



Spring – Autowiring

N neeraj26pathak

[Read](#)[Discuss](#)[Practice](#)

Spring is an open-source application development framework of Java that allows you to create robust enterprise applications using Plain Old Java Objects (**POJO** in short). The Spring framework can inject dependencies automatically. The Spring container detects those dependencies specified in the configuration file and @ the relationship between the beans. This is referred to as autowiring in Spring. An autowired application requires fewer lines of code comparatively but at the same time, it provides very little flexibility to the programmer.

Modes of Autowiring

There are five modes of autowiring:

1. No

This mode tells the framework that autowiring is not supposed to be done. It is the default mode used by Spring.

XML



We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

```
</bean>  
<bean id="city" class="sample.City"></bean>
```

2. byName

It uses the name of the bean for injecting dependencies. However, it requires that the name of the property and bean must be the same. It invokes the setter method internally for autowiring.

XML

```
<bean id="state" class="sample.State">  
  <property name="name" value="UP" />  
</bean>  
<bean id="city" class="sample.City" autowire="byName"></bean>
```

3. byType

It injects the dependency according to the type of the bean. It looks up in the configuration file for the class type of the property. If it finds a bean that matches, it injects the property. If not, the program throws an error. The names of the property and bean can be different in this case. It invokes the setter method internally for autowiring.

XML

```
<bean id="state" class="sample.State">  
  <property name="name" value="UP" />  
</bean>  
<bean id="city" class="sample.City" autowire="byType"></bean>
```

4. constructor

It injects the required dependencies by invoking the constructor. It works similar to the “byType” mode but it looks for the class type of the constructor arguments. If none or more than one bean are detected, then it throws an error, otherwise, it autowires the “byType” on all constructor arguments.

XML

```
<bean id="state" class="sample.State">  
  <property name="name" value="UP" />  
</bean>
```

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

5. autodetect

The autodetect mode uses two other modes for autowiring – constructor and byType. It first tries to autowire via the constructor mode and if it fails, it uses the byType mode for autowiring. It works in Spring 2.0 and 2.5 but is deprecated from Spring 3.0 onwards.

XML

```
<bean id="state" class="sample.State">
  <property name="name" value="UP" />
</bean>
<bean id="city" class="sample.City" autowire="autodetect"></bean>
```

Sample Program for the byType Mode

File Name: State.java

Java

```
public class State {
    private String name;
    public String getName() { return name; }
    public void setName(String s) { this.name = s; }
}
```

File Name: City.java

Java

```
class City {
    private int id;
    private String name;
    private State s;
    public int getID() { return id; }
    public void setId(int eid) { this.id = eid; }
    public String getName() { return name; }
    public void setName(String st) { this.name = st; }
    public State getState() { return s; }
    @Autowired public void setState(State sta)
    {
        this.s = sta;
    }
    public void showCityDetails()
    {

```

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

```

        System.out.println("State : " + s.getName());
    }
}

```

Spring bean configuration file:

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"
       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">
  <bean id="state" class="sample.State">
    <property name="name" value="UP" />
  </bean>
  <bean id="city" class="sample.City" autowire="byName"></bean>
</beans>

```

Test Program file: DemoApplication.java

Java

```

@SpringBootApplication
public class DemoApplication {
    public static void main(String[] args)
    {
        SpringApplication.run(DemoApplication.class, args);
        ApplicationContext context = new ClassPathXmlApplicationContext("applicationCo
City cty = context.getBean("city", City.class);
        cty.setId(01);
        cty.setName("Varanasi");
        State st = context.getBean("state", State.class);
        st.setName("UP");
        cty.setState(st);
        cty.showCityDetails();
    }
}

```

Output:

City ID : 01

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

Advantage of Autowiring

It requires less code because we don't need to write the code to inject the dependency explicitly.

Disadvantage of Autowiring

- No control of the programmer.
- It can't be used for primitive and string values.

Last Updated : 19 Nov, 2021

4

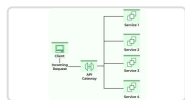
Similar Reads



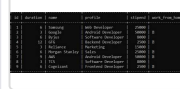
Spring Boot - Spring JDBC vs Spring Data JDBC



Difference Between Spring DAO vs Spring ORM vs Spring JDBC



Java Spring Boot Microservices - Develop API Gateway Using Spring Cloud Gateway



Spring Boot | How to access database using Spring Data JPA



Difference between Spring MVC and Spring Boot



How to Create a Spring Boot Project in Spring Initializr and Run it in IntelliJ IDEA?



How to Download and Install Spring Tool Suite (Spring Tools 4 for Eclipse) IDE?



Spring - Add User Name and Password in Spring Security

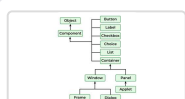


How to Create and Setup Spring Boot Project in Spring Tool Suite?



How to Run Your First Spring Boot Application in Spring Tool Suite?

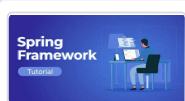
Related Tutorials



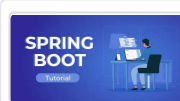
Java AWT Tutorial



Spring MVC Tutorial



Spring Tutorial



Spring Boot Tutorial



Java 8 Features - Complete Tutorial

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

[Previous](#)[Next](#)

Article Contributed By :

N

[neeraj26pathak](#)
neeraj26pathak
[Follow](#)

Vote for difficulty

Current difficulty : [Easy](#)

Article Tags : [Java-Spring](#), [Picked](#), [Java](#)

Practice Tags : [Java](#)



A-143, 9th Floor, Sovereign Corporate Tower, Sector-136, Noida, Uttar Pradesh - 201305

feedback@geeksforgeeks.org

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)



Company

[About Us](#)
[Legal](#)
[Careers](#)
[In Media](#)
[Contact Us](#)
[Advertise with us](#)

Languages

[Python](#)
[Java](#)
[C++](#)
[PHP](#)
[GoLang](#)
[SQL](#)
[R Language](#)
[Android Tutorial](#)

DSA Roadmaps

[DSA for Beginners](#)
[Basic DSA Coding Problems](#)
[Complete Roadmap To Learn DSA](#)
[DSA for FrontEnd Developers](#)
[DSA with JavaScript](#)
[Top 100 DSA Interview Problems](#)
[All Cheat Sheets](#)
[DSA Roadmap by Sandeep Jain](#)

Computer Science

[GATE CS Notes](#)

Explore

[Job-A-Thon Hiring Challenge](#)
[Hack-A-Thon](#)
[GfG Weekly Contest](#)
[Offline Classes \(Delhi/NCR\)](#)
[DSA in JAVA/C++](#)
[Master System Design](#)
[Master CP](#)

DSA Concepts

[Data Structures](#)
[Arrays](#)
[Strings](#)
[Linked List](#)
[Algorithms](#)
[Searching](#)
[Sorting](#)
[Mathematical](#)
[Dynamic Programming](#)

Web Development

[HTML](#)
[CSS](#)
[JavaScript](#)
[Bootstrap](#)
[ReactJS](#)
[AngularJS](#)
[NodeJS](#)
[Express.js](#)
[Lodash](#)

Python

[Python Programming Examples](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

[Computer Network](#)[Database Management System](#)[Software Engineering](#)[Digital Logic Design](#)[Engineering Maths](#)

Data Science & ML

[Data Science With Python](#)[Data Science For Beginner](#)[Machine Learning Tutorial](#)[Maths For Machine Learning](#)[Pandas Tutorial](#)[NumPy Tutorial](#)[NLP Tutorial](#)[Deep Learning Tutorial](#)

Competitive Programming

[Top DSA for CP](#)[Top 50 Tree Problems](#)[Top 50 Graph Problems](#)[Top 50 Array Problems](#)[Top 50 String Problems](#)[Top 50 DP Problems](#)[Top 15 Websites for CP](#)

Interview Corner

[Company Wise Preparation](#)[Preparation for SDE](#)[Experienced Interviews](#)[Internship Interviews](#)[Competitive Programming](#)[Aptitude Preparation](#)

Commerce

[Accountancy](#)[Business Studies](#)[Python Projects](#)[Python Tkinter](#)[OpenCV Python Tutorial](#)[Python Interview Question](#)

DevOps

[Git](#)[AWS](#)[Docker](#)[Kubernetes](#)[Azure](#)[GCP](#)

System Design

[What is System Design](#)[Monolithic and Distributed SD](#)[Scalability in SD](#)[Databases in SD](#)[High Level Design or HLD](#)[Low Level Design or LLD](#)[Top SD Interview Questions](#)

GfG School

[CBSE Notes for Class 8](#)[CBSE Notes for Class 9](#)[CBSE Notes for Class 10](#)[CBSE Notes for Class 11](#)[CBSE Notes for Class 12](#)[English Grammar](#)

UPSC

[Polity Notes](#)[Geography Notes](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#)

Management
Income Tax
Finance
Statistics for Economics

Science and Technology Notes
Economics Notes
Important Topics in Ethics
UPSC Previous Year Papers

SSC/ BANKING

SSC CGL Syllabus
SBI PO Syllabus
SBI Clerk Syllabus
IBPS PO Syllabus
IBPS Clerk Syllabus
Aptitude Questions
SSC CGL Practice Papers

Write & Earn

Write an Article
Improve an Article
Pick Topics to Write
Write Interview Experience
Internships

@geeksforgeeks , Some rights reserved