Name of Student: Pushkar Sane		
Roll Number: 45	Lab Assignment Number: 5	
Title of Lab Assignment: Implementation of java application using spring framework.		
DOP: 09-10-2023	DOS: 11-10-2023	
CO Mapped: CO3	PO Mapped: PO2, PO3, PO4, PO7, PO11, PSO1, PSO2	Signature:

Practical No. 5

Aim: Implementation of java application using spring framework.

1. Write a program to print “Hello World” using spring framework.
2. Write a program to create shape class and rectangle class Shape class has a relationship with rectangle class. Use constructor injection to invoke object of shape class in rectangle class.
3. Write a program to create address class and Student class address class has a relationship with student class. Use setter injection to invoke the object of address class in student class.

Codes:

1. Write a program to print “Hello World” using spring framework.

Message.java:

```
package SpringHello;
public class Message {
   void printMessage() {
      System.out.println("Hello World");
   }
}
```

TestClass.java

```
package SpringHello;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
public class TestClass {
        public static void main(String[] args) {
                ApplicationContext context = new
                        ClassPathXmlApplicationContext("configuration.xml");
                Message obj = (Message) context.getBean("messageBean");
                obj.printMessage();
        }
}
```

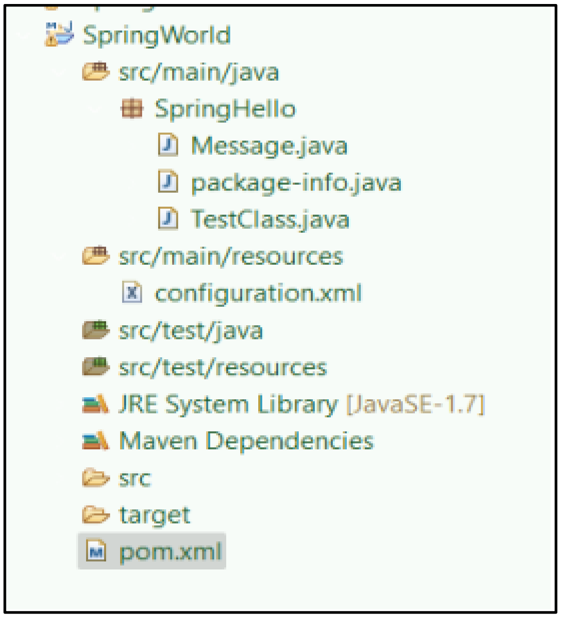
Configuration.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">
<bean id="messageBean" class="SpringHello.Message">
</bean>
</beans>
```

Pom.xml

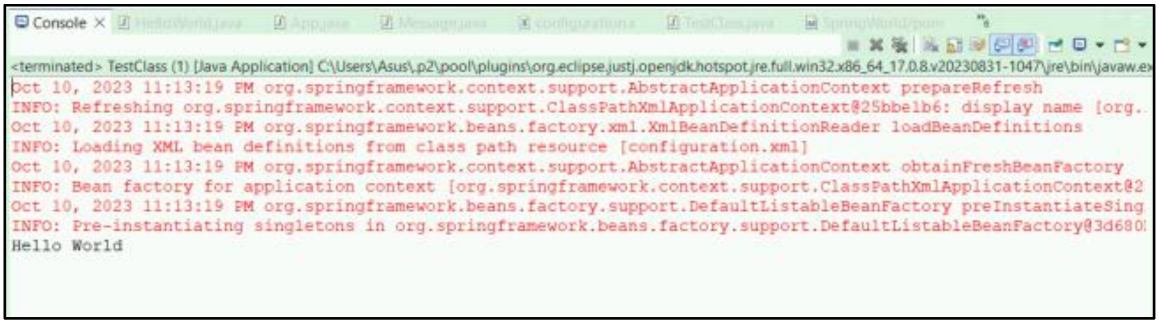
```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
 <modelVersion>4.0.0</modelVersion>
 <groupId>Hello_world</groupId>
 <artifactId>SpringWorld</artifactId>
 <version>0.0.1-SNAPSHOT</version>
 <dependencies>
 <dependency>
<groupId>org.springframework</groupId>
<artifactId>spring</artifactId>
<version>2.5.6</version>
</dependency>
 </dependencies>
</project>
```

Project Structure:



```
SpringWorld
    src/main/java
        SpringHello
            Message.java
            package-info.java
            TestClass.java
    src/main/resources
        configuration.xml
    src/test/java
    src/test/resources
    JRE System Library [JavaSE-1.7]
    Maven Dependencies
    src
    target
    pom.xml
```

Output:



```
<terminated> TestClass (1) [Java Application] C:\Users\Asus\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.ex
Oct 10, 2023 11:13:19 PM org.springframework.context.support.AbstractApplicationContext prepareRefresh
INFO: Refreshing org.springframework.context.support.ClassPathXmlApplicationContext@25bbe1b6: display name [org.
Oct 10, 2023 11:13:19 PM org.springframework.beans.factory.xml.XmlBeanDefinitionReader loadBeanDefinitions
INFO: Loading XML bean definitions from class path resource [configuration.xml]
Oct 10, 2023 11:13:19 PM org.springframework.context.support.AbstractApplicationContext obtainFreshBeanFactory
INFO: Bean factory for application context [org.springframework.context.support.ClassPathXmlApplicationContext@2
Oct 10, 2023 11:13:19 PM org.springframework.beans.factory.support.DefaultListableBeanFactory preInstantiateSing
INFO: Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@3d680
Hello World
```

2. Write a program to create shape class and rectangle class Shape class has a relationship with rectangle class. Use constructor injection to invoke object of shape class in rectangle class.

Shape.java

```
package Shape_constructor;
public class Shape {
    private Rectangle rectangle;
    public Shape(Rectangle rectangle) {
        this.rectangle = rectangle;
    }
    public void printShapeDetails() {
        System.out.println("Shape Details:");
        rectangle.printRectangleDetails();
    }
}
```

Rectangle.java

```
package Shape_constructor;
public class Rectangle {
    private int length;
    private int width;
    public Rectangle(int length, int width) {
        this.length = length;
        this.width = width;
    }
    public void printRectangleDetails() {
        System.out.println("Rectangle: Length = " + length + ", Width = " + width);
    }
}
```

MainApp.java

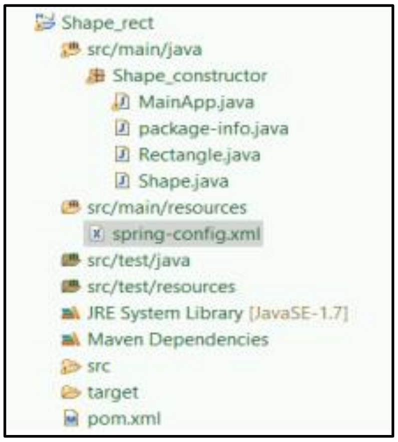
```
package Shape_constructor;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
```

```
public class MainApp {
    public static void main(String[] args) {
        ApplicationContext context = new
ClassPathXmlApplicationContext("springconfig.xml");
        Shape shape = (Shape) context.getBean("shape");
        shape.printShapeDetails();
    }
}
```

Pom.xml

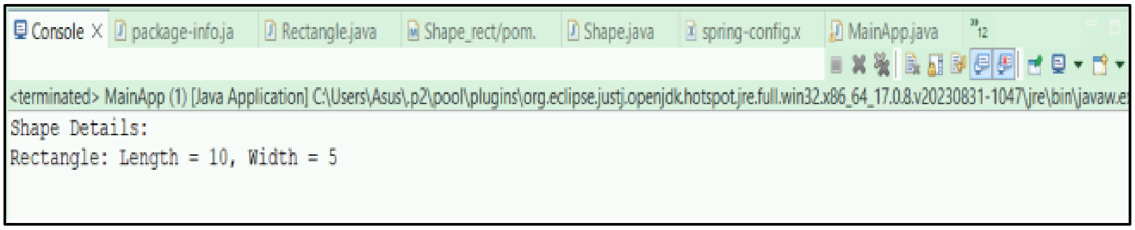
```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
<modelVersion>4.0.0</modelVersion>
<groupId>ShapeQ2</groupId>
<artifactId>Shape_rect</artifactId>
<version>0.0.1-SNAPSHOT</version>
<dependencies>
<dependency>
<groupId>org.springframework</groupId>
<artifactId>spring-context</artifactId>
<version>5.3.10</version> <!-- Use the latest Spring version --> </dependency>
</dependencies>
</project>
```

Project Structure:



```
Shape_rect
  src/main/java
    Shape_constructor
      MainApp.java
      package-info.java
      Rectangle.java
      Shape.java
  src/main/resources
    spring-config.xml
  src/test/java
  src/test/resources
  JRE System Library [JavaSE-1.7]
  Maven Dependencies
  src
  target
  pom.xml
```

Output:



```
<terminated> MainApp (1) [Java Application] C:\Users\Asus\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.e
Shape Details:
Rectangle: Length = 10, Width = 5
```

3. Write a program to create address class and Student class address class has a relationship with student class. Use setter injection to invoke the object of address class in student class.

Student.java

```
package Student_addresss;
public class Student {
    private Address address;
    // Setter method for address
    public void setAddress(Address address) {
        this.address = address;
    }
    // Getter method for address
    public Address getAddress() {
        return address;
    }
    public void printStudentDetails() {
        System.out.println("Student Details:");
        System.out.println("Student's " + address.toString());
    }
}
```

Address.java

```
package Student_addresss;
public class Address {
    private String street;
    private String city;
    private String state;
    // Getter and Setter methods for street
    public String getStreet() {
        return street;
    }
    public void setStreet(String street) {
        this.street = street;
    }
    // Getter and Setter methods for city
    public String getCity() {
        return city;
    }
    public void setCity(String city) {
        this.city = city;
    }
    // Getter and Setter methods for state
    public String getState() {
        return state;
    }
    public void setState(String state) {
        this.state = state;
    }
    @Override
    public String toString() {
        return "Address: " + street + ", " + city + ", " + state;
    }
}
```

MainApp.java

```
package Student_addresss;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
public class MainApp {
public static void main(String[] args) {
  ApplicationContext context = new ClassPathXmlApplicationContext("springconfig.xml");
  Student student = (Student) context.getBean("student");
  student.printStudentDetails();
  }
}
```

Spring-config.xml

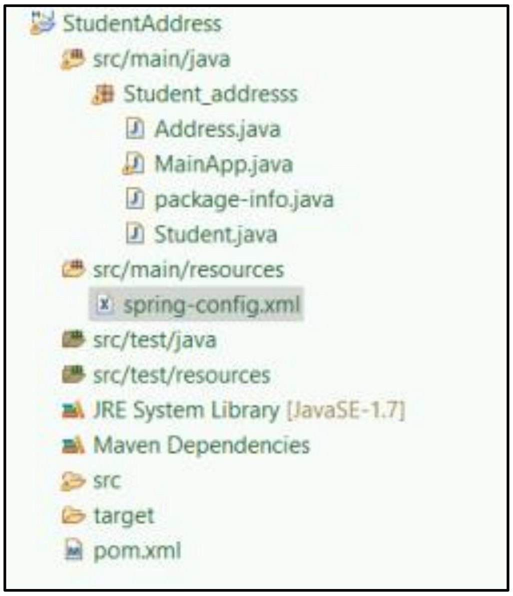
```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/springbeans.xsd">
    <bean id="address" class="Student_addresss.Address">
        <property name="street" value="Old Highway Street" />
        <property name="city" value="Badlapur" />
        <property name="state" value="Maharashtra" />
    </bean>
    <bean id="student" class="Student_addresss.Student">
        <property name="address" ref="address" />
    </bean>
</beans>
```

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>Student_address</groupId>
<artifactId>StudentAddress</artifactId>
<version>0.0.1-SNAPSHOT</version>
<dependencies>
<dependency>
<groupId>org.springframework</groupId>
<artifactId>spring-context</artifactId>
```

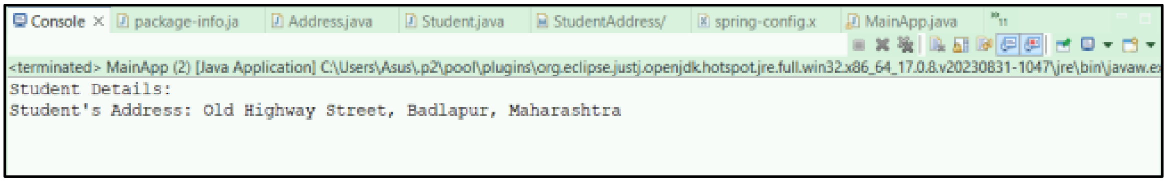
```
<version>5.3.10</version> <!-- Use the latest Spring version --> </dependency>
</dependencies>
</project>
```

Project Structure:



```
StudentAddress
    src/main/java
        Student_addresss
            Address.java
            MainApp.java
            package-info.java
            Student.java
    src/main/resources
        spring-config.xml
    src/test/java
    src/test/resources
    JRE System Library [JavaSE-1.7]
    Maven Dependencies
    src
    target
    pom.xml
```

Output:



```
<terminated> MainApp (2) [Java Application] C:\Users\Asus\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.e
Student Details:
Student's Address: Old Highway Street, Badlapur, Maharashtra
```