

```
1. Book.java
2.
3. Book class is a simple class consisting of the book details such title, author,
   publications and its corresponding POJO's. The getBookDetails() method will display the
   book information which is set.
4.
5. package com.javainterviewpoint;
6.
7. public class Book
8. {
9.     private String title;
10.    private String publications;
11.    private String author;
12.
13.    public Book()
14.    {
15.        super();
16.    }
17.
18.    public Book(String title, String publications, String author)
19.    {
20.        super();
21.        this.title = title;
22.        this.publications = publications;
23.        this.author = author;
24.    }
25.
26.    public String getTitle()
27.    {
28.        return title;
29.    }
30.
31.    public void setTitle(String title)
32.    {
33.        this.title = title;
34.    }
35.
36.    public String getPublications()
37.    {
38.        return publications;
39.    }
40.
41.    public void setPublications(String publications)
42.    {
43.        this.publications = publications;
44.    }
45.
46.    public String getAuthor()
47.    {
48.        return author;
49.    }
50.
51.    public void setAuthor(String author)
52.    {
53.        this.author = author;
54.    }
55.
56.    public void getBookDetails()
57.    {
58.        System.out.println("**Published Book Details**");
59.        System.out.println("Book Title      : " + title);
60.        System.out.println("Book Author    : " + author);
61.        System.out.println("Book Publications : " + publications);
62.    }
63. }
64.
65. Library.java
66.
```

67. Library class has the Book class instance as a property and its corresponding getters and setters. The book property will get its value through our configuration file.

```
68.  
69. package com.javainterviewpoint;  
70.  
71. public class Library  
72. {  
73.     private Book book;  
74.  
75.     public void setBook(Book book)  
76.     {  
77.         this.book = book;  
78.     }  
79.  
80.     public Book getBook()  
81.     {  
82.         return book;  
83.     }  
84. }
```

85. SpringConfig.xml

87.  
88. In our configuration file we have defined separate id for each bean Library and Book classes

```
89.  
90. <beans xmlns="http://www.springframework.org/schema/beans"  
91. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
92. xsi:schemaLocation="http://www.springframework.org/schema/beans  
93. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">  
94.  
95.     <bean id="library" class="com.javainterviewpoint.Library">  
96.         <property name="book" ref="book"></property>  
97.     </bean>  
98.  
99.     <bean id="book" class="com.javainterviewpoint.Book">  
100.         <property name="title" value="Spring XML Configuration"></property>  
101.         <property name="author" value="JavaInterviewPoint"></property>  
102.         <property name="publications" value="JIP Publication"></property>  
103.     </bean>  
104. </beans>
```

105.  
106. We inject primitives to the Book class properties such as title, author, publications.

```
107.  
108. <bean id="book" class="com.javainterviewpoint.Book">  
109.     <property name="title" value="Spring XML Configuration"></property>  
110.     <property name="author" value="JavaInterviewPoint"></property>  
111.     <property name="publications" value="JIP"></property>  
112. </bean>
```

113.  
114. We are referencing the Book class bean id to the property book of the Library class

```
115.  
116. <property name="book" ref="book"></property>
```

117.  
118. The ref passed to the property book should be the bean id of the Book Class. In short

```
119.  
120. ref =<<bean id of Book class>>  
121. Application.java
```

```
122.  
123. package com.javainterviewpoint;  
124.  
125. import org.springframework.context.ApplicationContext;  
126. import org.springframework.context.support.ClassPathXmlApplicationContext;  
127.  
128. public class Application  
129. {  
130.     public static void main(String args[])  
131.     {
```

```

132.         // Read the Spring configuration file [SpringConfig.xml]
133.         ApplicationContext appContext = new
ClassPathXmlApplicationContext("SpringConfig.xml");
134.         // Get the Library instance
135.         Library library = (Library) appContext.getBean("library");
136.         // Get the Book Details
137.         library.getBook().getBookDetails();
138.     }
139. }
140.

```

```

1. Book.java
2.
3. Book class is a simple class consisting of the book details such title, author,
publications and its corresponding POJO's. The getBookDetails() method will display the
book information which is set.
4.
5. package com.javainterviewpoint;
6.
7. public class Book
8. {
9.     private String title;
10.    private String publications;
11.    private String author;
12.
13.    public Book()
14.    {
15.        super();
16.    }
17.
18.    public Book(String title, String publications, String author)
19.    {
20.        super();
21.        this.title = title;
22.        this.publications = publications;
23.        this.author = author;
24.    }
25.
26.    public String getTitle()
27.    {
28.        return title;
29.    }
30.
31.    public void setTitle(String title)
32.    {
33.        this.title = title;
34.    }
35.
36.    public String getPublications()
37.    {
38.        return publications;
39.    }
40.
41.    public void setPublications(String publications)
42.    {
43.        this.publications = publications;
44.    }
45.
46.    public String getAuthor()
47.    {
48.        return author;
49.    }
50.
51.    public void setAuthor(String author)
52.    {
53.        this.author = author;
54.    }
55.
56.    public void getBookDetails()

```

```

57.     {
58.         System.out.println("***Published Book Details***");
59.         System.out.println("Book Title      : " + title);
60.         System.out.println("Book Author   : " + author);
61.         System.out.println("Book Publications : " + publications);
62.     }
63. }
64.
65. Library.java
66.
67. Library class has the Book class instance as a property and its corresponding getters
    and setters. The book property will get its value through our configuration file.
68.
69. package com.javainterviewpoint;
70.
71. public class Library
72. {
73.     private Book book;
74.
75.     public void setBook(Book book)
76.     {
77.         this.book = book;
78.     }
79.
80.     public Book getBook()
81.     {
82.         return book;
83.     }
84. }
85.
86. SpringConfig.xml
87.
88. In our configuration file we have defined separate id for each bean Library and Book
    classes
89.
90. <beans xmlns="http://www.springframework.org/schema/beans"
91. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
92. xsi:schemaLocation="http://www.springframework.org/schema/beans
93. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
94.
95.     <bean id="library" class="com.javainterviewpoint.Library">
96.         <property name="book" ref="book"></property>
97.     </bean>
98.
99.     <bean id="book" class="com.javainterviewpoint.Book">
100.         <property name="title" value="Spring XML Configuration"></property>
101.         <property name="author" value="JavaInterviewPoint"></property>
102.         <property name="publications" value="JIP Publication"></property>
103.     </bean>
104. </beans>
105.
106. We inject primitives to the Book class properties such as title, author,
    publications.
107.
108. <bean id="book" class="com.javainterviewpoint.Book">
109.     <property name="title" value="Spring XML Configuration"></property>
110.     <property name="author" value="JavaInterviewPoint"></property>
111.     <property name="publications" value="JIP"></property>
112. </bean>
113.
114. We are referencing the Book class bean id to the property book of the Library
    class
115.
116. <property name="book" ref="book"></property>
117.
118. The ref passed to the property book should be the bean id of the Book Class. In
    short
119.
120. ref =<<bean id of Book class>>
121. Application.java

```

```
122.
123. package com.javainterviewpoint;
124.
125. import org.springframework.context.ApplicationContext;
126. import org.springframework.context.support.ClassPathXmlApplicationContext;
127.
128. public class Application
129. {
130.     public static void main(String args[])
131.     {
132.         // Read the Spring configuration file [SpringConfig.xml]
133.         ApplicationContext appContext = new
ClassPathXmlApplicationContext("SpringConfig.xml");
134.         // Get the Library instance
135.         Library library = (Library) appContext.getBean("library");
136.         // Get the Book Details
137.         library.getBook().getBookDetails();
138.     }
139. }
140.
```