Q1. Spring Setter Injection Injectiong address from Address class customer class.

**package** com.helloworldexample;

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

**private** String street;

**private** String city;

**private** String state;

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

**private** String country;

**public** String getStreet() {

**return** street;

}

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

**this**.street = street;

}

**public** String getCity() {

**return** city;

}

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

**this**.city = city;

}

**public** String getState() {

**return** state;

}

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

**this**.state = state;

}

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

**return** zip;

}

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

**this**.zip = zip;

}

**public** String getCountry() {

**return** country;

}

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

**this**.country = country;

}

@Override

**public** String toString() {

**return** "Address [street=" + street + ", city=" + city + ", state=" + state + ", zip=" + zip + ", country="

+ country + "]";

}

}

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

**private** **int** id;

**private** String name;

**private** **long** contact;

**private** Address address;

**public** **int** getId() {

**return** id;

}

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

**this**.id = id;

}

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

**public** **long** getContact() {

**return** contact;

}

**public** **void** setContact(**long** contact) {

**this**.contact = contact;

}

**public** Address getAddress() {

**return** address;

}

**public** **void** setAddress(Address address) {

**this**.address = address;

}

**void** display() {

System.***out***.println(id+ " " +name+ " " +contact+ " " +address);

}

}

Main File code

package com.helloworldexample;

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.ClassPathResource;

import org.springframework.core.io.Resource;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

**package** com.helloworldexample;

**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.ClassPathResource;

**import** org.springframework.core.io.Resource;

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

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

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

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

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

Customer customer=(Customer)context.getBean("customer");

customer.display();

}

}

Spring.xml code

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

<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN"

"http://www.springframework.org/dtd/spring-beans.dtd">

<beans>

<bean id = *"customer\_address"* class= *"com.helloworldexample.Address"*>

<property name=*"street"* value=*"Peddar Road"*></property>

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

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

<property name=*"zip"* value=*"400026"*></property>

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

</bean>

<bean id = *"customer"* class= *"com.helloworldexample.Customer"*>

<property name=*"id"* value=*"1"*></property>

<property name=*"name"* value=*"Riddhi Rathod"*></property>

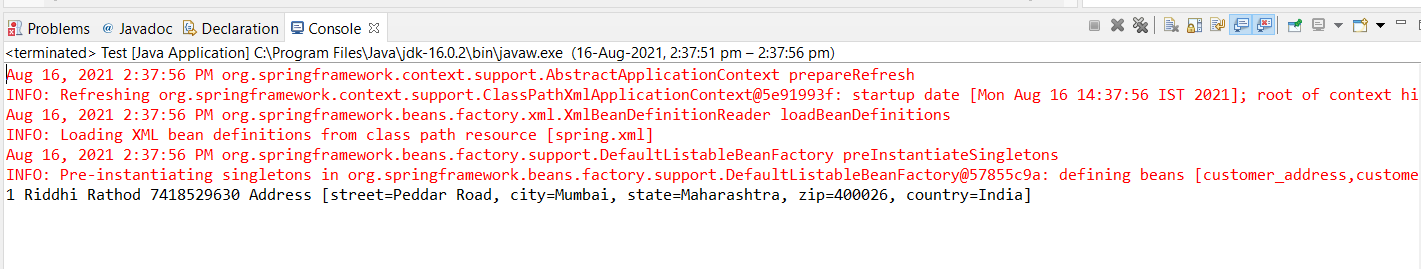
<property name=*"contact"* value=*"7418529630"*></property>

<property name=*"address"* ref=*"customer\_address"*></property>

</bean>

</beans>

Output:



Q9 : Bean Lifecycle Methods

**package** com.helloworldexample;

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

// This method executes

// automatically as the bean

// is instantiated

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

{

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

"Bean HelloWorld has been "

+ "instantiated and "

+ "init() method");

}

// This method executes

// when the spring container

// is closed

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

{

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

"Conatiner has been closed "

+ "and destroy() method");

}

}

**package** com.helloworldexample;

**import** org.springframework

.context

.ConfigurableApplicationContext;

**import** org.springframework

.context.support

.ClassPathXmlApplicationContext;

Main Class

**public** **class** HelloMain {

**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 display

= **new** ClassPathXmlApplicationContext(

"spring.xml");

// It will close the spring container

// and as a result invokes the

// destroy() method

display.close();

}

}

Xml file code

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

<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN"

"http://www.springframework.org/dtd/spring-beans.dtd">

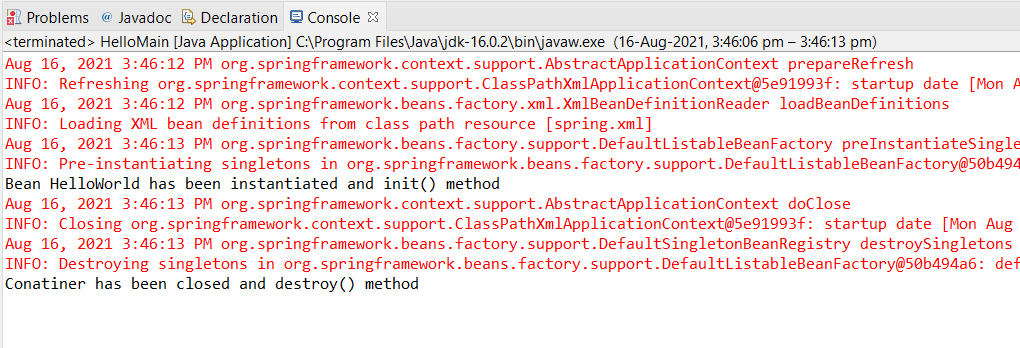
<beans>

<bean id=*"hello"* class=*"com.helloworldexample.HelloWorld"*

init-method=*"init"* destroy-method=*"destroy"*/>

</beans>

Output:



Q10. ApplicationContextAware Demo

**package** com.helloworldexample;

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

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

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

**public** **class** ApplicationContextAware1 **implements** ApplicationContextAware {

ApplicationContext context;

**public** ApplicationContext getContext() {

**return** context;

}

@Override

**public** **void** setApplicationContext(ApplicationContext context)

**throws** BeansException {

**this**.context=context;

}

}

**package** com.helloworldexample;

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

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

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

**public** **class** HelloMain {

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

AbstractApplicationContext context = **new** ClassPathXmlApplicationContext("spring.xml");

ApplicationContextAware1 appcontext= (ApplicationContextAware1)context.getBean("appcontext");

ApplicationContext appCon =appcontext.getContext();

A a= (A)appCon.getBean("testA");

a.display();

context.registerShutdownHook();

}

}

**package** com.helloworldexample;

**public** **class** A {

**public** **void** display(){

System.***out***.println("Application Context Program Demonstration.");

}

}

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

<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN"

"http://www.springframework.org/dtd/spring-beans.dtd">

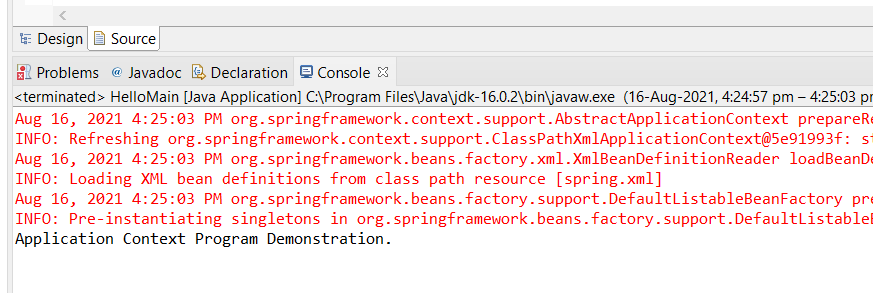
<beans>

<bean id=*"testA"* class=*"com.helloworldexample.A"*/>

<bean id=*"appcontext"* class=*"com.helloworldexample.ApplicationContextAware1"*/>

</beans>

Output:



Q8

**package** Annotation1;

**import** javax.annotation.PostConstruct;

**import** javax.annotation.PreDestroy;

**import** javax.annotation.Resource;

**public** **class** Circle

{

**private** Point center;

**public** Point getCenter() {

**return** center;

}

**public** **void** setCenter(Point center) {

**this**.center = center;

}

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

{

System.***out***.println("Circle is drawn of center ("+center.getX()+", "+center.getY()+")");

}

@PostConstruct

**public** **void** initializeCircle()

{

//populates the circle data cache upon initialization...

System.***out***.println("Init of Circle");

}

@PreDestroy

**public** **void** destroyCircle()

{

//clears the circle related cache upon destruction..

System.***out***.println("Destroy of Circle");

}

}

**package** Annotation1;

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

**int** x;

**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;

}

}

**package** Annotation1;

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

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

**public** **class** DrawingApp

{

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

{

AbstractApplicationContext context = **new** ClassPathXmlApplicationContext("spring.xml");

context.registerShutdownHook();

Circle circle = (Circle)context.getBean("circle");

circle.draw();

}

}

Xml file

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

<beans xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

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

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

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

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

*http://www.springframework.org/schema/beans/spring-beans-2.5.xsd*

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

*http://www.springframework.org/schema/context/spring-context-2.5.xsd"* >

<bean id=*"circle"* class=*"Annotation1.Circle"*>

<property name=*"center"* ref=*"Point"* />

</bean>

<bean id=*"Point"* class=*"Annotation1.Point"*>

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

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

</bean>

<context:annotation-config/>

<context:component-scan base-package=*"controller"* />

</beans>

**Pom file**

<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>SpringCore</groupId>

<artifactId>AnnotAssign</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>javax.annotation</groupId>

<artifactId>javax.annotation-api</artifactId>

<version>1.3.2</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>4.3.5.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

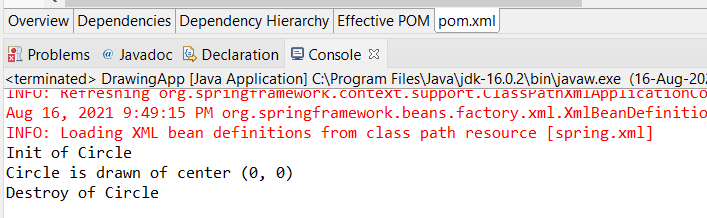
<version>4.3.5.RELEASE</version>

</dependency>

</dependencies>

</project>

**Output:**



Q2.

**package** Annotation1;

**import** java.util.Iterator;

**import** java.util.List;

**public** **class** Questions {

**private** **int** id;

**private** String name;

**private** List<String> answers;

**public** **int** getId() {

**return** id;

}

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

**this**.id = id;

}

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

**public** List<String> getAnswers() {

**return** answers;

}

**public** **void** setAnswers(List<String> answers) {

**this**.answers = answers;

}

**public** **void** displayInfo(){

System.***out***.println(id+" "+name);

System.***out***.println("answers are:");

Iterator<String> itr=answers.iterator();

**while**(itr.hasNext()){

System.***out***.println(itr.next());

}

}

}

Main File

**package** Annotation1;

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

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

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

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

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

AbstractApplicationContext context = **new** ClassPathXmlApplicationContext("spring.xml");

Questions myques=(Questions)context.getBean("");

myques.displayInfo();

}

}

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

<beans xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

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

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

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

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

*http://www.springframework.org/schema/beans/spring-beans-2.5.xsd*

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

*http://www.springframework.org/schema/context/spring-context-2.5.xsd"* >

<bean id=*"q"* class=*"Annotation1.Questions"*>

<property name=*"id"* value=*"1"*></property>

<property name=*"name"* value=*"What is Java?"*></property>

<property name=*"answers"*>

<list>

<value>Java is a programming language</value>

<value>Java is a platform independent language</value>

<value>Java is object oriented language</value>

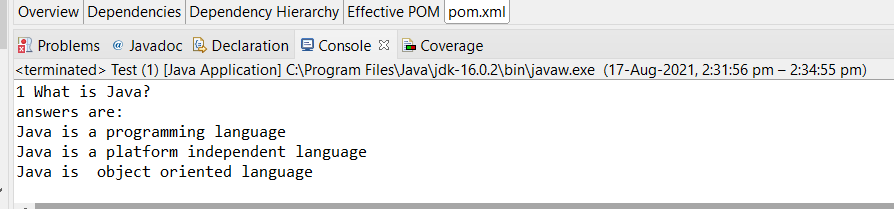
</list>

</property>

</bean>

</beans>

Output for list:



**package** Annotation1;

**import** java.util.Iterator;

**import** java.util.List;

**import** java.util.Set;

**public** **class** Questions {

**private** **int** id;

**private** String name;

**private** Set<String> answers;

**public** **int** getId() {

**return** id;

}

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

**this**.id = id;

}

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

**public** Set<String> getAnswers() {

**return** (Set<String>) answers;

}

**public** **void** setAnswers(Set<String> answers) {

**this**.answers = (Set<String>) answers;

}

**public** **void** displayInfo(){

System.***out***.println(id+" "+name);

System.***out***.println("answers are:");

Iterator<String> itr=answers.iterator();

**while**(itr.hasNext()){

System.***out***.println(itr.next());

}

}

}

<bean id=*"q"* class=*"Annotation1.Questions"*>

<property name=*"id"* value=*"1"*></property>

<property name=*"name"* value=*"What is Java?"*></property>

<property name=*"answers"*>

<set>

<value>Java is a programming language</value>

<value>Java is a platform independent language</value>

<value>Java is object oriented language</value>

<value>Java is a programming language</value>

</set>

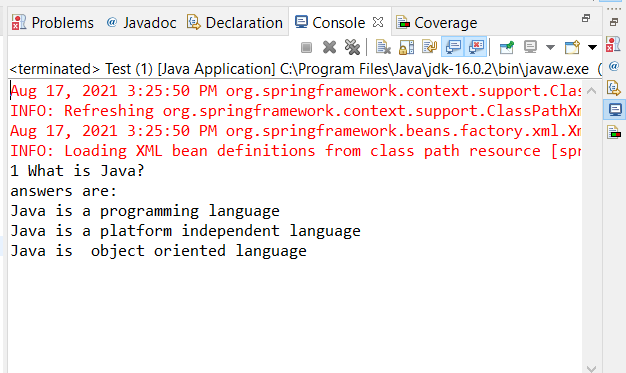
</property>

</bean>

</beans>

Main File Same as above:

Output: Duplicate Values from xml file are not included.



Map:

**package** Annotation1;

**import** java.util.Iterator;

**import** java.util.Map;

**import** java.util.Set;

**import** java.util.Map.Entry;

**public** **class** Question {

**private** **int** id;

**private** String name;

**private** Map<String,String> answers;

**public** **int** getId() {

**return** id;

}

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

**this**.id = id;

}

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

**public** Map<String, String> getAnswers() {

**return** answers;

}

**public** **void** setAnswers(Map<String, String> answers) {

**this**.answers = answers;

}

**public** **void** displayInfo(){

System.***out***.println("question id:"+id);

System.***out***.println("question name:"+name);

System.***out***.println("Answers....");

Set<Entry<String, String>> set=answers.entrySet();

Iterator<Entry<String, String>> itr=set.iterator();

**while**(itr.hasNext()){

Entry<String,String> entry=itr.next();

System.***out***.println("Answer:"+entry.getKey() + entry.getValue());

}

}

}

**package** Annotation1;

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

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

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

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

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

AbstractApplicationContext context = **new** ClassPathXmlApplicationContext("spring.xml");

Question myques=(Question)context.getBean("q");

myques.displayInfo();

}

}

<bean id=*"q"* class=*"Annotation1.Question"*>

<property name=*"id"* value=*"1"*></property>

<property name=*"name"* value=*"What is Java?"*></property>

<property name=*"answers"*>

<map>

<entry key=*"1 "* value=*"Java is a programming language"*></entry>

<entry key=*"2 "* value= *"Java is platform independent "*></entry>

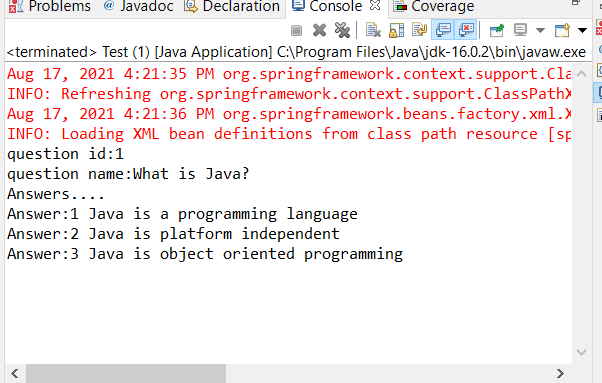
<entry key=*"3 "* value=*"Java is object oriented programming"*></entry>

</map>

</property>

</bean>

</beans>



Q3.

**package** Annotation1;

**public** **class** BankAcc {

**private** **long** accountId;

**private** String accountHolderName;

**private** String accountType;

**private** **int** accountBalance;

}

**package** Annotation1;

**public** **class** BankAccountRepositoryImpl **implements** BankAccountRepository{

**public** **double** getBalance(**long** accountId) {

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

**return** updateBalance(0, 0);

}

**public** **double** updateBalance(**long** accountId, **double** newBalance) {

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

**return** updateBalance(0,0);

}

}

**package** Annotation1;

**public** **class** BankAccountServiceImpl **implements** BankAccountService{

**public** **double** withdraw(**long** accountId,**double** balance) {

**return** 0;

}

**public** **double** deposit(**long** accountId, **double** balance) {

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

**return** 0;

}

**public** **double** getBalance(**long** accountId) {

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

**return** 0;

}

**public** **boolean** fundTransfer(**long** fromAccount, **long** toAccount, **double** amount) {

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

**return** **false**;

}

}

**package** Annotation1;

**public** **class** BankAccController {

**long** accountId;

**double** balance;

**long** fromAccount;

**long** toAccount;

**double** amount;

**public** **long** getAccountId() {

**return** accountId;

}

**public** **void** setAccountId(**long** accountId) {

**this**.accountId = accountId;

}

**public** **long** getFromAccount() {

**return** fromAccount;

}

**public** **void** setFromAccount(**long** fromAccount) {

**this**.fromAccount = fromAccount;

}

**public** **long** getToAccount() {

**return** toAccount;

}

**public** **void** setToAccount(**long** toAccount) {

**this**.toAccount = toAccount;

}

**public** **double** getAmount() {

**return** amount;

}

**public** **void** setAmount(**double** amount) {

**this**.amount = amount;

}

**public** **void** setBalance(**double** balance) {

**this**.balance = balance;

}

**public** **double** getBalance() {

**return** balance;

}

**public** **void** withdraw() {

balance=balance-amount;

System.***out***.println("BankAccount Id: "+getAccountId()+"--- Bank Balance before withdraw: "+getBalance());

}

**public** **void** deposit() {

balance=balance+amount;

System.***out***.println("BankAccount Id: "+getAccountId()+"--- Bank Balance after deposit: "+getBalance());

}

**public** **void** fundTransfer() {

balance=amount+balance;

System.***out***.println("Balance: "+balance);

}

**public** **void** display() {

System.***out***.println("Bank Details are: ");

}

}

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

**package** Annotation1;

**public** **interface** BankAccountRepository {

**public** **double** getBalance(**long** accountId);

**public** **double** updateBalance(**long** accountId,**double** newBalance);

}

**package** Annotation1;

**public** **interface** BankAccountService {

**public** **double** withdraw(**long** accountId,**double** balance);

**public** **double** deposit(**long** accountId,**double** balance);

**public** **double** getBalance(**long** accountId);

**public** **boolean** fundTransfer(**long** fromAccount,**long** toAccount,**double** amount);

}

<?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.xsd*

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

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

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

<bean id = *"b"* class= *"Annotation1.BankAccController"* >

<property name=*"accountId"* value=*"1001"*></property>

<property name=*"balance"* value=*"25000"*></property>

<property name=*"amount"* value=*"10000"*></property>

<property name=*"fromAccount"* value=*"1001"*></property>

<property name=*"toAccount"* value=*"1002"*></property>

</bean>

<context:annotation-config/>

</beans>

**package** Annotation1;

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

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

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

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

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

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

AbstractApplicationContext context = **new** ClassPathXmlApplicationContext("Bank.xml");

BankAccController b = (BankAccController)context.getBean("b");

b.display();

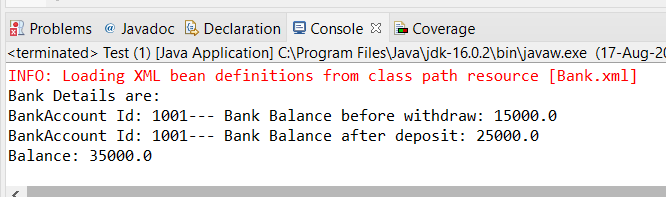
b.withdraw();

b.deposit();

b.fundTransfer();

}

}



SPRING MVC

Q1.

Web.xml file

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

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://Java.sun.com/xml/ns/javaee"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"* id=*"WebApp\_ID"* version=*"3.0"*>

<display-name>FirstSpringMVCProject</display-name>

<servlet>

<servlet-name>spring-dispatcher</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet

</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>spring-dispatcher</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

</web-app>

Spring dispatcher servlet file:

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

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://Java.sun.com/xml/ns/javaee"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"* id=*"WebApp\_ID"* version=*"3.0"*>

<display-name>FirstSpringMVCProject</display-name>

<servlet>

<servlet-name>spring-dispatcher</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet

</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>spring-dispatcher</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

</web-app>

Hello page.jsp file

<html>

<body>

<h1>First Spring MVC Application Demo</h1>

<h2>${welcomeMessage}</h2>

</body>

</html>

Hello Controller File:

**package** com.hellocontroller;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.servlet.ModelAndView;

**import** org.springframework.web.servlet.mvc.AbstractController;

**public** **class** HelloController **extends** AbstractController {

@Override

**protected** ModelAndView handleRequestInternal(HttpServletRequest arg0, HttpServletResponse arg1) **throws** Exception {

ModelAndView modelandview = **new** ModelAndView("HelloPage");

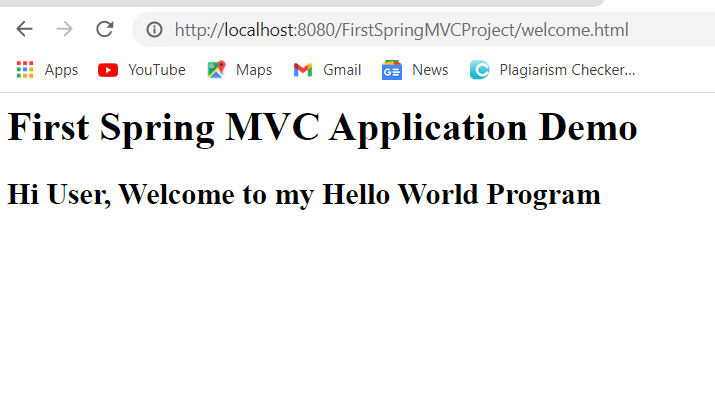
modelandview.addObject("welcomeMessage","Hi User, Welcome to my Hello World Program");

**return** modelandview;

}

}

Output:



**SPRING REST**

**Rest Assignment**

**import** org.springframework.http.HttpHeaders;

**import** org.springframework.http.HttpStatus;

**import** org.springframework.http.MediaType;

**import** org.springframework.http.ResponseEntity;

**import** org.springframework.web.bind.annotation.PathVariable;

**import** org.springframework.web.bind.annotation.RequestBody;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.ResponseBody;

**import** org.springframework.web.bind.annotation.RestController;

@RestController

**public** **class** StudenRestControllerApi {

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*RestAPI\*\*\*\*\*\*\*\*\*\*\*\*\*\*

@ResponseBody

@RequestMapping(value="/students", method = RequestMethod.***GET***)

**public** ArrayList<Student> getStudentList(){

Student student1 = **new** Student();

student1.setStudentName("The Great Khali");

Student student2= **new** Student();

student2.setStudentName("The Undertaker");

Student student3= **new** Student();

student3.setStudentName("John Cena");

ArrayList<Student> studentList = **new** ArrayList<Student> ();

studentList.add(student1);

studentList.add(student2);

studentList.add(student3);

**return** studentList;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Retrieving a student record\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*@ResponseBody

@RequestMapping(value = "/students/{name}", method = RequestMethod.GET)

public Student getStudent(@PathVariable("name") String studentName)

{

Student student = new Student();

student.setStudentName(studentName);

student.setStudentHobby("Riddhi Loves Cooking");

return student;

}

/\*@RequestMapping(value = "/students/{name}", method = RequestMethod.PUT)

public boolean updateStudent(@PathVariable("name") String studentName, @RequestBody Student student) {

System.out.println("student name: " +studentName);

System.out.println("student name: " +student.getStudentName() +"Student Hobby" +student.getStudentHobby());

return true;

}

@RequestMapping(value="/students/{name}", method = RequestMethod.PUT, consumes=MediaType.APPLICATION\_JSON\_VALUE)

public ResponseEntity<Boolean> updatestudent(@PathVariable("name") String studentName, @RequestBody Student student) {

System.out.println("Student Name: "+studentName);

System.out.println("Student Name: "+student.getStudentName()+"\n"

+"Student Hobby: "+student.getStudentHobby()+"\n"

+"Student Mobile: "+student.getStudentMobile());

HttpHeaders httpHeaders = new HttpHeaders();

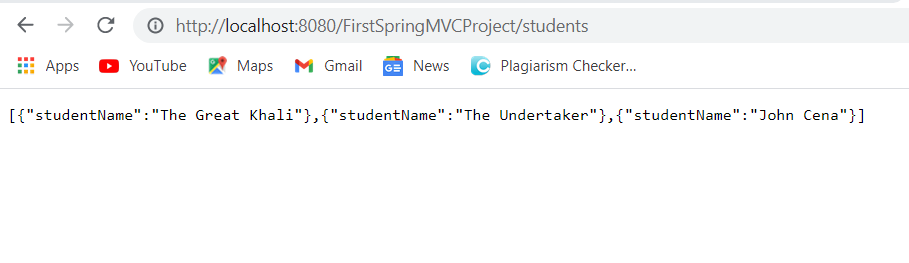
httpHeaders.add("Created By ", "Riddhi Rathod");

httpHeaders.add("DOB ", "01/04/1999");

return new ResponseEntity<Boolean>(true, httpHeaders, HttpStatus.OK);

}\*/

}



Q1.

**ackage** StudentAdmin;

**import** java.util.ArrayList;

**import** org.springframework.http.HttpHeaders;

**import** org.springframework.http.HttpStatus;

**import** org.springframework.http.MediaType;

**import** org.springframework.http.ResponseEntity;

**import** org.springframework.web.bind.annotation.PathVariable;

**import** org.springframework.web.bind.annotation.RequestBody;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.ResponseBody;

**import** org.springframework.web.bind.annotation.RestController;

@RestController

**public** **class** StudenRestControllerApi {

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*RestAPI\*\*\*\*\*\*\*\*\*\*\*\*\*\*

@ResponseBody

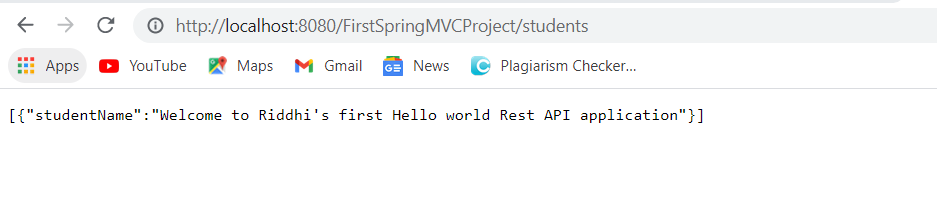
@RequestMapping(value="/students", method = RequestMethod.***GET***)

**public** ArrayList<Student> getStudentList(){

Student student1 = **new** Student();

student1.setStudentName("Welcome to Riddhi's first Hello world Rest API application");

output:



Q3.

**package** StudentAdmin;

**import** java.util.ArrayList;

**import** java.util.Date;

**import** com.fasterxml.jackson.annotation.JsonInclude;

**import** com.fasterxml.jackson.annotation.JsonProperty;

**import** com.fasterxml.jackson.annotation.JsonPropertyOrder;

@JsonInclude(JsonInclude.Include.***NON\_NULL***)

//@JsonPropertyOrder({"studentDOB", "studentName","studentMobile", "studentHobby","studentSkills"})

**public** **class** Student {

//@JsonProperty("Student\_name")

**private** String City;

//@Size(min=3,max=10)

**private** String State;

**private** String Country;

**public** String getCity() {

**return** City;

}

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

City = city;

}

**public** String getState() {

**return** State;

}

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

State = state;

}

**public** String getCountry() {

**return** Country;

}

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

Country = country;

}

**package** StudentAdmin;

**import** java.util.ArrayList;

**import** org.springframework.http.HttpHeaders;

**import** org.springframework.http.HttpStatus;

**import** org.springframework.http.MediaType;

**import** org.springframework.http.ResponseEntity;

**import** org.springframework.web.bind.annotation.PathVariable;

**import** org.springframework.web.bind.annotation.RequestBody;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RequestMethod;

**import** org.springframework.web.bind.annotation.ResponseBody;

**import** org.springframework.web.bind.annotation.RestController;

@RestController

**public** **class** StudenRestControllerApi {

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*RestAPI\*\*\*\*\*\*\*\*\*\*\*\*\*\*

@ResponseBody

@RequestMapping(value="/students", method = RequestMethod.***GET***)

**public** ArrayList<Student> getStudentList(){

Student student1 = **new** Student();

student1.setCity("Mumbai");

Student student2= **new** Student();

student2.setState("Maharashtra");

Student student3= **new** Student();

student3.setCountry("India");

ArrayList<Student> studentList = **new** ArrayList<Student> ();

studentList.add(student1);

studentList.add(student2);

studentList.add(student3);

**return** studentList;

}

