Example

Here’s an example to show you how to use **InitializingBean** and **DisposableBean**. A CustomerService bean to implement both InitializingBean and DisposableBean interface, and has a message property.

Java

package com.mkyong.customer.services;

import org.springframework.beans.factory.DisposableBean;

import org.springframework.beans.factory.InitializingBean;

public class CustomerService implements InitializingBean, DisposableBean

{ String message;

public String getMessage() {

return message;

}

public void setMessage(String message) {

this.message = message;

}

public void afterPropertiesSet() throws Exception {

System.out.println("Init method after properties are set : " + message);

}

public void destroy() throws Exception {

System.out.println("Spring Container is destroy! Customer clean up");

}}

*File : Spring-Customer.xml*

Markup

<beans xmlns="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-2.5.xsd">

<bean id="customerService" class="com.mkyong.customer.services.CustomerService">

<property name="message" value="i'm property message" />

</bean>

</beans>

Run it

Java

package com.mkyong.common;

import org.springframework.context.ConfigurableApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.mkyong.customer.services.CustomerService;

public class App

{ public static void main( String[] args )

{ ConfigurableApplicationContext context =

new ClassPathXmlApplicationContext(new String[] {"Spring-Customer.xml"});

CustomerService cust = (CustomerService)context.getBean("customerService");

System.out.println(cust);

context.close();

}

}

The **ConfigurableApplicationContext.close()** will close the application context, releasing all resources and destroying all cached singleton beans. It’s use for destroy() method demo purpose only :)

Output

Bash

Init method after properties are set : im property message

com.mkyong.customer.services.CustomerService@47393f

...

INFO: Destroying singletons in org.springframework.beans.factory.

support.DefaultListableBeanFactory@77158a:

defining beans [customerService]; root of factory hierarchy

Spring Container is destroy! Customer clean up