|  |  |  |  |
| --- | --- | --- | --- |
| **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>

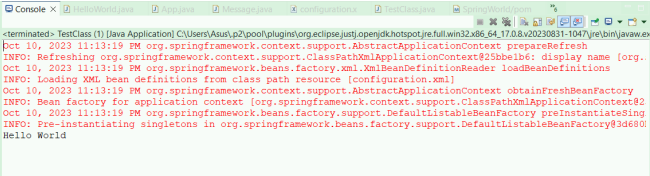
</dependencies>

</project>

**Project Structure:**



**Output:**



1. 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>

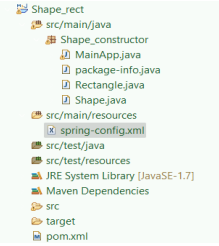
<artifactId>spring-context</artifactId>

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

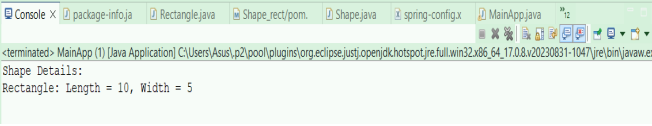
</dependencies>

</project>

**Project Structure:**



**Output:**



1. 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>

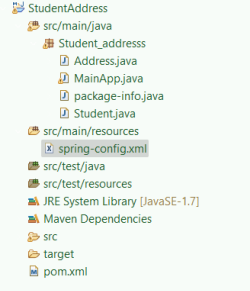
<artifactId>spring-context</artifactId>

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

</dependencies>

</project>

**Project Structure:**



**Output:**

