**Triangle.java**

**package** com.test;

**public** **class** Triangle {

**public** **void** init()

{

System.***out***.println("Init called");

}

**public** **void** destroy()

{

System.***out***.println("Destroy called");

}

**public** **void** draw()

{

System.***out***.println("Triangle draw ");

}

}

Spring1.xml

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"* default-init-method=*"init"* default-destroy-method=*"destroy"*>

<bean id=*"triangle"* class=*"com.test.Triangle"* >

</bean>

</beans>

Testapp.java

**package** com.test;

**import** org.springframework.context.ApplicationContext;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**public** **class** TestApp {

**public** **static** **void** main(String[] args) {

ClassPathXmlApplicationContext ctx = **new** ClassPathXmlApplicationContext("spring1.xml");

Triangle obj = (Triangle)ctx.getBean("triangle");

obj.draw();

/\*try {

obj.destroy();

} catch (Exception e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}\*/

ctx.close();

}

}

Spring.xml

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"*>

<bean id=*"triangle"* class=*"com.test.Triangle"* >

<property name=*"triangleName"* value=*"Equilateral"*/>

<property name=*"sides"* value=*"3"* />

</bean>

<bean id=*"triangle1"* class=*"com.test.Triangle"*>

<constructor-arg index=*"0"* value=*"4"*/>

<constructor-arg type=*"String"* value=*"euilateral"*/>

</bean>

</beans>

BPP

**package** com.test;

**import** org.springframework.beans.BeansException;

**import** org.springframework.beans.factory.config.BeanPostProcessor;

**public** **class** BPP **implements** BeanPostProcessor {

@Override

**public** Object postProcessAfterInitialization(Object obj, String name)

**throws** BeansException {

System.***out***.println("BPP after init called");

**return** obj;

}

@Override

**public** Object postProcessBeforeInitialization(Object obj, String name)

**throws** BeansException {

System.***out***.println("BPP before init called");

**return** obj;

}

}

BFPP

package com.test;

import org.springframework.beans.BeansException;

import org.springframework.beans.factory.config.BeanFactoryPostProcessor;

import org.springframework.beans.factory.config.ConfigurableListableBeanFactory;

public class BFPP implements BeanFactoryPostProcessor {

@Override

public void postProcessBeanFactory(ConfigurableListableBeanFactory arg0)

throws BeansException {

System.out.println("BFPP method called");

}

}

Spring3.xml

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"* default-init-method=*"init"* default-destroy-method=*"destroy"*>

<bean id=*"triangle"* class=*"com.test.Triangle"* >

</bean>

<bean class=*"com.test.BPP"*></bean>

<bean class=*"com.test.BFPP"*></bean>

</beans>

TestApp.java

package com.test;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class TestApp {

public static void main(String[] args) {

ClassPathXmlApplicationContext ctx = new ClassPathXmlApplicationContext("spring3.xml");

Triangle obj = (Triangle)ctx.getBean("triangle");

obj.draw();

/\*try {

obj.destroy();

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}\*/

ctx.close();

}

}

NEW Test app

**package** com.test;

**import** org.springframework.context.support.ClassPathXmlApplicationContext;

**public** **class** TestApp {

**public** **static** **void** main(String[] args) {

ClassPathXmlApplicationContext ctx = **new** ClassPathXmlApplicationContext("spring4.xml");

Triangle2 obj = (Triangle2)ctx.getBean("triangle");

obj.draw();

ctx.close();

}

}

Spring4.xml

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"* default-init-method=*"init"* default-destroy-method=*"destroy"*>

<bean id=*"triangle"* class=*"com.test.Triangle2"*>

<property name=*"point1"* ref=*"point1"* />

<property name=*"point2"* ref=*"point2"* />

<property name=*"point3"* ref=*"point3"* />

</bean>

<bean id=*"point1"* class=*"com.test.Point"*>

<property name=*"x"* value=*"10"*/>

<property name=*"y"* value=*"20"*/>

</bean>

<bean id=*"point2"* class=*"com.test.Point"*>

<property name=*"x"* value=*"30"*/>

<property name=*"y"* value=*"40"*/>

</bean>

<bean id=*"point3"* class=*"com.test.Point"*>

<property name=*"x"* value=*"50"*/>

<property name=*"y"* value=*"60"*/>

</bean>

</beans>

Triangle 2

**package** com.test;

**public** **class** Triangle2 {

**private** Point point1;

**private** Point point2;

**private** Point point3;

**public** Point getPoint1() {

**return** point1;

}

**public** **void** setPoint1(Point point1) {

**this**.point1 = point1;

}

**public** Point getPoint2() {

**return** point2;

}

**public** **void** setPoint2(Point point2) {

**this**.point2 = point2;

}

**public** Point getPoint3() {

**return** point3;

}

**public** **void** setPoint3(Point point3) {

**this**.point3 = point3;

}

**public** **void** draw()

{

System.***out***.println("Inside draw ");

System.***out***.println(point1.getX()+" "+point1.getY());

System.***out***.println(point2.getX()+" "+point2.getY());

System.***out***.println(point3.getX()+" "+point3.getY());

}

}

Point

**package** com.test;

**public** **class** Point {

**private** **int** x;

**private** **int** y;

**public** **int** getX() {

**return** x;

}

**public** **void** setX(**int** x) {

**this**.x = x;

}

**public** **int** getY() {

**return** y;

}

**public** **void** setY(**int** y) {

**this**.y = y;

}

}

New spring4.xml

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"* default-init-method=*"init"* default-destroy-method=*"destroy"*>

<bean id=*"triangle"* class=*"com.test.Triangle2"* autowire=*"constructor"*>

</bean>

<bean id=*"point1"* class=*"com.test.Point"*>

<property name=*"x"* value=*"10"*/>

<property name=*"y"* value=*"20"*/>

</bean>

<bean id=*"point2"* class=*"com.test.Point"*>

<property name=*"x"* value=*"30"*/>

<property name=*"y"* value=*"40"*/>

</bean>

<bean id=*"point3"* class=*"com.test.Point"*>

<property name=*"x"* value=*"50"*/>

<property name=*"y"* value=*"60"*/>

</bean>

</beans>

Triangle 2

**package** com.test;

**public** **class** Triangle2 {

**private** Point point1;

**private** Point point2;

**private** Point point3;

**public** Triangle2(Point point1, Point point2, Point point3) {

**super**();

**this**.point1 = point1;

**this**.point2 = point2;

**this**.point3 = point3;

}

**public** Triangle2() {

**super**();

}

**public** Point getPoint1() {

**return** point1;

}

**public** **void** setPoint1(Point point1) {

**this**.point1 = point1;

}

**public** Point getPoint2() {

**return** point2;

}

**public** **void** setPoint2(Point point2) {

**this**.point2 = point2;

}

**public** Point getPoint3() {

**return** point3;

}

**public** **void** setPoint3(Point point3) {

**this**.point3 = point3;

}

**public** **void** draw()

{

System.***out***.println("Inside draw ");

System.***out***.println(point1.getX()+" "+point1.getY());

System.***out***.println(point2.getX()+" "+point2.getY());

System.***out***.println(point3.getX()+" "+point3.getY());

}

}

How to inject collection

<beans xmlns=*"http://www.springframework.org/schema/beans"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans*

*http://www.springframework.org/schema/beans/spring-beans-3.0.xsd*

*http://www.springframework.org/schema/context*

*http://www.springframework.org/schema/context/spring-context-3.0.xsd"* default-init-method=*"init"* default-destroy-method=*"destroy"*>

<bean id=*"triangle"* class=*"com.test.Triangle2"* autowire=*"default"*>

<property name=*"points"*>

<list>

<ref bean=*"point1"*/>

<ref bean=*"point2"*/>

<ref bean=*"point3"*/>

</list>

</property>

</bean>

<bean id=*"point1"* class=*"com.test.Point"*>

<property name=*"x"* value=*"10"*/>

<property name=*"y"* value=*"20"*/>

</bean>

<bean id=*"point2"* class=*"com.test.Point"*>

<property name=*"x"* value=*"20"*/>

<property name=*"y"* value=*"30"*/>

</bean>

<bean id=*"point3"* class=*"com.test.Point"*>

<property name=*"x"* value=*"100"*/>

<property name=*"y"* value=*"200"*/>

</bean>

</beans>

Triangle2

**package** com.test;

**import** java.util.List;

**public** **class** Triangle2 {

**private** List<Point> points;

**public** List<Point> getPoints() {

**return** points;

}

**public** **void** setPoints(List<Point> points) {

**this**.points = points;

}

**public** **void** draw()

{

System.***out***.println("Inside draw ");

**for**(Point p : points)

{

System.***out***.println(p.getX()+" "+p.getY());

}

}

}