Multiple Configuration File:

<beans>

<import resource="teacher-spring.xml" />

<import resource="student-spring.xml" />

</beans>

Constructor Injection:

<constructor-arg value="1010" />

<constructor-arg value="1010" type="int" />

<constructor-arg value="Atul Rai" index="0" />

<constructor-arg value="MCA" name="course" />

<constructor-arg index="0" ref="e" />

Setter Injection:

<property name="engine" ref="e" />

<property name="modelYear" value="2018" />

Collection Injection:

<property name="carNames">

<list>

<value>Hindustan Moters</value>

<value>Tata Moters</value>

<value>Ashoka Leyland</value>

</list>

</property>

<constructor-arg index="0">

<list>

<value>Hindustan Moters</value>

<value>Tata Moters</value>

<value>Ashoka Leyland</value>

</list>

</constructor-arg>

<map>

<entry key="India" value="New Delhi" />

<entry key="Russia" value="Moscow" />

<entry key="France" value="Paris" />

</map>

<property name="carNames" value="Hindustan Moters,Tata Moters,Ashoka Leyland"/>

xmlns:util="<http://www.springframework.org/schema/util>"

<property name="empName">

<util:list list-class="java.util.Vector">

<value>Atul</value>

<value>Sandeep</value>

</util:list>

</property>

<property name="empId">

<util:set set-class="java.util.TreeSet">

<value>WSO1101</value>

<value>WSO1201</value>

</util:set>

</property>

<property name="empIdName">

<util:map map-class="java.util.Hashtable">

<entry key="WSO1" value="Vipul" />

<entry key="WSO2" value="Mukul" />

<entry key="WSO3" value="Ankita" />

</util:map>

</property>

<property name=*"strList"*>

<property name=*"myClassList"*>

<list>

<ref bean=*"mc1"* />

<ref bean=*"mc2"* />

</list>

</property>

<property name=*"myClassMap"*>

<map>

<entry key=*"1"* value-ref=*"mc1"* />

<entry key=*"2"* value-ref=*"mc2"* />

</map>

</property>

<props>

<prop key="class\_path">com.mysql.jdbc.Driver</prop>

<prop key="url">jdbc:mysql://127.0.0.1:3306/database</prop>

<prop key="user\_name">system</prop>

<prop key="password">manager</prop>

</props>

<util:properties location="classpath:org/websparrow/resources/database.properties"></util:properties>

@Required:

<bean class="org.springframework.beans.factory.annotation.RequiredAnnotationBeanPostProcessor" />

Depends-on:

<bean id="p" class="org.websparrow.beans.Principal" depends-on="t,l" />

P|C namespace:

<bean id="newTeacher" class="beans.Teacher" p:name="Vijay Pandey" p:qualification="PhD" p:student-ref="newStudent" />

<bean id="newboss" class="beans.Boss" c:name="Atul Rai" c:employee-ref="newemp" />

@Autowire:

<bean id="country" class="org.websparrow.beans.Country" autowire="no|byType|byName|constructor" />

<bean id="s1" class="org.websparrow.beans.State" autowire-candidate="false">

<property name="stateName" value="Uttar Pradesh" />

</bean>

<context:annotation-config />

Or

<bean class="org.springframework.beans.factory.annotation.AutowiredAnnotationBeanPostProcessor"/>

@Autowired(true)

public City(State state) {

this.state = state;

}

@Autowired

@Qualifier("s1") // eligible reference bean id

private State state;

@Component:

<!-- scan the package -->

<context:component-scan base-package="org.websparrow.beans" />

<!-- activate annotations -->

<context:annotation-config />

@Component

public class Country {

@Autowired

private State state;

Static Injection:

before spring 5

<bean class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">

<property name="staticMethod" value="org.websparrow.beans.Employee.setEmpName" />

<property name="arguments">

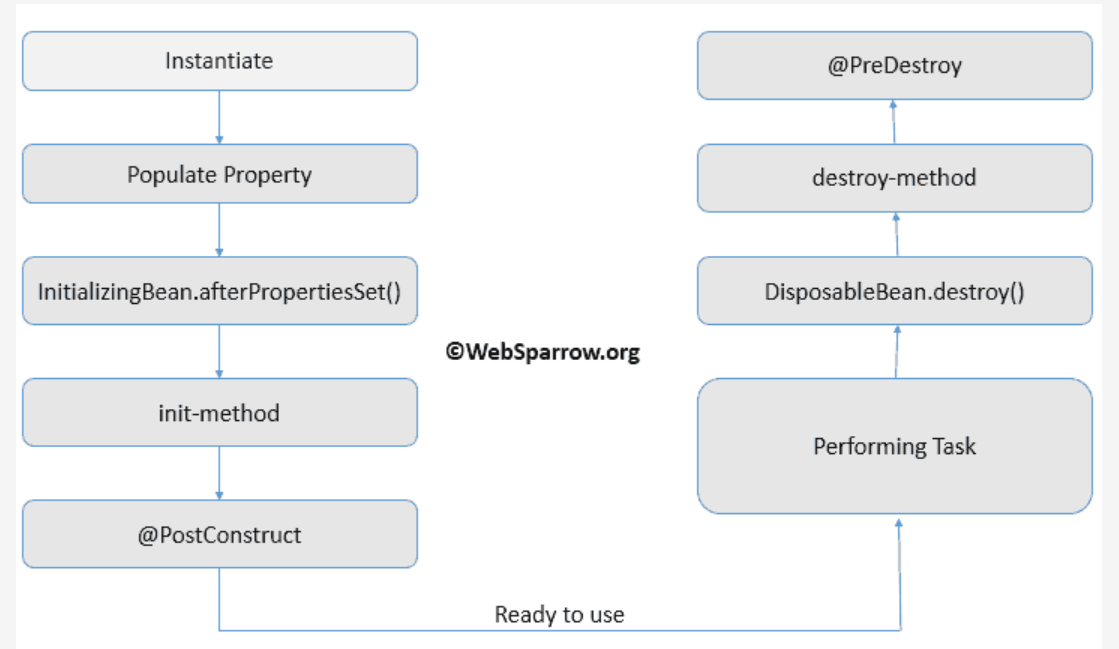
<list>

<value>Sandeep Sharma</value>

</list>

</property>

Bean Lifecycle:



public class Employee implements InitializingBean, DisposableBean {

@Override

public void afterPropertiesSet() throws Exception {}

@Override

public void destroy() throws Exception {}

<bean id="emp" class="org.websparrow.xml.beans.Employee" init-method="conInit" destroy-method="conCleanUp">

<beans default-init-method="myInit" default-destroy-method="myDestroy">

<bean class="org.springframework.context.annotation.CommonAnnotationBeanPostProcessor" />

@PostConstruct

public void conInit() throws Exception {}

@PreDestroy

public void conCleanUp() throws Exception {}

Method Replacement:

<bean id="break" class="org.websparrow.beans.NewBreak" />

<bean id="car" class="org.websparrow.beans.Car">

<replaced-method name="breaks" replacer="break" />

</bean>

public class NewBreak implements MethodReplacer {

@Override

public Object reimplement(Object obj, Method method, Object[] args) throws Throwable {

// new property of Car.breaks() method.

System.out.println("New car break.");

return obj;

}

}

public class Car {

public void design() {

System.out.println("Old car design.");

}

public void breaks() {

System.out.println("Old car break.");

}

}

Method Lookup:

<bean id="car" class="org.websparrow.beans.Car">

<lookup-method name="myCarEngine" bean="e" />

</bean>

public interface Car {

public Engine myCarEngine();

}

Injecting Properties:

driver=oracle.jdbc.driver.OracleDriver

url=jdbc:oracle:thin:@localhost:1521:xe

user=root

pwd=root

<bean class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">

<property name="location" value="driver.properties" />

</bean>

<!-- inject context scope data into required bean/class -->

<bean id="connect" class="org.websparrow.beans.ConnectionManager">

<property name="driver" value="${driver}" />

<property name="url" value="${url}" />

<property name="userName" value="${user}" />

<property name="password" value="${pwd}" />

</bean>