



BARDZO POPULARNY (NAJ...?) **KONTENER ZIAREN**

WSTRZYKIWANIE ZALEŻNOŚCI

PROGRAMOWANIE ASPEKTOWE

WSPIERA MODUŁOWOŚĆ ARCHITEKTURY

REDUKUJE ZBĘDNY / POWTARZALNY KOD

INTEGRACJA Z TECHNOLOGIAMI ZE ŚWIATA JEE

KONTROLA NAD OBIEKTEM LUB CZĘŚCIĄ PROGRAMU
ZOSTAJE PRZEKAZANA DO KONTENERA

IOC MOŻNA ZAIMPLEMENTOWAĆ NA RÓŻNE SPOSOBY
JEDNYM Z NICH JEST WZORZEC DI

"DEPENDENCY INJECTION" IS A 25-DOLLAR TERM FOR
A 5-CENT CONCEPT. (...) DEPENDENCY INJECTION
MEANS **GIVING AN OBJECT ITS INSTANCE VARIABLES.**
REALLY. THAT'S IT.

- JAMES SHORE

ROZDZIELENIE 'CO' OD 'JAK'

TO

LEPSZA SEGMENTACJA PROGRAMU

KORZYŚCI:

UŁATWIENIE ZMIANY IMPLEMENTACJI

UŁATWIENIE TESTOWANIA

@Autowired- wstrzykiwnie w Spring'u



```
@RestController
public class GreetController {

    @Autowired
    private GreetService greetService;

    // Użycie
}
```

Spring >= 4.3 && 1 konstruktor = @Autowired

```
@RestController
public class GreetController {

    private GreetService greetService;

    @Autowired
    public GreetController(GreetService greetService) {
        this.greetService = greetService;
    }

    // Użycie
}
```

```
@RestController
public class GreetController {

    private GreetService greetService;

    @Autowired
    public void setGreetService(GreetService greetService) {
        this.greetService = greetService;
    }

    @GetMapping("/greet")
    public String greet() { return greetService.getMsg(); }
}
```

UPROSZCZENIE KONFIGURACJI APLIKACJI
KONWENCJA PONAD KONFIGURACJĄ
PRZEJRZyste 'WEJŚCIE' DO APLIKACJI
STARTOWE POM'Y
MIKROSERWISY
INTEGRACJA Z BIBLIOTEKAMI (NP. NETFLIX)

[HTTPS://SPRING.IO/GUIDES/GS/ACCESSING-DATA-REST/](https://spring.io/guides/gs/accessing-data-rest/)





```
{  
  "firstName":"Frodo",  
  "lastName":"Baggins"  
}
```



```
@Entity  
public class Person {
```

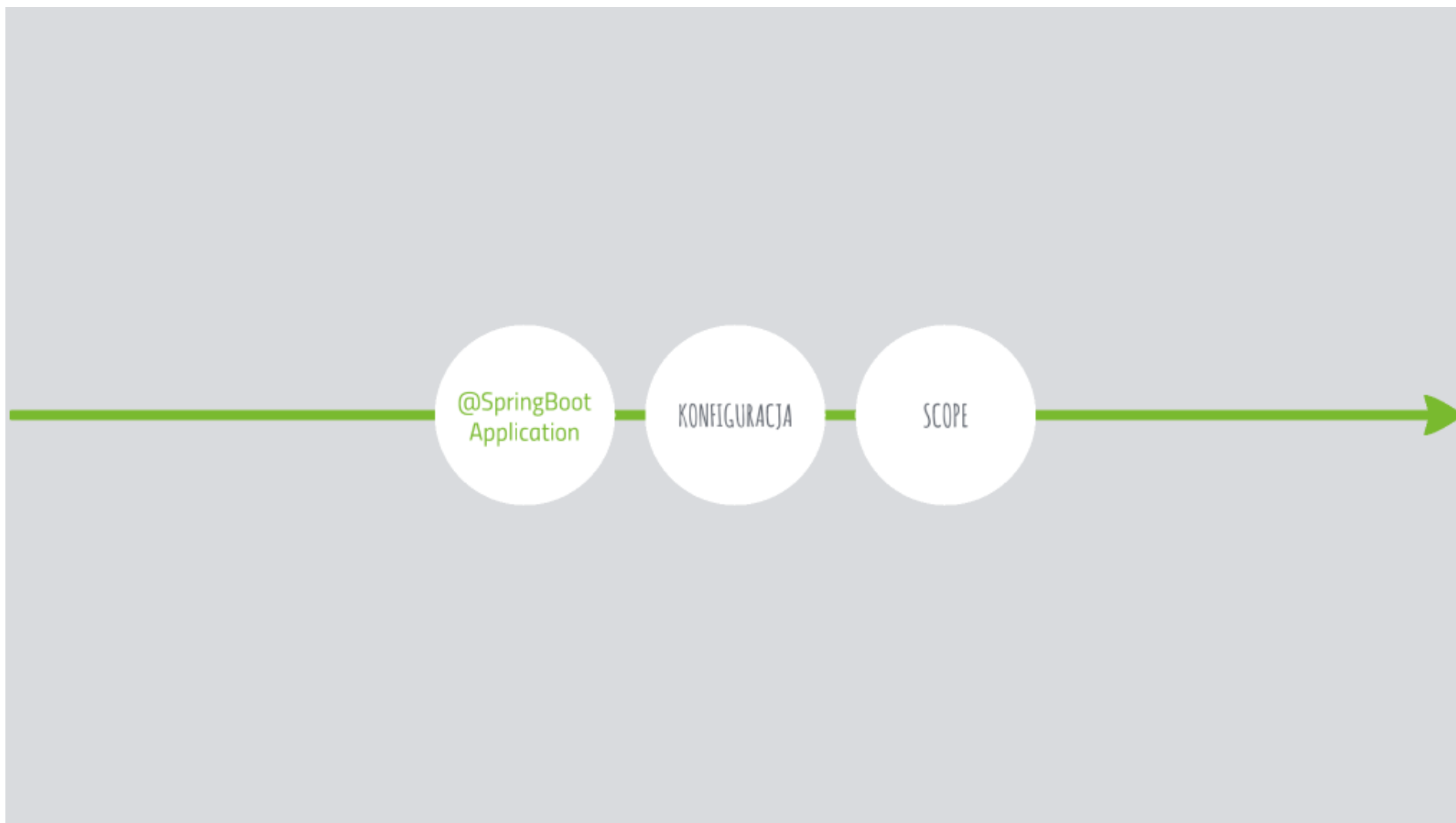
```
    @Id  
    private Long id;  
  
    private String firstName;  
    private String lastName;  
}
```

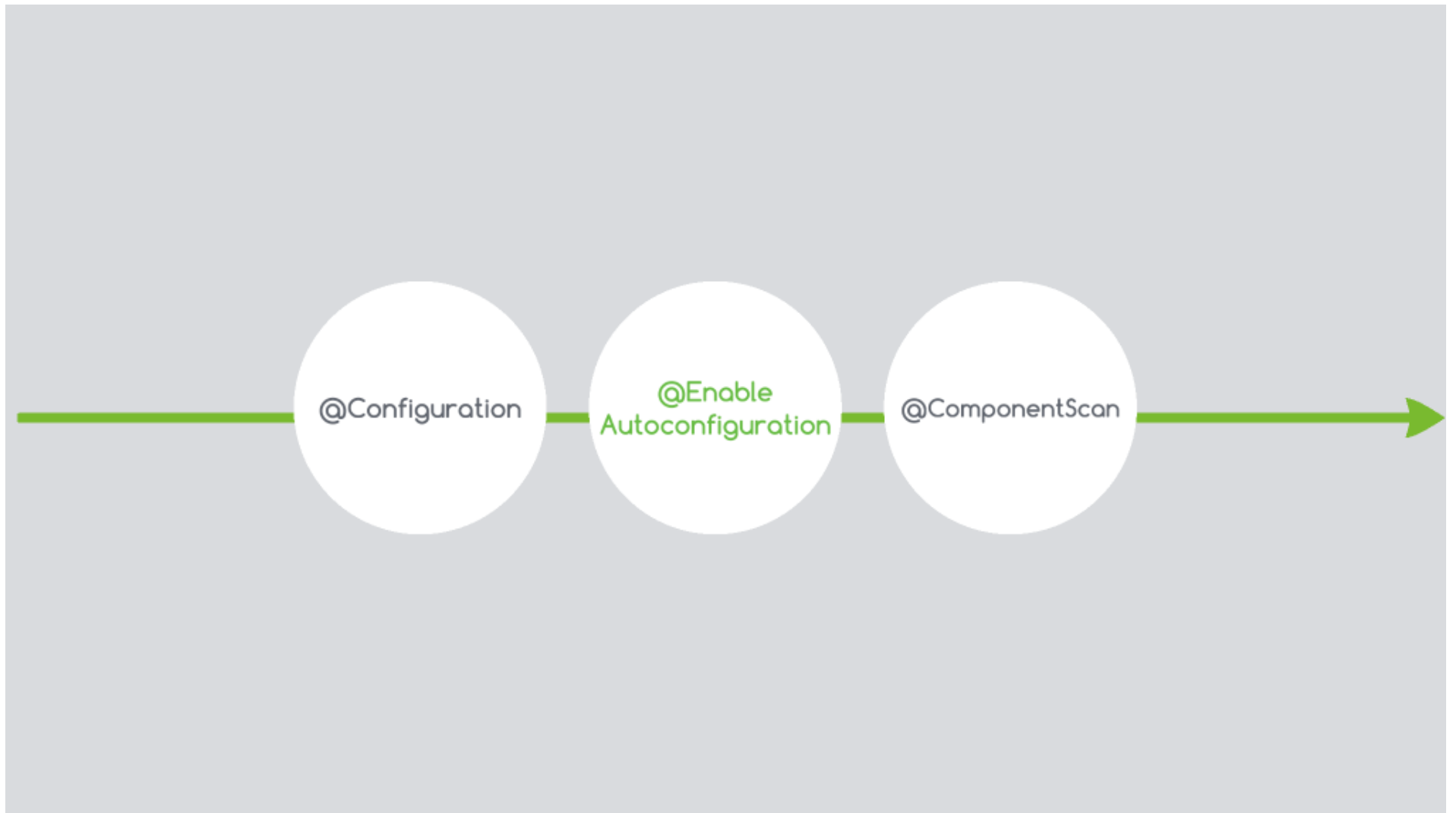
```
public interface PersonRepository extends  
CrudRepository<Person, Long> {  
}
```



ID	FIRST_NAME	LAST_NAME
----	------------	-----------

1	Frodo	Baggins
---	-------	---------





ŹRÓDŁO DEFINICJI ZIAREN SPRING'A

@Component
@Service
@Repository
@Controller
@RestController

KONWENCJA PONAD KONFIGURACJĄ

SPRINGBOOT **AUTOMATYCZNIE ZAŁADUJE NIEZBĘDNE ZIARNA**
W ZALEŻNOŚCI OD TEGO CO ZNALAZŁ NA CLASSPATH

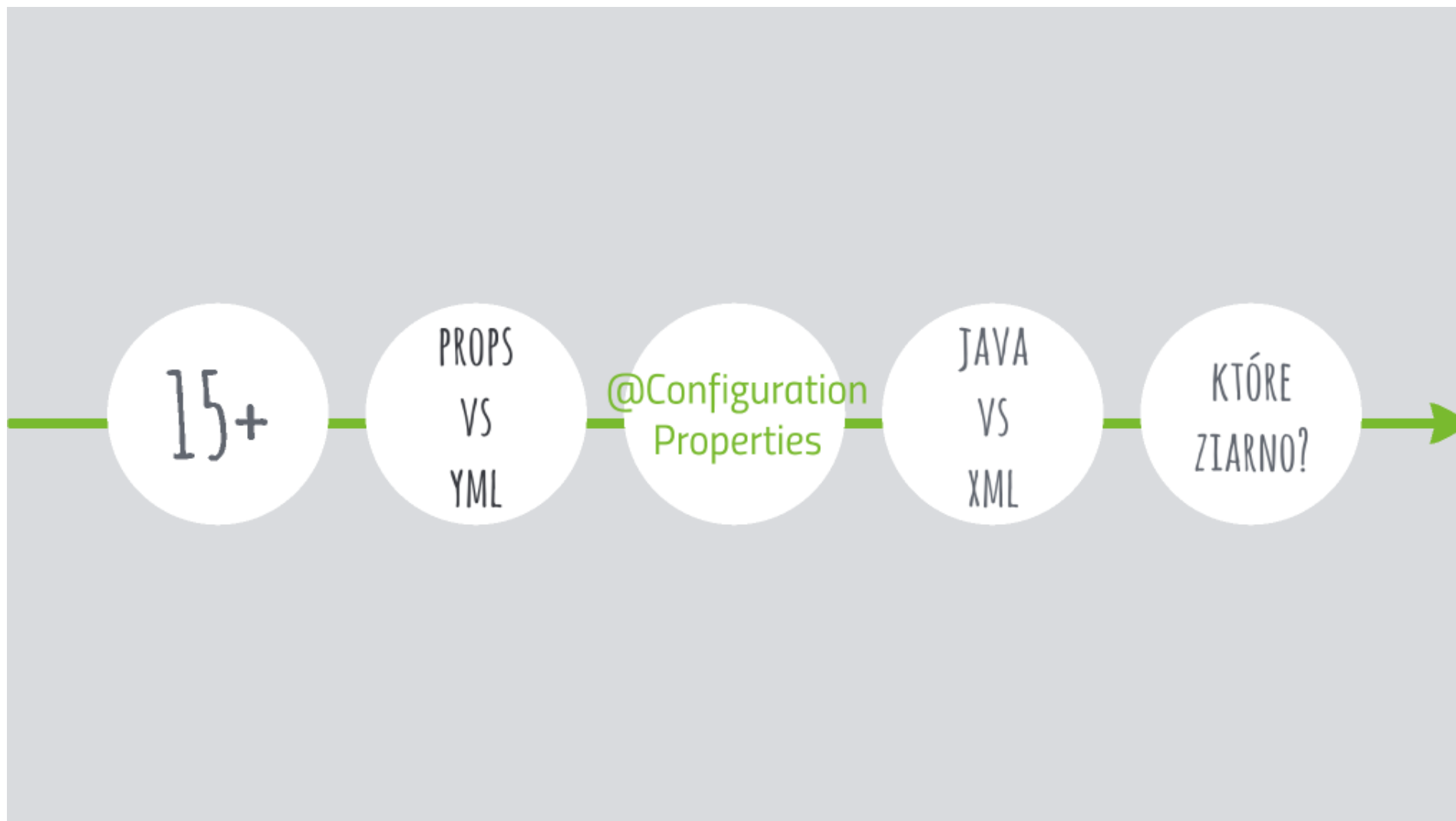
MOŻLIWOŚĆ MODYFIKACJI DOMYŚLNEGO ZACHOWANIA

CO TO OZNACZA W
PRAKTYCE?

JEŚLI NA CLASSPATH ZNAJDUJE SIĘ H2
A NIE ZDEFINIOWANO ZIARNA
SKONFIGUROWANIA POŁĄCZENIA Z BAZĄ,
SPRINGBOOT UTWORZY BAZĘ W PAMIĘCI

PRZESKANUJ OD BIEŻĄCEGO PAKIETU W DÓŁ
W POSZUKIWANIU ZIAREN SPRING'A

@Component
@Service
@Repository
@Controller
@RestController



<https://docs.spring.io/spring-boot/docs/current/reference/html/boot-features-external-config.html>

pageController.msg=Hello from properties

pageController.foo=foo

pageController.bar=bar

pageController:

msg: Hello from YAML

foo: foo

bar: bar

@ConfigurationProperties



MAPOWANIE PROPS'ÓW DO POJO

UŻYWAMY GDY WIELE @VALUE

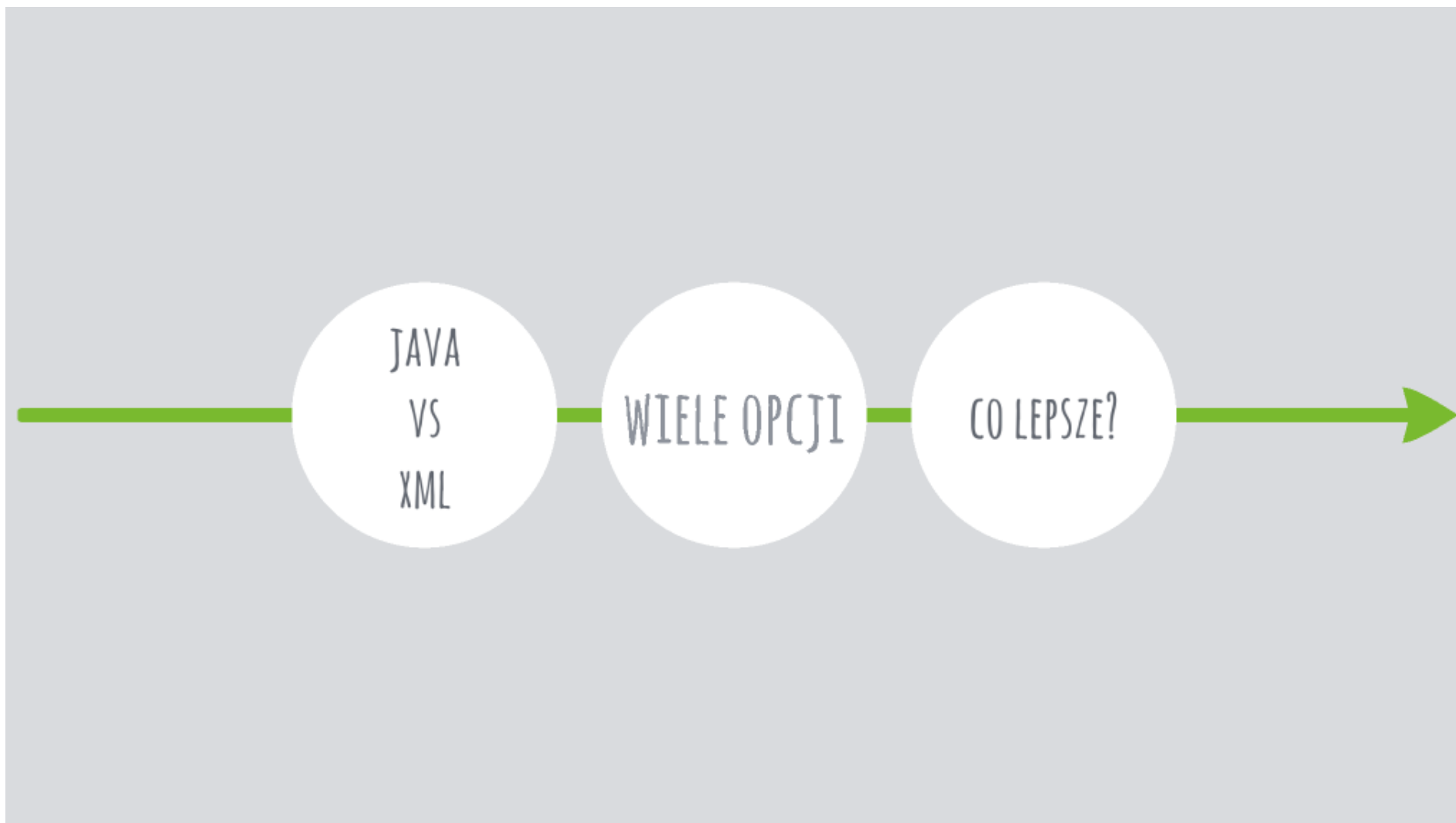
@VALID'ACJA

WSPARCIE IDE

@CONFIGURATIONPROPERTIES

DODANIE ZALEŻNOŚCI (IDEA PODPOWIADA)

SETTER'Y DLA USTAWIANYCH PÓL



JavaConfig.java

```

1 package workshop.sb.config;
2
3 import org.springframework.beans.factory.annotation.Value;
4 import org.springframework.context.annotation.*;
5 import workshop.sb.config.service.GreetService;
6 import workshop.sb.config.service.GreetServiceFirstImplementation;
7
8 @Configuration
9 @ComponentScan(basePackages = {"workshop.sb.config.controller"})
10 @PropertySource("classpath:service.properties")
11 public class JavaConfig {
12
13     @Value("${service.first.implementation.msg}")
14     public String firstImplMsg;
15
16     @Bean
17     @Profile("java")
18     public GreetService firstImpl() { return new GreetServiceFirstImplementation(firstImplMsg); }
19 }

```

spring.xml

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4       xmlns:context="http://www.springframework.org/schema/context"
5       xsi:schemaLocation="http://www.springframework.org/schema/beans
6                           http://www.springframework.org/schema/beans/spring-beans.xsd
7                           http://www.springframework.org/schema/context
8                           http://www.springframework.org/schema/context/spring-context.xsd">
9
10     <beans profile="xml">
11         <bean id="secondImpl" class="workshop.sb.config.service.GreetServiceSecondImplementation">
12             <constructor-arg value="from the second implementation"/>
13         </bean>
14         <context:component-scan base-package="workshop.sb.config.controller"/>
15         <context:property-placeholder location="classpath:service.properties" />
16     </beans>
17 </beans>

```

Spring pozwala na **wiele sposobów konfiguracji**

Java Configuration

XML

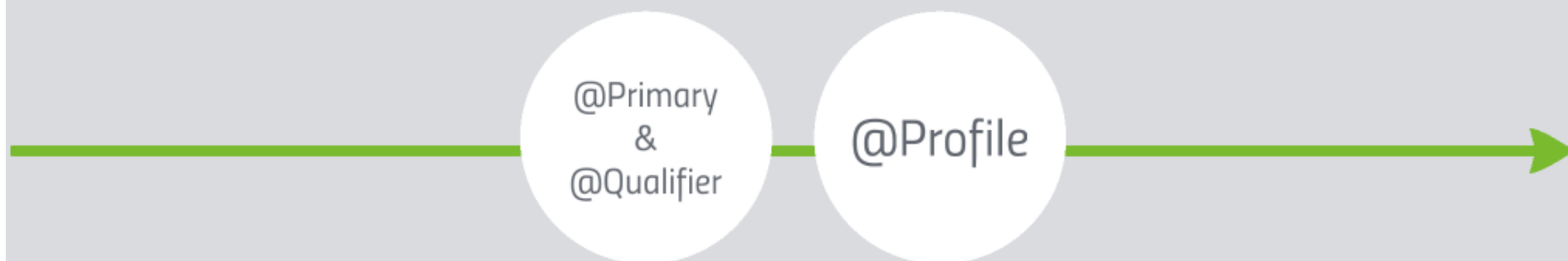
@dnotacje

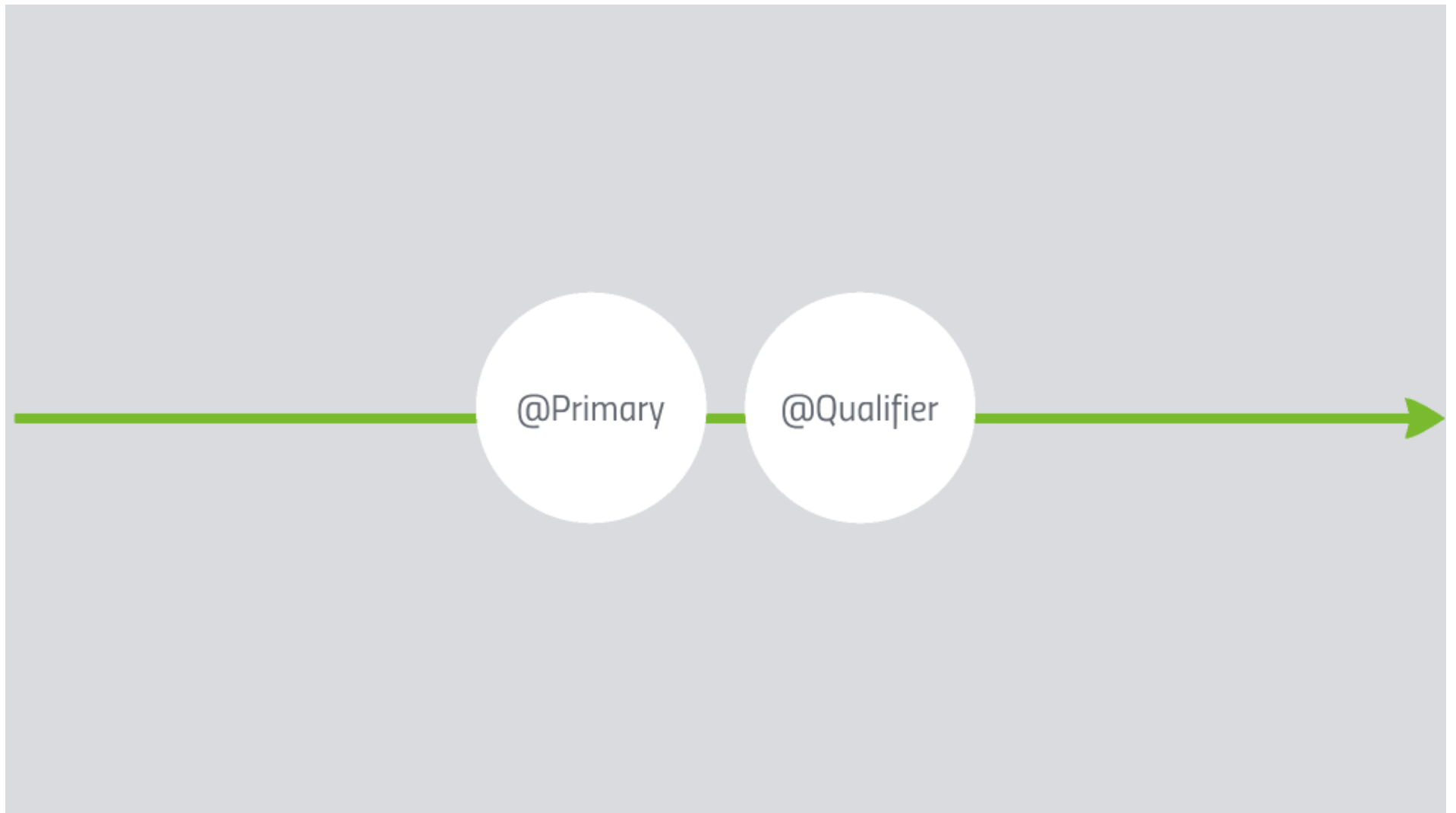
Groovy Configuration

TY DECYDUJESZ*

*JEŚLI JESTEM TEAMLEADEREM :)

1 IMPLEMENTACJA ALBO WYJĄTEK !





```
@Service  
@Primary  
public class GreetServiceFirstImplementation implements GreetService {
```

```
@Autowired  
private GreetService greetService;
```

```
@Service("second")  
public class GreetServiceSecondImplementation implements GreetService
```

```
@Autowired  
@Qualifier("second")  
private GreetService greetService;
```

I Adnotacja na springowym ziarnie:
`@Profile("moj-profil")`

II Uruchomienie Spring'a z profilem:

opcja 1 - application.properties
`spring.profiles.active=development`

opcja 2 - przekazanie flagi do VM:
`-Dspring.profiles.active=development`

opcja 3 - konfiguracje w IDE

@SCOPE("⟨WYBRANY SCOPE⟩")

SINGLETON (DEFAULT*)

PROTOTYPE

REQUEST

SESSION

GLOBAL SESSION

*ZARÓWNO W SPRING, JAK I ROZMOWACH O PRACĘ :)



HELLO WORLD

SZKIELET PROJEKTU:
SPRING INITIALIZR

DODAJ ZALEŻNOŚĆ WEB

UTWÓRZ KONTROLER

URUCHOM I PRZETESTUJ



WEB LUB KLIENT W IDEI

Generate a Maven Project with Java and Spring Boot 2.0.4

Project Metadata

Artifact coordinates

Group

com.example

Artifact

demo

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Web, SECURITY, JPA, ACTUATOR, DEVTOOLS...

Selected Dependencies

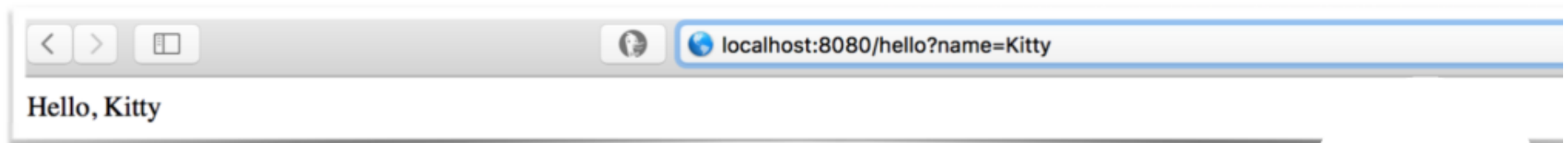
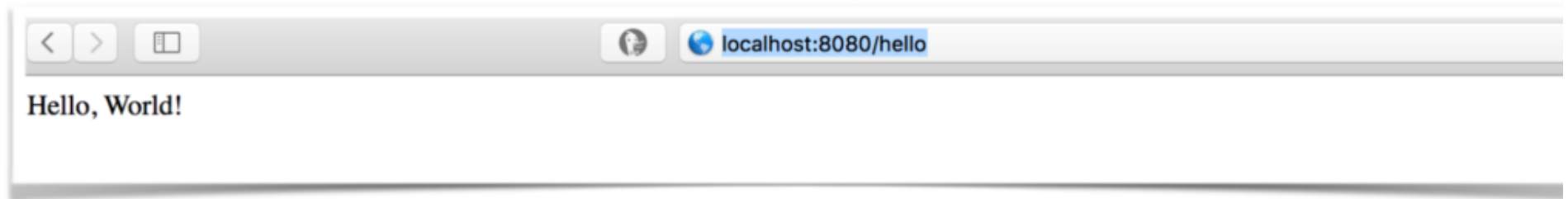
Web

Generate Project

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

```
@RestController
public class HelloController {

    @GetMapping("/hello")
    public String greetName(@RequestParam(value="name", required=false) String name) {
        String greeting = "Hello, ";
        String defaultName = "World!";
        return name != null ? greeting + name : greeting + defaultName;
    }
}
```



TEST

```
@RunWith(SpringRunner.class)
@WebMvcTest
public class HelloworldApplicationTests {

    @Autowired
    private MockMvc mockMvc;

    @Test
    public void shouldReturnDefaultMessage() throws Exception {
        this.mockMvc.perform(get( uriTemplate: "/hello"))
            .andExpect(status().isOk())
            .andExpect(content().string(containsString( substring: "Hello, World!"))));
    }

    @Test
    public void shouldReturnHelloWithNamePassedAsParam() throws Exception {
        this.mockMvc.perform(get( uriTemplate: "/hello?name=Kitty"))
            .andExpect(status().isOk())
            .andExpect(content().string(containsString( substring: "Hello, Kitty"))));
    }
}
```

