**Spring Assignment**

1. **For Setter Injection Main Class SpringQuestion1Test**
2. **package** springcore.com;
3. **import** org.springframework.context.ApplicationContext;
4. **import** org.springframework.context.support.ClassPathXmlApplicationContext;
5. **public** **class** SpringQuestion1
6. {
7. @SuppressWarnings("unused")
8. **private** **static** ApplicationContext *context*;
9. **public** **static** **void** main(String[] args)
10. {
11. @SuppressWarnings("resource")
12. ApplicationContext context = **new** ClassPathXmlApplicationContext("AddressBean.xml");
14. Customer customer = (Customer)context.getBean("Customer");
15. System.***out***.println(customer);
17. }
19. }

**Address.java**

**package** springcore.com;

**public** **class** Address {

**private** String city, street, state, country;

**private** **int** zip;

**public** String getCity()

{

**return** city;

}

**public** **void** setCity(String city)

{

**this**.city = city;

}

**public** String getStreet()

{

**return** street;

}

**public** **void** setStreet(String street)

{

**this**.street = street;

}

**public** String getState()

{

**return** state;

}

**public** **void** setState(String state)

{

**this**.state = state;

}

**public** String getCountry()

{

**return** country;

}

**public** **void** setCountry(String country)

{

**this**.country = country;

}

**public** **int** getZip()

{

**return** zip;

}

**public** **void** setZip(**int** zip)

{

**this**.zip = zip;

}

}

**AddressBean.xml**

<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-3.0.xsd"*>

<bean id = *"Address"* class = *"springcore.com.Address"*>

<property name = *"zip"* value = *"221"*/>

<property name = *"city"* value = *"London"*/>

<property name = *"street"* value = *"Baker Street"*/>

<property name = *"state"* value = *"World"*/>

<property name = *"country"* value = *"England"*/>

</bean>

<bean id = *"Customer"* class = *"springcore.com.Customer"*>

<property name = *"customerId"* value = *"123"*/>

<property name = *"customercontact"* value = *"123456"*/>

<property name = *"customerAddress"* ref = *"Address"*/>

<property name = *"customerName"* value = *"Sherlock Holmes"*/>

</bean>

</beans>

**Customer.java**

**package** springcore.com;

**public** **class** Customer {

**private** **int** customerId;

**private** **int** customercontact;

**private** String customerName;

**private** Address customerAddress;

**public** **int** getCustomerId()

{

**return** customerId;

}

**public** **void** setCustomerId(**int** customerId)

{

**this**.customerId = customerId;

}

**public** **int** getCustomercontact()

{

**return** customercontact;

}

**public** **void** setCustomercontact(**int** customercontact)

{

**this**.customercontact = customercontact;

}

**public** Address getCustomerAddress()

{

**return** customerAddress;

}

**public** **void** setCustomerAddress(Address customerAddress)

{

**this**.customerAddress = customerAddress;

}

**public** String getCustomerName()

{

**return** customerName;

}

**public** **void** setCustomerName(String customerName)

{

**this**.customerName = customerName;

}

**public** String toString()

{

**return** "Customer Id: "+getCustomerId()+ ", Customer Contact: "+getCustomercontact()+ ", Customer Address: {City: "+getCustomerAddress().getCity()+", Street: "+getCustomerAddress().getStreet()+", State: "+getCustomerAddress().getState()+", Country: "+getCustomerAddress().getCountry()+", Zip: "+getCustomerAddress().getZip()+"}, Customer Name: "+getCustomerName();

}

}

**For Constructor Main Class SpringQuestion1Test**

**package** springcore.com;

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

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

**public** **class** SpringQuestion1

{

@SuppressWarnings("unused")

**private** **static** ApplicationContext *context*;

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

{

@SuppressWarnings("resource")

ApplicationContext context1 = **new** ClassPathXmlApplicationContext("Customer.xml");

Customer customer2 = (Customer)context1.getBean("Customer1");

System.***out***.println(customer2);

}

}

**For Constructor Address.java**

**package** springcore.com;

**public** **class** Address {

**private** String city, street, state, country;

**private** **int** zip;

**public** String getCity()

{

**return** city;

}

**public** **void** setCity(String city)

{

**this**.city = city;

}

**public** String getStreet()

{

**return** street;

}

**public** **void** setStreet(String street)

{

**this**.street = street;

}

**public** String getState()

{

**return** state;

}

**public** **void** setState(String state)

{

**this**.state = state;

}

**public** String getCountry()

{

**return** country;

}

**public** **void** setCountry(String country)

{

**this**.country = country;

}

**public** **int** getZip()

{

**return** zip;

}

**public** **void** setZip(**int** zip)

{

**this**.zip = zip;

}

**public** Address(String city, String street, String state, String country, **int** zip) {

**super**();

**this**.city = city;

**this**.street = street;

**this**.state = state;

**this**.country = country;

**this**.zip = zip;

}

}

**For Constructor Customer.java**

**package** springcore.com;

**public** **class** Customer

{

**private** **int** customerId;

**private** **int** customercontact;

**private** String customerName;

**private** Address customerAddress;

**public** Customer(**int** customerId, **int** customercontact, String customerName, Address customerAddress)

{

**this**.customerId = customerId;

**this**.customercontact = customercontact;

**this**.customerName = customerName;

**this**.customerAddress = customerAddress;

}

**public** **int** getCustomerId()

{

**return** customerId;

}

**public** **void** setCustomerId(**int** customerId)

{

**this**.customerId = customerId;

}

**public** **int** getCustomercontact()

{

**return** customercontact;

}

**public** **void** setCustomercontact(**int** customercontact)

{

**this**.customercontact = customercontact;

}

**public** Address getCustomerAddress()

{

**return** customerAddress;

}

**public** **void** setCustomerAddress(Address customerAddress)

{

**this**.customerAddress = customerAddress;

}

**public** String getCustomerName()

{

**return** customerName;

}

**public** **void** setCustomerName(String customerName)

{

**this**.customerName = customerName;

}

**public** String toString()

{

**return** "Customer Id: "+getCustomerId()+ ", Customer Contact: "+getCustomercontact()+ ", Customer Address: {City: "+getCustomerAddress().getCity()+", Street: "+getCustomerAddress().getStreet()+", State: "+getCustomerAddress().getState()+", Country: "+getCustomerAddress().getCountry()+", Zip: "+getCustomerAddress().getZip()+"}, Customer Name: "+getCustomerName();

}

}

**For Constructor Customer.xml**

<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-3.0.xsd"*>

<bean id = *"Address"* class = *"springcore.com.Address"*>

<constructor-arg index = *"0"* name = *"city"* type = *"java.lang.String"* value = *"London"*/>

<constructor-arg index = *"1"* name = *"street"* type = *"java.lang.String"* value = *"Baker Street"*/>

<constructor-arg index = *"2"* name = *"state"* type = *"java.lang.String"* value = *"World"*/>

<constructor-arg index = *"3"* name = *"country"* type = *"java.lang.String"* value = *"England"*/>

<constructor-arg index = *"4"* name = *"zip"* type = *"int"* value = *"221"*/>

</bean>

<bean id = *"Customer1"* class = *"springcore.com.Customer"*>

<constructor-arg index = *"0"* name = *"customerId"* type = *"int"* value = *"123"*/>

<constructor-arg index = *"1"* name = *"customercontact"* type = *"int"* value = *"123456"*/>

<constructor-arg index = *"2"* name = *"customerName"* type = *"java.lang.String"* value = *"Sherlock Holmes"*/>

<constructor-arg index = *"3"* name = *"customerAddress"* ref = *"Address"*/>

</bean>

</beans>

**Main Class Question2.java**

**a.List**

**package** Spring\_Core;

**import** **static** org.junit.jupiter.api.Assertions.\*;

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

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

**import** org.junit.jupiter.api.Test;

**class** Question2

{

@SuppressWarnings("unused")

**private** **static** ApplicationContext *context*;

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

{

@SuppressWarnings("resource")

ApplicationContext context = **new** ClassPathXmlApplicationContext("Coll.xml");

List list = (List) context.getBean("Collections");

System.***out***.println("List: " + list.getList());

}

}

**Question2.java**

**package** Spring\_Core;

**public** **class** Question

{

**private** **int** questionId;

**private** String question, answers;

**public** **void** setQuestionId(**int** questionId) {

**this**.questionId = questionId;

}

**public** **void** setQuestion(String question) {

**this**.question = question;

}

**public** **void** setAnswers(String answers) {

**this**.answers = answers;

}

**public** String toString()

{

**return** "Question ID: "+questionId+ ", Questions: "+question+ ", Answers: "+answers+"\n";

}

}

**List.java**

package Spring\_Core;

import java.util.ArrayList;

import java.util.Map;

import java.util.Set;

public class List

{

private ArrayList<Object> list;

private Set<Object> set;

private Map<Object, Object> map;

public ArrayList<Object> getList() {

return list;

}

public void setList(ArrayList<Object> list) {

this.list = list;

}

public Set<Object> getSet() {

return set;

}

public void setSet(Set<Object> set) {

this.set = set;

}

public Map<Object, Object> getMap() {

return map;

}

public void setMap(Map<Object, Object> map) {

this.map = map;

}

}

**Coll.xml**

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

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

xmlns:aop=*"http://www.springframework.org/schema/aop"* xmlns:context=*"http://www.springframework.org/schema/context"*

xmlns:jee=*"http://www.springframework.org/schema/jee"* xmlns:tx=*"http://www.springframework.org/schema/tx"*

xmlns:task=*"http://www.springframework.org/schema/task"* xsi:schemaLocation=*"http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-3.2.xsd http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-3.2.xsd http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-3.2.xsd http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee-3.2.xsd http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-3.2.xsd http://www.springframework.org/schema/task http://www.springframework.org/schema/task/spring-task-3.2.xsd"*>

<bean id=*"Collections"* class=*"Spring\_Core.List"*>

<property name=*"list"*>

<list>

<ref bean=*"answers"* />

<ref bean=*"answers1"* />

<ref bean=*"answers2"* />

</list>

</property>

</bean>

<bean id=*"answers"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*”Teja"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"123"*/>

</bean>

<bean id=*"answers1"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Ritu"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"456"*/>

</bean>

<bean id=*"answers2"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Manu"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"789"*/>

</bean>

</beans>

**Main Class Question2.java**

**b.set**

**package** Spring\_Core;

**import** **static** org.junit.jupiter.api.Assertions.\*;

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

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

**import** org.junit.jupiter.api.Test;

**class** Question2

{

@SuppressWarnings("unused")

**private** **static** ApplicationContext *context*;

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

{

@SuppressWarnings("resource")

ApplicationContext context = **new** ClassPathXmlApplicationContext("Coll.xml");

List set = (List) context.getBean("Collections");

System.***out***.println("Set : " + set.getSet());

}

}

**Coll.xml**

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

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

xmlns:aop=*"http://www.springframework.org/schema/aop"* xmlns:context=*"http://www.springframework.org/schema/context"*

xmlns:jee=*"http://www.springframework.org/schema/jee"* xmlns:tx=*"http://www.springframework.org/schema/tx"*

xmlns:task=*"http://www.springframework.org/schema/task"* xsi:schemaLocation=*"http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-3.2.xsd http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-3.2.xsd http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-3.2.xsd http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee-3.2.xsd http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-3.2.xsd http://www.springframework.org/schema/task http://www.springframework.org/schema/task/spring-task-3.2.xsd"*>

<bean id=*"Collections"* class=*"Spring\_Core.List"*>

<property name=*"set"*>

<set>

<ref bean=*"answers"* />

<ref bean=*"answers1"* />

<ref bean=*"answers2"* />

</set>

</property>

</bean>

<bean id=*"answers"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Teja"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"123"*/>

</bean>

<bean id=*"answers1"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Ritu"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"456"*/>

</bean>

<bean id=*"answers2"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Manu"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"789"*/>

</bean>

</beans>

**Main Class Question2.java**

**c.map**

**package** Spring\_Core;

**import** **static** org.junit.jupiter.api.Assertions.\*;

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

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

**import** org.junit.jupiter.api.Test;

**class** Question2

{

@SuppressWarnings("unused")

**private** **static** ApplicationContext *context*;

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

{

@SuppressWarnings("resource")

ApplicationContext context = **new** ClassPathXmlApplicationContext("Coll.xml");

List map = (List) context.getBean("Collections");

System.***out***.println("Map : " + map.getMap());

}

}

**Coll.xml**

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

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

xmlns:aop=*"http://www.springframework.org/schema/aop"* xmlns:context=*"http://www.springframework.org/schema/context"*

xmlns:jee=*"http://www.springframework.org/schema/jee"* xmlns:tx=*"http://www.springframework.org/schema/tx"*

xmlns:task=*"http://www.springframework.org/schema/task"* xsi:schemaLocation=*"http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-3.2.xsd http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-3.2.xsd http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-3.2.xsd http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee-3.2.xsd http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-3.2.xsd http://www.springframework.org/schema/task http://www.springframework.org/schema/task/spring-task-3.2.xsd"*>

<bean id=*"Collections"* class=*"Spring\_Core.List"*>

<property name=*"map"*>

<map>

<entry key=*"1"* value-ref=*"answers"* />

<entry key=*"2"* value-ref=*"answers1"* />

<entry key=*"3"* value-ref=*"answers2"* />

</map>

</property>

</bean>

<bean id=*"answers"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Teja"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"123"*/>

</bean>

<bean id=*"answers1"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Ritu"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"456"*/>

</bean>

<bean id=*"answers2"* class=*"Spring\_Core.Question"*>

<property name=*"answers"* value=*"Manu"*/>

<property name = *"question"* value = *"What's your name?"*/>

<property name = *"questionId"* value = *"789"*/>

</bean>

</beans>

2d.

package test;

public class BeanClass{

int questionid;

String question;

String answers;

public int getQuestionid() {

return questionid;

}

public void setQuestionid(int questionid) {

this.questionid = questionid;

}

public String getQuestion() {

return question;

}

public void setQuestion(String question) {

this.question = question;

}

public String getAnswers() {

return answers;

}

public void setAnswers(String answers) {

this.answers = answers;

}

}

XML FILE:

Beans.xml

<?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:context="http://www.springframework.org/schema/context"

xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-4.3.xsd

http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.3.xsd"

default-lazy-init="true">

<bean id="myid" class="test.BeanClass">

<property name="questionid" value="1"></property>

<property name="question" value="Java Spring"></property>

<property name="answer" value="Java Spring"></property>

</bean>

</beans>

Main Class

package test;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class Tester {

public static void main(String[] args) {

ApplicationContext context=new ClassPathXmlApplicationContext("Beans.xml");

BeanClass beanClass=(BeanClass) context.getBean("myid");

beanClass.getName();

}

}

@Bean

package spring\_ex4;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

//@Configuration

public class ApplicationConfiguration {

@Bean(name="demoService")

public DemoManager helloWorld()

{

return new DemoManagerImpl();

}

}

package spring\_ex4;

public interface DemoManager {

public String getServiceName();

}

package spring\_ex4;

public class DemoManagerImpl implements DemoManager

{

public String getServiceName()

{

return "Hello!!!!";

}

}

package spring\_ex4;

import org.springframework.context.ApplicationContext;

import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class VerifySpringCoreFeature {

public static void main(String[] args)

{

ApplicationContext context = new AnnotationConfigApplicationContext(ApplicationConfiguration.class);

DemoManager obj = (DemoManager) context.getBean("demoService");

System.out.println( obj.getServiceName() );

}

}

Output:

Jan 25, 2022 2:01:42 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh

INFO: Refreshing org.springframework.context.annotation.AnnotationConfigApplicationContext@20322d26: startup date [Tue Jan 25 14:01:42 IST 2022]; root of context hierarchy

Addition of first and second = 4

Jan 25, 2022 2:01:43 PM org.springframework.context.support.AbstractApplicationContext doClose

INFO: Closing org.springframework.context.annotation.AnnotationConfigApplicationContext@20322d26: startup date [Tue Jan 25 14:01:42 IST 2022]; root of context hierarchy

@contoller

package spring\_exp4;

import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class SpringMainClass {

public static void main(String[] args) {

AnnotationConfigApplicationContext context = new AnnotationConfigApplicationContext();

context.scan("spring\_exp4");

context.refresh();

MathController ms = context.getBean(MathController.class);

int result = ms.add(2, 2);

System.out.println("Addition of first and second = " + result);

context.close();

}

}

package spring\_exp4;

package spring\_exp4;

import org.springframework.stereotype.Controller;

import org.springframework.stereotype.Service;

//@Service("ms")

@Controller

public class MathController {

public int add(int x, int y) {

return x + y;

}

}

Output:

Jan 25, 2022 2:04:52 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh

INFO: Refreshing org.springframework.context.annotation.AnnotationConfigApplicationContext@20322d26: startup date [Tue Jan 25 14:04:52 IST 2022]; root of context hierarchy

Addition of first and second = 4

Jan 25, 2022 2:04:52 PM org.springframework.context.support.AbstractApplicationContext doClose

INFO: Closing org.springframework.context.annotation.AnnotationConfigApplicationContext@20322d26: startup date [Tue Jan 25 14:04:52 IST 2022]; root of context hierarchy

@Service

package spring\_exp4;

import org.springframework.stereotype.Component;

import org.springframework.stereotype.Service;

@Service("ms")

//@Component

public class MathService {

public int add(int x, int y) {

return x + y;

}

} package spring\_exp4;

import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class SpringMainClass {

public static void main(String[] args) {

AnnotationConfigApplicationContext context = new AnnotationConfigApplicationContext();

context.scan("spring\_exp4");

context.refresh();

MathService ms = context.getBean(MathService.class);

int result = ms.add(2, 2);

System.out.println("Addition of first and second = " + result);

context.close();

}

}

Output:

Jan 25, 2022 2:01:42 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh

INFO: Refreshing org.springframework.context.annotation.AnnotationConfigApplicationContext@20322d26: startup date [Tue Jan 25 14:01:42 IST 2022]; root of context hierarchy

Addition of first and second = 4

Jan 25, 2022 2:01:43 PM org.springframework.context.support.AbstractApplicationContext doClose

INFO: Closing org.springframework.context.annotation.AnnotationConfigApplicationContext@20322d26: startup date [Tue Jan 25 14:01:42 IST 2022]; root of context hierarchy

@Autowired

package maths\_example;

import org.springframework.beans.factory.BeanFactory;

// org.springframework.beans.factory.xml.XmlBeanFactory;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Repository;

//import org.springframework.core.io.FileSystemResource;

//@Repository

public class Mainbean {

public static void main(String[] args) {

// TODO Auto-generated method stub

//BeanFactory factory= new XmlBeanFactory(new FileSystemResource("spring.xml"));

ApplicationContext context=new ClassPathXmlApplicationContext("spring.xml");

Shape shape=(Shape)context.getBean("circle");

shape.draw();

}

}

package maths\_example;

public class Triangle implements Shape {

private Point PointA;

private Point PointB;

private Point PointC;

public Point getPointA() {

return PointA;

}

public void setPointA(Point pointA) {

PointA = pointA;

}

public Point getPointB() {

return PointB;

}

public void setPointB(Point pointB) {

PointB = pointB;

}

public Point getPointC() {

return PointC;

}

public void setPointC(Point pointC) {

PointC = pointC;

}

public void draw()

{

System.out.println("Draw triangle");

System.out.println(getPointA().getX()+ " "+getPointA().getY());

System.out.println(getPointB().getX()+ " "+getPointB().getY());

System.out.println(getPointC().getX()+ " "+getPointC().getY());

}

} package maths\_example;

public interface Shape {

public void draw();

}

package maths\_example;

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;

}

}

import org.springframework.beans.factory.annotation.Required;

import org.springframework.stereotype.Component;

import org.springframework.stereotype.Repository;

//@Component

//@Repository

public class Circle implements Shape {

private Point center;

public void draw()

{

System.out.println("draw circle");

System.out.println("circle point" +center.getX() +center.getY());

}

public Point getCenter() {

return center;

}

//@Autowired

public void setCenter(Point center) {

this.center = center;

}

}

<?xml version="1.0" encoding="UTF-8"?>

<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-3.0.xsd">

<bean id = "triangle" class="maths\_example.Triangle">

<property name="PointA" ref="pointA"/>

<property name="PointB" ref="pointB"/>

<property name="PointC" ref="pointC"/>

</bean>

<bean id = "pointA" class="maths\_example.Point">

<property name="x" value="0"/>

<property name="y" value="10"/>

</bean>

<bean id = "pointB" class="maths\_example.Point">

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

<property name="y" value="10"/>

</bean>

<bean id = "pointC" class="maths\_example.Point">

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

<property name="y" value="10"/>

</bean>

<bean id = "center" class="maths\_example.Point">

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

<property name="y" value="10"/>

</bean>

<bean id = "circle" class="maths\_example.Circle">

<!-- <property names="center" ref="pointA"/> -->

</bean>

<bean class="org.springframework.beans.factory.annotation.AutowiredAnnotationBeanPostProcessor">

</bean>

</beans>

Output:

Jan 25, 2022 3:22:45 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh

INFO: Refreshing org.springframework.context.support.ClassPathXmlApplicationContext@2038ae61: startup date [Tue Jan 25 15:22:45 IST 2022]; root of context hierarchy

Jan 25, 2022 3:22:46 PM org.springframework.beans.factory.xml.XmlBeanDefinitionReader loadBeanDefinitions

INFO: Loading XML bean definitions from class path resource [spring4.xml]

draw circle

circle point2010

5. **@Resource Annotation**

**CLASS 1**

**import** org.springframework.beans.factory.annotation.Required;

**public** **class** Employee {

**private** String id;

**private** String name;

@Resource(name="mycompany")

**private** Company company;

**public** String getId() {

**return** id;

}

**public** **void** setId(String id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

@Required

**public** **void** setName(String name) {

**this**.name = name;

}

**public** Company getCompany() {

**return** company;

}

**public** **void** setCompany(Company company) {

**this**.company = company;

}

@Override

**public** String toString() {

**return** "Employee [id=" + id + ", name=" + name + ", company=" + company.toString() + "]";

}

}

CLASS 2

**package** New;

**public** **class** Company {

**private** String name;

**private** String location;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getLocation() {

**return** location;

}

**public** **void** setLocation(String location) {

**this**.location = location;

}

@Override

**public** String toString() {

**return** "Company [name=" + name + ", location=" + location + "]";

}

}

CLASS3

**package** New;

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

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

**public** **class** Test {

@SuppressWarnings("resource")

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

ApplicationContext context = **new** ClassPathXmlApplicationContext("n.xml");

Employee emp=(Employee) context.getBean("myemployee");

System.***out***.println(emp.toString());

}

}

XML FILE

<?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:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"*

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

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

<!-- To activate the '@Resource' annotation in the spring framework -->

<context:annotation-config />

<bean id=*"mycompany"* class=*"New.Company"*>

<property name=*"name"* value=*"Test Pvt. Ltd."* />

<property name=*"location"* value=*"India"* />

</bean>

<bean id=*"myemployee"* class=*"New.Employee"*>

<property name=*"id"* value=*"126"* />

<property name=*"name"* value=*"Nam"* />

</bean></beans>

@Inject Annotataion

Class 1

**public** **class** Employee {

**private** String id;

**private** String name;

@Inject

**private** Company company;

**public** String getId() {

**return** id;

}

**public** **void** setId(String id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

@Required

**public** **void** setName(String name) {

**this**.name = name;

}

**public** Company getCompany() {

**return** company;

}

**public** **void** setCompany(Company company) {

**this**.company = company;

}

@Override

**public** String toString() {

**return** "Employee [id=" + id + ", name=" + name + ", company=" + company.toString() + "]";

}

}

CLASS 2

**package** New;

**public** **class** Company {

**private** String name;

**private** String location;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getLocation() {

**return** location;

}

**public** **void** setLocation(String location) {

**this**.location = location;

}

@Override

**public** String toString() {

**return** "Company [name=" + name + ", location=" + location + "]";

}

}

CLASS 3

**package** New;

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

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

**public** **class** Test {

@SuppressWarnings("resource")

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

ApplicationContext context = **new** ClassPathXmlApplicationContext("n.xml");

Employee emp=(Employee) context.getBean("myemployee");

System.***out***.println(emp.toString());

}

}

XML FILE

<?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:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"*

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

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

<!-- To activate the '@Resource' annotation in the spring framework -->

<context:annotation-config />

<bean id=*"mycompany"* class=*"New.Company"*>

<property name=*"name"* value=*"Test Pvt. Ltd."* />

<property name=*"location"* value=*"India"* />

</bean>

<bean id=*"myemployee"* class=*"New.Employee"*>

<property name=*"id"* value=*"126"* />

<property name=*"name"* value=*"Nam"* />

</bean>

</beans>

OUTPUT:

Jan 25, 2022 2:27:39 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh

INFO: Refreshing org.springframework.context.support.ClassPathXmlApplicationContext@3b0143d3: startup date [Tue Jan 25 14:27:39 IST 2022]; root of context hierarchy

Jan 25, 2022 2:27:39 PM org.springframework.beans.factory.xml.XmlBeanDefinitionReader loadBeanDefinitions

INFO: Loading XML bean definitions from class path resource [n.xml]

Jan 25, 2022 2:27:39 PM org.springframework.beans.factory.annotation.AutowiredAnnotationBeanPostProcessor <init>

INFO: JSR-330 'javax.inject.Inject' annotation found and supported for autowiring

Employee [id=126, name=Nam, company=Company [name=Test Pvt. Ltd., location=India]]

Spring core Question 7:

Test.java

**package** com.maven7;

**import** org.springframework.expression.Expression;

**import** org.springframework.expression.ExpressionParser;

**import** org.springframework.expression.spel.standard.SpelExpressionParser;

**public** **class** Test {

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

/\*

ExpressionParser parser = new SpelExpressionParser();

Expression exp = parser.parseExpression("'Hello Maven'");

String message = (String)exp.getValue();

System.out.println(message);

\*/

/\*

ExpressionParser parser = new SpelExpressionParser();

Expression exp = parser.parseExpression("'Welcome To'.concat('SpringExpression')");

String message = (String) exp.getValue(); System.out.println(message);

\*/

/\*

ExpressionParser parser = new SpelExpressionParser();

Expression exp = parser.parseExpression("'Hello Abhi'.bytes");

byte[] bytes = (byte[]) exp.getValue();

for(int i=0;i<bytes.length;i++) {

System.out.print(bytes[i]+" ");

}

\*/

/\*

ExpressionParser parser = new SpelExpressionParser();

Expression exp =

parser.parseExpression("new String('hello world').toUpperCase()");

String message = exp.getValue(String.class); System.out.println(message);

\*/

ExpressionParser parser = **new** SpelExpressionParser();

Expression exp = parser.parseExpression("'Hello Abhi'.bytes.length");

**int** length = (Integer) exp.getValue();

System.***out***.println(length);

}

}

Pom.xml

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.maven7</groupId>

<artifactId>SPELexpression</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>5.3.15</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.springframework/spring-expression -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-expression</artifactId>

<version>5.3.15</version>

</dependency>

</dependencies>

</project>

8th question:

1 Way:

ANS:

Class 1:

**package** QuestionEight;

**import** org.springframework.beans.factory.DisposableBean;

**import** org.springframework.beans.factory.InitializingBean;

**public** **class** CustomerService **implements** InitializingBean, DisposableBean {

**private** String msg;

**public** String getMsg() {

**return** msg;

}

**public** **void** setMsg(String msg) {

**this**.msg = msg;

}

**public** **void** destroy() **throws** Exception {

// **TODO** Auto-generated method stub

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

}

**public** **void** afterPropertiesSet() **throws** Exception {

// **TODO** Auto-generated method stub

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

}}

Class 2:

**package** QuestionEight;

**import** org.springframework.context.ConfigurableApplicationContext;

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

**public** **class** Snippet {

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

ConfigurableApplicationContext context = **new** ClassPathXmlApplicationContext(

**new** String[] { "SpringCustomer.xml" });

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

System.***out***.println(cust);

context.close();

}

}

XML FILE

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<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=*"QuestionEight.CustomerService"*>

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

</bean>

</beans>

OUTPUT:

Init method after properties are set : i'm property message

Spring Container is destroy! Customer clean up

2nd Way:

Class1:

**package** EightB;

**import** javax.annotation.PostConstruct;

**import** javax.annotation.PreDestroy;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Component;

@Component

**public** **class** MyBean {

**public** MyBean() {

System.***out***.println("MyBean instance created");

}

@PostConstruct

**private** **void** init() {

System.***out***.println("Verifying Resources");

}

@PreDestroy

**private** **void** shutdown() {

System.***out***.println("Shutdown All Resources");

}

**public** **void** close() {

System.***out***.println("Closing All Resources");

}

}

Class 2:

**package** EightB;

**import** org.springframework.context.annotation.Bean;

**import** org.springframework.context.annotation.Configuration;

**import** org.springframework.context.annotation.Scope;

@Configuration

**public** **class** MyConfiguration {

@Bean

@Scope(value = "singleton")

**public** MyBean myBean() {

**return** **new** MyBean();

}

}

Class 3:

**package** EightB;

**import** org.springframework.context.annotation.AnnotationConfigApplicationContext;

**public** **class** SpringApp {

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

AnnotationConfigApplicationContext ctx = **new** AnnotationConfigApplicationContext();

ctx.register(MyConfiguration.**class**);

ctx.refresh();

MyBean mb1 = ctx.getBean(MyBean.**class**);

System.***out***.println(mb1.hashCode());

MyBean mb2 = ctx.getBean(MyBean.**class**);

System.***out***.println(mb2.hashCode());

ctx.close();

}

}

OUTPUT:

MyBean instance created

Verifying Resources

2145970759

2145970759

Shutdown All Resources

Closing All Resources

9. Write a Java program to demonstrate complete Bean Life cycle .

ANS:

**public** **class** HelloWorld {

**public** **void** init() **throws** Exception

{

System.***out***.println(

"Bean HelloWorld has been "

+ "instantiated and I'm "

+ "the init() method");

}

// This method executes

// when the spring container

// is closed

**public** **void** destroy() **throws** Exception

{

System.***out***.println(

"Container has been closed "

+ "and I'm the destroy() method");

}

}

Class 2: Main method Class

**import** org.springframework.context.ConfigurableApplicationContext;

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

**public** **class** Client {

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

// Loading the Spring XML configuration

// file into the spring container and

// it will create the instance of

// the bean as it loads into container

ConfigurableApplicationContext cap = **new** ClassPathXmlApplicationContext("spring.xml");

// It will close the spring container

// and as a result invokes the

// destroy() method

cap.close();

}

}

XML file :

<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.xsd"*>

<bean id=*"HelloBean"* class=*"springcore\_example.HelloBean"*>

<property name =*"name"* value=*"sarika"*>

</property>

</bean>

</beans>

OUTPUT:

Bean HelloWorld has been instantiated and I'm the init() method

Container has been closed and I'm the destroy() method

10 question:

drawing class

package com.springdemo.firstmodule;

import org.springframework.beans.factory.BeanFactory;

import org.springframework.beans.factory.xml.XmlBeanFactory;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.core.io.FileSystemResource;

public class DrawingApp {

public static void main(String[] args) {

// TODO Auto-generated method stub

//Triangle triangle=new Triangle();

//BeanFactory factory=new XmlBeanFactory(new FileSystemResource("spring.xml"));

ApplicationContext context=new ClassPathXmlApplicationContext("spring.xml");

Triangle triangle=(Triangle) context.getBean("triangle");

triangle.draw();

}

}

traingle class:

**package** com.springdemo.firstmodule;

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

**import** org.springframework.beans.factory.BeanNameAware;

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

**import** org.springframework.context.ApplicationContextAware;

**import** java.util.List;

**public** **class** Triangle **implements** ApplicationContextAware, BeanNameAware {

**private** Point pointA;

**private** Point pointB;

**private** Point pointC;

**private** ApplicationContext context=**null**;

/\*private String type;

private int height;

public int getHeight() {

return height;

}

public Triangle(String type) {

this.type=type;

}

public Triangle(int height) {

this.height=height;

}

public Triangle(String type,int height) {

this.type=type;

this.height=height;

}

public String getType() {

return type;

}\*/

/\*public void setType(String type) {

this.type = type;

}

\*/

**public** Point getPointA() {

**return** pointA;

}

**public** **void** setPointA(Point pointA) {

**this**.pointA = pointA;

}

**public** Point getPointB() {

**return** pointB;

}

**public** **void** setPointB(Point pointB) {

**this**.pointB = pointB;

}

**public** Point getPointC() {

**return** pointC;

}

**public** **void** setPointC(Point pointC) {

**this**.pointC = pointC;

}

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

//System.out.println(getType()+" triangle drawn of height" + getHeight());

System.***out***.println("Point A= ( " + getPointA().getX() +" ,"+ getPointA().getY() + ")");

System.***out***.println("Point B= ( " + getPointB().getX() +" ,"+ getPointB().getY() + ")");

System.***out***.println("Point C= ( " + getPointC().getX() +" ,"+ getPointC().getY() + ")");

}

@Override

**public** **void** setApplicationContext(ApplicationContext context) **throws** BeansException {

// **TODO** Auto-generated method stub

**this**.context=context;

}

@Override

**public** **void** setBeanName(String beanName) {

// **TODO** Auto-generated method stub

System.***out***.println("Bean name is:" + beanName);

}

}

Point class:

**package** com.springdemo.firstmodule;

**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;

}

}

Spring.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN 2.0//EN" "http://www.springframework.org/dtd/spring-beans-2.0.dtd">

<beans>

<bean id=*"triangle"* class=*"com.springdemo.firstmodule.Triangle"* autowire=*"byName"*>

</bean>

<bean id=*"pointA"* class=*"com.springdemo.firstmodule.Point"*>

<property name=*"x"* value=*"0"* />

<property name=*"y"* value=*"0"* />

</bean>

<bean id=*"pointB"* class=*"com.springdemo.firstmodule.Point"*>

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

<property name=*"y"* value=*"0"* />

</bean>

<bean id=*"pointC"* class=*"com.springdemo.firstmodule.Point"*>

<property name=*"x"* value=*"0"* />

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

</bean>

</beans>