

Spring[®] Framework

Notes for Professionals



50+ pages
of professional hints and tricks

Contents

About	1
Chapter 1: Getting started with Spring Framework	2
Section 1.1: Setup (XML Configuration)	2
Section 1.2: Showcasing Core Spring Features by example	3
Section 1.3: What is Spring Framework, why should we go for it?	6
Chapter 2: Spring Core	8
Section 2.1: Introduction to Spring Core	8
Section 2.2: Understanding How Spring Manage Dependency?	9
Chapter 3: Spring Expression Language (SpEL)	12
Section 3.1: Syntax Reference	12
Chapter 4: Obtaining a SqlRowSet from SimpleJdbcCall	13
Section 4.1: SimpleJdbcCall creation	13
Section 4.2: Oracle Databases	14
Chapter 5: Creating and using beans	16
Section 5.1: Autowiring all beans of a specific type	16
Section 5.2: Basic annotation autowiring	17
Section 5.3: Using FactoryBean for dynamic bean instantiation	18
Section 5.4: Declaring Bean	19
Section 5.5: Autowiring specific bean instances with @Qualifier	20
Section 5.6: Autowiring specific instances of classes using generic type parameters	21
Section 5.7: Inject prototype-scoped beans into singletons	22
Chapter 6: Bean scopes	25
Section 6.1: Additional scopes in web-aware contexts	25
Section 6.2: Prototype scope	26
Section 6.3: Singleton scope	28
Chapter 7: Conditional bean registration in Spring	30
Section 7.1: Register beans only when a property or value is specified	30
Section 7.2: Condition annotations	30
Chapter 8: Spring JSR 303 Bean Validation	32
Section 8.1: @Valid usage to validate nested POJOs	32
Section 8.2: Spring JSR 303 Validation - Customize error messages	32
Section 8.3: JSR303 Annotation based validations in Springs examples	34
Chapter 9: ApplicationContext Configuration	37
Section 9.1: Autowiring	37
Section 9.2: Bootstrapping the ApplicationContext	37
Section 9.3: Java Configuration	38
Section 9.4: Xml Configuration	40
Chapter 10: RestTemplate	43
Section 10.1: Downloading a Large File	43
Section 10.2: Setting headers on Spring RestTemplate request	43
Section 10.3: Generics results from Spring RestTemplate	44
Section 10.4: Using Preemptive Basic Authentication with RestTemplate and HttpClient	44
Section 10.5: Using Basic Authentication with HttpComponent's HttpClient	46
Chapter 11: Task Execution and Scheduling	47
Section 11.1: Enable Scheduling	47
Section 11.2: Cron expression	47

Section 11.3: Fixed delay	49
Section 11.4: Fixed Rate	49
Chapter 12: Spring Lazy Initialization	50
Section 12.1: Example of Lazy Init in Spring	50
Section 12.2: For component scanning and auto-wiring	51
Section 12.3: Lazy initialization in the configuration class	51
Chapter 13: Property Source	52
Section 13.1: Sample xml configuration using PropertyPlaceholderConfigurer	52
Section 13.2: Annotation	52
Chapter 14: Dependency Injection (DI) and Inversion of Control (IoC)	53
Section 14.1: Autowiring a dependency through Java configuration	53
Section 14.2: Autowiring a dependency through XML configuration	53
Section 14.3: Injecting a dependency manually through XML configuration	54
Section 14.4: Injecting a dependency manually through Java configuration	56
Chapter 15: JdbcTemplate	57
Section 15.1: Basic Query methods	57
Section 15.2: Query for List of Maps	57
Section 15.3: SQLRowSet	58
Section 15.4: Batch operations	58
Section 15.5: NamedParameterJdbcTemplate extension of JdbcTemplate	59
Chapter 16: SOAP WS Consumption	60
Section 16.1: Consuming a SOAP WS with Basic auth	60
Chapter 17: Spring profile	61
Section 17.1: Spring Profiles allows to configure parts available for certain environment	61
Chapter 18: Understanding the dispatcher-servlet.xml	62
Section 18.1: dispatcher-servlet.xml	62
Section 18.2: dispatcher servlet configuration in web.xml	62
Credits	64
You may also like	65

About

Please feel free to share this PDF with anyone for free,
latest version of this book can be downloaded from:

<https://goalkicker.com/SpringFrameworkBook>

This *Spring® Framework Notes for Professionals* book is compiled from [Stack Overflow Documentation](#), the content is written by the beautiful people at Stack Overflow. Text content is released under Creative Commons BY-SA, see credits at the end of this book whom contributed to the various chapters. Images may be copyright of their respective owners unless otherwise specified

This is an unofficial free book created for educational purposes and is not affiliated with official Spring® Framework group(s) or company(s) nor Stack Overflow. All trademarks and registered trademarks are the property of their respective company owners

The information presented in this book is not guaranteed to be correct nor accurate, use at your own risk

Please send feedback and corrections to web@petercv.com

Chapter 1: Getting started with Spring Framework

Version Release Date

5.0.x	2017-10-24
4.3.x	2016-06-10
4.2.x	2015-07-31
4.1.x	2014-09-14
4.0.x	2013-12-12
3.2.x	2012-12-13
3.1.x	2011-12-13
3.0.x	2009-12-17
2.5.x	2007-12-25
2.0.x	2006-10-04
1.2.x	2005-05-13
1.1.x	2004-09-05
1.0.x	2003-03-24

Section 1.1: Setup (XML Configuration)

Steps to create Hello Spring:

1. Investigate Spring Boot to see if that would better suit your needs.
2. Have a project set up with the correct dependencies. It is recommended that you are using Maven or Gradle.
3. create a POJO class, e.g. `Employee.java`
4. create a XML file where you can define your class and variables. e.g `beans.xml`
5. create your main class e.g. `Customer.java`
6. Include [spring-beans](#) (and its transitive dependencies!) as a dependency.

`Employee.java`:

```
package com.test;

public class Employee {

    private String name;

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void displayName() {
        System.out.println(name);
    }
}
```

`beans.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-4.3.xsd">

  <bean id="employee" class="com.test.Employee">
    <property name="name" value="test spring"></property>
  </bean>

</beans>
```

Customer.java:

```
package com.test;

import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class Customer {
  public static void main(String[] args) {
    ApplicationContext context = new ClassPathXmlApplicationContext("beans.xml");

    Employee obj = (Employee) context.getBean("employee");
    obj.displayName();
  }
}
```

Section 1.2: Showcasing Core Spring Features by example

Description

This is a self-contained running example including/showcasing: minimum *dependencies* needed, Java *Configuration*, *Bean declaration* by annotation and Java Configuration, *Dependency Injection* by Constructor and by Property, and *Pre/Post* hooks.

Dependencies

These dependencies are needed in the classpath:

1. [spring-core](#)
2. [spring-context](#)
3. [spring-beans](#)
4. [spring-aop](#)
5. [spring-expression](#)
6. [commons-logging](#)

Main Class

Starting from the end, this is our Main class that serves as a placeholder for the `main()` method which initialises the Application Context by pointing to the Configuration class and loads all the various beans needed to showcase particular functionality.

```
package com.stackoverflow.documentation;

import org.springframework.context.ApplicationContext;
```

This is a sample, click download link to get the full Tutorial

CLICK BELOW

