

Steps to follow:

1. Create spring project and add the dependencies
2. Create the pojo class
3. Generate springConfiguration file
It contains configuration details of pojo class
4. Generate client Application

Create spring project and add the dependencies

1. Create a maven project
2. New->other->maven-archetypequickstart 1.1
Group id : com.takeo
Artifact id : 001_SF_helloWorld
Package: com.takeo

Google: spring core maven dependency -> 5.3.9

Google: spring bean maven dependency ->5.3.12

3. pom.xml-> copy the above dependencies and paste underneath the dependencies

```
<!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-core</artifactId>
  <version>5.3.9</version>
</dependency>
```

```
<!-- https://mvnrepository.com/artifact/org.springframework/spring-beans -->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-beans</artifactId>
  <version>5.3.12</version>
</dependency>
```

Create the pojo class

4. Save it and update it
5. src/main/java-> new->class->
Package: com.takeo.pojo
Class: HelloWorld

```
package com.takeo.pojo;
```

```

public class HelloWorld {
    private String message;

    public String getMessage() {
        return message;
    }

    public void setMessage(String message) {
        this.message = message;
    }
}

```

Generate springConfiguration file

6. com.takeo->package:
Com.takeo.config

com.takeo.config->other->spring Bean configuration file
 We can give any logical name;
 File name: spring.xml->next->next->finish

```

<?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 class ="com.takeo.HelloWorld" id = "hello"> <!-- package/classname -->
    <property name="message"> <!-- setter->
    <value> Welcome to Rajesh's World</value> <!--outputValue -->

```

```

    </property> <!-- ctrl+space/settermethod -->

```

```

</bean>

```

```

</beans>

```

Generate client Application

7. Client Class: App.java

```
package com.takeo;

import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.ClassPathResource;

/**
 * Hello world!
 *
 */
public class App
{
    public static void main( String[] args )
    {
        // instantiate Spring Core container
        BeanFactory factory = new XmlBeanFactory(new
ClassPathResource("com\\takeo\\config\\spring.xml"));
        // update it if you dont see the import things. For path
        // go to the spring.xml ->properties-> copy from com//.....
        HelloWorld hello =(HelloWorld) factory.getBean("hello");// id is " hello" in
spring
        // HelloWorld: Object or classname, Bean: hello or id name
        // we need to do typcaste
        // now we can access
        System.out.println(hello.getMessage());

    }
}
```

Next Project

002_SF_Student

```
<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
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
```

```
<groupId>com.takeo</groupId>
<artifactId>002_SF_Student</artifactId>
<version>0.0.1-SNAPSHOT</version>
<packaging>jar</packaging>

<name>002_SF_Student</name>
<url>http://maven.apache.org</url>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
</properties>

<dependencies>

  <!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->
  <dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-core</artifactId>
    <version>5.3.9</version>
  </dependency>

  <!-- https://mvnrepository.com/artifact/org.springframework/spring-beans -->
  <dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-beans</artifactId>
    <version>5.3.12</version>
  </dependency>

  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>3.8.1</version>
    <scope>test</scope>
  </dependency>
</dependencies>
</project>
```

Student

```
package com.takeo.pojo;
```

```
public class Student {  
    private int sno;  
    private String sname;  
    private String sadd;  
    public int getSno() {  
        return sno;  
    }  
    public void setSno(int sno) {  
        this.sno = sno;  
    }  
    public String getSname() {  
        return sname;  
    }  
    public void setName(String sname) {  
        this.sname = sname;  
    }  
    public String getSadd() {  
        return sadd;  
    }  
    public void setSadd(String sadd) {  
        this.sadd = sadd;  
    }  
}
```

spring.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 class="com.takeo.pojo.Student" id="stu">  
        <property name="sno" value="100"></property>  
        <property name="sname" value="Rajesh"></property>  
        <property name="sadd" value="USA"></property>
```

```
</bean>
```

```
</beans>
```

App.Java

```
package com.takeo;
```

```
import com.takeo.pojo.Student;
```

```
import org.springframework.beans.factory.BeanFactory;
```

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

```
import org.springframework.core.io.ClassPathResource;
```

```
/**
```

```
 * Hello world!
```

```
 *
```

```
 */
```

```
public class App
```

```
{
```

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

```
    {
```

```
        BeanFactory factory = new XmlBeanFactory(new  
ClassPathResource("com\\takeo\\spring.xml"));
```

```
        Student stu = (Student)factory.getBean("stu");
```

```
        System.out.println(stu.getSno()+"\t"+stu.getSname()+"\t"+stu.getSadd());
```

```
    }
```

```
}
```

Next Project:

003_SF_Employee

```
<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
```

```
http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<groupId>com.takeo</groupId>
<artifactId>003_SF_Employee</artifactId>
<version>0.0.1-SNAPSHOT</version>
<packaging>jar</packaging>
```

```
<name>003_SF_Employee</name>
<url>http://maven.apache.org</url>
```

```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
</properties>
```

```
<dependencies>
```

```
<!-- https://mvnrepository.com/artifact/org.springframework/spring-core -->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-core</artifactId>
  <version>5.3.9</version>
</dependency>
```

```
<!-- https://mvnrepository.com/artifact/org.springframework/spring-beans -->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-beans</artifactId>
  <version>5.3.12</version>
</dependency>
```

```
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>3.8.1</version>
  <scope>test</scope>
```

```
</dependency>
</dependencies>
</project>
```

Address

```
package com.takeo;

public class Address {
    private int streetno;
    private String streetname;
    private String streetAddress;
    public int getStreetno() {
        return streetno;
    }
    public void setStreetno(int streetno) {
        this.streetno = streetno;
    }
    public String getStreetname() {
        return streetname;
    }
    public void setStreetname(String streetname) {
        this.streetname = streetname;
    }
    public String getStreetAddress() {
        return streetAddress;
    }
    public void setStreetAddress(String streetAddress) {
        this.streetAddress = streetAddress;
    }
}

}
```

Employee

```
package com.takeo;

public class Employee {
    private int eno;
    private String ename;
    private Address address;
    public int getEno() {
```



```

        return eno;
    }
    public void setEno(int eno) {
        this.eno = eno;
    }
    public String getEname() {
        return ename;
    }
    public void setEname(String ename) {
        this.ename = ename;
    }
    public Address getAddress() {
        return address;
    }
    public void setAddress(Address address) {
        this.address = address;
    }
}
}

```

spring.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 class="com.takeo.Address" id="addr">
        <property name="streetno" value="1"></property>
        <property name="streetname" value="RajMarg"></property>
        <property name="streetAddress" value="Sitamarhi"></property>

    </bean>
    <bean class="com.takeo.Employee" id="emp">
        <property name="eno" value="100"></property>
        <property name="ename" value="Rajesh"></property>
        <property name="address"> <ref bean="addr"/></property>

</bean>

```

</beans>

App.java

```
package com.takeo;
```

```
import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.ClassPathResource;
```

```
/**
```

```
 * Hello world!
```

```
 *
```

```
 */
```

```
public class App
```

```
{
```

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

```
    {
```

```
        BeanFactory factory = new XmlBeanFactory(new
        ClassPathResource("com\\takeo\\spring.xml"));
```

```
        Address addr = (Address)factory.getBean("addr");
```

```
        System.out.println(addr.getStreetno()+"\t"+addr.getStreetname()+"\t"+addr.getStreetAddress());
```

```
        Employee emp = (Employee)factory.getBean("emp");
```

```
        System.out.println(emp.getEno()+"\t"+emp.getEname()+"\t"+emp.getAddress());
```

```
    }
```

```
}
```