

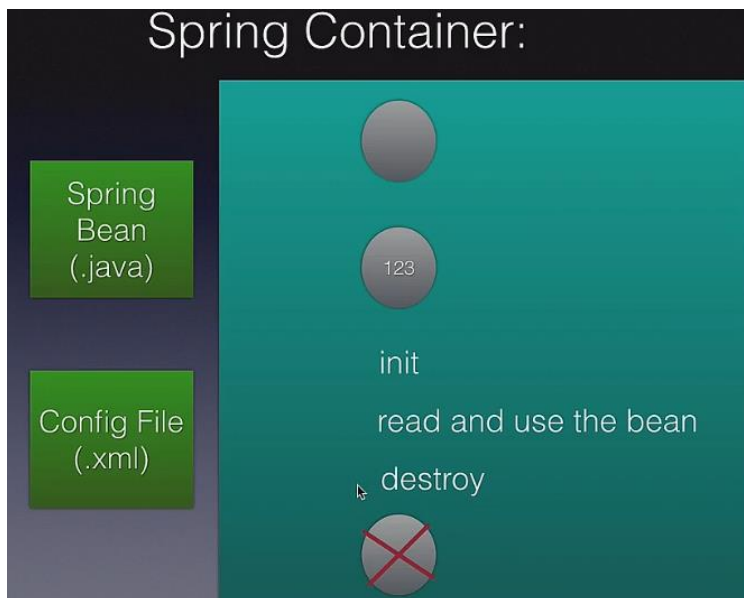
1. 라이프 사이클 메소드 개념

```
public void init()  
public void destroy()
```

가. 사용 예

- 데이터베이스 접속
- 데이터베이스 접속해제

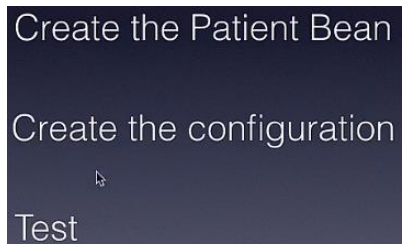
나. 스프링 컨테이너



다. 설정

- 1) XML Configuration
- 2) Spring Interfaces
- 3) Annotations

2. XML 설정을 이용하는 라이프 사이클



3. 섀다운 전 혹은

4. 애노테이션을 이용한 라이프 사이클

질문: JDK11: 내 프로젝트에서 @PostConstruct 및 @PostDestroy 주석을 사용 X

답:

- @PostConstruct 및 @PreDestroy 주석은 모두 Java EE 의 일부
- Java EE 는 Java 9 에서 더 이상 사용 X, Java 11 에서 제거
- 종속성에서 추가해야 함.

```
<dependency>
  <groupId>javax.annotation</groupId>
  <artifactId>javax.annotation-api</artifactId>
  <version>1.3.2</version>
</dependency>
```

5. 스프링 XML 스키마

가. DTD(Document Type Definition)와 스키마

- xml 문서는 미리 정해진 구조를 따라서 작성됐는지 검사할 수 있음
- **xml 문서의 구조를 정의 하는 방법:** DTD, 스키마 schema
- 스프링의 xml 설정 파일은 이 두 가지 방식을 모두 지원한다.

나. DTD(Document Type Definition): 작성하는 문서가 어떤 문서인지에 대해서 정의

- **하나의 시스템 내에서** 사용할 xml 데이터의 구조를 정의하여 유효성을 점검할 때 사용.

다. 스키마(Schema)

- 스프링은 DI를 위한 특별목적에 위해 별도의 태그를 사용할 수 있는 방법을 제공
- 이 태그들은 각각 별개의 스키마 파일에 정의되어있고 독립적인 네임스페이스를 사용해야 함
- 따라서 이런 태그를 사용하려면 DTD대신 네임스페이스가 지원되는 스키마를 사용해야 함
- **서로 다른 시스템 사이**의 데이터를 주고받을 수 있는 데이터 표준화 제공.

```

<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:tx="http://www.springframework.org/schema/tx"
       xmlns:mvc="http://www.springframework.org/schema/mvc"
       xmlns:util="http://www.springframework.org/schema/util"
       xsi:schemaLocation="
         http://www.springframework.org/schema/beans
         http://www.springframework.org/schema/beans/spring-beans-4.1.xsd
         http://www.springframework.org/schema/util
         http://www.springframework.org/schema/util/spring-util-4.1.xsd
         http://www.springframework.org/schema/mvc
         http://www.springframework.org/schema/mvc/spring-mvc-4.1.xsd
         http://www.springframework.org/schema/tx
         http://www.springframework.org/schema/tx/spring-tx-4.1.xsd
         http://www.springframework.org/schema/context
         http://www.springframework.org/schema/context/spring-context-4.1.xsd"
>
.
.
.
예) </beans>

```

6. 모든 애노테이션 지원 설정

Patient.java **lcconfig.xml** Test.java

```

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       xmlns:p="http://www.springframework.org/schema/p"
6       xsi:schemaLocation="http://www.springframework.org/schema/beans
7         http://www.springframework.org/schema/beans/spring-beans.xsd
8         http://www.springframework.org/schema/context
9         http://www.springframework.org/schema/context/spring-context.xsd">
10
11   <bean class="com.springcore.lc.annotations.Patient" name="patient"
12         p:id="123" />
13
14   <context:annotation-config />
15
16   <!-- <bean
17         class="org.springframework.context.annotation.CommonAnnotationBeanPostProcessor" /> -->
18
19 </beans>

```

7. 라이프 사이클 메소드 정리

	XML Configuration
init	init-method destroy-method
destroy	Annotations
	@PostConstruct @PreDestroy
	Spring Interfaces
	InitializingBean DisposableBean
<context:annotation-config/>	

8. 과제

TicketReservation	
initialize	Annotations
cleanUp	