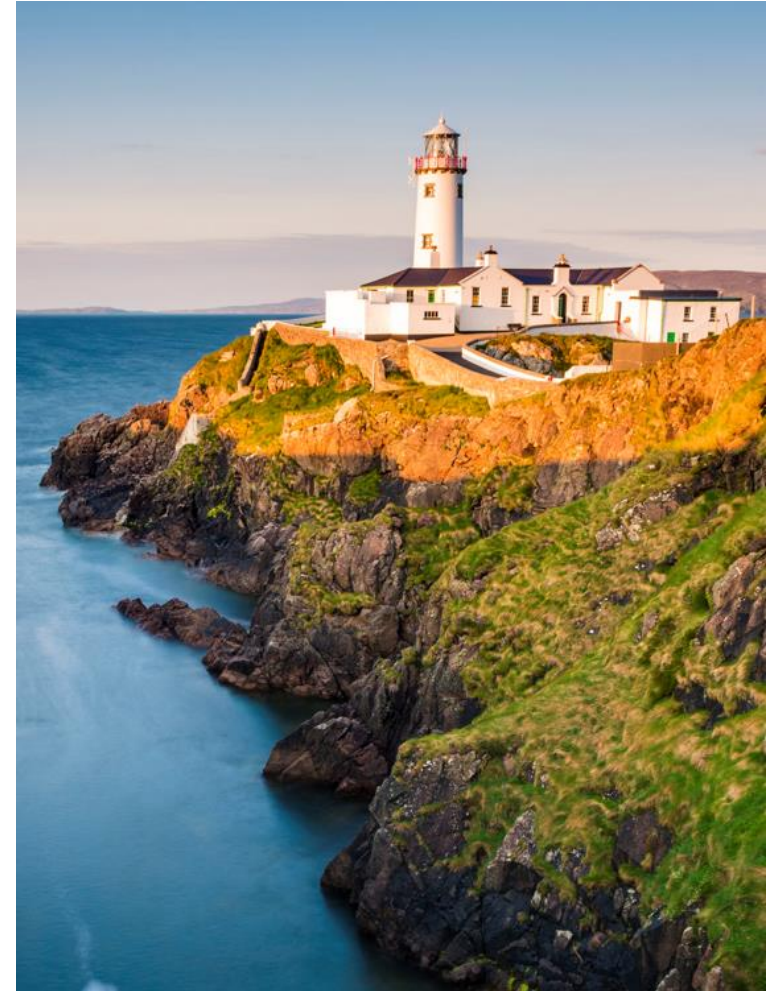
- [Spring Initializer](#)
- [The Spring Guides](#)



Spring Boot – Motivation and Introduction

What you've learned in this unit

- What Spring Boot is
- What the difference between the Spring Framework and Spring Boot is
- Why it makes sense to use Spring Boot
- How to get started with Spring Boot



Thank you.

Contact information:

open@sap.com

© 2018 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

See <http://global.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.