

# Spring XML Namespace Kabiliyeti



### Spring XML Şema Kabiliyeti

- Temel amaç XML tabanlı
   ApplicationContext konfigürasyonunu
   kolaylaştırmaktır
- Bean kavramının üstüne kuruludur
- Bean tanımlarını daha kolay ve anlaşılır biçimde yapmayı sağlar
- Namespace konfigürasyonu olarak da bilinir
- XML şema tanımları ile yapılabilen herşey klasik bean tanımları ile yine yapılabilir

### Namespace Kabiliyeti Olmadan MVC Konfigürasyonuna Örnek

```
<bean name="handlerAdapter"</pre>
class="org.springframework.web.servlet.mvc.method.annotation.RequestMappingHan
dlerAdapter">
roperty name="webBindingInitializer" ref="webBindingInitializer" />
property name="messageConverters" ref="messageConverters"/>
</bean>
<bean name="handlerMapping"</pre>
class="org.springframework.web.servlet.mvc.method.annotation.RequestMappingHan
dlerMapping">
cproperty name="useSuffixPatternMatch" value="false" />
</bean>
<bean id="webBindingInitializer"</pre>
class="org.springframework.web.bind.support.ConfigurableWebBindingInitializer"
conversionService" ref="conversionService">
property name="validator">
<bean
class="org.springframework.validation.beanvalidation.LocalValidatorFactoryBean
" />
</property>
</bean>
```

## Namespace Kabiliyeti Olmadan MVC Konfigürasyonuna Örnek

```
<bean id="messageConverters"</pre>
class="org.springframework.beans.factory.config.ListFactoryBean">
property name="sourceList">
st>
<bean
class="org.springframework.http.converter.ByteArrayHttpMessageConverter" />
<bean
class="org.springframework.http.converter.StringHttpMessageConverter" />
<bean
class="org.springframework.http.converter.ResourceHttpMessageConverter" />
<br/>bean
class="org.springframework.http.converter.xml.SourceHttpMessageConverter" />
<bean
class="org.springframework.http.converter.xml.XmlAwareFormHttpMessageConverter"
/>
<br/>bean
class="org.springframework.http.converter.xml.Jaxb2RootElementHttpMessageConvert
er" />
<br/>bean
class="org.springframework.http.converter.json.MappingJacksonHttpMessageConverte
r" />
</list>
</property>
</bean>
```

#### Namespace Kabiliyeti ile MVC Konfigürasyonu



Spring'in değişik modüllerine karşılık gelen çeşitli namespace elemanları vardır. Kullanılabilmeleri için konfigürasyon dosyasına XSD tanımının eklenmesi gerekir

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:mvc="http://www.springframework.org/schema/mvc"
xsi:schemaLocation="http://www.springframework.org/schema/mvc
http://www.springframework.org/schema/mvc/spring-mvc-4.0.xsd
http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">
```

#### <mvc:annotation-driven/>

</beans>

JSR-303 validator, conversionService, httpMessageConverters gibi pek çok bean tanımını tek bir ifadeye indirger

#### Namespace Elemanlarının Annotasyon Karşılıkları



<pre><context:component-scan base-="" package="com.javaegitimleri"></context:component-scan></pre>	<pre>@ComponentScan(base- packages={"com.javaegitimleri"})</pre>
<tx:annotation-driven></tx:annotation-driven>	@EnableTransactionManagement
<context:spring-configured></context:spring-configured>	@EnableSpringConfigured
<cache:annotation-driven></cache:annotation-driven>	@EnableCaching
<aop:aspectj-autoproxy></aop:aspectj-autoproxy>	@EnableAspectJAutoProxy
<context:load-time-weaver></context:load-time-weaver>	@EnableLoadTimeWeaving
<mvc:annotation-driven></mvc:annotation-driven>	@EnableWebMvc



### İletişim

- Harezmi Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- http://www.java-egitimleri.com
- info@java-egitimleri.com

