**2.**

Question 4:

Example on @service:

// Content: @Service annotation is used with classes that provide some business functionalities.It can be applied only to class. The spring context will autodelete these class

when annotation-based configuration and classpath scanning is used.

// Annotation based configuration is used here.

**// pom.xml dependency to add:**

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.0.6.RELEASE</version>

</dependency>

**//Create a service class:**

package springCoreTest;

import org.springframework.stereotype.Service;

@Service("ms")

public class MathService {

public int add(int x, int y) {

return x + y;

}

public int subtract(int x, int y) {

return x - y;

}

}

**//Create a Main class:**

package springCoreTest;

import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class SpringMainClass {

public static void main(String[] args) {

AnnotationConfigApplicationContext context = new AnnotationConfigApplicationContext();

context.scan("springCoreTest");

context.refresh();

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

int add = ms.add(9, 5);

System.out.println("Addition = " + add);

int subtract = ms.subtract(10, 2);

System.out.println("Subtraction = " + subtract);

context.close();

}

}

**output:**

Addition = 14

Subtraction = 8

Question 4:

**Example on @Configuration:**

//Content: This annotation indicates that a class declares one or more @Bean methods and may be processed by the Spring container to generate bean definitions

and service requests for those beans at runtime.

**// pom.xml dependency to add:**

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.9</version>

</dependency>

**//Create a class to config:**

package com.SpringAnno.ConfigAnno;

public class Audi {

public void config() {

System.out.println("A4, Black Colour, 73L fuel capacity");

}

}

**//Create a Main class:**

package com.SpringAnno.ConfigAnno;

import org.springframework.context.ApplicationContext;

import org.springframework.context.annotation.AnnotationConfigApplicationContext;

public class App

{

public static void main( String[] args )

{

ApplicationContext factory= new AnnotationConfigApplicationContext(AudiCar.class);

Audi A4= factory.getBean(Audi.class);

A4.config();

}

}

**// Create a Config class:**

package com.SpringAnno.ConfigAnno;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

@Configuration

public class AudiCar {

@Bean

public Audi getCar() {

return new Audi();

}

}

**Output:**

A4, Black Colour, 73L fuel capacity

//Example of Autowired using Annotations

//JavaProject name : AutowiredAnnotation

//class file name: Capital

**package** com.springautowired;

**public** **class** Capital {

**private** String name;

**public** String getName() {

**return** name;

}

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

**this**.name = name;

}

@Override

**public** String toString() {

**return** "Capital [name=" + name + "]";

}

}

//class file name: Country

**package** com.springautowired;

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

**public** **class** Country {

**private** String countryName;

@Autowired

**private** Capital capital;

**public** String getCountryName() {

**return** countryName;

}

**public** **void** setCountryName(String countryName) {

**this**.countryName = countryName;

}

**public** Capital getCapital() {

**return** capital;

}

**public** **void** setCapital(Capital capital) {

**this**.capital = capital;

}

@Override

**public** String toString() {

**return** "Country [countryName=" + countryName + ", capital=" + capital.toString() + "]";

}

}

//class file name:Country2

**package** com.springautowired;

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

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

**public** **class** Country2 {

**private** String countryName;

@Autowired

@Qualifier(value="myCapital2")

**private** Capital capital;

**public** String getCountryName() {

**return** countryName;

}

**public** **void** setCountryName(String countryName) {

**this**.countryName = countryName;

}

**public** Capital getCapital() {

**return** capital;

}

**public** **void** setCapital(Capital capital) {

**this**.capital = capital;

}

@Override

**public** String toString() {

**return** "Country [countryName=" + countryName + ", capital=" + capital.toString() + "]";

}

}

//Main class file: AppMain

//package name:com.spring

**package** com.spring;

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

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

**import** com.springautowired.Country;

**import** com.springautowired.Country2;

**public** **class** AppMain {

**private** **static** **void** autowired(String file) {

ApplicationContext ac = **new** ClassPathXmlApplicationContext(file);

**if** (file.equalsIgnoreCase("autowired.xml")) {

Country country = ac.getBean("myCountry", Country.**class**);

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

} **else** {

Country2 country = ac.getBean("myCountry", Country2.**class**);

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

}

}

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

**int** choice = Menu.*displayMenu*();

**switch** (choice) {

**case** 1:

System.***out***.println("'Autowired annotation' selected");

*autowired*("autowired.xml");

**break**;

**case** 2:

System.***out***.println("'Autowired with Qualifier annotation' selected");

*autowired*("autowired-qualifier.xml");

**break**;

**default**:

System.***err***.println("Invalid choice.");

}

}

}

//class file name:Menu

**package** com.spring;

public class Menu {

**public** **static** **int** displayMenu() {

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

**return** 0;

}

}

//pom.xml 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>com.autowired</groupId>

<artifactId>AutowiredAnnotation</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

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

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-beans</artifactId>

<version>5.0.6.RELEASE</version>

</dependency>

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

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.0.6.RELEASE</version>

</dependency>

</dependencies>

<build>

<finalName>${project.artifactId}</finalName>

</build>

</project>

//autowired.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"*>

<!-- Used to activate the @Autowired annotation in Spring -->

<context:annotation-config />

<bean id=*"myCapital"* class=*"com.springautowired.Capital"*>

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

</bean>

<bean id=*"myCountry"* class=*"com.springautowired.Country"*>

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

</bean>

</beans>

///autowired-qualifier.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"*>

<!-- Used to activate the @Autowired annotation in Spring -->

<context:annotation-config />

<bean id=*"myCapital1"* class=*"com.springautowired.Capital"*>

<property name=*"name"* value=*"The Hague"* />

</bean>

<bean id=*"myCapital2"* class=*"com.springautowired.Capital"*>

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

</bean>

<bean id=*"myCountry"* class=*"com.springautowired.Country2"*>

<property name=*"countryName"* value=*"Netherlands"* />

</bean>

</beans>

Output: Invalid choice.