CLOUDZ LABS

Spring Boot Introduction

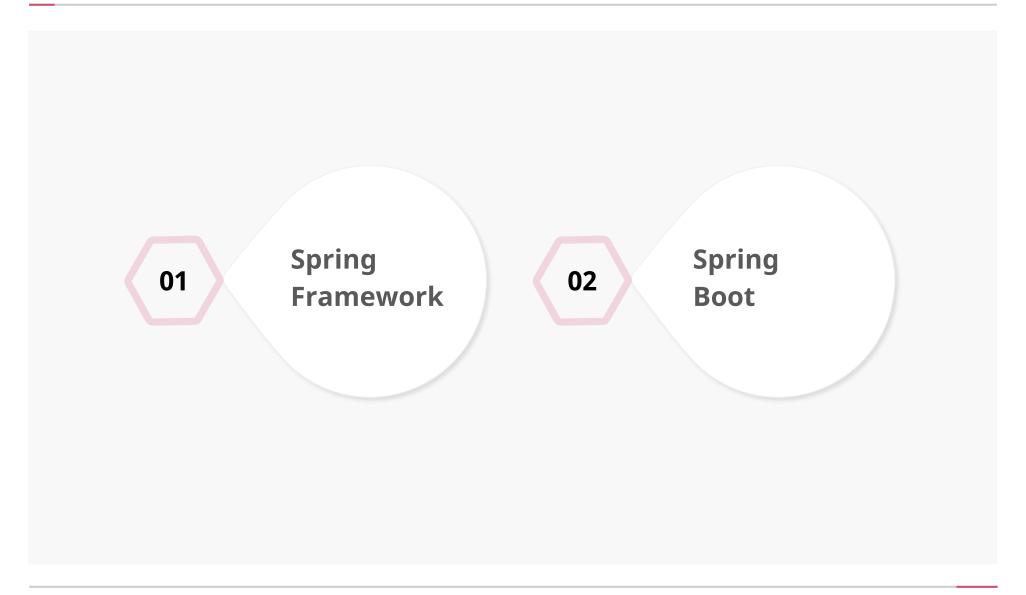




Table of Contents

01 Spring Boot
02 Spring Boot Features
03 Spring VS Spring Boot

PART 01 Spring Boot



1. Spring Framework



https://spring.io/

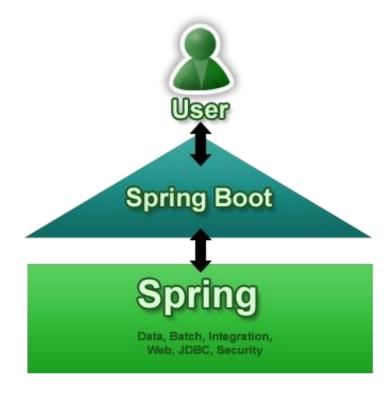


The Spring Framework is a Java platform that provides comprehensive infrastructure support for developing Java application.

Spring handles infrastructure so you focus on **your application.**

2. Spring Boot 1. Spring Boot

Spring Boot is designed to get **you** up and running as **quickly** as possible.



https://projects.spring.io/spring-boot/

2. Spring Boot

1. Spring Boot



Spring Boot makes it easy to create stand-alone, production-grade
Spring based Application that

you can "just run".

1. Spring Framework



SPRING BOOT

Takes an opinionated view of building Spring applications and gets you up and running as quickly as possible.

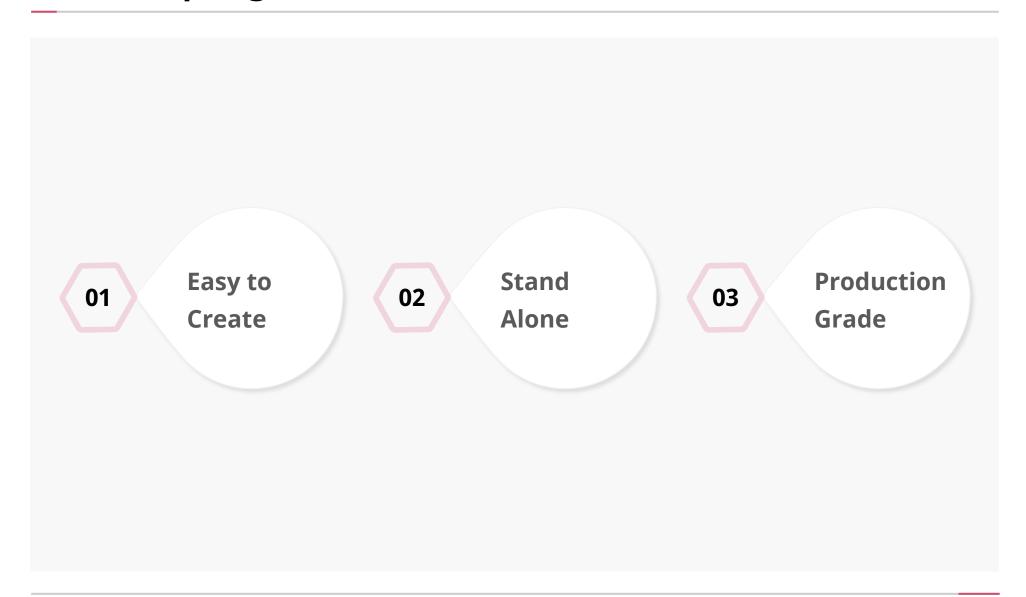


SPRING FRAMEWORK

Provides core support for dependency injection, transaction management, web apps, data access, messaging and more.

https://spring.io/projects

PART 02 Spring Boot Features



Spring Boot makes it **easy to create** stand-alone, production-grade Spring based Application that

you can "just run".

project

Application

project

Application

project

```
<!-- Spring -->
<dependency>
        <groupId>org.springframework</groupId>
        <version>4.3.10.RELEASE
        <exclusions>
                <exclusion>
                         <groupId>commons-logging
                         <artifactld>commons-logging</artifactld>
                 </exclusion>
        </exclusions>
</dependency>
<dependency>
        <aroupld>ora.springframework</aroupld>
        <artifactld>spring-webmvc</artifactld>
         Aversian 12 10 DELEASE/Versian
</dependency>
<dependency>
    <groupId>org.springframework.data</groupId>
    <a tilfactid>spring-data-jpa</artifactid>

<
</dependency>
<dependency>
        <aroupld>ora.sprinaframework</aroupld>
        <artifactid>spring-aspects</artifactid>
        (vargion ) / 3 10 RELEASE (vargion)
</dependency>
```

```
<!-- AspectJ -->
<depe dency>
       (groupld) arg aspectic/groupld)
       <artifactld>aspectirt</artifactld>
       <version>1.8.10
</dependency>
<!-- JPA -->
<depe dency>
       Carounld ora hibernate (arounld)
       <artifactId>hibernate-core</artifactId>
       <version>5.0.12.Final
       <exclusions>
              <exclusion>
                     <groupId>org.apache.geronimo.specs</groupId>
                     <artifactId>geronimo-jta_1.1_spec</artifactId>
              </exclusion>
       </exclusions>
</dependency>
<dependency>
       <around>org.hibernate</around>
      <artifactId>hibernate-entitymanager</artifactId>
       Augraian \ 5 0 12 Final / Augraian
       <exclusions>
              <exclusion>
                     <groupId>org.apache.geronimo.specs/groupId>
       <!-- dbms. logging, test, servlet & etc... -->
```

project

```
<!-- Spring -->
<dependency>
      <groupId>org.springframework</groupId>
      <artifactio>spring-context</artifactio>
      <version>4 3 10 RELEASE/version>
      <exclusions>
            <exclusion>
                   <groupld>commons-logging</groupld>
                   <artifactld>commons-logging</artifactld>
             </exclusion>
      </exclusions>
</dependency>
<dependency>
      <groupId>org.springframework
      <artifactId>spring-webmvc</artifactId>
      <version>4.3.10.RELEASE
<dependency>
   <groupId>org.springframework.data</groupId>
   <artifactId>spring-data-jpa</artifactId>
   < ersion>1.11.6.RELEASE
<dependency>
      <groupId>org.springframework
      <artifactld>spring-aspects</artifactld>
      <version>4.3.10.RELEASE
</dependency>
```

```
<!-- AspectJ -->
<dependency>
      <groupia>org.aspectj</groupia>
      <artifactId>aspectirt</artifactId>
      <version>1.8.10
</dependency>
<!-- JPA -->
<dependency>
      <groupId>org.hibernate
       Cartifactld hiharnato-caro (artifactld)
      <version>5.0.12.Final
      <exclusions>
             <exclusion>
                    <groupId>org.apache.geronimo.specs</groupId>
                    <artifactId>geronimo-jta_1.1_spec</artifactId>
             </exclusion>
      </exclusions>
</dependency>
<dependency>
      <aroupld>ora.hibernate</aroupld>
      <artifactId>hibernate-entitymanager</artifactId>
      <version>5.0.12.Final
             <exclusion>
                    <groupId>org.apache.geronimo.specs/groupId>
      <!-- dbms. logging, test, servlet & etc... -->
```

spring-boot-starter-project

```
<parent>
       <artifactld>spring-boot-starter-parent</artifactld>
        Velsion 1.3.0. NELEASE / Velsion
       <relativePath/> <!-- lookup parent from repository -->
</parent>
<dependencies>
       <dependency>
              <artifactld>spring-boot-starter-data-jpa</artifactld>
       </dependency>
       <dependency>
              <artifactId>spring-boot-starter-web</artifactId>
       </dependency.
       <dependency>
               <artifactId>mysql-connector-java</artifactId>
               ~2cobe\initille/2cobe\
       </dependency>
       <dependency>
              <artifactId>spring-boot-starter-test</artifactId>
               <scope>test</scope>
       </dependency>
</dependencies>
```

spring-boot-starter-project

```
<parent>
       <groupId>org.springframework.boot</groupId>
                   uning boot starter parent (/prtifactId>
       <version>1.5.6.RELEASE</version>
       <reiativePatn/> <!-- lookup parent from repository -->
</parent>
<dependencies>
       <dependency>
              <aroupld>ora.springframework.boot</aroupld>
              <artifactId>spring-boot-starter-data-jpa</artifactId>
       </dependency>
       <dependency>
              <groupId>org.springframework.boot</groupId>
              <artifactId>spring-boot-starter-web</artifactId>
       </dependency>
       <dependency>
              <aroupld>mvsal</aroupld>
              <artifactId>mysgl-connector-java</artifactId>
              <scope>runtime</scope>
       </dependency>
       <dependency>
              <groupId>org.springframework.boot</groupId>
              <artifactId>spring-boot-starter-test</artifactId>
              <scope>test</scope>
       </dependency>
</dependencies>
```

spring-boot-starter-project

Application

web.xml

```
<!-- next... ->
       <filter-mapping>
              <filter-name>EncodingFilter</filter-name>
              <url-pattern>/*</url-pattern>
       </filter-mapping>
       <!-- Processes application requests -->
       <servlet>
              <servlet-name>appServlet</servlet-name>
              <servlet-class>org.springframework.web.servlet.DispatcherServlet/servlet-class>
              <init-param>
                     <param-name>contextConfigLocation</param-name>
                     <param-value>/WEB-INF/spring/appServlet/servlet-context.xml</param-value>
              </init-param>
              <load-on-startup>1</load-on-startup>
       </servlet>
       <servlet-mapping>
              <servlet-name>appServlet</servlet-name>
              <url-pattern>/</url-pattern>
       </servlet-mapping>
</web-app>
```

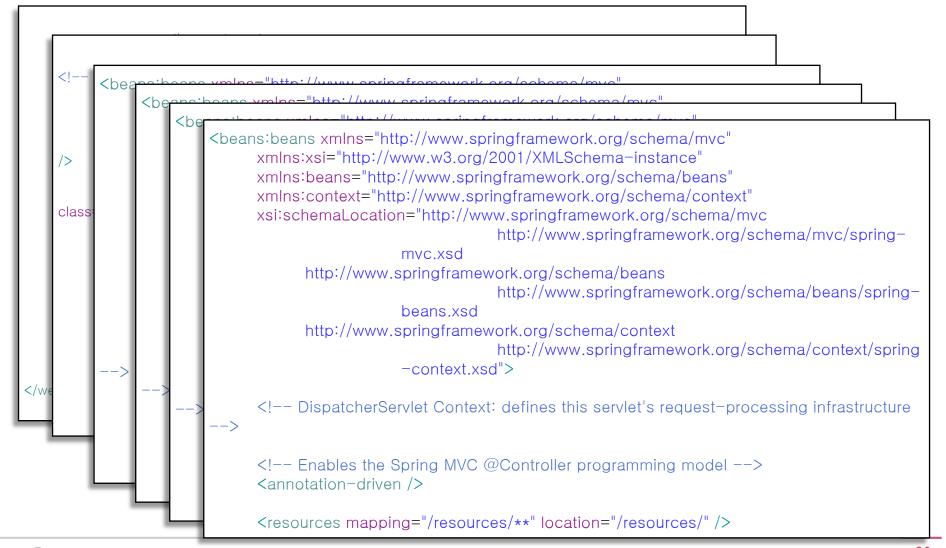
root-context.xml

```
<!-- JPA settings -->
      <bean id="transactionManager" class="org.springframework.orm.jpa.JpaTransactionManager">
             cproperty name="entityManagerFactory" ref="entityManagerFactory" />
      </bean>
      <bean class="org.springframework.dao.annotation.PersistenceExceptionTranslationPostProcessor"</pre>
      <bean id="entityManagerFactory"</pre>
class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean">
             cproperty name="dataSource" ref="dataSource" />
             cproperty name="packagesToScan" value="com.sample.spring" />
             property name="jpaVendorAdapter">
                    <bean class="org.springframework.orm.jpa.vendor.HibernateJpaVendorAdapter" />
             </property>
             property name="ipaProperties">
                    props>
                           prop key="hibernate.hbm2ddl.auto">update
                    </props>
             </property>
      </bean>
```

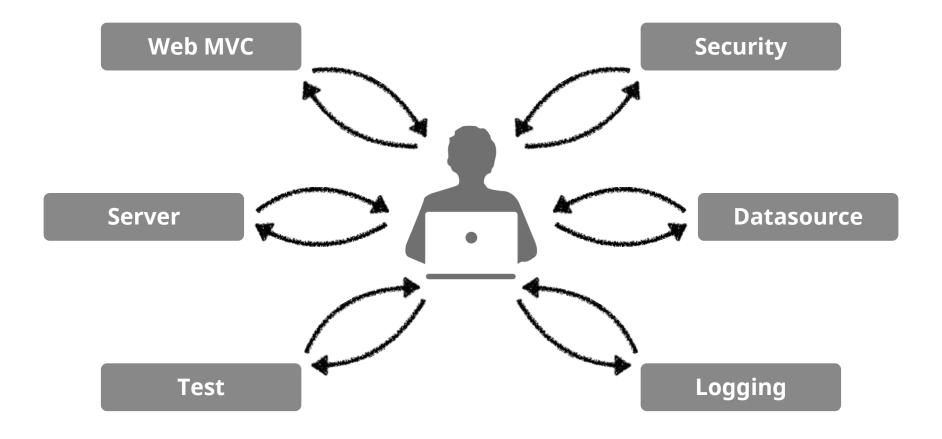
servlet-context.xml

```
<beans:beans xmlns="http://www.springframework.org/schema/mvc"</pre>
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xmlns:beans="http://www.springframework.org/schema/beans"
           xmlns:context="http://www.springframework.org/schema/context"
           xsi:schemaLocation="http://www.springframework.org/schema/myc
                                         http://www.springframework.org/schema/mvc/spring-
                             mvc xsd
class
                http://www.springframework.org/schema/beans
                                         http://www.springframework.org/schema/beans/spring-
                             beans.xsd
                http://www.springframework.org/schema/context
                                         http://www.springframework.org/schema/context/spring
                             -context xsd">
           <!-- DispatcherServlet Context: defines this servlet's request-processing infrastructure</p>
           <!-- Enables the Spring MVC @Controller programming model -->
           <annotation-driven />
           <resources mapping="/resources/**" location="/resources/" />
```

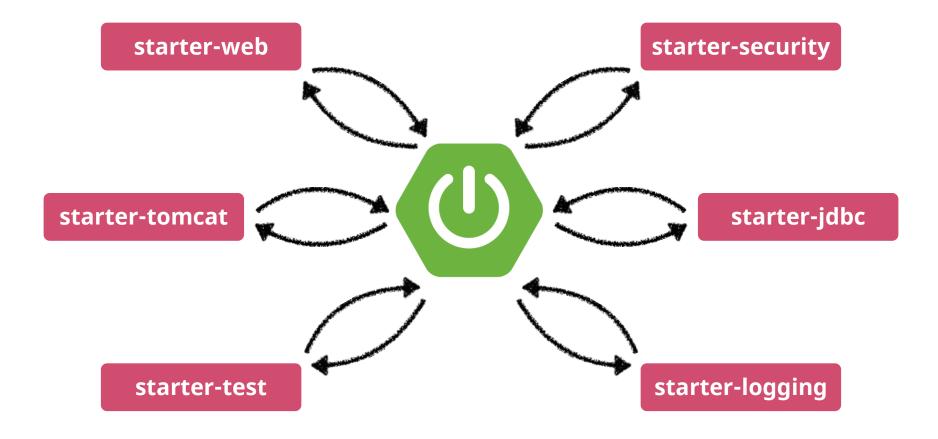
xml....



Application



SpringBootApplication

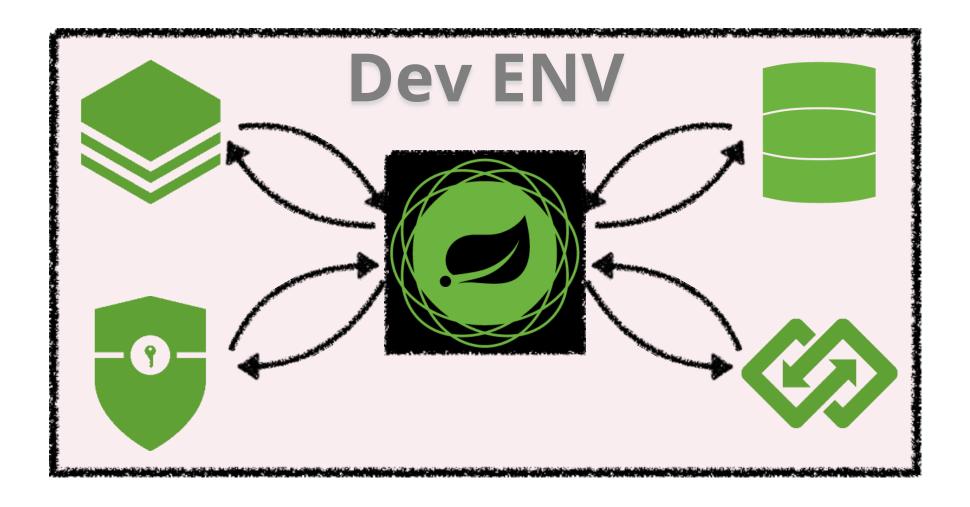


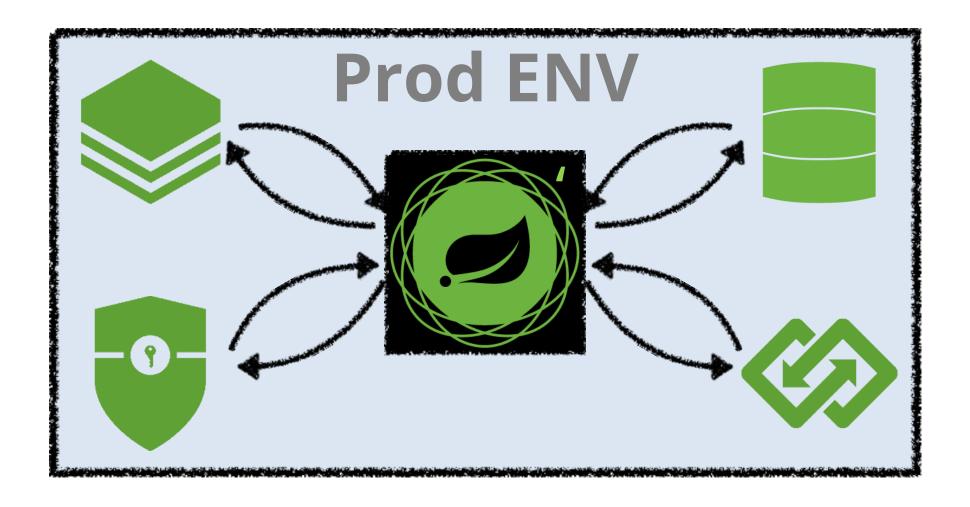
@SpringBootApplication

```
package com.sample.boot;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
@SpringBootApplication
public class SpringBootSampleApplication {
      public static void main(String[] args) {
            SpringApplication.run(SpringBootSampleApplication.class, args);
```

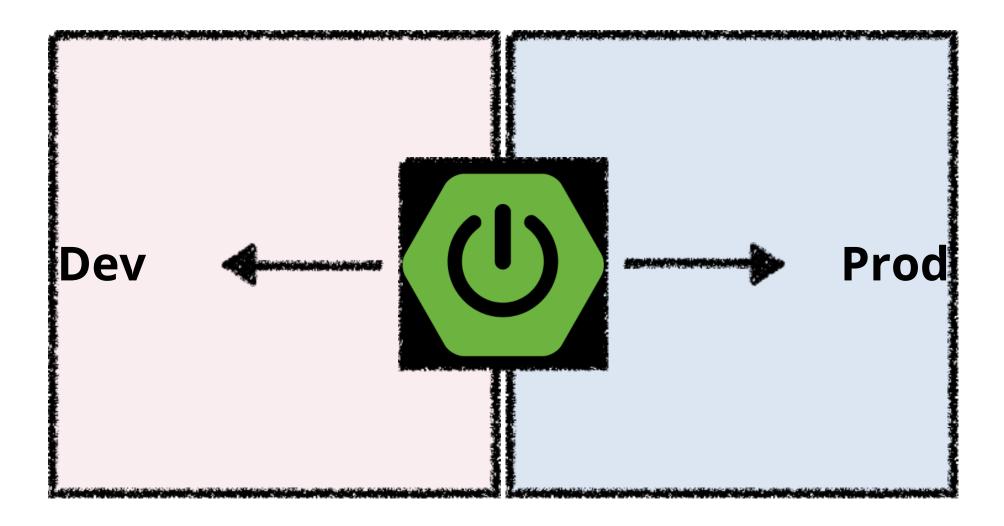
spring-boot-starter-project

@SpringBootApplication





application-{profile}.yml



application-{profile}.yml

```
# Common
spring:
ipa:
  hibernate:
    ddl-auto: update
# Localhost
spring:
 profiles: default
 datasource:
  url: jdbc:mysgl://localhost:3306/testdb
  username: root
  password:
  driver-class-name: com.mysql.jdbc.Driver
# Production
spring:
 profiles: prod
 datasource:
  url: jdbc:mysql://${DB_SERVICE_HOST}:${DB_SERVICE_PORT}/${MYSQL_DATABASE}
  username: ${MYSQL_USER}
  password: ${MYSQL_PASSWORD}
  driver-class-name: com.mysql.idbc.Driver
```

spring-boot-starter-project

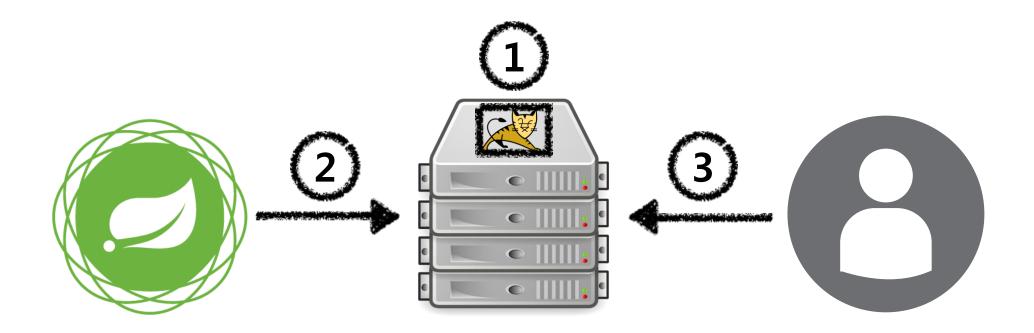
@SpringBootApplication

application-{profile}.yml

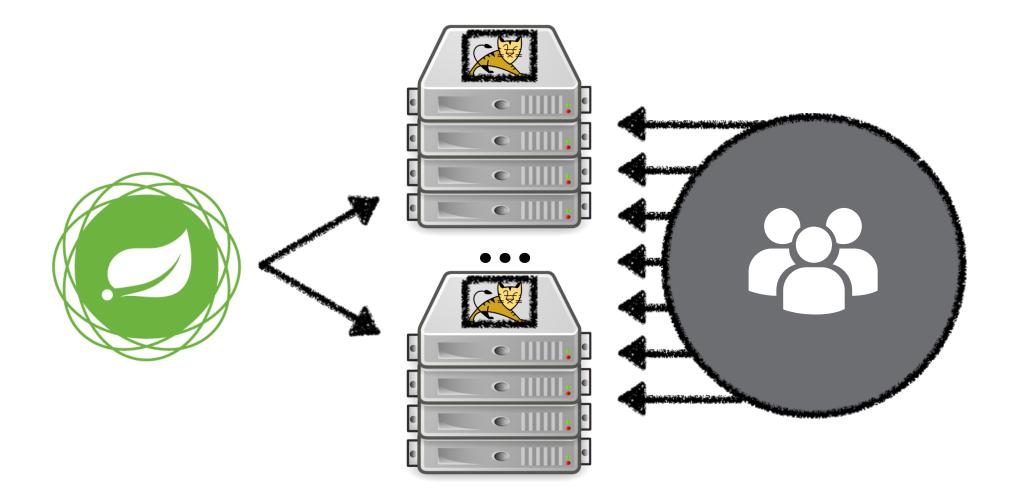
Spring Boot makes it easy to create stand-alone, production-grade Spring based Application that

you can "just run".

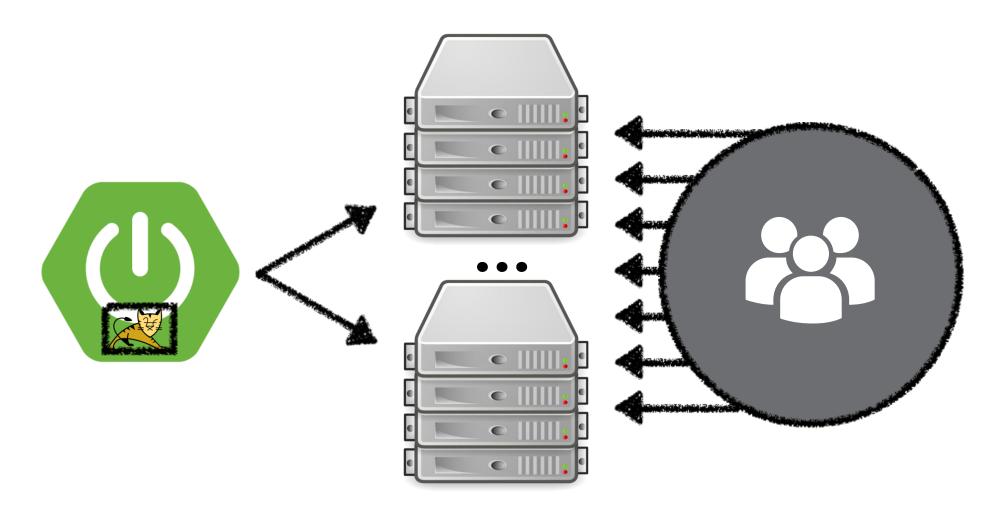
WAS Setting



n times WAS Setting



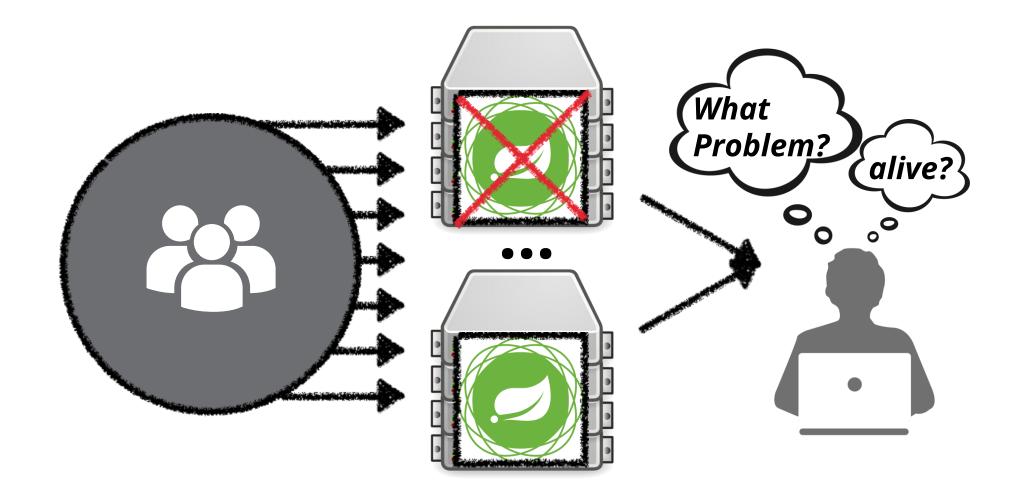
1 times WAS Setting



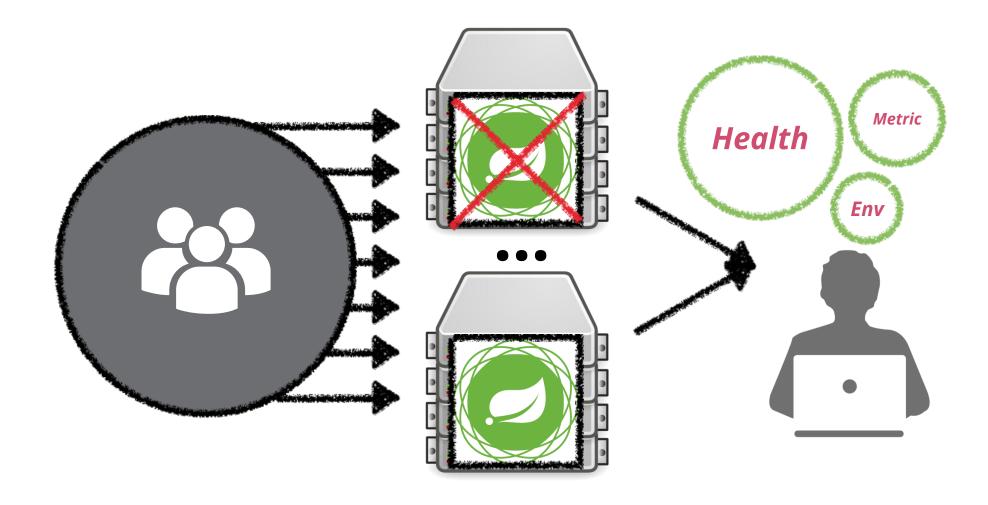
Spring Boot makes it easy to create stand-alone, **production-grade** Spring based Application that

you can "just run".

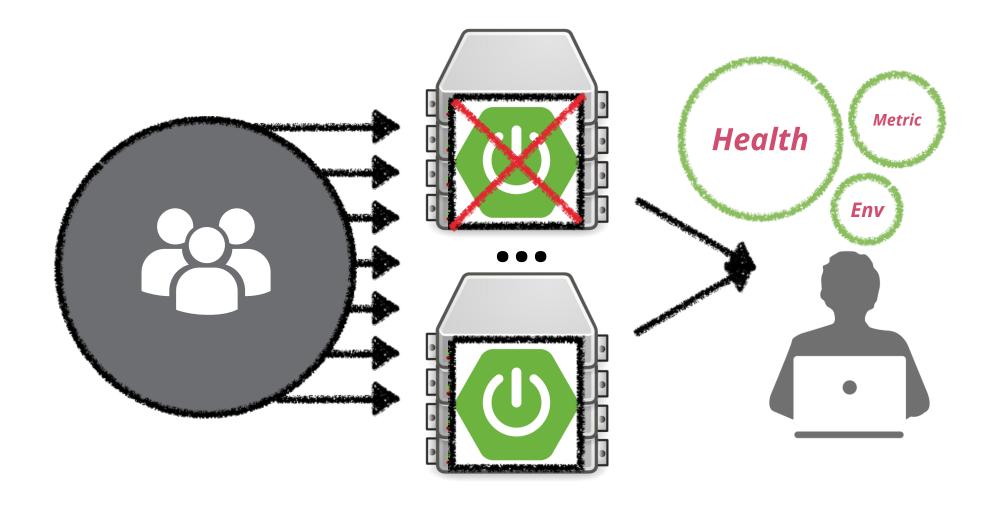
a number of additional features



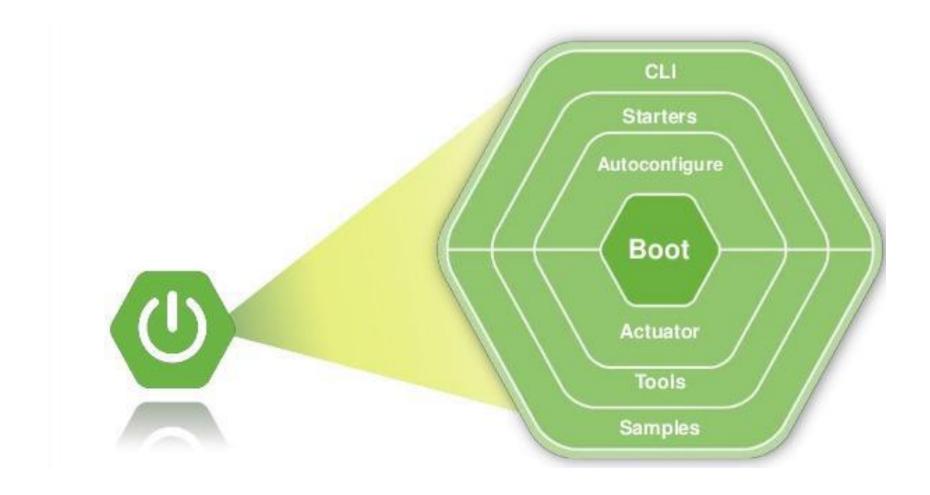
Production-ready features



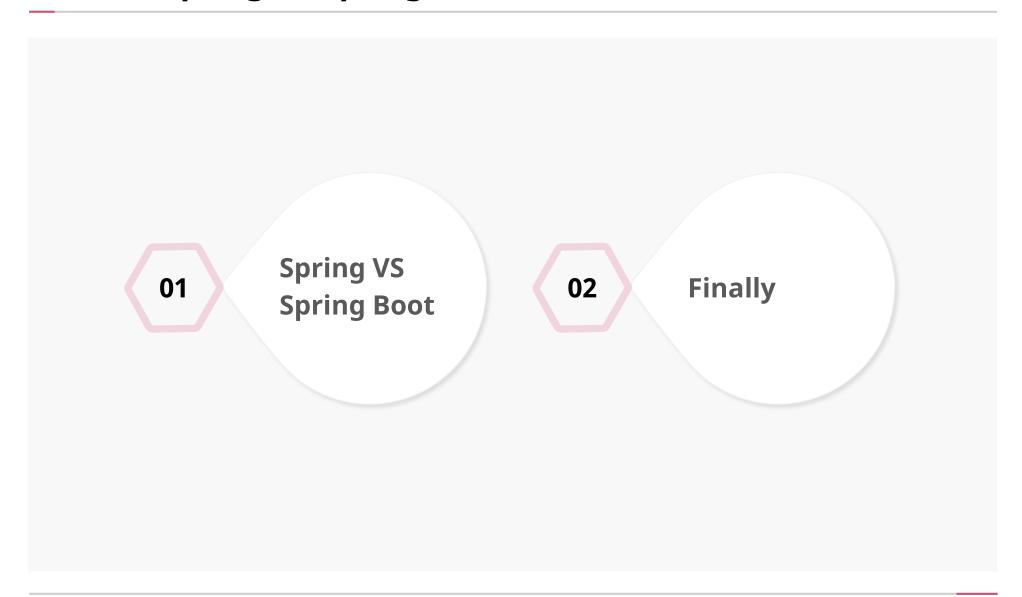
spring-boot-starter-actuator



2. Spring Boot



PART 03 Spring VS Spring Boot



1. Spring VS Spring Boot 3. Spring VS Spring Boot



1. Spring VS Spring Boot3. Spring VS Spring Boot



2. Finally 3. Spring VS Spring Boot

you can "just run".



